Implementação de um Serviço de Sincronização Multimídia para Comunicação de Dados Pré-orquestrados[1]

Ronaldo Mussauer de Lima
mussauer@gta.ufrj.br

Otto Carlos Muniz B. Duarte
otto@gta.ufrj.br

Grupo de Teleinformática e Automação (GTA)
Universidade Federal do Rio de Janeiro
COPPE/EE - Prog. de Eng. Elétrica/Depto. de Eletrônica
C.P. 68504 - CEP 21945-970 - Rio de Janeiro - RJ - Brasil

RESUMO

Este trabalho apresenta a proposta e a implementação de um serviço de sincronização multimídia que oferece facilidades de suporte à comunicação e apresentação, em tempo real, de informações multimídias pré-orquestradas armazenadas em bases de dados remotas. Como base para os mecanismos de sincronização foi utilizado o modelo estendido da rede de Petri para composição de objetos (XOCPN). A aplicação alvo é descrita, o serviço é definido através de suas primitivas de serviço e o protocolo é especificado através dos tipos e formatos de suas unidades de dados de protocolo. Foi utilizada uma arquitetura de implementação de alto desempenho, contendo módulos de escalonamento de tarefas com prioridades, gerenciamento de memória otimizado e gerenciamento de temporizações. O artigo apresenta ainda a definição das estruturas de dados para o modelo XOCPN e um diagrama de implementação do mecanismo de sincronização, descrevendo algumas rotinas relativas aos módulos de sincronização relacionados à XOCPN.

ABSTRACT

This work presents the proposal and the implementation of a multimedia synchronization service that offers facilities to real time communication and presentation of pre-orchestrated multimedia information, that is stored in remote data bases. As a basis for synchronization mechanisms, the eXtended Object Composition Petri Net (XOCPN) has been employed. The target application is described, the synchronization service is defined by its service primitives and the protocol is specified through its data unit's types and field formats. A high performance implementation architecture has been used, defining specialized modules of priority-based task scheduling, optimized memory management and timer management. This paper also presents the definition of the data structures for the XOCPN model and an implementation diagram of the synchronization's mechanism, describing some procedures of the XOCPN-related synchronization's modules.

1. INTRODUÇÃO

O avanço nos sistemas de comunicação, incluindo redes e protocolos de alta velocidade, tem motivado o desenvolvimento e a implementação de aplicações multimídias distribuídas, que integrem diversos tipos de mídias como áudio, vídeo, texto, gráficos etc [Stei90]. A necessidade de integração decorre principalmente das dependências existentes entre as informações contidas nas mídias, tais como dependências temporais, espaciais e relações causais [CCO96]. Desta forma, todos os procedimentos que envolvam a integração de mídias, como a captura, a manipulação, o armazenamento, a comunicação e, principalmente, a apresentação devem refletir estas dependências [BlSt96]. Cada tipo de mídia possui requisitos

[1]Este trabalho foi realizado com recursos da UFRJ, FUJB, CNPq, PROTEM-CC, CAPES e COFECUB.

específicos de desempenho relativos às restrições de tempo real para apresentação, armazenamento e comunicação, e podem sofrer atrasos randômicos durante a transmissão e recuperação dos dados. De forma a dar suporte à comunicação de dados multimídias e a apresentação coerente no destino, é essencial a garantia da sincronização dos elementos de dados. No contexto de multimídia, o conceito de sincronização é amplo, incluindo o planejamento do acesso a recursos compartilhados, e a especificação e o controle de atividades conjuntas de processos cooperantes [Stei90]. Mais restritamente, técnicas de sincronização são mecanismos destinados a coordenar a ordenação temporal e espacial dos eventos relacionados à aplicação. Considerando as relações temporais, derivadas da integração de mídias dependentes no tempo, como um dos principais aspectos dos sistemas multimídias, a abordagem do conceito de sincronização multimídia feita neste trabalho se baseia nas relações temporais.

Diversas técnicas têm sido propostas na literatura. Apesar de nenhuma delas oferecer uma solução geral para o problema, elas procuram satisfazer os mais diferentes requisitos de sincronização, apresentando vantagens e desvantagens [EFI94], dependendo do tipo de aplicação que se deseja usar, do sistema de comunicação utilizado e dos recursos disponíveis. Dependendo da técnica proposta, implementam o controle da sincronização na fonte e/ou destino, utilizam esquemas para sincronização local ou distribuída, utilizam técnicas baseadas em protocolos, utilizam linguagem ou modelos específicos, aplicam-se a fontes de dados ao vivo e/ou armazenados etc.

A proposta de Nicolau [Nico90] utiliza técnicas de sincronização baseadas em protocolos e identifica dois níveis de elementos de dados, denominados Quadros de Sincronização Física (*Physical Synchronization Frames* - PSFs) e Quadros de Sincronização Lógica (*Logical Synchronization Frames* - LSFs), para definir pontos de sincronização. Os pontos de sincronização são representados por variáveis e são definidas três primitivas para estes pontos. A proposta de Escobar et al. [EPD94] também utiliza técnicas de sincronização baseadas em protocolos, usadas principalmente para configurações de múltiplas fontes e/ou destinos. O protocolo insere estampas de tempo nos dados na fonte e utiliza um atraso de equalização (fixo ou adaptativo) em todos os destinos, com o objetivo de fixar um atraso igual para todos os fluxos e, conseqüentemente, sincronizá-los. Ramanathan & Rangan [RaRa93] propõem técnicas de realimentação para sincronizar sistemas multimídias distribuídos. A sincronização é implementada em servidores. A realimentação é utilizada para manter a sincronização intramídia - através de uma estimativa, feita pelo servidor, do menor e do maior tempo que a apresentação das unidades de mídia poderia ser iniciada - como para manter a sincronização intermídias - através da associação, a cada unidade de mídia, de estampas de tempo relativo, objetivando a determinação das relações temporais entre fluxos relacionados. Carmo et al. [CCS93] propõem um conjunto de primitivas de serviço para uma camada de sincronização que utilize serviços da camada transporte. Courtiat & Carmo [CCO96] definem um mecanismo de propósito geral, para sincronização intra e intermídias, baseado na identificação das relações causais entre unidades de informação de um ou vários fluxos. A técnica proposta por Qazi et al. [QWG93], para comunicação de objetos multimídias distribuídos, utiliza um modelo estendido (XOCPN) da rede de Petri temporizada para composição de objetos, proposto por Little & Ghafoor [LiGh90a, LiGh91]. Woo et al. [WQG94] propõem duas camadas de protocolos de sincronização, baseadas no modelo XOCPN, uma para intramídia e outra para intermídias, as quais destinam-se à comunicação de informações multimídias pré-orquestradas.

O presente artigo apresenta a implementação de um serviço de sincronização multimídia, desenvolvido no Grupo de Teleinformática e Automação (GTA) da UFRJ, que destina-se a aplicações distribuídas, em tempo real, e provê um completo esquema para apresentação e comunicação de dados multimídias armazenados e pré-orquestrados [LiDu96,

Lima96]. As técnicas de sincronização utilizadas se baseiam nas propostas de Qazi et al. [QWG93] e Woo et al. [WQG94], por se aplicarem bem para a finalidade deste trabalho. O artigo define o serviço utilizado para prover funções referentes à comunicação dos dados e apresentação sincronizada dos mesmos no destino, bem como o protocolo de sincronização; apresenta a arquitetura de implementação proposta, que inclui módulos especializados e estruturas de dados específicas; e descreve algumas rotinas relativas aos módulos de sincronização relacionados à XOCPN.

Este artigo está organizado da seguinte maneira. Na Seção 2 são apresentados os conceitos básicos. Na Seção 3 é descrita a aplicação alvo. A Seção 4 descreve o mecanismo de sincronização, incluindo a técnica para modelagem e comunicação de informações multimídias. A Seção 5 apresenta o serviço proposto, incluindo suas primitivas, e o protocolo de sincronização, incluindo a descrição das unidades de dados de protocolo, relacionando-os com o modelo XOCPN. Na Seção 6 é apresentada a implementação do mecanismo de sincronização, incluindo a arquitetura de implementação de alto desempenho utilizada; a descrição das estruturas de dados da XOCPN; o diagrama das rotinas envolvidas na execução da XOCPN e o pseudo-código de algumas dessas rotinas. A Seção 7 apresenta as conclusões.

