

Estruturação de Documentos Multimídia através de Relacionamentos Semânticos para Apresentação Adaptativa em um Ambiente de Ensino à Distância.

*Eduardo Carneiro da Cunha, Luiz Fernando Rust da Costa Carmo
& Luci Pirmez*

E-mail: {educ, rust, luci}@nce.ufrj.br

Núcleo de Computação Eletrônica - UFRJ
P.O.Box 2324, 20001-970
Rio de Janeiro - RJ - Brasil

Resumo: *Uma apresentação multimídia distribuída envolve a recuperação de objetos a partir de um ou mais servidores e a apresentação destes nos sistemas clientes. A recuperação dos objetos multimídia é influenciada por fatores como a especificação dos relacionamentos entre os objetos, a largura de banda e o atraso máximo oferecidos pela rede. O projeto ServiMídia[Carm97] busca desenvolver um ambiente propício para a criação e recuperação adaptativa de documentos multimídia em redes corporativas. O objetivo final consiste em possibilitar uma apresentação multimídia coordenada, garantindo a preservação da coerência entre os diferentes fluxos de mídias, mesmo quando o processamento é confrontado com uma insuficiência temporária de recursos oferecidos pela rede de comunicação. Este artigo propõe uma estratégia de composição de documentos multimídia que trata os requisitos de QoS da apresentação do ponto de vista do autor e que pode ajudar no gerenciamento de variações nos recursos disponíveis na rede.*

1. Introdução

Sistemas multimídia são sistemas computacionais que manipulam de forma integrada vários tipos de mídias de representação de informação. Frequentemente, sistemas multimídia são distribuídos, isto é, seus componentes estão localizados em diferentes nós de processamento numa rede local ou de longa distância. Qualidade de serviço neste contexto pode ser definida intuitivamente como uma medida de quão satisfeito está o usuário com respeito a um serviço prestado por um sistema multimídia distribuído (SMD). Embora a noção de QoS seja intuitiva, uma série de parâmetros mensuráveis pode ser estabelecida para se definir tal conceito objetivamente. Estes parâmetros se dividem em dois níveis: usuário e sistema.

A maioria das arquiteturas de SMD desenvolvidas até o momento trata a questão da qualidade de serviço do ponto de vista do sistema computacional, ou do provedor de acesso, e empregam políticas efetivas de monitoramento e gerência de recursos para prover suporte à qualidade de serviço [Aurr98]. Apesar de haver uma divisão em dois níveis, é importante ressaltar que parâmetros em nível de sistema estão relacionados aos parâmetros em nível de usuário, e vice-versa. Por outro lado, estas arquiteturas falham por não abstraírem a noção de QoS até o nível de usuário, fazendo com que a especificação da QoS seja feita forçosamente através dos parâmetros do nível do sistema.

O trabalho apresentado neste artigo é resultado de um projeto de pesquisa em ensino à distância e trata, basicamente, com a autoria e o armazenamento de documentos multimídia e com a infraestrutura de rede para a recuperação remota dos mesmos. A questão básica é como acomodar clientes com diferenças substanciais de disponibilidade de recursos de comunicação em um mesmo ambiente cliente/servidor. Um mesmo documento, recuperado por clientes distintos, pode ser interpretado de

forma completamente diferente de um cliente para outro, de acordo com a qualidade da apresentação que é percebida pelo usuário. A qualidade da apresentação, por sua vez, está diretamente relacionada à disponibilidade de recursos encontrada na rede ao longo do caminho que atinge o cliente, já que alguns dos fluxos multimídia podem ser perdidos ou parcialmente danificados durante o processo de recuperação do documento.

Neste trabalho, nós investigamos uma estratégia para a composição, armazenamento e recuperação de documentos multimídia que nos permite, a partir de uma única especificação (documento multimídia), gerar diferentes formatos de apresentações nas estações clientes (fig.1). Estes formatos de apresentações são adaptados da especificação primária de acordo com a disponibilidade dos recursos de comunicação verificadas dinamicamente nas rotas para os respectivos clientes. Nós consideramos a questão de adaptação como um problema de transformação: de acordo com a disponibilidade de recursos, a especificação original é transformada resultando em um formato (visão) personalizado da apresentação. O ponto chave nesta estratégia é que cada formato gerado deve, necessariamente, conservar as propriedades semânticas das mídias originalmente especificadas pelo autor do documento. Isto é, o documento multimídia deve manter a sua coerência e o seu valor semântico mesmo quando sofrer alguma degradação com relação ao documento (especificação) primário.

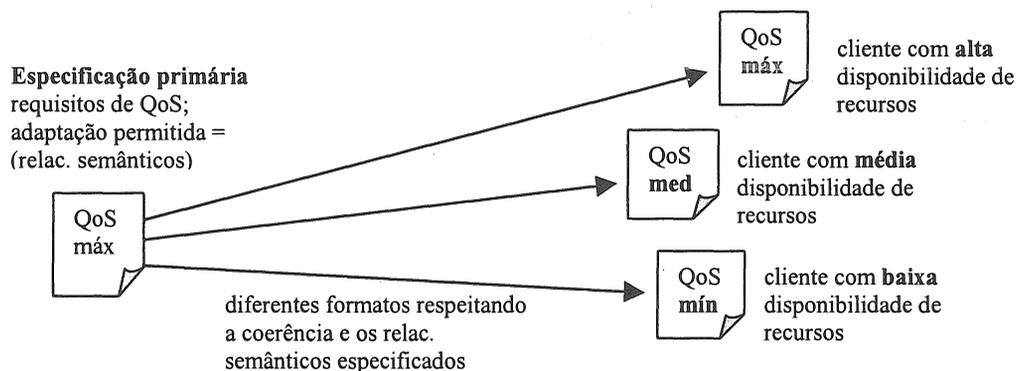


Figura 1 – geração de formatos de apresentação

Este artigo está organizado da seguinte forma: a seção 2 apresenta a estratégia de autoria desenvolvida neste projeto para lidar com as características introduzidas acima. A seção 3 mostra como esta estratégia pode ser implementada utilizando uma linguagem de descrição de apresentações multimídia. A seção 4 apresenta a ferramenta de autoria desenvolvida, e a seção 5 mostra um exemplo de aplicação. Na seção 6 discutimos alguns trabalhos relacionados enquanto na seção 7 relacionamos algumas conclusões deste trabalho.

