UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

Proposta de Extensão do Modelo de Expressões Intervalares para Sincronização Multimídia na Web

por

LETICIA LOPES LEITE

Dissertação submetida à avaliação,
como requisito parcial para a obtenção do grau de Mestre
em Ciência da Computação

Prof. Dr. José Valdeni de Lima
Orientador

Porto Alegre, dezembro de 2000.

CIP – CATALOGAÇÃO NA PUBLICAÇÃO

Leite, Leticia Lopes

Proposta de extensão do modelo de expressões intervalares para sincronização multimídia na Web/por Leticia Lopes Leite. – Porto Alegre: PPGC da UFRGS, 2000.

85p.: il.

Dissertação (mestrado) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR – RS, 2000. Orientador: Lima, José Valdeni de.

1. Sincronização multimídia. 2. Modelos para sincronização. 3. Expressões Intervalares. I. Lima, José Valdeni de. II. Título.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
Reitora: Profa. Wrana Panizzi
Pró-Reitor de Ensino: Prof. José Carlos Ferraz Hennemann
Superintendente de Pós-Graduação: Prof. Philippe Olivier Alexandre Navaux
Diretor do Instituto de Informática: Prof. Philippe Olivier Alexandre Navaux
Coordenadora do PPGC: Profa. Carla Maria Dal Sasso Freitas
Bibliotecária-Chefe do Instituto de Informática: Beatriz Haro

Agradecimentos

Aos meus pais, Adão e Irdes que sempre estiveram ao meu lado suportando dias de mau humor e constantes choramingos com muito amor.

Ao meu irmão, Paulo Ricardo (Paulinho) pela paciência, amizade e carinho que sempre me dedica.

Ao meu namorado, Paulo Ricardo pela ajuda com a dissertação, paciência, companheirismo e amor.

A minha avó Leticia, que mesmo não estando presente, continua olhando por mim.

Ao meu orientador Prof. Dr. José Valdeni de Lima, pelos conselhos, correções e dedicação à minha dissertação.

Ao bolsista Niccholas Vidal, pela ajuda com o estudo de caso.

Ao CNPq pela bolsa de pesquisa concedida e sem a qual não seria possível fazer este mestrado.

Finalmente, gostaria de agradecer a todos aqueles que de uma forma ou outra estiveram ao meu lado apoiando, corrigindo e dando conselhos sempre muito valiosos.

Obrigada,

Leticia.

Sumário

Lista de Símbolos **7**

Lista de Figuras **8**

Lista de Tabelas **10**

Resumo **11**

Abstract **12**

1 Introdução **13**

2 Sincronização Multimídia **16**

2.1 Sincronização Espacial (SE) 17
2.2 Sincronização Temporal (ST) 17
2.3 O Problema da Sincronização 19
2.4 Consistência da Apresentação 20

3 Modelos para Especificação de Sincronismo **22**

3.1 Modelo *Timeline* 23
3.2 Modelo Baseado em Eventos 24
3.3 Modelo de Sincronização Baseado em Intervalos 25
3.4 Modelo Avançado de Sincronização Baseado em Intervalos 26
3.5 Modelo *Script* 29
3.6 Modelo de Expressões Intervalares 30

4 Ferramentas de Autoria para Apresentações Multimídia Sincronizadas **36**

4.1 CMIF (CWI *Multimedia Interchange Format*) 37
4.1.1 O Modelo de Dados do CMIF *37*
4.1.2 A Vista Hierárquica *38*
4.1.3 A Vista de Canais *38*
4.1.4 A Vista da Execução *39*
4.2 Macromedia Flash 40
4.3 Assimetrix Multimedia Toolbook 43
4.4 A Linguagem SMIL (*Synchronized Multimedia Integration Language*) 45
4.4.1 Player *46*
4.4.2 A Sincronização Espacial *46*
4.4.3 A Sincronização Temporal *47*
4.4.4 Largura de Banda *48*
4.4.5 Vantagens no Uso do SMIL *49*
4.4.6 Desvantagens no Uso do SMIL *50*
4.4.7 O Modelo de Expressões Intervalares e a Linguagem SMIL *50*

5 Proposta de Extensão do Modelo de Expressões Intervalares **52**

5.1 Definição do Segmento de Mídia Conjunto V 54
5.1.1 Funcionamento do Controlador "Mídia Invisível" *56*
5.1.2 O Conjunto V e o Conjunto Vazio *58*
5.1.3 Implementação do Conjunto V *59*
5.2 Ponto de Sincronização 59

6 Estudo de Caso: SIGHMA 63

6.1 A Estrutura do SIGHMA 63
6.2 A Implementação do SIGHMA 64
6.3 A reestruturação do SIGHMA 65
6.4 Funcionamento do SIGHMA 66

7 Conclusões e Trabalhos Futuros 72

Anexo 1 Código do SIGHMA 74

Anexo 2 Testes realizados com o Macromedia Flash 78

Bibliografia 80

Lista de Abreviaturas

BMP	Formato de arquivos gráficos
CMIF	CWI *Multimedia Interchange Format*
CMIFed	Editor CMIF
CSS1	*Cascading Style Sheets* nível 1
CSS2	*Cascading Style Sheets* nível 2
CWI	*Centrum voor Wiskunde en Informatica*
DA	Duração da Apresentação
DA	Duração da apresentação
DHTML	*Dynamic HiperText Markup Language*
DM	Duração da Mídia
DM	Duração da mídia
DTD	Definição de Tipo de Documento
EBNF	*Extended Backus-Naur Form*
FA	Fim da Apresentação
GIF	Formato de arquivo de imagens
GriNS	*Graphical iNterface to SMIL*
HPAS	*Hypermedia Presentation and Authoring System*
HTML	*HyperText Markup Language*
IA	Início da Apresentação
SE	Sincronização Espacial
SIGHMA	Sistema de Gerência de Hiperdocumentos Multimídia Ativos
SMIL	*Synchronized Multimedia Integration Language*
ST	Sincronização Temporal
SWF	Extensão de arquivos do tipo *Shockwave*
W3C	*World Wide Web Consortium*
WWW	*World Wide Web*
XML	*eXtensible Markup Language*

Lista de Símbolos

δ_1	Primeiro Parâmetro de Atraso
δ_2	Segundo Parâmetro de Atraso
∞	Infinito
t	Tempo de Duração de um Intervalo
I_{base}	Representação de um Segmento de Mídia
Kbps	Kilobits por segundo
α	Alfa
α_p	Alfa base p
β	Beta
/	Tal que
$\leq$	Menor ou igual
=	Igual
-	Menos
$\rightarrow$	Equivale
{Ø}	Conjunto Vazio
δ_3	Terceiro Parâmetro de Atraso

Lista de Figuras

FIGURA 2.1 – Exemplo de sincronização espacial 17
FIGURA 2.2 - Sincronização de Componentes Independentes do Tempo [GAS 98] .. 18
FIGURA 2.3 - Exemplo de inconsistência qualitativa [KER 95] 20
FIGURA 2.4 - Exemplo de inconsistência quantitativa [KER 95] 21
FIGURA 3.1 - Interface genérica dos sistemas que utilizam *timeline* [COS 96] 23
FIGURA 3.2 - Relações Temporais Binárias definidas em [ALL 83] 25
FIGURA 3.3 - Representação gráfica da relação *Before* 26
FIGURA 3.4 - Representação gráfica da relação *During* 26
FIGURA 3.5 - Representação gráfica de inconsistência gerada pelo Modelo Avançado Baseado em Intervalos [GAS 98] 27
FIGURA 3.6 - Operadores do Modelo Avançado de Sincronização Baseado em Intervalos [WAH 94] 28
FIGURA 3.7 - Implementação utilizando o Modelo Avançado Baseado em Intervalos 28
FIGURA 3.8 - Apresentação descrita através de um *script* [GAS 98] 30
FIGURA 3.9 - Representação gráfica dos operadores do Modelo de Expressões Intervalares 33
FIGURA 3.10 - Composição temporal de um roteiro guiado a um museu utilizando Expressões Intervalares [KER 97] 34
FIGURA 4.1 - Vista hierárquica do CMIFed [GAS 98] 38
FIGURA 4.2 - A vista de canais do CMIFed [BUL 93] 39
FIGURA 4.3 – Interface do CMIFed [BUL 93] 40
FIGURA 4.4 - *Timeline* do *Macromedia Flash* 4.0 41
FIGURA 4.5 - Apresentação construída com o *Macromedia Flash* 4.0 42
FIGURA 4.6 - Interface do *Assimetrix Multimedia Toolbook* 44
FIGURA 4.7 - Interface do *RealPlayer* G2 8.0 46
FIGURA 4.8 - Exemplo de sincronização espacial utilizando SMIL [STA 99] 47
FIGURA 4.9 - Exemplo de sincronização temporal com o operador "*par*" utilizando SMIL [STA 99] 47
FIGURA 4.10 - Exemplo de sincronização temporal com o operador "*seq*" utilizando SMIL [STA 99] 48
FIGURA 5.1 - Exemplo de utilização do controlador "**mídia invisível**" 54
FIGURA 5.2 - Composição temporal de um roteiro guiado a um museu com a utilização de um controlador de tempo 55
FIGURA 5.3 - Representação gráfica dos operadores "*seq*" e "*ident*" utilizando mídia pertencente ao Conjunto V 57
FIGURA 5.4 - Representação gráfica de uma apresentação utilizando o Conjunto V . 57
FIGURA 5.5 - Exemplo de composição temporal utilizando o Conjunto Ø [KER 97]. 58
FIGURA 5.6 - Representação da operação "*overlaps*" 59
FIGURA 5.7 – Representação gráfica de uma composição temporal utilizando o operador SINC-POINT 61
FIGURA 5.8 – Representação gráfica da modificação de composição temporal utilizando o operador SINC-POINT 61
FIGURA 6.1 - Representação gráfica da relação dos blocos no SIGHMA 64
FIGURA 6.2 - Representação gráfica da primeira implementação do SIGHMA 65
FIGURA 6.3 - Representação gráfica da ordem da apresentação após a reestruturação 66
FIGURA 6.4 - Botões para navegação no SIGHMA 67