2. CONCEITOS BÁSICOS

Através de dispositivos e interfaces específicos, os sistemas de computação capturam e registram eventos do ambiente externo. As informações provenientes desta captura podem ser apresentadas imediatamente ou transferidas através de uma rede de computadores para serem apresentadas em instalações remotas; ou podem, também, ser armazenadas em dispositivos apropriados, para posterior recuperação e apresentação. Os dispositivos dos quais se obtém os dados podem ser classificados em dois tipos:

- **Fontes de dados ao vivo** - são dispositivos que capturam os eventos do meio externo, convertendo-os em dados para o sistema. Estas fontes geram dados em tempo real, na mesma taxa em que são requeridos para consumo (e.g., câmeras de vídeo);
- **Fontes de dados armazenados** - são dispositivos que armazenam os dados capturados. Estas fontes podem gerar dados a taxas diferentes da qual foram capturados e/ou serão apresentados. Com isto, o sistema possui uma flexibilidade maior para planejar os tempos nos quais os dados serão comunicados para a aplicação [LiGh91]. Um exemplo deste tipo de fonte é o CD-ROM.

Os dados multimídias podem ser classificados em dois tipos:

- **Dados isócronos**, ou contínuos, referem-se aos dados gerados a partir da captura de um evento que se sucede continuamente. A seqüência de informações inerente ao evento é capturada à uma taxa constante. Por possuírem uma dependência temporal implícita em sua geração, estes dados devem ser apresentados continuamente no tempo para que possuam significado [EFI94]. Dados relativos a áudio e vídeo são um exemplo deste tipo;
- **Dados não-isócronos** são dados que não possuem relação com taxa de captura, geração, comunicação ou apresentação. O mais importante no que se refere a estes dados é a composição espacial dos mesmos no momento da apresentação. Alguns exemplos deste tipo de dados são textos, gráficos e desenhos.

Um conjunto de dados pode possuir uma relação temporal sintética com outro conjunto, ou seja, é possível existir entre eles uma relação temporal formulada explicitamente. Neste caso, estes dados recebem a denominação de **pré-orquestrados**. Desta forma, fontes de dados ao vivo não geram dados pré-orquestrados. A relação temporal sintética entre um determinado conjunto de dados é formulada e os dados, com as respectivas relações temporais, são armazenados para posterior recuperação. Um exemplo é uma apresentação de gráficos

com respectivos textos explicativos. Os dados pré-orquestrados podem também ser referentes a uma única mídia, por exemplo uma apresentação de *slides*.

Um termo largamente utilizado para se referenciar aos elementos de dados em multimídia é **objeto multimídia**. Um objeto multimídia, ou simplesmente objeto, é uma unidade de informação multimídia, complexa ou simples, sendo constituída por elementos de dados relativos à uma determinada mídia. Os objetos podem ser armazenados, transmitidos e apresentados para o usuário de diversas formas. Um objeto pode corresponder, por exemplo, a um segmento de vídeo, ou a um único quadro do segmento do vídeo ou, até mesmo, a um *pixel* do quadro do vídeo [LiGh90b]. Ou seja, não existe um tamanho ou um tempo de duração fixo para os objetos, variando de acordo com a aplicação. Uma seqüência temporal de objetos multimídias é denominada **fluxo**.

A sincronização de dados multimídias pode ser classificada em dois tipos:

- **Sincronização intramídia** - trata as dependências temporais dos dados relativos à uma mídia individual. Tomando como exemplo uma mídia isócrona, a qual deve ser apresentada continuamente e à uma taxa constante, a sincronização intramídia coordena a comunicação dos dados referentes à mídia de forma ordenada e temporizada. Este tipo de sincronização também é aplicável a dados não-isócronos pré-orquestrados;
- **Sincronização intermídias** - trata as relações temporais entre mídias. Neste caso, mais de uma mídia estão envolvidas. As relações temporais entre elas podem ser contínuas no tempo, as quais ocorrem implicitamente entre dados isócronos capturados simultaneamente, como os de áudio e vídeo em um videofone; ou podem ser sintéticas, onde a seqüência e duração de apresentação dos dados e as relações temporais entre as mídias são formuladas explicitamente, relacionando eventos que devem ser apresentados conjuntamente. Este tipo de sincronização é responsável por manter as relações temporais entre mídias na recuperação e na apresentação dos dados relativos à elas.

Uma apresentação multimídia de dados pré-orquestrados pode envolver diversos fluxos. Este tipo de apresentação possui características específicas de sincronização, dependendo de como a relação temporal dos dados é formulada (ou pré-orquestrada). Tais características dizem respeito, por exemplo, à ordem de exibição dos objetos, ao instante inicial de exibição e à duração de exibição de cada objeto, à taxa de exibição dos objetos isócronos, à exibição simultânea de objetos relacionados no tempo etc. Um caminho para que a apresentação multimídia transcorra corretamente é a representação ou especificação das características de sincronização, através de um método para modelar as dependências temporais dos objetos envolvidos na apresentação. Além disto, quando a aplicação realiza a transmissão dos objetos a serem apresentados no destino, a modelagem relativa à fonte deve levar em conta as características do tráfego de dados, pois os atrasos randômicos e, principalmente, as variações de atraso (*jitter*) acabam por modificar as características de sincronização [RPT96]. Na Seção 4 é apresentado um método para modelar a sincronização intramídia e intermídias de fluxos de objetos multimídias relativos a dados pré-orquestrados, isócronos ou não.

3. A APLICAÇÃO

A aplicação visada consiste de um sistema acessado pelo usuário final, denominado **sistema local**, que deseja realizar diferentes apresentações multimídias cujos dados se encontram previamente armazenados em uma base de dados acessada por um outro sistema, denominado **sistema remoto**. Desta forma, a Aplicação oferece ao usuário as diferentes apresentações multimídias que estão compostas e armazenadas e ele seleciona qual deseja

assistir e os parâmetros de qualidade de serviço da apresentação, como número de quadros por segundo, dimensão da imagem, padrão de compressão, *bits* por *pixel* etc, dependendo da mídia.

A composição da apresentação multimídia se refere às diversas formas que os dados (objetos multimídias) podem ser grupados. A especificação completa de cada apresentação multimídia fica também armazenada na base de dados e pode ser identificada e acessada, pelo sistema remoto. Após informada sobre a apresentação escolhida, o sistema remoto recupera as informações referentes à apresentação multimídia correspondente e, posteriormente, recupera os dados para serem transmitidos ao sistema local onde será efetuada a apresentação. A Figura 1 apresenta o lugar da Aplicação na arquitetura do sistema.

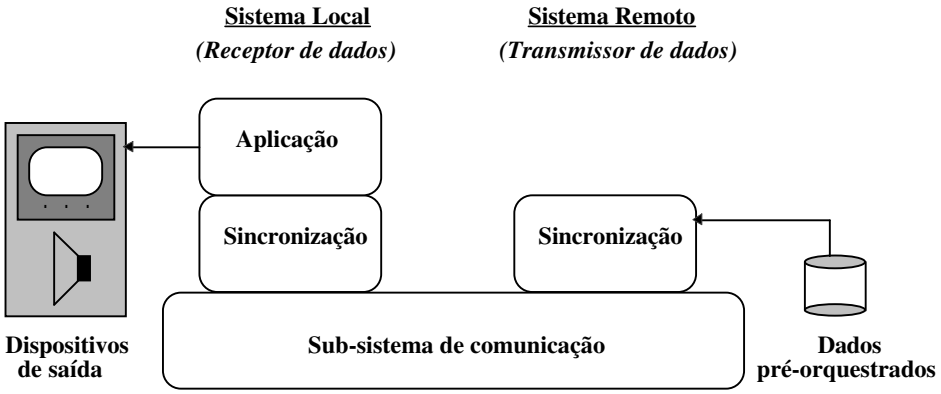

Figura 1: Lugar da Aplicação na arquitetura do sistema.

Tendo em vista a aplicação alvo, este trabalho define o serviço e especifica o protocolo relativo à camada de sincronização, que é responsável pela transferência sincronizada de informações do sistema remoto para o sistema local através de uma rede de computadores. A seção seguinte descreve o mecanismo de sincronização utilizado na implementação do serviço proposto.

4. O MECANISMO DE SINCRONIZAÇÃO MULTIMÍDIA

Foi utilizado o modelo XOCPN, envolvido nas propostas de Qazi et al. [QWG93] e Woo et al. [WQG94], como base para o mecanismo de sincronização do serviço proposto, pelo fato de ser bem adequado ao tipo de aplicação escolhido. O modelo XOCPN é uma extensão do modelo OCPN, cujas descrições se encontram a seguir.