2. Modelo de autoria multimídia

A especificação de documentos multimídia é realizada pelos sistemas de autoria atuais com base em três aspectos fundamentais: estruturação lógica da apresentação, estabelecimento de relações temporais e definição espacial entre os objetos multimídia que a compõem. A estruturação lógica se preocupa em estabelecer mecanismos de abstração, objetivando a obtenção de uma visão ampla e estruturada da apresentação. A especificação do comportamento temporal, por sua vez, implica na definição de relações de sincronização temporal (instantes iniciais e durações) entre os objetos

envolvidos. A sincronização espacial, por fim, prima por ajustar o posicionamento destes mesmos objetos de acordo com os dispositivos de saída.

É importante destacar que a preocupação, aqui mencionada, com a manutenção da coerência de uma apresentação, associada ao controle de degradação da mesma, não é explorada pelas arquiteturas de SMD atuais. Em [Sant98], é discutido uma extensão às classes de MHEG-5 [Mheg96] com o objetivo de manter informações semânticas sobre os objetos multimídia armazenados em um servidor de objetos MHEG-5. Porém, estas informações são utilizadas apenas durante a fase de edição das apresentações com o propósito de se realizar uma busca (consulta) no servidor de objetos baseada no conteúdo das mídias armazenadas.

Para permitir a especificação dos relacionamentos semânticos entre os componentes da apresentação, utilizamos uma versão modificada do modelo de sincronização causal apresentado em [Cour96]. Como veremos a seguir, a combinação do modelo de sincronização causal com o modelo de sincronização temporal baseado em linha de tempo vem suprir as principais limitações deste último. Com o objetivo de minimizar o antagonismo existente entre poder de expressão e facilidade de uso, propõe-se, neste trabalho, uma estratégia de autoria que agrega:

- mecanismos de estruturação lógica das apresentações;
- um modelo de sincronização espacial baseado na definição de regiões de playback;
- um modelo de sincronização temporal baseado em linhas de tempo; associado a
- um modelo de sincronização causal baseado em relacionamentos condicionais entre os objetos envolvidos.

2.1 Estruturação lógica

A complexidade das apresentações multimídia está diretamente relacionada com o número de objetos multimídia envolvidos e, conseqüentemente, com os diversos relacionamentos estabelecidos entre eles. Esta é a razão fundamental pela qual deve ser adotada uma forma de estruturação lógica da apresentação. Na estratégia proposta neste trabalho, utilizamos uma estrutura lógica para a apresentação que está baseada no conceito de grupos de clips, onde os clips são os objetos das mídias que compõem o documento. Estes grupos são representados em uma árvore hierárquica (fig.2) semelhante a uma árvore de diretórios, onde cada grupo pode conter clips ou outros grupos. Os grupos de clips podem ser de dois tipos: *grupo paralelo*, onde os clips devem ser apresentados em paralelo; e *grupo sequencial*, onde os clips devem ser apresentados em seqüência. Na realidade, o próprio documento é considerado um grupo sequencial, pois os elementos no nível logo abaixo ao dele devem ser apresentados em seqüência.

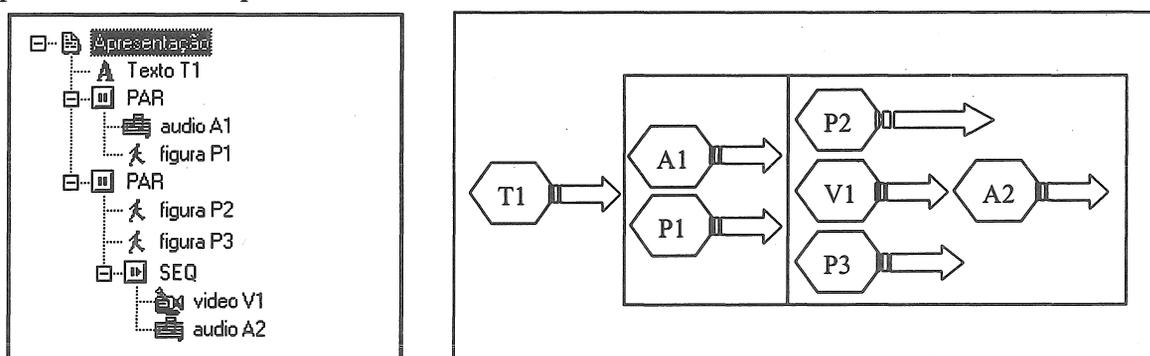


Figura 2 - estrutura lógica da apresentação

2.2 Layout da apresentação

A sincronização espacial permite ao autor organizar o posicionamento dos objetos das mídias visíveis de uma apresentação. Isto é, o autor pode configurar o layout da apresentação através da definição de regiões na área de apresentação onde os clips deverão ser tocados. Se a apresentação só apresenta um clip de cada vez, não é necessária a criação de um layout para a mesma. Na maioria das vezes cada clip é automaticamente apresentado na janela principal da ferramenta de apresentação (player), e a janela se ajusta automaticamente para cada clip novo. Se for desejado que o tamanho da área de playback permaneça estável, ou se a apresentação tocar vários clips ao mesmo tempo, é interessante definir áreas de playback (regiões) dentro da janela principal.