FIGURA 6.5 - Implementação do funcionamento dos botões de navegação.....67
FIGURA 6.6 - Implementação da reestruturação do SIGHMA 68
Figura 6.7 - Janela de menu do SIGHMA 69
FIGURA 6.8 - Janela de apresentação do SIGHMA.....70
FIGURA 6.9 - Janela de animação do SIGHMA 70
FIGURA 6.10 – Exemplo de Janela do módulo "*Prizes*" 71

Lista de Tabelas

TABELA 3.1 – Eventos Intrínsecos do HTML.....25
TABELA 3.2 - Representação dos operadores definidos em [DUD 95].....32
TABELA 4.1 - Comparação entre os operadores das Expressões Intervalares e a Linguagem SMIL.....50
TABELA 5.1 - Relação entre operadores do modelo e o controlador “mídia invisível”56

Resumo

O desenvolvimento de apresentações multimídia sincronizadas para Web se baseia na definição e especificação de relações entre objetos advindos de diversas mídias, que possuem requisitos específicos de desempenho, estando ainda sujeitos à atrasos randômicos durante sua transmissão. Neste sentido, foram definidos modelos para especificação de sincronismo que têm por objetivo estruturar a ordenação destas aplicações de maneira a executar as relações definidas pelo autor quando da exibição coerente no destino.

Este trabalho realiza um estudo dos principais modelos utilizados no desenvolvimento de apresentações multimídia sincronizadas, principalmente relacionando-se ao ambiente Web. Desta forma, tornando possível identificar características, vantagens e desvantagens de seus usos, bem como verificar que o Modelo de Expressões Intervalares é adequado a este trabalho, embora ainda apresente algumas deficiências no controle de mídias e interações do usuário.

Finalmente, visando solucionar as deficiências encontradas no Modelo de Expressões Intervalares, foi desenvolvida uma proposta de extensão para o mesmo. Ela consiste na criação de um controlador para a interação do usuário e para a manipulação de mídias, chamado **mídia invisível**; e na definição do operador **Sinc-Point,** que proporciona ao autor maior flexibilidade na definição da sincronização. Esta proposta apresenta as seguintes vantagens: possibilita ao autor criar sincronizações do tipo "*overlaps*", que baseia o disparo de uma mídia durante a apresentação de outra; possibilita ao autor realizar um controle sobre o intervalo para a interação do usuário e para a apresentação de mídias; mantém a consistência da apresentação; não acarreta muitas mudanças no modelo original.

Palavras chave: Sincronização Multimídia, Modelos para Sincronismo, Proposta de Extensão do Modelo de Expressões Intervalares.

TITLE: "SUGGEST EXTENSION OF INTERVAL EXPRESSION MODEL FOR MULTIMEDIA SYNCHRONIZATION ON THE WEB"

Abstract

The development of synchronized multimedia presentations is based on definitions and specifications of the relationships between objects obtained from several types of media, which are susceptible to eventual delays during transmission. In this sense, there is a necessity to define models specifying synchronism with the objective of structuring the order of execution of these sources of media, according to the relationships defined by the author.

This thesis analyses the main characteristics, advantages and disadvantages of the most important existing models, specially those related with the Internet. It also verifies that the Interval Expressions Model is adequate for this purpose, even thought it presents some deficiencies on managing media and user interaction.

To solve these deficiencies, an extension to the Interval Expressions Model is suggested. This extension consists in the creation of two handlers. The first is an handler responsible for user interaction and media, called **invisible media.** The other is an handler that offers the author more flexibility on the definition of synchronization, called **Sinc-Point**. This extension presents the following advantages: (1) permits the author to create "overlap" synchronizations; (2) permits a better control of user interaction and presentation of media; (3) maintains a presentation consistent ; (4) the original model can be easily adapted.

Keywords: Multimedia Synchronization, Models of Synchronism, Extension of Interval Expressions Model.

1 Introdução

Segundo [STE 90], multimídia pode ser definida como a geração, a representação, o processamento, o armazenamento e a disseminação de informações expressas por meio de múltiplas mídias (tais como áudio, vídeo, imagem, animação e texto), dependentes ou independentes do tempo, de forma integrada e controlada por computador. Diz-se então, que Sistemas Multimídia são aqueles que processam informações multimídia. No presente trabalho, foi adotada a definição proposta por [STE 90] aliada à inclusão de um novo conjunto de mídias proposto em [DUD 95], do qual fazem parte ainda os *links* e programas executáveis.

O advento da multimídia tem-se mostrado importante auxiliar em diversos setores da economia, desde o ensino, passando pelo comércio eletrônico ou ainda como colaborador no processo de desenvolvimento e manutenção na indústria. Segundo [INF 98], a integração de mídias como som, imagem e texto, gera uma interação muitas vezes mais efetiva que esses diversos meios separadamente, fazendo com que os processos de assimilação e compreensão das informações tornem-se muito mais rápidos.

Atualmente, auxiliado pelo aumento do uso das redes de computadores, a demanda por informações multimídia cresceu muito rapidamente, fato que é salientado no artigo "Desenvolvendo Aplicações Multimídia", de [INF 98]:

"Há poucos anos, a multimídia estava relegada a um plano secundário, apenas utilizada em jogos e em aplicações isoladas e muito específicas. Mas, os rápidos avanços das tecnologias de informação a estão tornando viável comercialmente para aplicação em alta escala, e os desenvolvedores devem realmente considerar a utilização em seus produtos".

A necessidade de integração dos diversos tipos de mídias em uma aplicação decorre principalmente das dependências existentes entre as informações contidas nas mídias, tais como dependências temporais, espaciais e relações causais. A tarefa de ordenar a apresentação dos dados multimídia baseada nas dependências existentes é denominada Sincronização Multimídia [CAR 96].

Segundo [GON 98], a sincronização em aplicações multimídia pode ser dividida em duas categorias:

- Sincronização ao Vivo (sincronização em tempo real): quando o objetivo é apresentar as mídias da mesma forma como foram originalmente capturadas.
- Sincronização Sintética: quando as relações das mídias que compõem a apresentação são definidas de forma independente. Nesse caso, é necessário utilizar um modelo para a especificação e a manipulação das operações de sincronização. Exemplos comuns de operações são: apresentação de mídias em paralelo e apresentação de mídias em seqüência.

A sincronização pode referenciar aspectos temporais e/ou espaciais em uma apresentação multimídia. Na sincronização temporal são definidas as relações temporais entre as mídias, ou seja, ela é responsável pelo estabelecimento da ordem de

apresentação dos objetos. Enquanto na sincronização espacial, são definidas as relações de *layout* da apresentação, ou seja, ela é responsável pela definição da forma de apresentação da mídia em um dispositivo de saída.

A definição da sincronização de diversas mídias em uma aplicação requer a utilização de técnicas específicas que possam prover o suporte à comunicação de dados multimídia e à apresentação coerente no destino. Nesse sentido, as técnicas de sincronização utilizam mecanismos para coordenar a ordenação temporal e espacial dos eventos relacionados à aplicação [GON 98]. O sincronismo pode ser aplicado para uma apresentação de fluxos de dados concorrentes (paralelos) ou seqüenciais, assim como para eventos gerados externamente pelo usuário. Essas técnicas ainda devem considerar as atuais necessidades do mercado e possibilitar o trabalho na Web, que por apresentar informações multimídia armazenadas remotamente, podem ocasionar atrasos randômicos durante a transmissão e recuperação de dados.

Sendo assim, utilizar uma técnica que permita não somente especificar a sincronização entre objetos multimídia, mas também modelar a apresentação por completo é uma necessidade vital para a implantação de um Sistema Multimídia.

Diante da necessidade observada, este trabalho se propõe a fazer um estudo dos principais modelos utilizados para o trabalho com Sincronização Temporal Multimídia na Web, fazendo um levantamento de vantagens e desvantagens do seu uso, ferramentas mais utilizadas que os implementam, assim como sugerir o emprego de um ou mais modelos para a definição deste tipo de aplicação.