Modelo OCPN

A apresentação de múltiplas mídias envolve tarefas com atividades paralelas e seqüenciais. Um caminho para especificar de forma clara e tratar mais facilmente as relações temporais entre as mídias é utilizar um método para modelar estas relações. Um método usado para especificar a sincronização e as relações entre os dados é proposto por Little et al. [LiGh90a], o qual utiliza um modelo para composição de objetos multimídias, baseado em redes de Petri temporizadas (*Object Composition Petri Net* - OCPN). O modelo OCPN foi elaborado a partir dos modelos de redes de Petri temporizadas pelo fato destes possuírem atributos de representação de eventos concorrentes e assíncronos. As redes de Petri temporizadas atribuem um tempo de duração de disparo para cada transição. As redes de Petri temporizadas estendidas representam processos por lugares, ao invés de transições. Além disto, estas redes atribuem um tempo de duração diferente de zero para cada lugar e disparo instantâneo para as transições. A OCPN incorpora, modifica e acrescenta características a

estes modelos. No modelo OCPN, os lugares estão relacionados com a utilização de recursos e possuem um tempo de duração. O disparo de uma transição na OCPN é assumido como sendo um evento instantâneo. Através da OCPN, é possível especificar as diversas relações temporais entre objetos multimídias [LiGh91], já que ela descreve efetivamente relações temporais do tipo "antes", "depois", "durante", "inicia junto", "termina junto" etc. Os recursos associados aos lugares da OCPN são relacionados com os tipos de objetos, tais como áudio, vídeo, texto etc. As OCPNs são redes do tipo série-paralela (não possuindo realimentação), são representada por grafos e especificadas pela tupla {T, P, A, D, R, M}, explicada adiante. A Figura 2 mostra o diagrama de uma OCPN, com tempo de apresentação total de 30s, no qual estão indicados, para cada lugar, o tipo de objeto e a duração do mesmo em segundos.

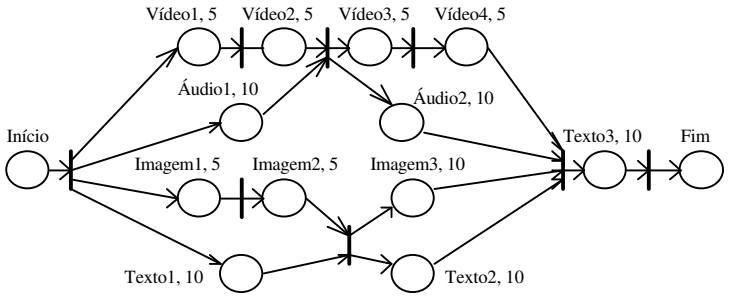

Figura 2: Diagrama de uma OCPN.

Modelo XOCPN

O modelo OCPN, descrito anteriormente, especifica os requisitos temporais ao nível da aplicação, mas se mostra insuficiente para suprir as demandas de comunicação de um sistema distribuído interconectado através de uma rede de computadores. Desta forma, a sincronização multimídia necessita de um modelo que especifique os mecanismos de comunicação e sincronização de dados considerando este ambiente, principalmente para prevenir e/ou compensar irregularidades provocadas por atrasos randômicos introduzidos pela rede. Além disto, a sincronização de objetos relacionados que apresentam uma dependência temporal contínua é mais crítica, uma vez que exige comunicação e apresentação com taxa controlada.

Qazi et al. [QWG93] propõem um modelo estendido da OCPN, denominado XOCPN, o qual divide os objetos em sub-objetos, com a finalidade de melhor coordenar e manter as relações temporais contínuas entre diversas mídias. Cada sub-objeto é chamado de unidade de intervalo de sincronização (*Synchronization Interval Unit* - SIU). O modelo estendido é gerado a partir das informações contidas na OCPN e visa uma sincronização mais eficiente de dados isócronos, apesar de também ser utilizado para dados não-isócronos. As XOCPNs são, também, representadas por grafos e especificadas pela tupla {T, P, A, D, R, M, Y, Z}, onde:

$T = \{t_1, t_2, ..., t_n\}$ é um conjunto de transições;

$P = \{p_1, p_2, ..., p_m\}$ é um conjunto de *m* lugares;

$A = \{TxP\} \cup \{PxT\} \rightarrow I$, $I = \{1, 2, ...\}$ é um conjunto de arcos diretos;

$D: P \rightarrow \{d_1, d_2\}$ onde d_1, d_2 são números reais e representam o atraso antes da execução de uma ação e a duração de execução da ação, respectivamente;

$R: P \rightarrow \{r_1, r_2, ..., r_k\}$ é um mapeamento a partir do conjunto de lugares para um conjunto de recursos, correspondendo a objetos de vídeo, áudio, imagem, gráfico ou texto;

$M: P \rightarrow I$, $I = \{0, 1, 2, ...\}$ é um mapeamento a partir do conjunto de lugares para o conjunto de inteiros, que representam o número de fichas em um lugar especificado;

$Y: P \rightarrow$ {Iniciar-recursos, Liberar-recursos, Apresentar-SIU, Transmitir-SIU, Sincronizar-intermídias} é um mapeamento a partir do conjunto de lugares para o conjunto de tipos de ações a serem desempenhadas durante a comunicação;

Z: P → {endereço dos parâmetros de qualidade de serviço, endereço de uma SIU, endereço dos requisitos de sincronização} é um mapeamento a partir do conjunto de lugares para endereços de parâmetros e dados respectivos.

Para realizar a comunicação de informações multimídias pré-orquestradas, é fundamental que se estabeleça um planejamento da transmissão e da recepção dos dados [WoGh94], pois independente dos atrasos randômicos da rede, os dados devem ser entregues ao destino em tempo para sua apresentação. E para que o sistema receptor de dados possa melhor controlar a apresentação dos objetos multimídias, o sistema transmissor de dados deve realizar a transmissão de cada sub-objeto no tempo apropriado. Desta forma, as XOCPNs são definidas no sistema local e no remoto, possuindo funções para sincronização de múltiplos fluxos. No sistema remoto, a XOCPN responsabiliza-se por planejar a transmissão dos sub-objetos, utilizando informações sobre os atrasos da rede e a capacidade do canal de comunicação. No sistema local, a XOCPN responsabiliza-se por compensar irregularidades ocorridas durante a comunicação dos dados, tais como perda ou atraso de sub-objetos.

As XOCPNs são geradas a partir da OCPN. A Figura 3 apresenta um exemplo de uma OCPN simples, com 40 segundos de duração total de apresentação, e as correspondentes XOCPNs de recepção e de transmissão, considerando a utilização de canais independentes para cada mídia.

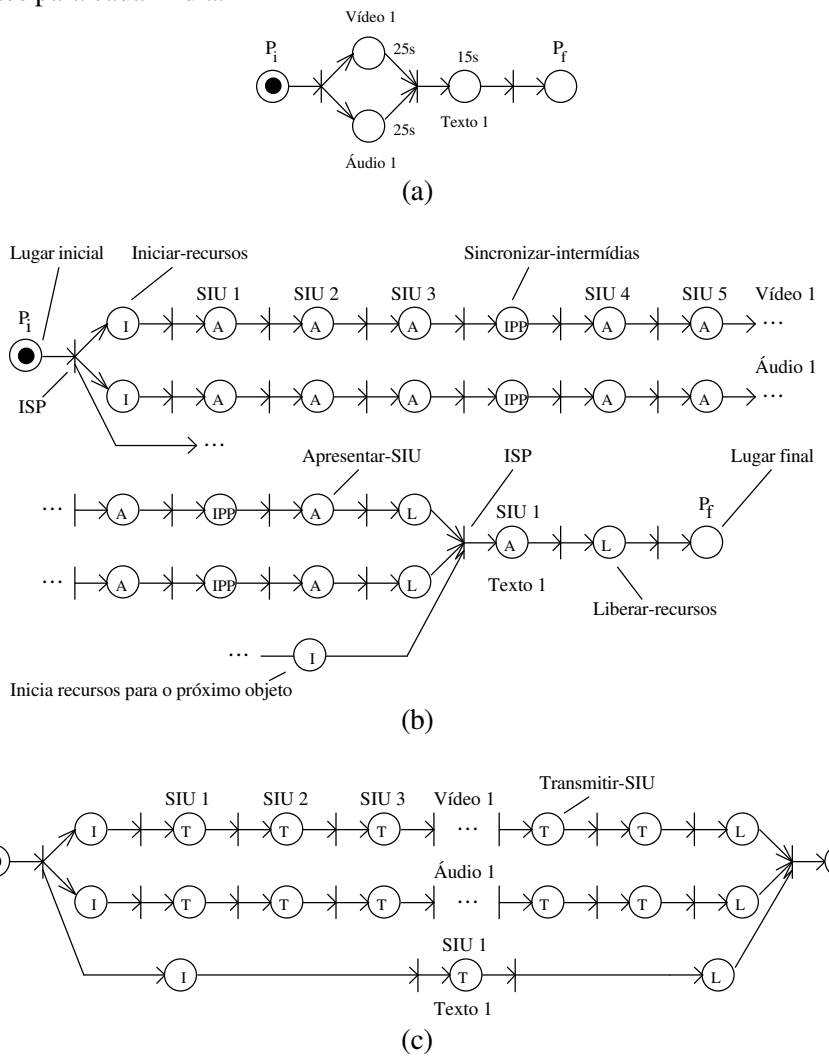

Figura 3: Representação da (a) OCPN, (b) XOCPN receptora e (c) XOCPN transmissora.