2.3 Relacionamentos temporais

Um documento multimídia consiste de diferentes tipos de objetos de mídias que devem ser apresentados em diferentes instantes de tempo por diferentes durações. Os instantes de início e as durações das apresentações destes objetos podem ser especificados de uma maneira rígida ou flexível. No caso de uma especificação rígida, estes instantes e durações são fixos. Em uma especificação flexível, é permitido que estes variem desde que eles respeitem certos relacionamentos que foram especificados.

Uma especificação temporal flexível é obtida, neste trabalho, com o estabelecimento de margens de tolerância para o início da apresentação dos objetos, isto é, estabelecendo um intervalo temporal que indique que a apresentação de um objeto poderá ser iniciada em qualquer instante dentro deste intervalo. Desta forma, os aspectos temporais são expressos por uma margem de tolerância para o início da apresentação mais a duração desejada para o respectivo objeto.

A vantagem do uso de especificações temporais flexíveis é a de facilitar o uso de técnicas de relaxamento ou aceleração das apresentações com propósitos de sincronização, auxiliando na derivação de um escalonamento da apresentação, como discutido em [Cand98]. A figura 3, por exemplo, mostra dois objetos posicionados nas suas respectivas linhas de tempo. Esta figura ilustra a seguinte especificação:

- comece a tocar o áudio A1 em algum instante entre 10:00 e 10:05; e
- comece a apresentar a figura P1 em algum instante entre 9:28 e 9:32.

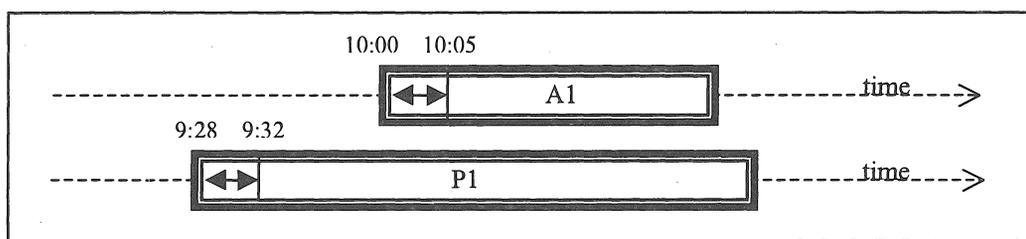


Figura 3 - especificação temporal flexível

2.4 Relacionamentos causais

Como brevemente introduzido, o problema que guiou o desenvolvimento deste trabalho foi a diferença verificada na disponibilidade de recursos de comunicação durante a fase de apresentação de um documento multimídia distribuído em um ambiente de ensino à distância. A maneira escolhida para

se lidar com este problema foi permitir ao autor do documento multimídia especificar a importância de cada objeto que ele inserir no documento e delinear as condições que devem ser respeitadas com o objetivo de preservar a consistência do documento. Por exemplo, se o autor é capaz de informar que uma dada informação é essencial ou não, e informar os seus requisitos de QoS, então, esta informação pode ser usada para realizar uma degradação coerente da QoS no caso de uma sobrecarga em um dos canais de comunicação usados.

O ambiente ServiMídia [Carm97] garante a apresentação dos objetos essenciais e permite o estabelecimento de relacionamentos causais entre os objetos qualitativos. Estes relacionamentos são especificados através da definição de links condicionais que interconectam os objetos qualitativos formando uma malha de causalidade que descreve a consistência desejada para o documento, ao mesmo tempo que possibilita uma adaptação coerente da apresentação. Para ver isto, considere o exemplo da seção 2.3 incrementado com relacionamentos causais:

- *comece a tocar o áudio A1 em algum instante entre 10:00 e 10:05 e comece a tocá-lo apenas se a figura P1 já foi iniciada;*
- *comece a apresentar a figura P1 em algum instante entre 9:28 e 9:32 e continue mostrando até que o áudio A1 termine.*

Nos dois casos existe um relacionamento causal entre os objetos A1 e P1. Na primeira declaração, temos uma especificação relacionada ao instante de início da apresentação de P1, isto é, a apresentação de A1 está condicionada ao fato da apresentação de P1 ter sido iniciada. Na segunda declaração, temos uma especificação relacionada ao instante de fim da apresentação de A1. Desta forma, haverá dois links, um de A1 para P1 e outro de P1 para A1. Se, por exemplo, a figura P1 não puder ser apresentada por algum motivo, a apresentação do áudio A1 será descartada, economizando os recursos da rede. Note que o descarte da informação é realizado com base nas especificações fornecidas pelo autor do documento, resultando em uma apresentação coerente ao mesmo tempo que não sobrecarrega a rede com informações que se tornaram desnecessárias (perderam o significado).

Neste trabalho, definimos um objeto Link que consiste de um LinkSource e um LinkTarget. O LinkSource, por sua vez, consiste de uma lista de SourceConditions associadas a diferentes objetos de mídia. O LinkTarget é associado a um único objeto e é formado por uma LinkAction e uma LinkDexpr. A LinkDexpr combina as SourceConditions em uma expressão de dependência para descrever os requisitos a garantir durante a apresentação do documento. Quando um link é disparado, sua LinkDexpr é verificada e quando esta se torna verdadeira a LinkAction é executada. A figura 4 representa um link com um LinkSource onde existem quatro SourceConditions associadas aos objetos N1, N2 e N3, e um LinkTarget com uma certa LinkAction associado ao objeto N4.