Desta forma, a organização do presente trabalho baseia-se na estrutura descrita a seguir.

O segundo capítulo introduz os aspectos básicos relacionados à sincronização multimídia/hipermídia, tais como: tipos de sincronização (espacial e temporal), problemas inerentes (por exemplo, a consistência da apresentação), assim como algumas características desejáveis para os modelos utilizados na especificação do sincronismo.

No terceiro capítulo são descritos os modelos utilizados no trabalho de sincronização multimídia, suas principais características e a análise segundo os requisitos definidos em [COU 96]. Baseados nessas características e a adequação aos requisitos especificados, definir-se-á um modelo, sobre o qual serão sugeridas modificações visando proporcionar maior flexibilidade ao trabalho com Sincronização Temporal Multimídia na Web.

O quarto capítulo é responsável por apresentar as ferramentas que se baseiam nos modelos estudados e que são comumente utilizadas para definição de aplicações multimídia sincronizadas. O estudo dessas ferramentas foi base para a determinação da aplicabilidade dos modelos ao trabalho de sincronização na Web, bem como para verificação do mais adequado a este tipo de aplicação.

No quinto capítulo é apresentada a Proposta de Extensão do Modelo de Expressões Intervalares, cujo objetivo é solucionar algumas deficiências do mesmo, tais

como o controle da interação com o usuário e do disparo de mídias e a sincronização baseada no transcorrer de mídias.

No sexto capítulo é apresentado o protótipo desenvolvido para a verificação da aplicabilidade do Modelo de Expressões Intervalares e da proposta de extensão desenvolvida. Esse protótipo chama-se SIGHMA (**S**istema de **G**erência de **H**iperdocumentos **M**ultimídia **A**tivos) e foi implementado para uso na Web através da Linguagem SMIL (*Synchronized Multimedia Integration Language*).

O sétimo capítulo é responsável por apresentar os anexos, que incluem o código SMIL do protótipo desenvolvido, assim como algumas páginas de teste construídas para avaliação do *Macromedia Flash*.

Finalmente, o oitavo capítulo é responsável por apresentar as conclusões obtidas com a realização desta dissertação, assim como por determinar algumas metas para trabalhos futuros.

2 Sincronização Multimídia

A Sincronização Multimídia se refere aos mecanismos usados na ordenação de eventos temporais e espaciais para a apresentação de diferentes mídias de um hiperdocumento[1]. Segundo [COS 96], para que a sincronização possa ser definida por um sistema hipermídia, esse deve possuir uma representação interna adequada e fornecer um mecanismo que capture os relacionamentos de sincronização entre os componentes.

A apresentação de fluxos sincronizados é essencial para se obter a impressão de naturalidade, pois fluxos fora de sincronismo são percebidos como sendo algo artificial, estranho ou até mesmo "enjoado" [STE 96].

Escobar [ESC 95] descreve experimentos, datados do começo do século, relacionados à influência da sincronização na percepção humana. Mais recentemente, Steinmetz [STE 96] também realizou alguns experimentos. Em um deles, várias pessoas assistiam a um noticiário de TV com *skews*[2] artificiais introduzidas em intervalos de 40 milisegundos (ms), de modo a observar a influência da sincronização labial na percepção humana. Os espectadores deveriam dar um parecer a respeito da sincronização em três visões: só da cabeça, da cabeça e do ombro, e do corpo inteiro. Primeiro foi perguntado se havia algo de estranho ou artificial no noticiário. Depois, se fosse detectado um erro de sincronização, era indagado se a pessoa era capaz de dizer se o áudio estava à frente ou atrás do vídeo. Por último, foi perguntado se o erro era aceitável ou não. A partir da análise dos resultados do experimento, delimitou-se empiricamente a região de sincronização labial entre –80ms (áudio atrasado em relação ao vídeo) e +80ms (áudio adiantado em relação ao vídeo). A região de assincronia foi delimitada abaixo de –160ms e acima de +160ms. Para essa região, freqüentemente os espectadores se distraíam, ficando mais atraídos pelo efeito de assincronia do que pelo conteúdo do noticiário em si.

A composição de objetos multimídia depende das características espaciais e temporais dos elementos multimídia [LIT 90]. Por isso, a sincronização é dividida em dois tipos: sincronização espacial e sincronização temporal (seções 2.1 e 2.2 respectivamente).

Segundo [XIA 97], um sistema hipermídia deve satisfazer as seguintes exigências:

- alto nível de sincronização temporal entre objetos multimídia;
- provisão de sincronização espacial dependente do tempo dentro da arquitetura do documento hipermídia;
- integração e manipulação transparente de vários tipos de mídia;
- interface com o usuário simples, mas poderosa, que manipule a informação temporal e espacial dentro do documento hipermídia.

---

[1] No presente trabalho, os termos hiperdocumento, aplicação hipermídia, aplicação multimídia, documento hipermídia e apresentação multimídia são utilizados como sinônimos.

[2] O *skew* é um parâmetro utilizado para medir a diferença entre os tempos de chegada de diferentes mídias que deveriam estar sincronizadas [MOU 99].

2.1 Sincronização Espacial (SE)

A Sincronização Espacial corresponde à combinação de objetos a serem exibidos em um dispositivo de saída, num dado instante de tempo em uma apresentação multimídia [SOA 92]. Nesse caso, a sincronização envolve operações de mudanças de escala, corte, conversões de cores e posicionamento dentro da janela da aplicação.

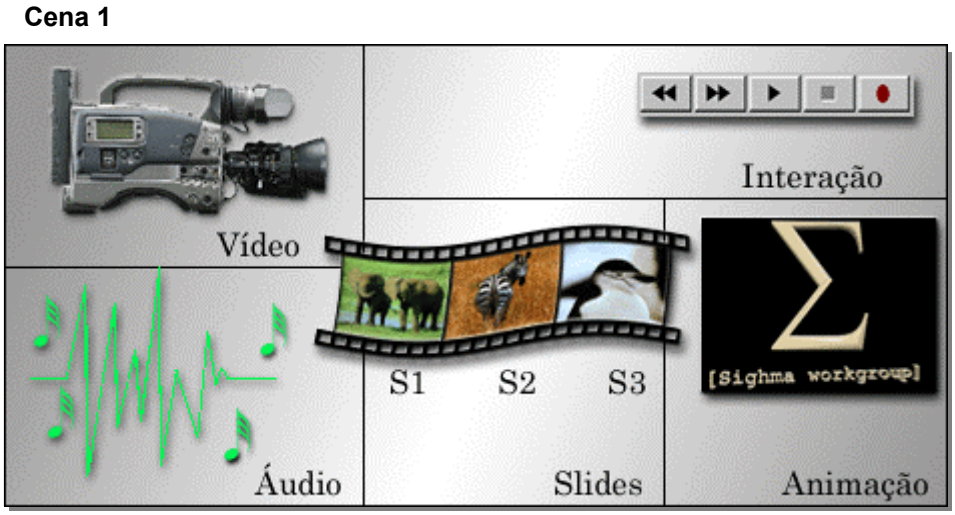


FIGURA 2.1 – Exemplo de sincronização espacial

A Figura 2.1 apresenta um exemplo de sincronização espacial, onde se tem a definição das regiões de apresentação de áudio (canto inferior esquerdo da janela), vídeo (canto superior esquerdo), animação (canto inferior direito) e uma seqüência de *slides* (S1, S2 e S3). Observe-se que, no canto direito superior da janela, é apresentada sob a forma de botão uma possibilidade de interação com o usuário.

A definição da sincronização espacial para apresentação em um monitor de vídeo irá definir como os componentes de um documento hipermídia estarão dispostos (posição ocupada na tela, prioridade de sobreposição, etc.). No caso de um dispositivo de áudio, a sincronização espacial envolve mixagem de canais de áudio em um dispositivo de saída de som, com ajustes de ganho e tom.

2.2 Sincronização Temporal (ST)

A Sincronização Temporal baseia-se no alinhamento de relacionamentos no domínio do tempo entre os constituintes do documento multimídia, ou seja, é o estabelecimento da ordem de apresentação dos objetos, criando regras para a exibição, inibição ou modificação dos mesmos de acordo com a especificação do autor[3] [ABR 97]. Segundo [XIA 97], a definição da sincronização temporal é um passo crucial no desenvolvimento de uma aplicação multimídia.

Como exemplo desse tipo de sincronização, temos um hiperdocumento que corresponda a uma aula. Durante a explanação, descrições e ilustrações devem ser apresentadas ao usuário (aluno), visando uma melhor assimilação do conteúdo. Torna-

[3] Os desenvolvedores de apresentações multimídia são chamados de autores.

se necessária então, a definição de relacionamentos temporais entre a voz do professor (áudio) e as ilustrações (imagens gráficas, textos e vídeo) que são referenciadas, visando manter a consistência temporal da apresentação [COS 96].