As SIUs são devidamente indexadas por números de seqüência para que a XOCPN na recepção possa identificá-las e apresentá-las no instante planejado. Para melhor controlar a sincronização intermídias, os objetos relacionados são divididos em sub-objetos (SIUs) de igual duração. No entanto, apesar de dois sub-objetos possuírem a mesma duração, o tamanho em *bytes* de cada um não necessariamente será igual, principalmente quando corresponderem a mídias distintas.

A função Y especifica os tipos de ação que podem caracterizar os lugares na XOCPN. A seguir, a descrição de cada uma das ações:

- **Iniciar-recursos (I)** - abre um canal para comunicação dos dados, negocia os atributos do canal de acordo com a qualidade de serviço desejada, reserva *buffers* de recepção ou transmissão, e prepara os dispositivos de saída para a apresentação dos dados;
- **Liberar-recursos (L)** - libera recursos após a apresentação completa do objeto que os utiliza;
- **Apresentar-SIU (A)** - apresenta o sub-objeto (SIU), associado ao lugar, no dispositivo de saída correspondente;
- **Transmitir-SIU (T)** - transmite o sub-objeto (SIU), associado ao lugar, através do canal correspondente.
- **Sincronizar-intermídias (IPP)** - Implementa uma política de sincronização intermídias.

Assim definido, o modelo XOCPN é um refinamento do modelo OCPN que inclui um completo esquema para comunicação, transmissão, apresentação e sincronização de dados multimídias, apropriado para um ambiente distribuído de redes de computadores.

Para o planejamento da recepção e transmissão dos dados, é necessário determinar os valores de d_1 e d_2, do componente D, da tupla das respectivas XOCPNs. Na recepção, estes valores são determinados tomando-se como base os próprios tempos de apresentação (início e duração) de cada objeto, especificados pela correspondente OCPN. Na transmissão, o planejamento deve ser feito, não só considerando as especificações temporais da OCPN, relativas à apresentação dos objetos, como o máximo possível de fatores que influenciam no tempo de transferência dos sub-objetos, desde a base de dados até o recebimento pela XOCPN receptora. Desta forma, este planejamento é fortemente dependente da capacidade oferecida pelo ambiente utilizado, incluindo a rede, o canal de comunicação, o processador, a base de dados etc. Little e Ghafoor [LiGh91, LiGh92] propõem um método para planejar a recuperação de dados armazenados, visando principalmente a utilização de canais com capacidade limitada.

Estratégias de Sincronização

O planejamento da recuperação dos dados por si só não garante uma apresentação síncrona no destino, principalmente quando os dados são provenientes de fontes distribuídas. O comportamento não determinístico da rede pode introduzir erros e atrasos durante a comunicação. Neste sentido, é essencial que o sistema receptor de dados corrija ou compense problemas relacionados com fluxos lentos, mantendo as relações temporais entre os dados, especificadas pela XOCPN, e provendo a qualidade de apresentação desejada [WQG94, YaHu96]. As ações a serem tomadas incluem diferentes estratégias (ou políticas) de sincronização [QWG93], tais como estratégia bloqueante, não-bloqueante e bloqueante restrita. A escolha por uma estratégia específica depende do tipo de sincronização: intramídia ou intermídias.

Na estratégia bloqueante, o processo de apresentação suspende até o sub-objeto esperado chegar. Na estratégia não-bloqueante, o processo de apresentação continua mesmo que o sub-objeto esperado não tenha chegado. Na estratégia bloqueante restrita, o processo de

apresentação suspende por um período de tempo pré-determinado. Após este período, o processo descarta o sub-objeto atrasado e, por exemplo, repete a apresentação do último sub-objeto.

Neste trabalho foi adotada a estratégia bloqueante restrita para sincronização intramídia e para a sincronização intermídias relacionada com as transições entre diferentes fluxos. Por outro lado, foi adotada a estratégia não-bloqueante para a sincronização intermídias relacionada com os Pontos de Compasso Intermídias (*Interstream Pacing Points* - IPPs). Estes pontos são, na realidade, os lugares **Sincronizar-intermídias**, os quais aparecem na Figura 3(b), que são inseridos na rede de Petri para permitir a implementação de uma determinada política de sincronização intermídias. A periodicidade destes lugares depende das mídias envolvidas e da qualidade de apresentação desejada. Neste caso, sempre que este lugar for alcançado por um dos fluxos, o processo de apresentação deste informa ao dos outros fluxos relacionados a ele que o lugar foi alcançado. A partir desta informação, os processos de apresentação descartarão os sub-objetos anteriores ao lugar e continuarão a apresentação a partir deste ponto. Assim, as apresentações das mídias com fluxo lento são adiantadas de forma a manter a sincronização entre as mídias envolvidas. A estratégia bloqueante não foi utilizada para evitar interrupções durante a apresentação dos objetos multimídias.

5. A CAMADA SINCRONIZAÇÃO

Na implementação realizada neste trabalho as funções de sincronização estão concentradas em uma camada, denominada **Sincronização**. Esta camada oferece serviços para aplicações que requerem comunicação e apresentação sincronizada de dados pré-orquestrados e armazenados em uma base de dados remota. As principais funções desta camada são as sincronizações intramídia e intermídias, utilizando estratégias específicas, a partir do modelo XOCPN.

A especificação completa de cada apresentação multimídia é feita através de uma OCPN, que fica também armazenada na base de dados e pode ser identificada e acessada, pelo sistema remoto, através de um índice próprio. Após informado sobre qual apresentação o usuário deseja assistir, o sistema remoto recupera as informações referentes à OCPN correspondente, gera a XOCPN transmissora e envia os dados relativos à OCPN para o sistema local gerar a XOCPN receptora. Posteriormente, a partir do planejamento determinado para a XOCPN transmissora, o sistema remoto recupera os objetos multimídias da base de dados para serem transmitidos e apresentados no sistema local.

Primitivas de Serviço

Observando a Figura 1, pode ser visto que as funções da camada Sincronização são claramente assimétricas e, portanto, as funções executadas pelas entidades local e remota (e, conseqüentemente, suas Máquinas de Estados) são distintas. Desta forma, as primitivas de serviço são definidas para a entidade de Sincronização local.

A Tabela 1 apresenta todas as primitivas definidas para o acesso aos serviços da entidade de Sincronização local, com uma breve descrição e os respectivos parâmetros de cada uma delas.

Primitiva	Descrição	Parâmetros
SINC-INITIALIZE.Request	Pedido para iniciar a apresentação multimídia selecionada	Endereço Destino Apresentação selecionada Qualidade de serviço
SINC-PLAYOUT.Indication	Indicação de apresentação de um sub-objeto (SIU)	Tipo do objeto Número do objeto Número do sub-objeto Tamanho do sub-objeto Sub-objeto (SIU)
SINC-STOP.Request	Pedido de interrupção da apresentação	Não tem
SINC-STOP.Indication	Indicação de falha ou de término dos dados relativos ao objeto	Condição de interrupção

Tabela 1: Primitivas de serviço de sincronização.

Protocolo de Sincronização

O protocolo segue o conceito de protocolos de alto desempenho com apenas cinco tipos de unidades de dados de protocolo (PDUs), e usando alinhamento em 32 *bits*.

A assimetria das funções da camada sincronização, citada anteriormente, levou a uma definição do conceito aplicado aos dados do usuário, que difere do conceito utilizado pelos protocolos convencionais. No esquema assimétrico proposto neste trabalho, *dados do usuário* significam os dados que são entregues à Aplicação no sistema local. Estes dados, neste caso, são os sub-objetos recuperados da base de dados pela entidade de Sincronização remota e inseridos na PDU de dados relativos ao sub-objeto.