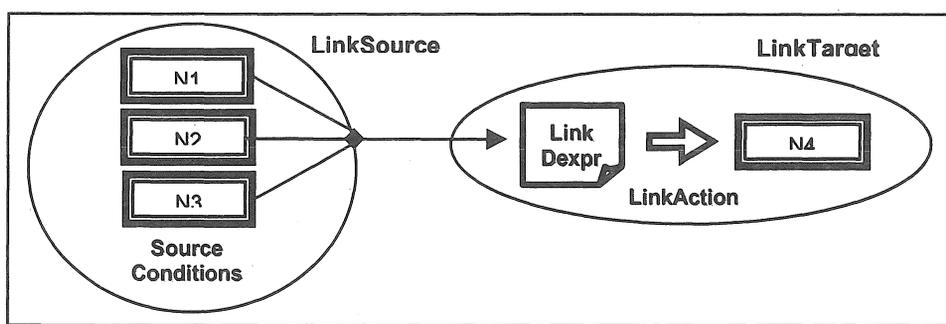


Figura 4 - representação de um link

Dois tipos de links foram definidos: *startlink* e *stoplink*, com as *LinkAction* definidas como “apresentar objeto” e “interromper objeto”, respectivamente. Note que os links são disparados uma única vez durante a apresentação do documento. Isto é, quando o instante de início da apresentação de um objeto for atingido, seu *startlink* é disparado e todas as suas condições são verificadas. Se o *startlink* foi disparado com sucesso, o *stoplink* é então disparado em seguida.

Repare que o disparo dos links é executado pela sincronização temporal. Tanto a sincronização temporal quanto a causal devem ser respeitadas. Se nenhum link for especificado para um objeto significa que ele não possui relacionamentos condicionais e sua apresentação deve respeitar apenas a sincronização temporal. Porém, ele não será considerado um objeto essencial pois estes devem ser especificados explicitamente. Lembre que mesmo não dependendo de nenhum objeto, ele pode vir a compor uma *SourceCondition* de um link para outro objeto. Por outro lado, quando um objeto é declarado explicitamente como essencial, sua apresentação é garantida pelo ambiente e ele não participa de relacionamentos causais, seja como origem ou destino de um link. Isto porque testar uma condição que deve ser garantida não faz sentido.

A tabela 1 a seguir resume os possíveis valores (estados) que podem ser testados através das *SourceConditions*, como estas podem ser combinadas através da *LinkDexpr*, a sintaxe empregada em cada caso, e um exemplo de especificação.

SourceConditions		
Estados	Descrição	Exemplo
<i>started</i>	verificado se a apresentação do objeto foi iniciada com sucesso	(P1:started)
<i>stopped</i>	verificado se a apresentação do objeto foi interrompida	
<i>concluded</i>	verificado se a apresentação do objeto chegou ao fim com sucesso	
LinkDexpr		
Descrição		Exemplo
corresponde a uma expressão Boolean que combina as <i>SourceConditions</i> descrevendo os relacionamentos condicionais a serem respeitados para a ativação do respectivo link.		(A1:concluded) and (P1:started)
Links		
Tipos	Descrição	Exemplo
<i>startlink</i>	estabelece um link condicional para o início da apresentação de um objeto qualitativo.	startlink = "(A1:concluded) and (P1:started)"
<i>stoplink</i>	estabelece um link condicional para que se interrompa a apresentação de um objeto qualitativo.	

Tabela 1 – Especificando relacionamentos causais (links)

3. Implementando a estratégia de autoria

Com relação a criação de apresentações multimídia, não há um consenso ou padrão amplamente aceito para a especificação de documentos multimídia a serem recuperados e/ou apresentados via servidores remotos. A linguagem SMIL [Smil98] introduz muitas idéias valiosas. Como veremos a seguir, estas idéias vêm de encontro à estratégia de autoria proposta neste trabalho e se encaixaram muito bem ao contexto do projeto ServiMídia [Carm97]. A seguir estão listadas algumas das vantagens de se usar SMIL:

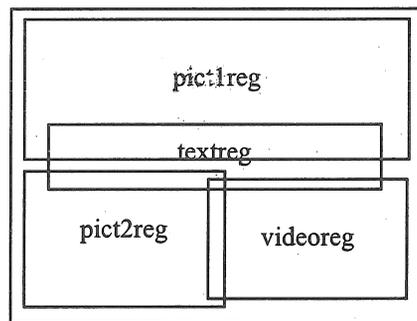
1. Usar clips em localidades diferentes: como em um arquivo SMIL cada objeto de mídia é descrito por uma URL, o autor pode compor apresentações usando clips de qualquer localidade.
2. Através do uso do elemento switch possibilita: suporte para múltiplas línguas, suporte a diferentes larguras de banda, suporte a apresentações personalizadas.
3. Controle temporal da apresentação
4. Controle espacial da apresentação

Desta forma, decidimos por adotar a linguagem SMIL como base para a especificação dos documentos criados pela ferramenta de autoria do ambiente ServiMídia. Entretanto, certas características específicas da nossa estratégia de autoria não são consideradas pela linguagem SMIL 1.0. Por exemplo, a especificação temporal flexível e os relacionamentos causais através de links não são considerados. O exemplo 1 mostra o documento multimídia estruturado na figura 2 especificado em SMIL. Nosso objetivo, nesta seção, é descrever as extensões que criamos à linguagem SMIL para que fosse possível implementar nossa estratégia de autoria.

```

<smil>
<head>
  <meta name="title" content="Multimedia Power"/>
  <meta name="copyright" content="(c)1999 Eduardo Cunha"/>
</layout>
  <root-layout background-color="maroon" width="450" height="425" />
  <region id="textreg" top="150" left="50" width="350" height="100" />
  <region id="videoreg" top="210" left="210" width="240" height="180" />
  <region id="pict1reg" top="10" left="10" width="430" height="200" />
  <region id="pict2reg" top="205" left="10" width="210" height="200" />
</layout>
</head>
<body>
  <text id="T1" src="title.rt" region="textreg" dur="20s"/>
  <par>
    <audio id="A1" src="audio1.ra" />
    
  </par>
  <par>
    
    
  <seq>
    <video id="V1" src="narration.rm" region="videoreg" />
    <audio id="A2" src="audio2.ra" />
  </seq>
</par>
</body>
</smil>