Segundo [COS 96], uma das maiores vantagens da Sincronização Temporal refere-se ao aumento do poder de comunicação de uma aplicação. Isso se deve ao fato de que as mídias utilizadas se relacionam de forma a determinar uma ordenação entre elas, proporcionando ao leitor uma apresentação de melhor qualidade e com mais flexibilidade.

Pode-se classificar as relações de Sincronização Temporal de duas formas: as internas a um componente, ou intramídia, e as entre componentes, ou entre-mídias.

- Intramídia: refere-se às relações de tempo entre várias unidades de apresentação de uma única mídia contínua[4]. Como exemplo deste tipo de sincronização podem ser citados os quadros de uma seqüência de vídeo, onde cada quadro corresponde a uma unidade de apresentação do vídeo que possui uma ordem para exibição.

- Entre-mídias: refere-se à sincronização entre dois ou mais componentes multimídia distintos, dependentes ou não do tempo. Um exemplo de sincronização entre-mídias é o "*LIP-SYNC*"[5] entre áudio e vídeo; ou ainda, em se tratando de uma sincronização não tão rígida, tem-se a apresentação de três *slides* com tempos individuais de 10 segundos (Figura 2.2). No quarto segundo de cada um deles, começa a ser apresentado um texto explicativo, que permanece no cenário até o final do período da apresentação do respectivo *slide* [GAS 98].

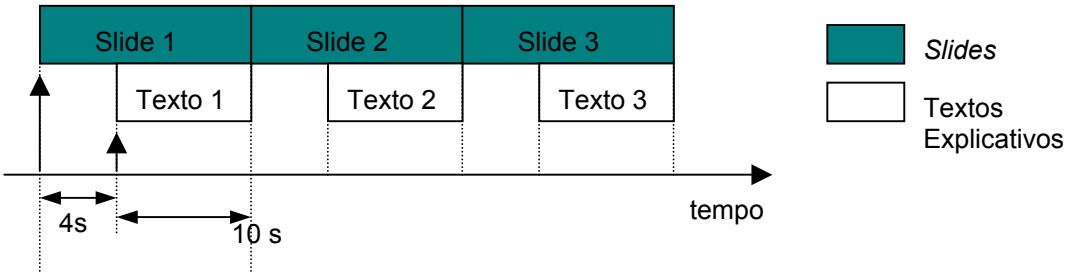


FIGURA 2.2 - Sincronização de Componentes Independentes do Tempo [GAS 98]

O presente trabalho tem como enfoque principal o estudo da sincronização temporal multimídia, abordando principalmente aspectos relacionados aos modelos utilizados para definição deste tipo de apresentação.

[4] Mídias contínuas são aquelas dependentes do tempo [RUB 96].
[5] *Lip-Sync* é a sincronização labial, refere-se à sincronização da voz falada com o movimento dos lábios da pessoa que está falando [NIC 90].

2.3 O Problema da Sincronização

A tarefa de sincronizar uma aplicação multimídia, de forma a satisfazer as relações temporais existentes entre as mídias e manter a consistência da apresentação, apresenta algumas dificuldades como, por exemplo, a ordenação da apresentação, a duração das mídias, a interação com o usuário e os dispositivos físicos utilizados pelo leitor.

Embora seja possível elaborar uma apresentação multimídia definindo com antecedência a duração absoluta das mídias envolvidas, muitas vezes o processo de autoria torna-se muito trabalhoso ou mesmo impraticável, principalmente quando se estende o uso da aplicação para a *World Wide Web* (WWW). Algumas razões para se evitar a definição da duração absoluta das mídias em uma apresentação são:

- Redes de computadores: o congestionamento da rede de transmissão de dados afeta diretamente a velocidade de execução da apresentação.
- Máquina utilizada pelo leitor[6]: a velocidade de apresentação de uma mídia está relacionada à máquina utilizada pelo leitor. Em máquinas mais lentas, a apresentação poderá sofrer um retardo.
- Interação com o usuário: o autor não pode pré-determinar o tempo utilizado pelo usuário para interagir com a aplicação. Observa-se então, a ocorrência de um intervalo com duração indefinida. Muitas vezes a duração é estimada, desfavorecendo usuários que por algum motivo ultrapassem esses limites.

Visando satisfazer tais especificidades foram propostos alguns modelos de sincronização, tais como: *Timeline*, *Script*, Baseado em Eventos, Baseado em Intervalos, Modelo Avançado Baseado em Intervalos, Expressões Intervalares, etc. Esses modelos devem considerar alguns requisitos existentes para o trabalho com sincronização em ambientes multimídia [COU 96]:

- suportar a consistência de objetos (seção 2.4) e a manutenção das especificações de sincronização;
- fornecer abstração do conteúdo do componente multimídia, de forma a possibilitar a referência a uma parte dele, assim como sua manipulação como uma única unidade lógica;
- possibilitar a sincronização baseada na interação com o usuário;
- descrever todos os tipos de relações de sincronização de forma fácil;
- possibilitar a integração de componentes dependentes ou independentes do tempo;
- possibilitar e suportar o trabalho de sincronização em níveis hierárquicos, para tornar possível o trabalho com cenários de sincronização grandes e complexos.

[6] Os usuários de apresentações multimídia são chamados leitores [BEI 96].

2.4 Consistência da Apresentação

O projeto de sincronização de uma aplicação multimídia requer atenção especial por parte do autor, uma vez que na definição de um documento são adicionadas informações que introduzem novos objetos de mídia ou que especificam novas restrições temporais. Mas, a modificação de um documento requer que seu autor assegure a consistência da apresentação e que a inclusão de novos objetos e relações introduzam restrições temporais adicionais.

Segundo [KER 95], as inconsistências são classificadas de duas formas:

– Inconsistência Qualitativa: ocorre quando a inconsistência é resultante de relações independentes da duração dos objetos utilizados (Figura 2.3).

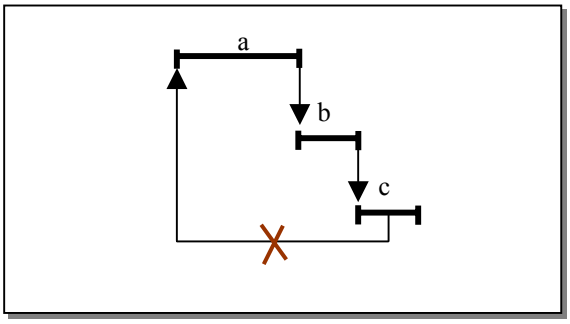


FIGURA 2.3 - Exemplo de inconsistência qualitativa [KER 95]

De acordo com o exemplo da Figura 2.3, dadas três mídias “a”, “b” e “c”; e supondo-se que as relações existentes entre elas sejam as seguintes:

1. a *meets* b – o final da apresentação da mídia “a” inicia a apresentação da mídia “b”;
2. b *meets* c - o final da apresentação da mídia “b” inicia a apresentação da mídia “c”;
3. c *overlaps* a – a mídia “c” é apresentada durante a apresentação da mídia “a”.

A especificação é **inconsistente qualitativamente**, pois é produto de operações independentes das durações dos objetos por ela relacionados. Dessa forma, a inconsistência é determinada na condição 3 (*c overlaps a*), que propõe a apresentação da mídia “c” durante a mídia “a”, contradizendo a condição 2 (*b meets c*), que estabelece a execução da mídia “c” somente após as mídias “a” e “b” [KER 95].

– Inconsistência Quantitativa: ocorre quando a inconsistência é resultante de relações dependentes da duração dos objetos por ela relacionados (Figura 2.4). Dessa forma, especificações consistentes qualitativamente, podem ser incompatíveis em relação às durações explícitas dos objetos relacionados, ou seja, inconsistentes quantitativamente [KER 95].

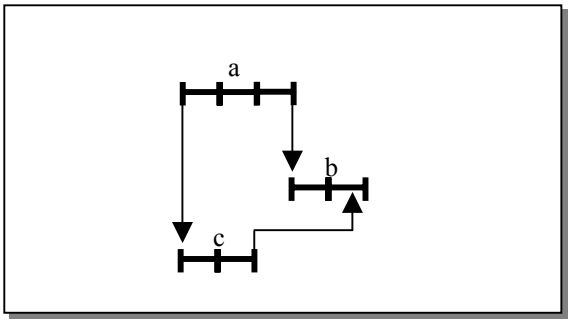


FIGURA 2.4 - Exemplo de inconsistência quantitativa [KER 95]

De acordo com o exemplo da Figura 2.4, dadas três mídias "a", "b" e "c"; e supondo-se que as relações existentes entre elas sejam as seguintes:

1. a *meets* b - o final da apresentação da mídia "a" inicia a apresentação da mídia "b";
2. a *starts* c – a apresentação da mídia "a" inicia a apresentação da mídia "c";
3. c *overlaps* b - a mídia "c" é apresentada durante a apresentação da mídia "b".

A especificação é inconsistente quantitativamente, caso as durações das mídias sejam, por exemplo, (a)=3 e (c)=2 unidades de tempo, pois para que a última relação (*c overlaps b*) seja satisfeita, o tempo de duração da mídia "c" deve ser maior que o tempo da mídia "a" [KER 95].