Todas as PDUs definidas possuem um campo de 32 *bits*, que identifica o tipo da PDU, os quais podem ser:

- 0x00: **IREQ** Pedido de inicialização dos procedimentos de apresentação;
- 0x01: **INETDT** Pedido de inicialização de geração da XOCPN;
- 0x02: **IREADY** Confirmação de término da geração da XOCPN;
- 0x03: **IFAIL** Indicação de falha na inicialização;
- 0x04: **SIUDT** Dados relativos ao sub-objeto e sua identificação.

Cada PDU, exceto a PDU IREADY, possui um segmento de dados com um ou mais campos contendo informações de controle específicas de cada PDU. A PDU SIUDT é a única que contém, dentro do segmento de dados, um campo de *dados do usuário*.

6. IMPLEMENTAÇÃO DO MECANISMO DE SINCRONIZAÇÃO

A implementação da técnica de sincronização escolhida envolve a representação de lugares e transições da rede de Petri em linguagem de programação, e as operações com estes elementos; a geração automática da XOCPN a partir da OCPN; a execução específica de cada lugar; o correto disparo das transições; o tratamento e a compensação do atraso dos sub-objetos; a aplicação das políticas específicas de sincronização intermídias e controle da sincronização intramídia etc. A implementação completa do serviço inclui ainda os procedimentos inerentes aos protocolos de comunicação, como a construção das diferentes Máquinas de Estados, envolvendo as primitivas e PDUs definidas; construção das estruturas de dados e das interfaces de comunicação entre as camadas; o gerenciamento da comunicação dos

dados (manipulação das estruturas de dados e interfaces, da memória, das conexões, das temporizações e da execução de tarefas). Pelo fato da aplicação envolver dados multimídias, foi necessário estudar as formas de captura, compressão, armazenamento e recuperação de imagem e vídeo, para o ambiente de implementação escolhido. Similarmente, foi feito o mesmo estudo para a parte referente a áudio. A partir deste estudo, foram desenvolvidas rotinas para manipulação e comunicação desses dados pela rede.

Arquitetura de Implementação

Uma arquitetura de implementação de alto desempenho foi utilizada visando a implementação dos mecanismos de sincronização. A definição desta arquitetura (Figura 4) baseou-se na arquitetura desenvolvida pelo GTA, denominada GTA/MM. Ela serve como base para o desenvolvimento de sistemas de comunicação multimídias, com garantias de qualidade de serviço e transporte de alta velocidade [Albu95, NuDu96], e define três módulos especializados:

- um módulo de escalonamento de tarefas (ET) baseado em prioridades, que elimina testes de filas desnecessários e possui funções para inserir, executar e retirar tarefas relativas a fluxos de tempo real e tarefas de alta prioridade, como o estouro de temporizadores referentes a determinados controles de sincronização.
- um módulo de gerenciamento de memória (GM) otimizado que realiza um processamento diferenciado para alocação e liberação de *buffers* de tamanho fixo e variável e separa a memória para transmissão e recepção em regiões independentes, a fim de reduzir a fragmentação ao mínimo possível.
- um módulo de gerenciamento de temporizações (GT) otimizado e flexível que elimina testes desnecessários a cada interrupção do relógio de tempo real e oferece funcionalidades requeridas pelos mecanismos de sincronização utilizados, como inicialização, criação, interrupção, estouro, reinicialização e finalização de temporizadores.

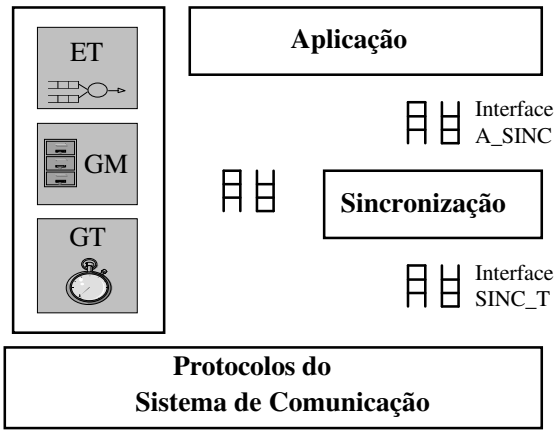

Figura 4. Arquitetura de implementação para comunicação sincronizada.

Quanto ao protocolo do sistema de comunicação, a implementação inicial foi realizada utilizando protocolos da família TCP/IP. O acesso a estes protocolos é feito através de interfaces específicas, denominadas *sockets*. O ambiente de implementação utiliza estações de trabalho *Sparc Sun*, interconectadas por rede local tipo *Ethernet*. Atualmente, está sendo realizada a migração para um sub-sistema de comunicação que utiliza a tecnologia *Asynchronous Transfer Mode* (ATM). A implementação está sendo desenvolvida na linguagem de programação C, devido a sua portabilidade e eficiência. Para a captura de vídeo, foi utilizada a *SunVideo Card*, que captura vídeo colorido à 30 quadros/s e realiza compressão por *hardware*. Foram utilizadas as funções da *XIL Imaging Library* [Sun94], para o

desenvolvimento de sistemas que utilizam a *SunVideo*. Para a interface gráfica com o usuário, está sendo utilizado o *toolkit Tcl/Tk*.

Os ítens a seguir descrevem a implementação das estruturas e de alguns procedimentos envolvidos no mecanismo de sincronização considerado neste trabalho.

Armazenamento da OCPN

Conforme explicado anteriormente, as OCPNs referentes a cada apresentação multimídia ficam armazenadas na base de dados. Desta forma, todas as informações relativas a cada OCPN, necessárias para a geração das XOCPNs transmissora e receptora, são armazenadas em um arquivo. Através de uma tabela de correspondência, é possível identificar o arquivo que contém os dados da OCPN correspondente à apresentação multimídia selecionada pelo usuário. O referido arquivo contém uma lista de todos os lugares e transições da OCPN, com as respectivas informações de cada um. As informações são as especificadas na tupla referente à OCPN. Através de uma outra tabela, é possível associar o arquivo identificado com a localização, na base de dados, de cada recurso especificado pela OCPN, bem como as características das mídias correspondentes (e.g., tamanho, número de quadros, resolução, dimensão, formato etc).

Estruturas dos Lugares e Transições da XOCPN

Na implementação da XOCPN, foram criadas duas estruturas de tamanho fixo, uma para representar os lugares, denominada **Estrutura_Lugar**, contendo campos correspondentes

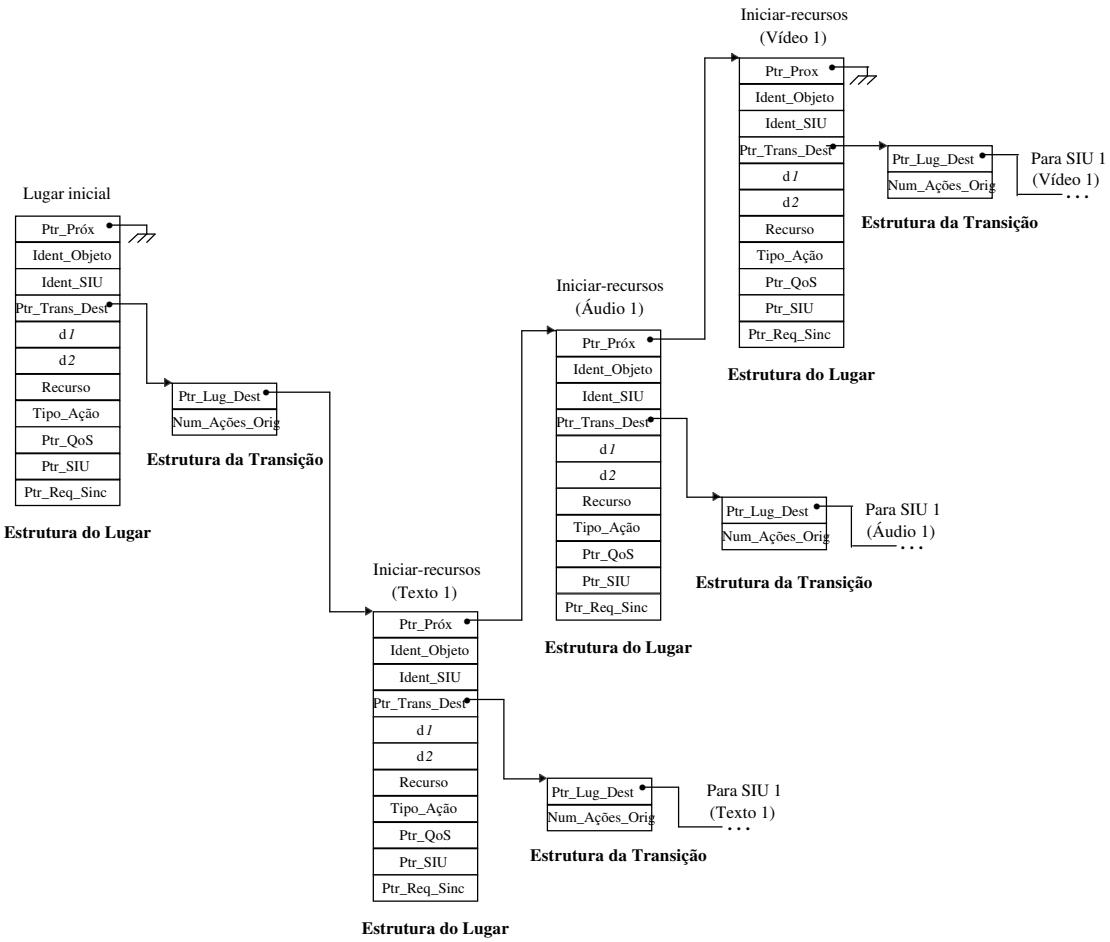

Figura 5: Estruturas dos lugares e transições da XOCPN.