```



Exemplo 1 – exemplo de documento smil

Podemos notar por este exemplo que o documento é dividido em dois blocos principais: *head* e *body*. No *head* estão especificados os elementos que não são relacionados com temporização como os elementos *meta* e o *layout* da apresentação. No *body* estão especificados os elementos temporizados como os grupos *par* e *seq* e os clips propriamente ditos. Neste exemplo, todos os clips estão armazenados localmente, mas o atributo *src* poderia especificar uma localidade remota de onde o clip seria recuperado. Para atingir o nosso objetivo, comentaremos a seguir alguns atributos da linguagem SMIL apresentando outros atributos que criamos para implementar a estratégia apresentada nas seções anteriores. Obviamente, não será possível, explicar os vários atributos internos da linguagem SMIL.

Em SMIL 1.0, os instantes de início para qualquer clip ou grupo de clips são definidos através do atributo *begin* que define um ponto específico dentro de um eixo temporal. O início da apresentação pode ser especificado de duas formas:

- por um intervalo temporal (*delay-value*): *begin*="5s";
- ou por um evento (*event-value*), que pode ser de três tipos:
begin="id(x) (begin)"; *begin*="id(x) (45s)"; *begin*="id(x) (end)"

Para permitir uma especificação temporal flexível criamos o atributo *can-begin* que define um intervalo de tolerância para o início da apresentação de qualquer clip. Por exemplo, na figura 5, utilizando o primeiro grupo *par* do exemplo 1, especificamos que A1 pode começar dentro da tolerância de 5s e que P1 pode começar dentro da tolerância de 4s. Adicionalmente você pode especificar um atributo *end*, ou um atributo *dur*, sozinho ou combinado com um *begin*.

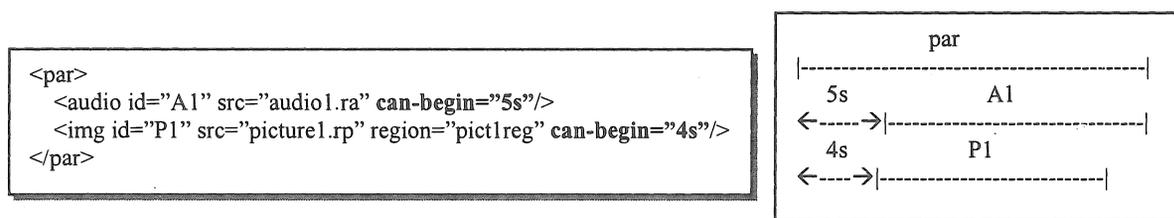


Figura 5 - atributo can-begin

Na figura 6, mostramos como especificar os relacionamentos causais entre os clips do documento. Para especificar os links que descrevem estes relacionamentos criamos dois atributos (um para cada tipo de link): *startlink* e *stoplink*. Os relacionamentos especificados na figura 6 são os mesmo do exemplo apresentado na seção 2.4.

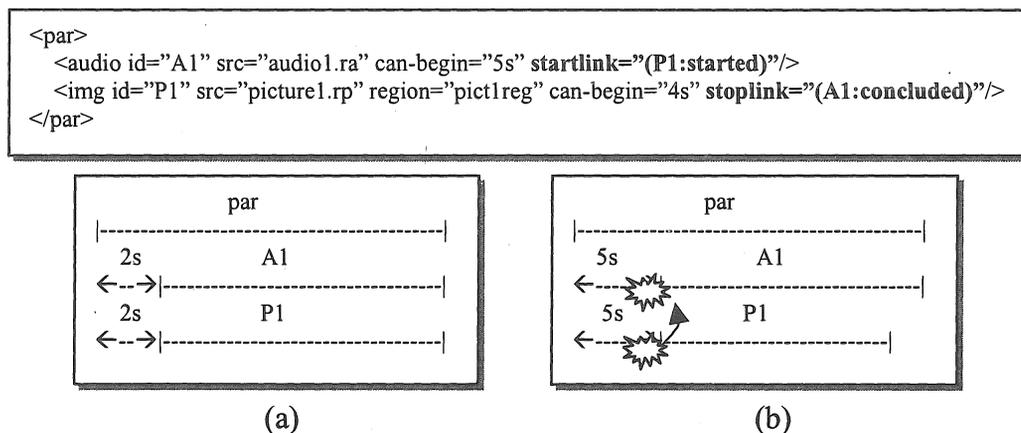


Figura 6 - atributos startlink e stoplink: (a) P1 e A1 começam sincronizados em 2s e terminam sincronizados quando A1 termina; (b) não é possível começar P1 dentro da tolerância, P1 e A1 são descartados.

Além de ser usado para sincronizar o fim da apresentação de dois ou mais clips, o atributo *stoplink* pode ser utilizado também para interromper um clip que perdeu o significado devido a algum problema ocorrido em outro clip do qual ele dependia. Desta forma, estaremos economizando os

recursos da rede. O *startlink*, por sua vez, também pode ser usado para iniciar um objeto alternativo, com requisitos de QoS menores, quando um certo objeto não puder ser restituído.