A manutenção da consistência da apresentação, muitas vezes é comprometida quando estendemos seu uso para a Web. Isso se deve ao fato de ocorrerem atrasos intramídia e inter-mídia durante a transferência dos dados, e que nem sempre são previstos pelos modelos utilizados na sincronização e/ou suportados pelas ferramentas de exibição. Por exemplo, quando ocorrer um atraso durante a transferência dos dados de uma mídia pertencente à determinada sincronização, as demais mídias relacionadas devem aguardar até que esta esteja disponível para apresentação.

O capítulo seguinte é responsável por apresentar os principais modelos utilizados para o trabalho com sincronização multimídia, assim como por especificar suas principais características, vantagens e desvantagens.

3 Modelos para Especificação de Sincronismo

Os modelos para especificação de sincronismo são mecanismos destinados a coordenar a ordenação temporal dos eventos relacionados à aplicação [LIM 97]. Atualmente, os trabalhos com sincronização multimídia têm se intensificado, e alguns modelos foram definidos visando apresentar caminhos para especificação de cenários temporais.

De acordo com [HAR 93] e [BUC 93], pode-se destacar os seguintes Modelos de Tempo: *Timeline*, S*cript*, Baseado em Eventos e Baseado em Intervalos. Para uma maior abrangência deste estudo, decidiu-se acrescentar dois outros modelos: Avançado Baseado em Intervalos e de Expressões Intervalares, propostos por [WAH 94] e [DUD 95] respectivamente. Desta forma, a seguir serão especificados alguns termos utilizados no presente trabalho:

- Início da Apresentação (IA): instante definido pelo autor para o início da apresentação multimídia.

- Fim da Apresentação (FA): instante definido pelo autor para o término da apresentação multimídia.

- Duração da Mídia (DM): duração real de apresentação de uma mídia, independente de fatores externos a mesma, tais como: velocidade da rede, interação com o usuário e máquina em que o usuário está executando a aplicação. Por exemplo, um vídeo que tem duração de 10 segundos, terá a sua DM=10seg., mesmo que a rede esteja congestionada, pois este fator não é considerado no cálculo.

- Duração da Apresentação (DA): duração da apresentação da mídia, dependente dos fatores externos anteriormente citados. Esse valor pode ser estimado pelo autor da aplicação. Por exemplo, para um vídeo com duração de 10 segundos, trafegando em uma rede congestionada, pode-se definir que DA=13seg., pois o autor estimou um atraso de 3 segundos na rede. Logo,

$$DA=DM+Atraso$$

Onde "Atraso" é um valor estimado pelo autor visando possibilitar a execução completa da apresentação, mesmo que ocorram problemas de rede e de periféricos. Vale ressaltar que nem sempre este valor será obedecido. Nesse caso, o autor deverá propor alternativas de solução quando os valores definidos forem ultrapassados.

Os modelos apresentados nessa seção foram analisados observando-se algumas características desejáveis para o trabalho com apresentações multimídia sincronizadas propostas em [COU 96], aliadas a outras necessárias para o trabalho com esse tipo de aplicação na Web. São elas:

- Consistência da apresentação: o modelo deve suportar a consistência de objetos e a manutenção das especificações de sincronização.

- Abstração do conteúdo: o modelo deve fornecer abstração do conteúdo do componente multimídia, de forma a possibilitar a referência a uma parte dele e também a possibilidade de manipulá-lo como uma única unidade lógica.
- Interação com o usuário: o modelo deve possibilitar a definição de sincronização baseada na interação com o usuário.
- Definição da sincronização: o modelo deve possibilitar a definição de relações de sincronização de forma fácil.
- Ferramenta para implementação: o modelo deve apresentar uma ferramenta que possibilite a implementação de aplicações baseadas no mesmo.
- Utilização na Web: o modelo deve possibilitar a definição de uma apresentação de boa qualidade a ser disponibilizada na Web e que atenda a algumas necessidades básicas, dentre elas destaca-se a geração de arquivos pequenos (devido ao fato dele ser transferido na rede).

3.1 Modelo *Timeline*

A principal característica do modelo é o posicionamento dos componentes da apresentação em relação ao eixo do tempo, chamado *timeline*. De acordo com esse modelo, os relacionamentos de sincronização não possuem vinculação relativa a outros componentes, mas estão vinculados de forma absoluta ao próprio eixo do tempo [GAS 98].

O Modelo *Timeline* foi um dos primeiros a ser utilizado para a definição de apresentações multimídia, sendo comumente usado por ferramentas de autoria, tais como *Director*, *MAEstro* e *Macromedia Flash*.

Um mecanismo de sincronização para a representação *timeline* deve interpretar a linha de tempo e executar as ações apropriadas como indicado nos instantes definidos para cada mídia.

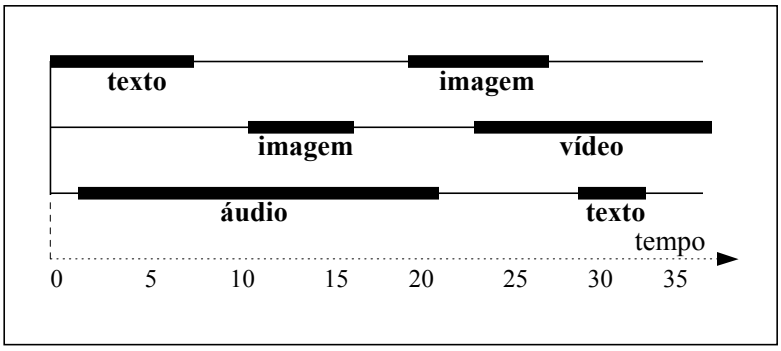


FIGURA 3.1 - Interface genérica dos sistemas que utilizam *timeline* [COS 96]

Na Figura 3.1 é apresentada uma composição temporal utilizando o Modelo *Timeline*, onde a linha mais espessa representa o tempo de apresentação de cada objeto de mídia.

Segundo [COS 96], [GON 98] e [GAS 98], podem ser citadas como vantagens no uso desse modelo:

- o decorrer do tempo é o responsável pela ativação das mudanças de comportamento de apresentação e pelo acionamento dos relacionamentos temporais definidos na aplicação;
- a facilidade de entendimento do modelo;
- a facilidade de manutenção, devido à independência mútua dos objetos que compõem a especificação;
- a possibilidade de referenciar partes do objeto na sincronização, ou seja, o modelo possibilita a definição da sincronização intramídia;
- a exclusão de um componente multimídia da linha temporal não interfere na sincronização estabelecida entre os componentes que restaram;
- a existência de excelentes ferramentas que possibilitem a implementação de aplicações para a disponibilização na Web.

Dentre os inconvenientes no uso do modelo, podem ser destacados [GON 98] e [GAS 98]:

- a necessidade de conhecimento prévio da duração absoluta das mídias envolvidas na apresentação;
- a consistência da apresentação pode ser comprometida quando ocorrem atrasos na transmissão dos dados multimídia pela rede;
- a representação de relacionamentos que dependem de interações com o usuário torna-se muitas vezes impossibilitada, uma vez que não se pode definir o tempo absoluto utilizado pelo mesmo para interagir com a aplicação.

A utilização do Modelo *Timeline* possibilita ao autor definir os tempos IA, FA, DM e DA. No caso do tempo FA, se for utilizado um comando "*Loop*[7]", a apresentação será finalizada ao término do número de *Loop's* definidos pelo autor.

3.2 Modelo Baseado em Eventos

O Modelo Baseado em Eventos é fundamentado na ocorrência de eventos de sincronização, que se caracterizam pela escolha de trechos nos documentos sujeitos à exibição, parada ou seleção, e representam as ações possíveis a serem realizadas sobre as mídias [GAS 98]. O sincronismo da apresentação é obtido através do relacionamento desses eventos. Sendo assim, os tempos definidos na seção 3 (IA, FA, DM, DA e Atraso), não são utilizados nesse modelo.

Dentre as vantagens na utilização do Modelo Baseado em Eventos, podem ser destacadas [GON 98]:

- a utilização de objetos sem tempo de apresentação fixo;
- a definição de mecanismos de ordenação em função do início e fim dos objetos;
- a facilidade na definição da interação com o usuário;
- a existência de ferramentas para a Web que se baseiam no modelo possibilitando a criação de apresentações de boa qualidade e a geração de arquivos pequenos.

[7] Procedimento ou conjunto de instruções que são executadas repetidamente até que uma condição específica seja satisfeita ou até que o programa seja concluído.

Os eventos definidos no HTML (***H**ypertext **M**arkup **L**anguage*) e apresentados na Tabela 3.1, são um exemplo de representação de interação com o usuário através do Modelo Baseado em Intervalos.