aos elementos da tupla, e outra para as transições, denominada **Estrutura_Transição**. Um dos campos da **Estrutura_Transição**, *Ptr_Lug_Dest*, aponta para os lugares de saída da transição. Como este campo assume apenas um valor, foi acrescentado o campo *Ptr_Próx* na **Estrutura_Lugar** para encadear todos os lugares de saída de uma mesma transição (lugares concorrentes). O outro campo da **Estrutura_Transição**, *Num_Ações_Orig*, contém o número de ações origem (lugares) que devem ser executadas antes do seu disparo. Este campo é decrementado ao término da execução de cada lugar de origem e a transição dispara quando este campo chega a zero. A Figura 5 apresenta estas estruturas encadeadas, tomando-se como base a XOCPN da Figura 3 (b).

Planejamento da Transmissão dos Sub-objetos

O planejamento da transmissão dos sub-objetos consiste basicamente em determinar os valores dos campos d_1 e d_2, da **Estrutura_Lugar**, dos lugares da XOCPN transmissora. A realização deste planejamento depende de diversas informações à respeito do ambiente de implementação e dos recursos que podem ser garantidos. Dependendo do ambiente utilizado, principalmente do sub-sistema de comunicação, muitas destas informações não são simples de mensurar e a reserva dos recursos necessários nem sempre é possível. Alguns exemplos destas informações/recursos são quantidade de canais disponíveis, capacidade de cada canal (vazão, atraso máximo e mínimo fim-a-fim, variação de atraso, taxa de perda de pacotes, taxa de erro binária etc [SHGC92]), tempo máximo e mínimo de recuperação de cada informação da base de dados etc.

Devido às limitações do sub-sistema de comunicação utilizado, que não permite reservas de recursos e não garante os parâmetros de qualidade de serviço requeridos, optou-se por utilizar o método para planejamento da recuperação de dados proposto por Little e Ghafoor [LiGh91, LiGh92], de forma a viabilizar a transmissão de dados com restrições temporais, através de um canal com capacidade limitada. O planejamento é calculado a partir dos algoritmos extraídos do método e das informações à respeito da capacidade do canal, do atraso de propagação dos pacotes e da distribuição cumulativa dos atrasos. Os cálculos do planejamento da recuperação de dados armazenados e sua aplicação ao planejamento da transmissão de dados utilizando a XOCPN pode ser visto com mais detalhes em [LiDu96].

Geração da XOCPN

As XOCPNs transmissora e receptora são geradas a partir do arquivo de dados relativos à OCPN e das características das mídias correspondentes. A geração das XOCPNs ocorre durante o procedimento inicial do sistema. O resultado da geração é a construção, na memória, das estruturas dos lugares e transições da XOCPN de forma encadeada e o correto preenchimento dos campos das estruturas. Após a geração de cada XOCPN o ponteiro para a **Estrutura_Lugar** referente ao lugar inicial é armazenado em uma variável global, denominada *Ptr_Lugar_Inicial*, para que possa ser usado no início da execução da respectiva XOCPN.

Execução da XOCPN

Com o objetivo de facilitar a compreensão dos módulos que foram desenvolvidos para prover as funcionalidades do serviço de sincronização baseado no modelo XOCPN, foram criados dois diagramas das rotinas envolvidas na execução, respectivamente, da XOCPN transmissora e da receptora. Estes diagramas servem como base para a implementação dos mecanismos e estratégias de sincronização propostos neste trabalho. O diagrama relativo à XOCPN receptora é apresentado na Figura 6. Foi escolhido o sistema local como referência para as descrições das rotinas envolvidas na execução da XOCPN, pelo fato deste possuir o

maior número de funcionalidades relativas à sincronização. A execução da XOCPN do sistema remoto é similar a do sistema local, no entanto, o que difere são as funcionalidades dos lugares de cada XOCPN. As rotinas referentes à XOCPN receptora estão relacionadas com a apresentação síncrona dos sub-objetos e as da XOCPN transmissora estão relacionadas com a recuperação e transmissão planejada dos sub-objetos.

Inicialmente, o serviço é solicitado na entidade local com a seleção da apresentação multimídia, que possui a respectiva OCPN armazenada na base de dados remota. A partir desta seleção, os dados da OCPN são identificados e lidos. Com isto, as XOCPNs transmissora e receptora são geradas, construindo-se as estruturas de dados definidas para os lugares e transições da XOCPN, levando em consideração, para a XOCPN transmissora, o planejamento da transmissão dos dados. Por fim, cada XOCPN gerada é executada.

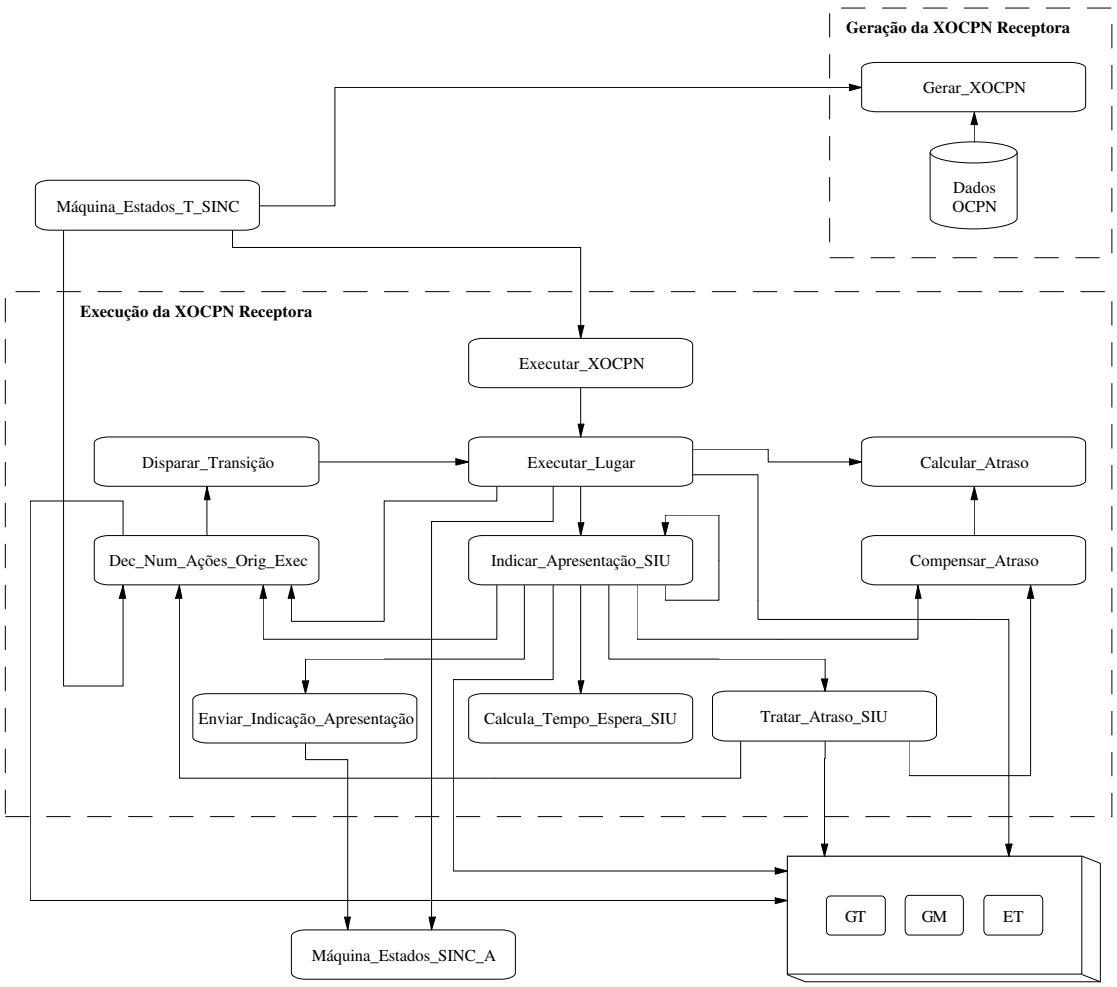

Figura 6: Diagrama das rotinas envolvidas na execução da XOCPN receptora.