Com o tag `<switch>` você pode especificar múltiplas opções dentre as quais a sua ferramenta de apresentação (player) irá escolher uma para ser apresentada. A escolha de qual clip apresentar é feita da seguinte forma: o player examina cada opção na ordem em que aparecem, avaliando cada atributo de teste e seus valores para selecionar um clip válido. Dentre os atributos de teste definidos na especificação da linguagem SMIL podemos citar: *system-bitrate*, *system-caption*, *system-language*, *system-screen-size*, *system-screen-depth*.

O tag `<switch>` realiza a escolha baseado em variáveis estáticas que são configuradas ou armazenadas pelas ferramentas de apresentação (players) dos clientes. Utilizando os relacionamentos causais, isto é, os atributos *startlink* e *stoplink*, estamos verificando variáveis dinâmicas que mudam de estado durante a apresentação. Na verdade, podemos associar um relacionamento causal ao próprio grupo switch (fig.7), obtendo assim, uma adaptação dinâmica da apresentação diante de certas dificuldades de comunicação. Só depende do autor do documento especificar as possíveis variações nos formatos da apresentação.

```
<switch startlink="not (V1:stoped)">
  <audio src="audio-portuguese" system-language="pt-BR"/>
  <audio src="audio-english" system-language="en"/>
</switch>
```

Figura 7 – especificando um link condicional para o switch

Por fim, criamos o atributo *ess* para podermos especificar um objeto como essencial ou qualitativo, através dos valores *true* ou *false*, respectivamente. Este atributo, como exemplificado na figura 8, declara que o objeto em questão é essencial e que sua apresentação deve ser garantida pelo ambiente. Seu valor default é "false".

```

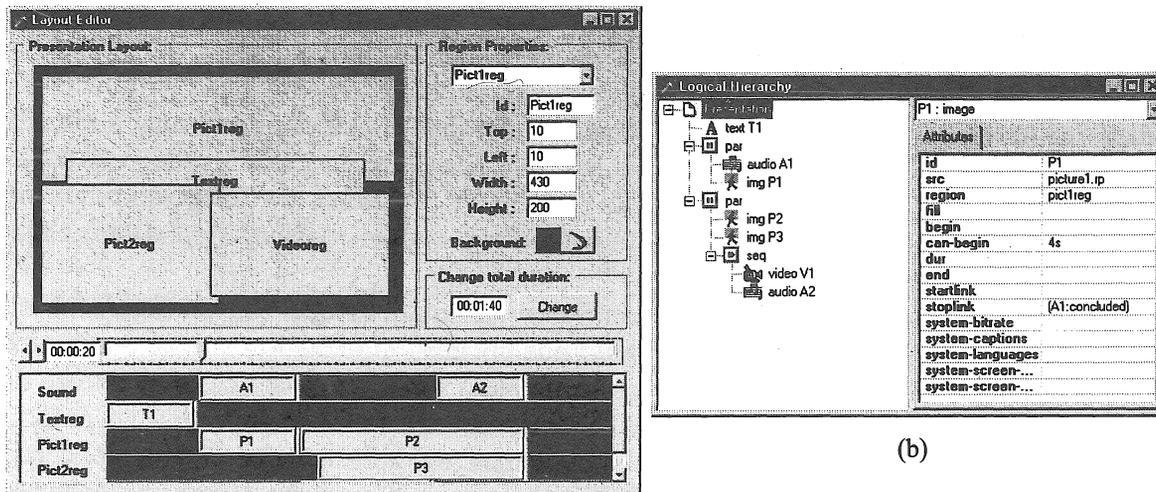

```

Figura 8 – especificando objetos essenciais

4. Ferramenta de autoria

Nesta seção apresentaremos uma visão geral da ferramenta de autoria que foi desenvolvida neste projeto para implementar a estratégia proposta utilizando a linguagem SMIL incrementada com as extensões que foram criadas e apresentadas na seção anterior. Esta ferramenta foi desenvolvida utilizando o ambiente de desenvolvimento de aplicações Delphi 4.0 da Inprise™ para a plataforma Windows98®.

A figura 9(a) ilustra uma das interfaces da ferramenta de autoria. Através desta interface, o autor especifica o layout da apresentação definindo as regiões de playback onde os clips deverão ser apresentados. É possível criar novas regiões, movê-las, redimensioná-las e definir suas cores de fundo. Para cada nova região é adicionada uma linha de tempo onde podem ser vistos os objetos (clips). Uma única linha temporal para a trilha sonora da apresentação está sempre presente, podendo, entretanto, permanecer vazia. Então, é possível visualizar o andamento da apresentação em um único plano (eixo temporal), tendo uma idéia da duração de cada objeto.



(a)

(b)

Figura 9 – (a) interface do editor de layout; (b) interface do editor da apresentação

Em uma outra interface, mostrada na figura 9(b), o autor especifica a estrutura lógica da apresentação através da árvore hierárquica discutida na seção 2.1. Nesta interface é possível criar novos grupos (par, seq e switch), novos clips, organizá-los, e definir todos os seus atributos e propriedades temporais/causais.

5. Exemplo de aplicação

Nesta seção, descrevemos um exemplo simples na área de ensino a distância. O treinamento é composto das duas fases a seguir: uma introdução (os servidores enviam informações para apresentar o tema da aula, os autores, copyright, etc) e o treinamento em si (os servidores enviam informações para apresentar o conteúdo da aula).