TABELA 3.1 – Eventos Intrínsecos do HTML

Evento	Função
Onclick	Testar click do mouse
Ondbclick	Testar click duplo do mouse
Onkeydown	Testar tecla pressionada
Onmousemove	Testar movimento do mouse

Como inconveniente no uso do modelo, cita-se a dificuldade de trabalho com cenários grandes, uma vez que o usuário pode ficar desorientado ao tentar manipular as especificações, tornando a criação e manutenção da apresentação uma tarefa bastante complexa [GAS 98]. Aliado a esse, está dificuldade na criação de cenários mais realísticos, onde não se pode conhecer todos os eventos possíveis.

Dentre os sistemas que utilizam o Modelo Baseado em Eventos, destacam-se o *Firefly* [BUC 93], o *CMIFed* [COS 96] e a Linguagem HTML.

3.3 Modelo de Sincronização Baseado em Intervalos

Segundo [ALL 83], um **intervalo** corresponde à duração da apresentação de um componente multimídia. Na especificação baseada em intervalos, a duração da apresentação de um objeto está relacionada com um intervalo de tempo. Dois intervalos de tempo podem ser sincronizados de treze maneiras diferentes, sendo que na Figura 3.2 serão demonstradas apenas sete, pois as outras seis são relações inversas às apresentadas. Por exemplo, *A before B* é a relação inversa de *B after A*.

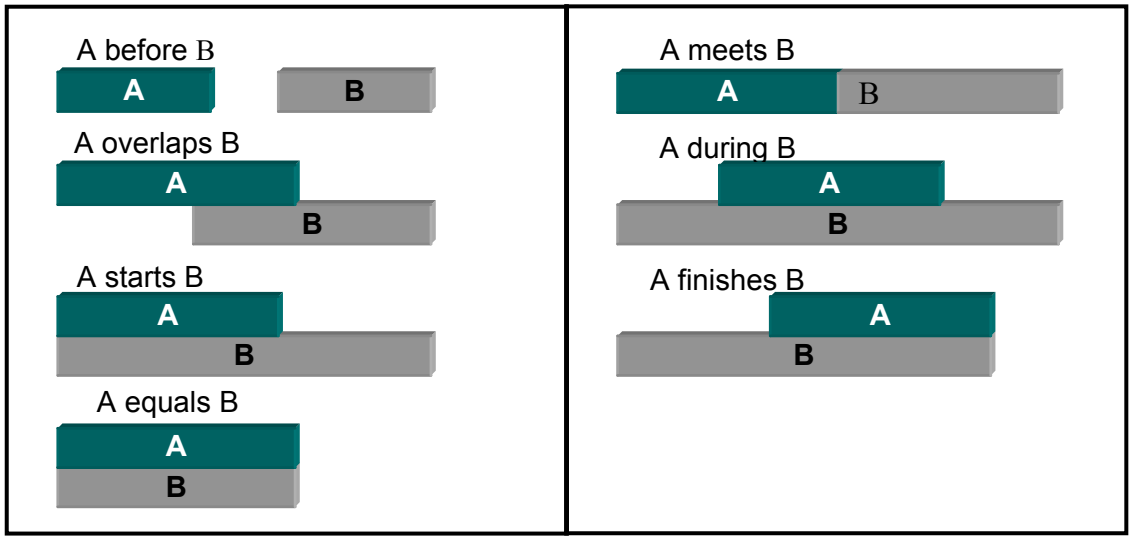


FIGURA 3.2 - Relações Temporais Binárias definidas em [ALL 83]

Assim como no *Timeline*, o Modelo Baseado em Intervalos necessita de um conhecimento prévio a respeito da Duração da Mídia (DM), pois a ocorrência de variações na DM podem ocasionar alterações nas relações estabelecidas entre as mídias (Figuras 3.3 e 3.4):

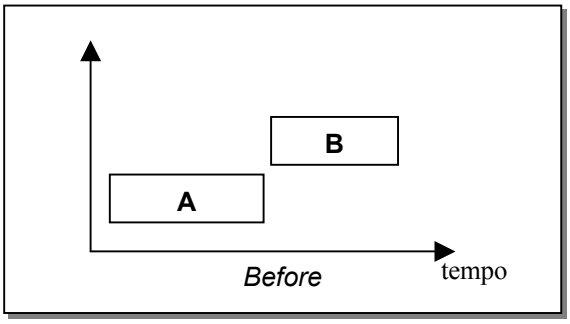


FIGURA 3.3 - Representação gráfica da relação *Before*

Caso a duração da mídia **A** seja aumentada, a relação passará a ser "*During*", caracterizando assim a ocorrência de uma inconsistência [KER 97].

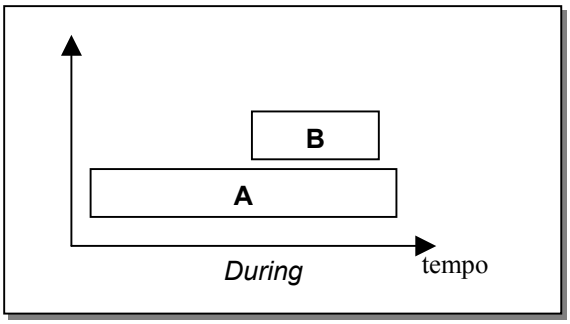


FIGURA 3.4 - Representação gráfica da relação *During*

Os demais tempos: IA, FA e DA não precisam necessariamente ser definidos pelo autor.

As relações definidas no Modelo Baseado em Intervalos podem ser utilizadas para a definição de apresentações simples. Mas, devido à necessidade de representação de apresentações mais complexas, Wahl [WAH 94] propôs o modelo estendido capaz de representar 29 relações distintas entre intervalos (seção 3.4).

3.4 Modelo Avançado de Sincronização Baseado em Intervalos

As treze relações binárias que compõem o Modelo Baseado em Intervalos serviram como base para a definição do Modelo Avançado de Sincronização Baseado em Intervalos, proposto por Wahl [WAH 94]. Visando expressar sincronizações mais complexas, Wahl sugeriu a criação de um conjunto composto por 29 relações possíveis entre intervalos de uma apresentação multimídia.

Segundo esse modelo, pode-se definir a duração da apresentação de objetos tanto de mídias contínuas, quanto de mídias discretas[8], assim como a definição da duração de interações com o usuário.

O modelo é bastante flexível e permite a especificação de apresentações sujeitas a diversas variações em tempo de execução [STE 95]. Mas, essa maleabilidade do modelo

[8] Mídias discretas são aquelas que independem do tempo como, por exemplo, a apresentação de textos e imagens estáticas [RUB 96].

possibilita a ocorrência de especificações inconsistentes. Um exemplo desse problema foi apresentado por Gaspary [GAS 98]:

Considere-se a existência de dois vídeos **A** e **B** relacionados com o operador "*not parallel*[9]". A apresentação de **B**, no entanto, está associada também a uma interação com o usuário através da relação "*before (0)*". No momento da execução desta aplicação, **A** está sendo apresentado. Quando a interação com o usuário é realizada, o vídeo **B** está apto para também ser apresentado, satisfazendo a relação "*before(0)*". Mas em contrapartida, não pode ser iniciado, pois infringiria a relação "*not parallel*" estabelecida por **A** (Figura 3.5-a). Os possíveis cenários consistentes são apresentados nas Figuras 3.5-b, 3.5-c.

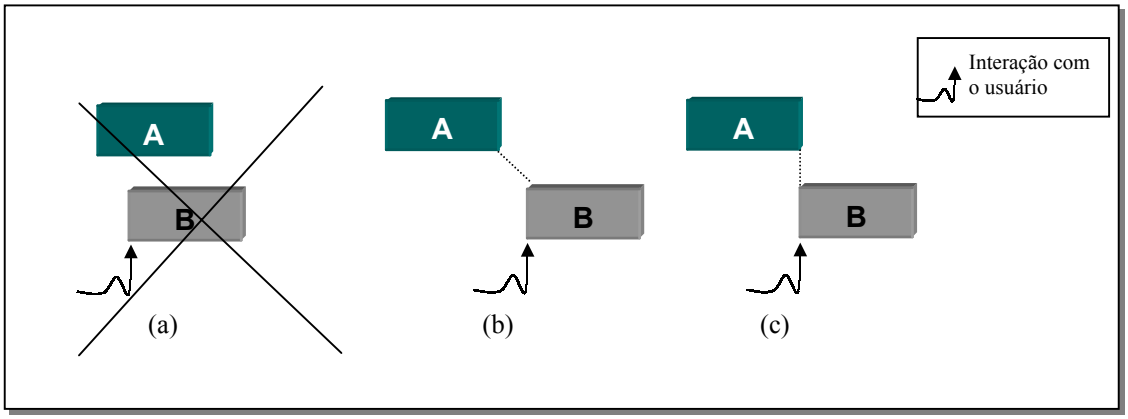


FIGURA 3.5 - Representação gráfica de inconsistência gerada pelo Modelo Avançado Baseado em Intervalos [GAS 98]

A linha pontilhada na Figura 3.5 está representada apenas para visualização da diferença dos cenários (b) e (c).

A utilização de um elevado número de relações, observada em apresentações complexas, caracteriza-se como uma desvantagem do modelo, pois compromete a legibilidade da sincronização especificada e propicia a geração de inconsistências.