A execução da XOCPN receptora inicia logo após a entidade de Sincronização local enviar a PDU IREADY para a entidade de Sincronização remota. A execução da XOCPN transmissora inicia logo após o recebimento da PDU IREADY pela entidade de Sincronização remota. A execução da XOCPN consiste basicamente em executar os lugares (partindo do lugar inicial) e disparar as transições, de acordo com as regras da rede de Petri. Assim, duas principais rotinas que fazem parte da execução das XOCPNs são **Executar_Lugar** e **Disparar_Transição**, que possuem como parâmetros *Ptr_Lugar* (ponteiro para a **Estrutura_Lugar** correspondente ao lugar a ser executado) e *Ptr_Transição* (ponteiro para a

Estrutura_Transição correspondente a transição a ser disparada) respectivamente. A rotina que inicia a execução da XOCPN, partindo do ponteiro para o lugar inicial, chama-se **Executar_XOCPN**. Como pode ser visto na Figura 6, a rotina que dispara uma transição é chamada posteriormente, no instante de disparo da mesma, e a rotina que executa os outros lugares da XOCPN será chamada, em seus respectivos instantes de tempo, pela rotina de disparo de transição, até que seja atingido o lugar final, quando então é finalizada a execução da XOCPN. A seguir, o pseudo-código da rotina **Executar_XOCPN**.

Executar_XOCPN (*Ptr_Lugar_Inicial***)**
Inicio
Executar_Lugar (*Ptr_Lugar_Inicial***)**
Fim

Execução dos Lugares da XOCPN

Quando a rotina **Executar_Lugar (***Ptr_Lugar***)** é chamada, primeiramente é feita a identificação do lugar, através do campo *Tipo_Ação* da **Estrutura_Lugar** apontada por *Ptr_Lugar*, e depois a ação correspondente ao lugar é executada. Na realidade, quando a ação correspondente for `TRANSMITIR_SIU` ou `APRESENTAR_SIU` (o que acontece na maioria dos casos), ela é inserida no escalonador de tarefas (na fila de prioridades de tempo real), pois estas ações nem sempre podem ser executadas imediatamente, porque dependem da disponibilidade, da temporização de espera e do tratamento do atraso do sub-objeto. No caso da ação ser `INICIAR_XOCPN`, `INICIAR_RECURSOS`, `LIBERAR_RECURSOS` ou `FINALIZAR_XOCPN`, ela é executada imediatamente, pois estas ações acontecem no início ou no fim da transmissão ou apresentação do objeto, ou da própria execução da XOCPN. Caso a ação seja `SINCRONIZAR_INTERMÍDIAS`, ela também é executada imediatamente, pois a própria natureza do lugar exige uma ação de correção (sincronização) imediata. Após o término da execução do lugar, a rotina **Dec_Num_Acoes_Orig_Exec (***Ptr_Lugar***)** é executada. Esta rotina decrementa o campo *Num_Ações_Orig* da transição destino (apontada pelo campo *Ptr_Trans_Dest* de *Ptr_Lugar*). Se este campo chegar a zero após o decremento, a rotina **Disparar_Transição** é enviada para a fila do escalonador de tarefas, tendo como parâmetro o campo *Ptr_Trans_Dest*. A seguir, o pseudo-código simplificado da rotina **Executar_Lugar**, para a XOCPN receptora, e o pseudo-código da rotina **Dec_Num_Acoes_Orig_Exec**:

Executar_Lugar (*Ptr_Lugar***)**
Inicio
caso campo *Tipo_Açao* **de** *Ptr_Lugar* **seja**
INICIAR_XOCPN**:**
<Inicializa as variáveis>
INICIAR_RECURSOS**:**
< Inicia os recursos>
APRESENTAR_SIU**:**
diferenca = Calcular_Atraso (campo *d1* **de** *Ptr_Lugar***)**
se (diferenca < 0)
Criar_Temporizador (*Ptr_Lugar***, Indicar_Apresentacao_SIU, (|diferenca|))**
senao
Enviar_Rotina_Tempo_Real_Fila_ET (Indicar_Apresentacao_SIU, *Ptr_Lugar***)**
fim-se
SINCRONIZAR_INTERMIDIAS**:**
<Aplica estratégia de sincronização intermídias>

```
    LIBERAR_RECURSOS:
      <Realiza procedimentos para liberação dos recursos>
    FINALIZAR_XOCPN:
      <Realiza procedimentos para finalização da execução da XOCPN>
  fim-caso
  se Tipo_Acao ≠ APRESENTAR_SIU e Tipo_Acao ≠ FINALIZAR_XOCPN
    Dec_Num_Acoes_Orig_Exec (Ptr_Lugar)
  fim-se
fim


Dec_Num_Acoes_Orig_Exec (Ptr_Lugar)
Inicio
   Ptr_Transicao_Dest = campo Ptr_Trans_Dest de Ptr_Lugar
   decrementa campo Num_Acoes_Orig de Ptr_Transicao_Dest
   se campo Num_Acoes_Orig de Ptr_Transicao_Dest = 0
      Enviar_Rotina_Tempo_Real_Fila_ET (Disparar_Transicao, Ptr_Transicao_Dest)
   fim-se
fim
```

A rotina **Calcular_Atraso** retorna a diferença entre o tempo atual e o tempo em que o sub-objeto deveria ser apresentado. Se a diferença for positiva, significa que a execução do lugar correspondente está atrasada e, conseqüentemente, a rotina de indicação de apresentação do sub-objeto (**Indicar_Apresentação_SIU (***Ptr_Lugar***)**) é enviada para o escalonador de tarefas imediatamente (quando executada, esta rotina chamará as rotinas de tratamento e compensação do atraso de sub-objetos). Caso a diferença seja negativa, significa que ainda resta um tempo para que o lugar seja executado. Neste caso, é criado um temporizador, com tempo igual ao módulo da diferença calculada, para que a rotina seja enviada para o escalonador de tarefas no momento previsto. Desta forma, procura-se manter o tempo total da apresentação como o estipulado pela XOCPN.

Armazenamento dos Sub-objetos Recebidos

Quando a entidade de Sincronização no destino recebe uma PDU SIUDT, é feita uma análise da validade do sub-objeto, ou seja, se o mesmo chegou a tempo para sua apresentação. Se o sub-objeto chegou atrasado, ele é descartado. Caso contrário, é realizada uma procura na XOCPN para identificar a **Estrutura_Lugar** correspondente ao sub-objeto. Depois disto, é alocada uma região de memória que será apontada pelo campo *Ptr_SIU* da **Estrutura_Lugar** identificada. Utilizando as informações contidas na PDU SIUDT, a região de memória alocada é preenchida com os dados do sub-objeto. Esta forma de armazenar os dados relativos aos sub-objetos facilita sobremaneira o manuseio com os mesmos, pois elimina o uso de filas (provavelmente uma para cada mídia) e a manipulação com as mesmas. Para verificar se o sub-objeto relativo a um determinado lugar já chegou, basta apenas testar se o campo *Ptr_SIU* da **Estrutura_Lugar** é diferente de nulo. Para descartá-lo, basta apenas liberar a região de memória onde ele está armazenado. Para apresentá-lo, basta apenas passar o ponteiro dos dados relativos ao sub-objeto, evitando mais uma cópia de memória.