Na introdução, um vídeo de abertura (“v1”) apresenta a empresa responsável pelo treinamento e os autores. Em paralelo com o vídeo existe uma trilha sonora de fundo (“a1”). Uma figura alternativa (“p1”) é apresentada se o vídeo não puder ser apresentado com os requisitos de QoS especificados. A seguir, o material da aula é apresentado. Nesta fase, a apresentação contém um vídeo (“v2”), que apresenta o conteúdo da aula, um áudio (“a2”) relacionado ao vídeo, e uma figura (p2”) que apresenta alguns diagramas e imagens. Nesta fase existe um forte requisito pela apresentação do vídeo. Em outras palavras, apresentar apenas a informação do áudio e da figura é considerada sem utilidade do ponto de vista da aplicação (autor). Se apenas o áudio não puder ser apresentado, ele será substituído pela apresentação de uma informação textual (“t1”) para o aluno (usuário).

Nós consideramos que em um ambiente de ensino a distância, onde as apresentações são na maioria das vezes relativamente longas, o usuário não será incomodado por um atraso adicional causado pelo mecanismo de adaptação, visto que nós consideramos que a interpretação correta do documento é mais crítica do que o atraso, e que o aluno preferirá assistir uma apresentação coerente com algum atraso casual do que uma apresentação contínua com uma qualidade insatisfatória gerando uma interpretação equivocada. Esta situação vem enfatizar a especificação temporal flexível mencionada na seção 2.3.

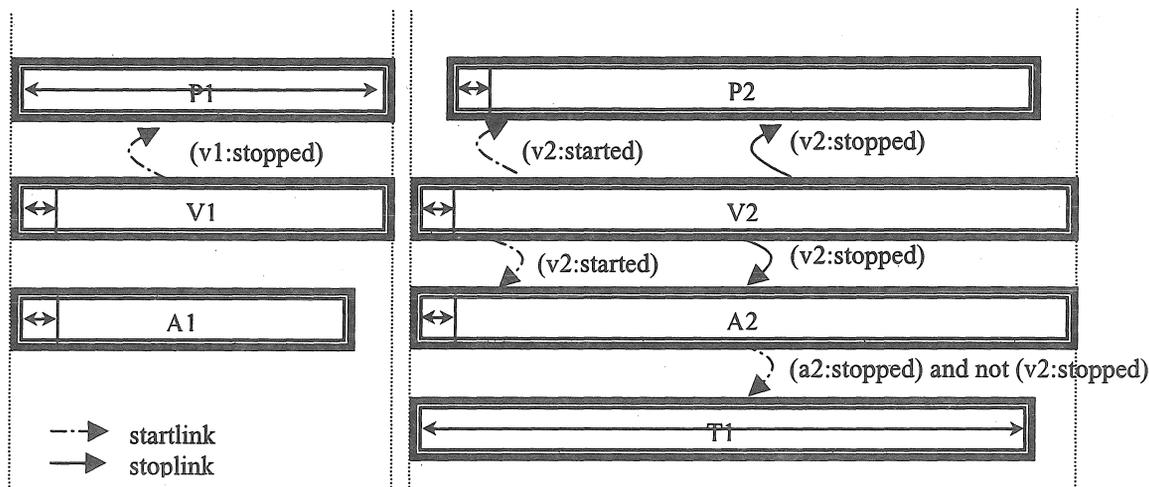


Figura 10 – Exemplo de aplicação

6. Trabalhos relacionados

6.1 MHEG

O padrão MHEG-5 [Mheg96] define um objeto Link que consiste de uma LinkCondition e de um LinkEffect. O LinkEffect, que é uma lista de ações elementares, é executado quando a LinkCondition se torna verdadeira. Em MHEG-5, uma LinkCondition é sempre disparada como resultado da ocorrência de um evento, e um evento sempre emana de exatamente um único objeto. Neste trabalho, os links têm características semelhantes a MHEG-5, porém, diferentemente de MHEG-5, que usa os links apenas para descrever os aspectos de sequenciamento de uma apresentação, nossa abordagem usa os links, e os relacionamentos semânticos criados por eles, para prover um melhor uso dos recursos de comunicação disponíveis ao mesmo tempo que preserva a coerência do documento especificado pelo autor.

O padrão MHEG-5 foi desenvolvido para suportar a distribuição de aplicações multimídia interativas em uma arquitetura cliente/servidor através de plataformas de diferentes tipos e modelos. MHEG-5 define uma forma final de representação para o intercâmbio das aplicações entre as diferentes plataformas. O escopo global de MHEG-5 é definir a sintaxe e o significado de um conjunto de classes de objetos que podem ser usados para interoperabilidade de aplicações multimídia. Entretanto, não existem, no padrão MHEG-5, considerações sobre especificação e suporte de QoS, ou adaptabilidade baseada em requisitos de QoS.

6.2 SMIL

A linguagem SMIL está sendo desenvolvida pelo W3C, o qual liberou a especificação de sua versão SMIL 1.0 através de uma recomendação em junho de 1998 [Smil98]. A linguagem SMIL permite integrar um conjunto de objetos multimídia independentes em uma apresentação multimídia sincronizada através de uma especificação textual, com uso de tags e muito similar ao HTML.

SMIL define o elemento switch para especificar um conjunto de objetos alternativos. Entretanto, a escolha de uma alternativa depende de um atributo de teste. Uma apresentação multimídia não é o local certo para se especificar este tipo de condição, porque as condições dependem da

plataforma e do tipo de conexão que um usuário pode ter com o servidor de mídias, e de variáveis fixas que são definidas pelo usuário na sua ferramenta de apresentação. Como as condições da apresentação são conhecidas durante o decorrer da mesma, uma adaptação da apresentação deve se dar com base em condições dinâmicas a serem monitoradas ao longo da apresentação.