Tendo em vista a simplificação do processo de especificação da sincronização, foram definidos dez operadores para a manipulação dessas relações, representados na Figura 3.6. A concatenação das vinte e nove relações em dez operadores é viabilizada pelo fato de vários casos poderem ser combinados em uma única operação [GAS 98].

[9] Este operador não foi definido nos modelos anteriormente estudados, mas ele determina que duas mídias não podem ser apresentadas em paralelo.

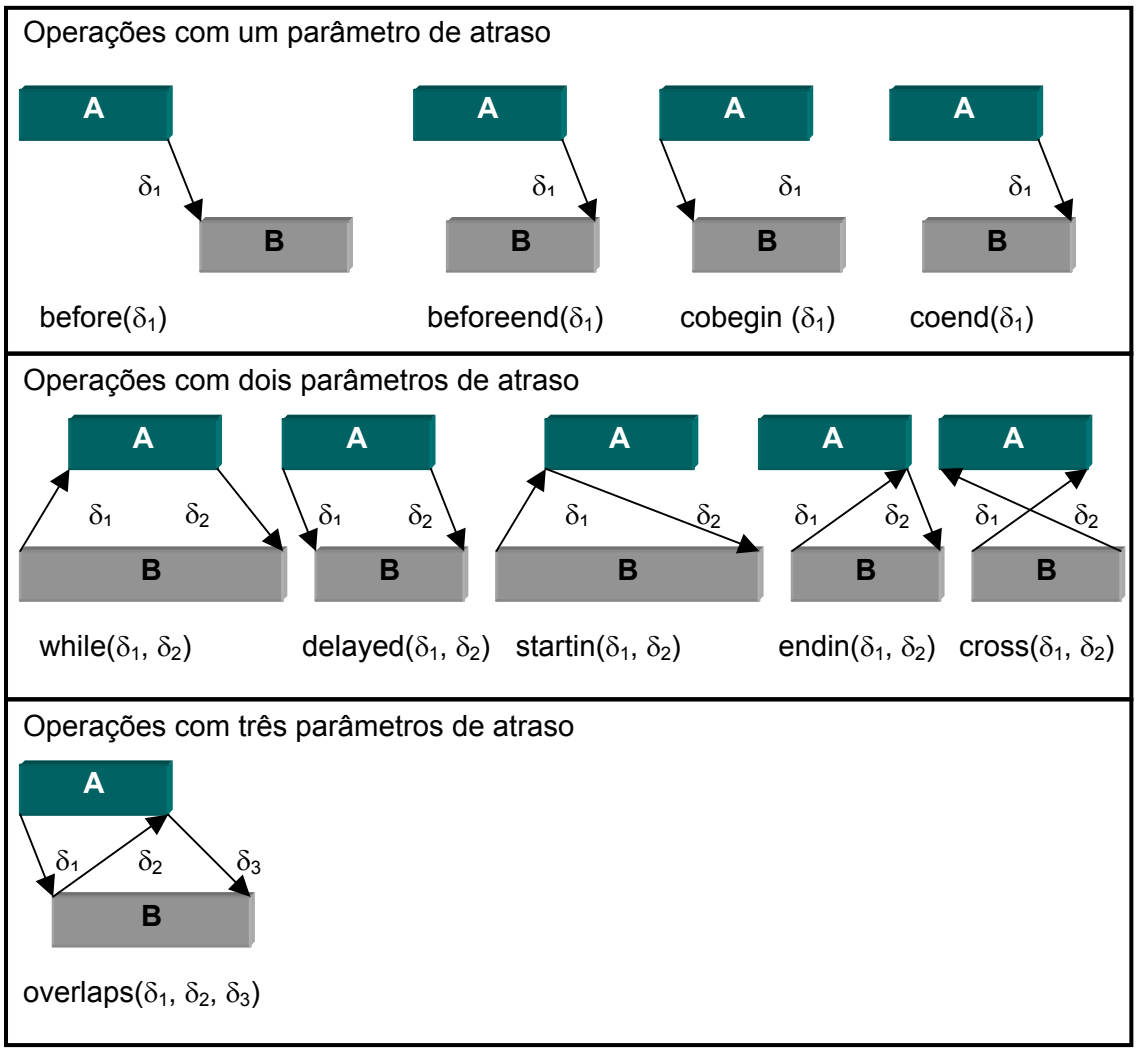


FIGURA 3.6 - Operadores do Modelo Avançado de Sincronização Baseado em Intervalos [WAH 94]

Onde δ representa o atraso aceitável para execução da mídia.

A Figura 3.7 mostra a utilização do Modelo Avançado Baseado em Intervalos para a definição de uma apresentação multimídia sincronizada [GON 98]:

```
Audio1 while(0,0)  Video
I1 before(0) I2
I2 before (0) I3
I3 before(0) Animation
Animation while(2,5) Audio2
```

FIGURA 3.7 - Implementação utilizando o Modelo Avançado Baseado em Intervalos

No exemplo demonstrado na Figura 3.7, observa-se a apresentação de um áudio "*Audio1*", enquanto o vídeo "*Video*" é apresentado. A exibição da imagem "*I1*" está relacionada à apresentação da imagem "*I2*" através de um operador "*before* (0)", que determinará a possibilidade de execução das duas imagens paralelamente. A imagem "*I3*" também está relacionada a "*I2*" e a uma animação "*Animation*" da mesma maneira. Enquanto essa animação está sendo executada, será apresentado o áudio "*Audio2*", que pode sofrer um atraso entre 2 e 5 unidades de tempo [GON 98].

Dentre as vantagens do uso desse modelo destacam-se [GAS 98] e [GON 98]:

- possibilidade de definição de mídias dependentes e independentes do tempo;
- boa abstração para o conteúdo da mídia;
- possibilidade de integração de objetos interativos.

Dentre as desvantagens no uso desse modelo destacam-se [GAS 98] e [GON 98]:

- especificação complexa;
- impossibilidade de referenciar sub-unidades do objeto, ou seja, é possível somente definir a sincronização entre-mídias e não a intramídias;
- geração de inconsistências devido ao não-determinismo em tempo de execução;
- inexistência de ferramentas que façam uso do modelo para o trabalho na Web.

3.5 Modelo *Script*

O Modelo *Script* baseia-se na definição da apresentação através de uma linguagem de especificação do seu comportamento, chamada *script*. Os elementos multimídia e a sincronização são especificados através de uma lista de comandos que irão definir como os objetos devem se comportar. Durante a apresentação, o sistema interpreta os *scripts* e executa as ações especificadas nos mesmos.

A utilização desse modelo é mais complexa devido ao fato da apresentação ser definida por intermédio de uma linguagem de programação. Mas, essa característica proporciona aos sistemas baseados no Modelo *Script* facilidades em baixo-nível para especificar componentes, temporizações, posicionamentos e interações em uma apresentação [GON 98].

Apresentações curtas podem ser facilmente construídas com essa estratégia. Mas, em se tratando de apresentações extensas, as tarefas de programação e manutenção podem se tornar complexas e difíceis de manusear, aumentando, consideravelmente, a possibilidade de geração de inconsistências.

O Modelo *Script* apresenta ferramentas que o implementam, inclusive para a Web, mas estas apresentam alguns inconvenientes, tais como a geração de arquivos grandes e de aplicações de baixa qualidade (fazendo-se uma comparação entre tamanho do arquivo e qualidade da apresentação).

Dentre os problemas encontrados no uso do Modelo *Script* podem ser citados [GAS 98]:

- pouca portabilidade das aplicações criadas;
- possibilidade de geração de inconsistências;
- dificuldade de realização da manutenção;
- complexa descrição da sincronização.

Segundo [GAS 98], o modelo é mais comumente utilizado em casos onde requisitos estritos de performance devam ser garantidos.

```
clear win
put background "pastel.pic"
put text "heading.txt" at 10,0
put picture "gables.pic" at 20,0
put picture "logo.pic" at 40,10
to handle ButtonClick
        get playsound "Mozart.wav"
        show field "Explica Música"
end ButtonClick
```

FIGURA 3.8 - Apresentação descrita através de um *script* [GAS 98]

No exemplo descrito na Figura 3.8, são definidas as sincronizações espacial e temporal de uma apresentação. Inicialmente são estabelecidas a cor do fundo, o posicionamento do texto "*heading.txt*", e das figuras "*gables.pic*" e "*logo.pic*". A seguir, é definida a função de um botão, que ao clique do usuário, acionará um áudio "*Mozart.wav*" juntamente com um campo de texto "*Explica Música*".

A utilização do modelo exige que o autor defina o início da apresentação (IA), enquanto os demais tempos FA, DM e DA podem ou não ser pré-definidos. No caso do tempo DM, o autor deve ter um conhecimento prévio a respeito da mídia para que a sincronização temporal seja estabelecida corretamente.

Como exemplos de sistemas que fazem uso desse modelo destacam-se o *Metacar*, o *Director* e o *Toolbook*, sendo que os dois últimos possibilitam a definição de aplicações para disponibilização na Internet.