Disparo das Transições da XOCPN

Quando uma transição da XOCPN possui lugares de saída concorrentes, estes devem ser executados concorrentemente após o disparo da referida transição. Como o desenvolvimento deste sistema é orientado para realizar tarefas seqüencialmente, de acordo

com o ordenamento da fila do escalonador de tarefas, após o disparo de uma transição, o lugar de saída apontado pela mesma é executado através da rotina **Executar_Lugar**. Como o campo *Ptr_Próx* da **Estrutura_Lugar** deste lugar de saída realiza o encadeamento com os outros lugares concorrentes, é possível determinar facilmente o ponteiro para estes lugares e, com isto, executá-los. A rotina **Executar_Lugar**, dependendo do lugar, enviará a ação correspondente para o escalonador de tarefas ou a executará imediatamente, como pode ser visto em seu pseudo-código. Obviamente que as ações concorrentes não estarão sendo executadas simultaneamente, no entanto, quando a velocidade de processamento das ações relativas ao lugares é suficientemente grande para que o atraso entre as apresentações das mídias envolvidas não ultrapasse os limites máximos da percepção audiovisual humana, a qualidade da apresentação não é degradada. A seguir, o pseudo-código da rotina **Disparar_Transição**:

```
Disparar_Transicao (Ptr_Transicao)
Inicio
   Ptr_Lugar = campo Ptr_Lug_Dest de Ptr_Transicao
   enquanto Ptr_Lugar ≠ NULO fazer
      Executar_Lugar (Ptr_Lugar)
      Ptr_Lugar = campo Ptr_Prox de Ptr_lugar
   fim-enquanto
fim
```

A implementação deste trabalho encontra-se em fase avançada de desenvolvimento, com a elaboração de um protótipo contendo aproximadamente 11000 linhas de código.

7. CONCLUSÕES

Este artigo apresentou a implementação de um serviço de sincronização multimídia para comunicação de dados pré-orquestrados. O mecanismo de sincronização utilizado é baseado no modelo XOCPN e considera as relações temporais sintéticas entre objetos multimídias armazenados. Foram apresentadas as definições das primitivas de serviço e das unidades de dados do protocolo de sincronização. Foi descrita a arquitetura de implementação utilizada, incluindo seus módulos de gerenciamento otimizado de memória, gerenciamento de temporizações e escalonamento de tarefas com prioridades. Foi apresentada a implementação das estruturas de lugares e transições da XOCPN. Foi apresentado um diagrama contendo as rotinas de execução da XOCPN receptora, bem como o pseudo-código de algumas rotinas envolvidas na execução das XOCPNs.

O modelo XOCPN é aplicado somente para fontes de dados armazenados. Isto ocorre porque na construção da XOCPN transmissora é necessário calcular o planejamento da recuperação/transmissão e o resultado deste planejamento, dependendo da capacidade do canal, exige uma recuperação de dados a uma taxa superior a qual foram capturados. Além disto, para o cálculo do planejamento e da construção da XOCPN transmissora e receptora, é necessário o conhecimento prévio das características das mídias envolvidas e das relações existentes entre os objetos. A divisão dos objetos multimídias referentes a dados isócronos em sub-objetos, no modelo XOCPN, aumenta o nível de granularidade dos objetos, proporcionando um melhor controle da sincronização intramídia e, principalmente, da sincronização intermídias.

As estruturas de dados baseadas em ponteiros tornou fácil o acesso aos lugares e transições da XOCPN, facilitando inclusive a implementação da estratégia de sincronização intermídias adotada. Alto desempenho é uma das mais importantes características da arquitetura de implementação utilizada, devido principalmente à redução de cópias de memória.

REFERÊNCIAS

[Albu95] Albuquerque, C.V.N., "Arquitetura de Implementação de Alto Desempenho para Sistemas de Comunicação Multimídia", *Tese de Mestrado*, PEE-COPPE/UFRJ, abril de 1995.

[BlSt96] Blakowski, G. e Steinmetz, R., "A Media Synchronization Survey: Reference Model, Specification, and Case Studies", *IEEE J. Select. Areas in Commun.*, vol. 14, nº 1, pp. 5-35, janeiro de 1996.

[CCO96] Courtiat, J.-P.; Carmo, L.F.R.C. e Oliveira, R.C., "A General-Purpose Multimedia Synchronization Mechanism Based on Causal Relations", *IEEE J. Select. Areas in Commun.*, vol. 14, nº 1, pp. 185-95, janeiro de 1996.

[CCS93] Carmo, L.F.R.C.; Courtiat, J.-P. e de Saqui-Sannes, P., "Towards Multimedia Commuincation Services", *Proc. of the Fourth Workshop on Future Trends of Distributed Computing Systems*, Lisboa, Portugal, 22 a 24 de setembro de 1993.

[EFI94] Ehley, L.; Furht, B. e Ilyas, M., "Evaluation of Multimedia Synchronization Techniques", *Proc. of the International Conference on Multimedia Computing and Systems*, pp. 514-19, 14 a 19 de maio de 1994.

[EPD94] Escobar, J.; Partridge, C. e Deutsch, D., "Flow Synchronization Protocol", *IEEE/ACM Trans. on Networking*, vol. 2, nº 2, pp. 111-21, abril de 1994.

[LiDu96] De Lima, R.M. e Duarte, O.C.M.B., "Comunicação Sincronizada de Dados Multimídias Pré-orquestrados", a ser publicado no *XXIII Seminário Integrado de Software e Hardware (SEMISH96)*, Recife, PE, agosto de 1996.

[LiGh90a] Little, T.D.C. e Ghafoor, A., "Synchronization and Storage Models for Multimedia Objects", *IEEE J. Select. Areas in Commun.*, vol. 8, nº 3, pp. 413-27, abril de 1990.

[LiGh90b] Little, T.D.C. e Ghafoor, A., "Network Considerations for Distributed Multimedia Object Composition and Communication", *IEEE Network*, vol. 4, nº 6, pp. 32-49, novembro de 1990.

[LiGh91] Little, T.D.C. e Ghafoor, A., "Multimedia Synchronization Protocols for Broadband Integrated Services", *IEEE J. Select. Areas in Commun.*, vol. 9, nº 9, pp. 1368-82, dezembro de 1991.

[LiGh92] Little, T.D.C. e Ghafoor, A., "Scheduling of Bandwidth-Constrained Multimedia Traffic", *Comput. Commun.*, vol. 15, nº 6, pp. 381-87, julho de 1992.

[Lima96] De Lima, R.M., "Um Serviço de Sincronização Multimídia de DDados Pré-orquestrados", *Tese de Mestrado*, PEE-COPPE/UFRJ, junho de 1996.

[Nico90] Nicolaou, C., "An Architecture for Real-Time Multimedia Communication Systems", *IEEE J. Select. Areas in Commun.*, vol. 8, nº 3, pp. 391-400, abril de 1990.

[NuDu96] Nunes, M.D. e Duarte, O.C.M.B., "MCF: Um Protocolo Multidestinatário Configurável com Gerenciamento de Acesso por Fichas de Permissão", *XIV Simpósio Brasileiro de Redes de Computadores*, Fortaleza, CE, pp. 159-77, maio de 1996.

[QWG93] Qazi, N.U.; Woo, M. e Ghafoor, A., "A Synchronization and Communication Model for Distributed Multimedia Objects", *Proc. ACM Multimedia '93,* Anaheim, CA, pp. 147-56, agosto de 1993.

[RaRa93] Ramanathan, S. e 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.

[RPT96] Raghavan, S.V.; Prabhakaran, B. e Tripathi, S.K., "Synchronization Representation and Traffic Source Modeling in Orquestrated Presentation", *IEEE J. Select. Areas in Commun.*, vol. 14, nº 1, pp. 104-13, janeiro de 1996.

[SHGC92] Shepherd, D.; Hutchinson, D.; Garcia, F. e Coulson, G., "Protocol Support for Distributed Multimedia Applications", *Comput. Commun.*, vol. 15, nº 6, pp. 359-66, julho de 1992.

[Stei90] Steinmetz, R., "Synchronization Properties in Multimedia Systems", *IEEE J. Select. Areas Commun.*, vol. 8, nº 3, pp. 401-12, abril de 1990.

[Sun94] Sun Microsystems Inc., "XIL Programmer's Guide", 1994.

[WoGh94] Woo, M. e Ghafoor, A., "Multichannel Scheduling for Communication of Pre-orchestrated Multimedia Information (Homogeneous Channels Case)", *Proc. IEEE INFOCOM '94*, pp. 920-27, 1994.

[WQG94] Woo, M.; Qazi, N.U. e Ghafoor, A., "A Synchronization Framework for Communication of Pre-orchestrated Multimedia Information", *IEEE Network*, vol. 8, nº 1, pp. 52-61, janeiro de 1994.

[YaHu96] Yang, C.-C e Huang, J.-H., "A Multimedia Synchronization Model and Its Implememtation in Transport Protocols", *IEEE J. Select. Areas in Commun.*, vol. 14, nº 1, pp. 212-25, janeiro de 1996.