6.3 Outros trabalhos

O problema de adaptação dos formatos de apresentações surgiu inicialmente com as conferências multimídia. Quando uma conferência é feita em multicast no MBONE, importantes benefícios de performance podem ser alcançados com “vídeos escalonados”, a codificação baseada no princípio de sucessivos refinamentos da qualidade. A informação básica é enviada primeiro, então, camadas de refinamento adicionam mais informação para incrementar a qualidade da apresentação. Destinos (clientes) que estão conectados usando diferentes bitrates podem selecionar as camadas de acordo com os seus requisitos. Mais uma vez, o usuário decide quais camadas ele deseja receber, provendo certo grau de adaptação.

Existem muitos trabalhos que lidam com a questão da adaptação de apresentações multimídia. A maioria realiza a adaptação através de decisões tomadas pelo usuário, seja no instante de pedido do documento ao servidor (content negotiation), seja na verificação de variáveis fixas (user's profiles), seja na seleção de canais ou filtros (proxys). Porém, verificamos uma deficiência com relação à especificação de uma adaptabilidade baseada nos requisitos de QoS dos objetos que compõem o documento. Neste trabalho, nós propomos a especificação dos requisitos de QoS e da possível adaptação que o documento pode sofrer ao mesmo tempo que construímos uma malha de causalidade através de links condicionais que descrevem a coerência do documento, resultando em uma correta interpretação da apresentação.

7. Conclusões

Neste artigo, apresentamos uma nova estratégia de autoria de documentos multimídia que permite ao autor estabelecer a semântica do documento que deve ser respeitada durante a sua apresentação nas estações clientes. Isto veio possibilitar que, a partir de um único documento criado com base nesta estratégia, possam ser gerados diferentes formatos de apresentação de acordo com a disponibilidade dinâmica dos recursos de comunicação, mantendo-se, entretanto a consistência desejada para o documento. Essa característica se mostra de grande valor principalmente em um ambiente de ensino a distância. Tanto a problemática da sincronização quanto o controle de degradação da qualidade da apresentação são considerados em uma abordagem original que combina o tradicional modelo temporal com um modelo baseado na especificação de relacionamentos causais entre os objetos que compõem o documento.

Uma ferramenta de autoria que implementa as funcionalidades desta estratégia foi também desenvolvida. Esta ferramenta gera um documento multimídia baseado na linguagem SMIL e incrementada com as extensões (atributos) criadas neste projeto para modelar nossa estratégia. A linguagem SMIL foi escolhida como base do documento por apresentar idéias valiosas que vieram de encontro a algumas características definidas no projeto, por ser uma proposta recente e estar recebendo um grande foco atualmente dentro da Internet. Outro fator é que existem poucas ferramentas de autoria nesta linguagem, e as que existem não implementam todas as suas funcionalidades.

Nossa próxima direção dentro do projeto é investigar as questões de reserva de recursos associada ao roteamento de QoS. Nós estamos estudando as complexidades do RSVP dentro do

Modelo de Serviços Integrados e do recente Modelo de Serviços Diferenciados. Nós pretendemos definir uma arquitetura que usará efetivamente os protocolos de tempo-real (RTP/RTCP e RTSP) não apenas para transferir mídias contínuas mas também prover sinalização de reserva de recursos e roteamento abordando os problemas de complexidade e escalabilidade mencionados em [Rose97a, Rose97b, Pan97, Mara98].

8. Referências

- [Carm97] L.F.R.C. Carmo, L.Pirmez, "ServiMídia: Ambiente Integrado de Criação e Recuperação de Documentos Multimídia com Controle Adaptativo de QoS", 2º Seminário Franco-Brasileiro em Sistema Informáticos Distribuídos, Novembro, 1997
- [Cour96] J.P. Courtiat, L.F.R.C. Carmo, R.C. de Oliveira, "A General-purpose Multimedia Synchronization Mechanism Based on Causal Relations", IEEE Journal on Selected Areas in Communications - Synchronization Issues in Multimedia Communications, Vol. 14, N. 1, January, 1996.
- [Aurr98] Aurrecoechea C, Campbell AT, Hauw L, "A survey of QoS architectures", ACM Multimedia Systems 6:138-151, 1998.
- [Voge95] Vogel, A. et al., "Distributed Multimedia and QoS: A Survey", IEEE Multimedia, verão, 1995, pp. 10-18.
- [MHEG96] ISO/IEC DIS 13522-5, "Information Technology Coding of Multimedia and Hypermedia Information, Part 5: Support for Base-Level Interactive Applications, MHEG-5 IS Document Pre-release 5", 1996.
- [Sant98] Santos M. T. P., Vieira M. T. P., "Sistema de Recuperação de Informações em um Servidor de Objetos Multimídia", IV Simpósio Brasileiro de Sistemas Multimídia e Hipermídia, Maio, 1998.
- [Cand98] Candan KS, Prabhakaran B, Subrahmanian VS, "Retrieval schedules based on resource availability and flexible presentation specifications", ACM Multimedia Systems 6: 232-250, 1998.
- [Smil98] Synchronized Multimedia Working Group of the World Wide Web Consortium, "Synchronized Multimedia Integration Language (SMIL) 1.0 Specification", W3C Recommendation, June, 1998.
- [Rose97a] J. Rosenberg, H. Schulzrine, "Timer Reconsideration for Enhanced RTP Scalability", draft-ietf-avt-reconsider-00.ps, July 1997.
- [Rose97b] J. Rosenberg, H. Schulzrine, "New Results in RTP Scalability", draft-ietf-avt-byerecon-00.ps, November 1997.
- [Pan97] P. Pan, H. Schulzrine, "YESSIR: A Simple Reservation Mechanism for the Internet", August 1997
- [Mara98] R. El-Marakby, D. Hutchison, "A Scalability Scheme for the Real-time Control Protocol", Proceedings of the 8th IFIP International Conference on High Performance Networking (HPN'98), pp.147-162, Vienna, Austria, September 1998.