3.6 Modelo de Expressões Intervalares

O modelo proposto por Duda e Kéramane [DUD 95] para composição da estrutura temporal de apresentações multimídia é baseado em operadores binários para representar intervalos de duração indefinida. Um operador tem dois intervalos como argumento e gera um outro intervalo como resultado:

Intervalo **operador** *Intervalo* ⟶ *Intervalo*

As entidades elementares do modelo são os **segmentos de mídia**, que apresentam comportamento temporal próprio. Esses segmentos de mídia são vistos pela composição temporal como **intervalos de tempo**, independentemente de seu conteúdo ou outras características.

Dessa forma, um intervalo de tempo a é definido pelos pontos delimitadores ($\underline{a} \leq \bar{a}$), com $a=\{t \ / \ \underline{a} \leq \bar{a}\}$. A duração de um intervalo a é ($\bar{a} - \underline{a}$) e depende do tipo de segmento de mídia [KER 97]. No Modelo de Expressões Intervalares são definidos os seguintes tipos de segmento de mídia [KER 97]:

⇒ **Conjunto *M***: abrange objetos multimídia tradicionais, tais como: áudio, vídeo, animação e texto. A duração desses objetos pode ser modificada durante a apresentação. O instante de ativação de uma mídia é determinado pela aplicação, e o de final é determinado pela duração do segmento.

⇒ **Conjunto *S***: abrange fluxos de mídia com alimentação, fluxos de teleconferência ou apresentações complexas que necessitem de conexão com a rede. A duração dos segmentos, o instante de sua ativação e o final da apresentação são *a priori* desconhecidos e, consistem em recursos de dados assíncronos.

⇒ **Conjunto *P***: abrange os programas. Quando ativado o programa inicia sua execução. O final da apresentação corresponde ao final da execução.

⇒ **Conjunto *L*:** abrange os *links*. Um *link* referencia outro segmento de mídia que é ativado quando o *link* é ativado. O instante de ativação é determinado pela apresentação.

⇒ $\{\varnothing\}$: corresponde a um segmento vazio.

Diante desses diferentes tipos, um segmento de mídia é definido como:

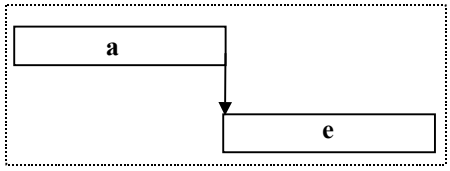
$$I_{base} = M \cup S \cup P \cup L \cup \{\varnothing\}$$

Duda e Kéramane [DUD 95] ainda prevêem um conjunto de operadores funcionais em alto nível que definem uma composição temporal multimídia. Esses operadores estão representados na Tabela 3.2:

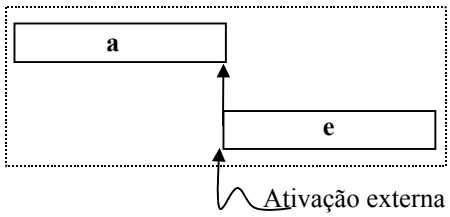
TABELA 3.2 - Representação dos operadores definidos em [DUD 95]

Operador	Função
Seq (a, e) → (a, ē)	O final do intervalo "a" inicia o intervalo "e".
Follow (a, e) → (a, ē)	O início do intervalo "e" termina o intervalo "a", sendo que o intervalo "e" é ativado externamente.
Par-begin (a, e) → (a, ā)	O início do intervalo "a" inicia o intervalo "e".
Par-end (a, e) → (a, ā)	O final do intervalo "a" termina o intervalo "e".
Par-min (a, e) → (a, min (ā, ē))	O início do intervalo "a" inicia o intervalo "e", o final da apresentação é determinado pelo intervalo que terminar primeiro.
Par-max (a, e) → (a, max (ā, ē))	O início do intervalo "a" inicia o intervalo "e", o final da apresentação é determinado pelo intervalo que terminar por último.
Equal (a, e) → (a, ā)	O intervalo "a" inicia e termina o intervalo "e".
Ident (a, e) → (a, ē)	O início do intervalo "a" determina o início do intervalo "e", e o final do intervalo "e" determina o final do intervalo "a".
Alternative $((a_1, e_1), \ldots, (a_n, e_n)) \rightarrow (a_1, \emptyset)$	Define um operador no qual os intervalos: a_i, i=1,…,n não são ativados em paralelo; o primeiro destes intervalos que terminar desativa os outros.
Loop a → (a, ∝)	Define um intervalo ou a execução indefinida que não pode ser parada pelos intervalos externos.

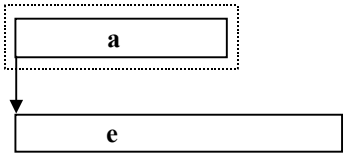
A Figura 3.9 representa graficamente alguns dos operadores descritos por Duda e Kéramane [KER 97a]:



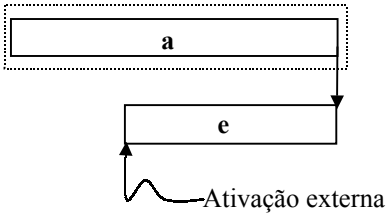
seq(a,e)



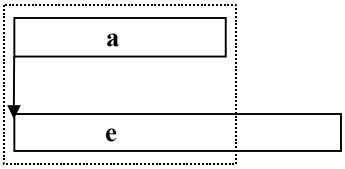
follow(a,e)



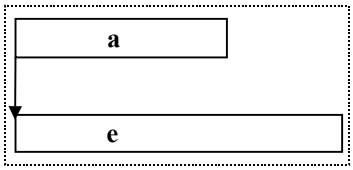
par-begin(a,e)



par-end(a,e)



par-min(a,e)



par-max(a,e)

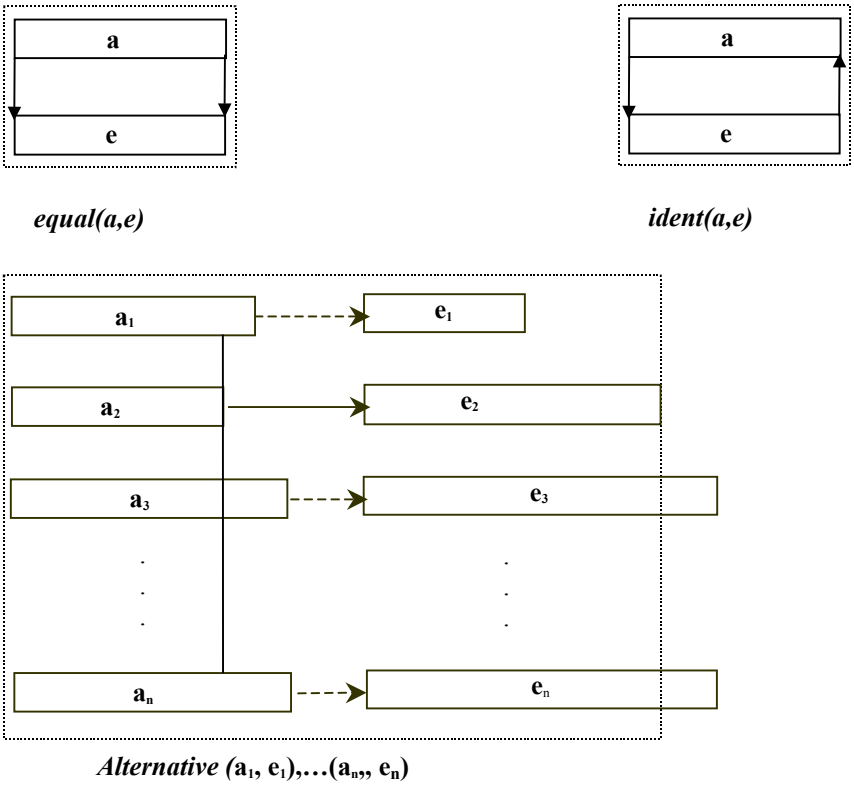


FIGURA 3.9 - Representação gráfica dos operadores do Modelo de Expressões Intervalares

Nas Figuras 3.9 e 3.10, as linhas pontilhadas representam o intervalo resultante da composição.

O exemplo demonstrado na Figura 3.10 se baseia num sistema que implementa um roteiro guiado a um museu. O passeio começa com um fluxo de vídeo *online* vindo de uma câmera localizada numa sala principal. Um programa de análise de conteúdo de vídeo é executado em paralelo com o fluxo de vídeo e tem como função controlar o fluxo por meio de um operador "*ident*". Quando as duas portas são detectadas, o programa pára e ativa dois outros programas que detectam interações com o usuário – quando o usuário clica na imagem de uma das portas, o *click* termina o programa correspondente àquela porta. O operador "*alternative*" associa diferentes segmentos de mídia a cada programa de detecção de *click*. Os segmentos de mídia são fluxos ao vivo vindo de câmeras que mostram as salas atrás das portas. O usuário poderá ver a sala desejada, clicando na porta correspondente.

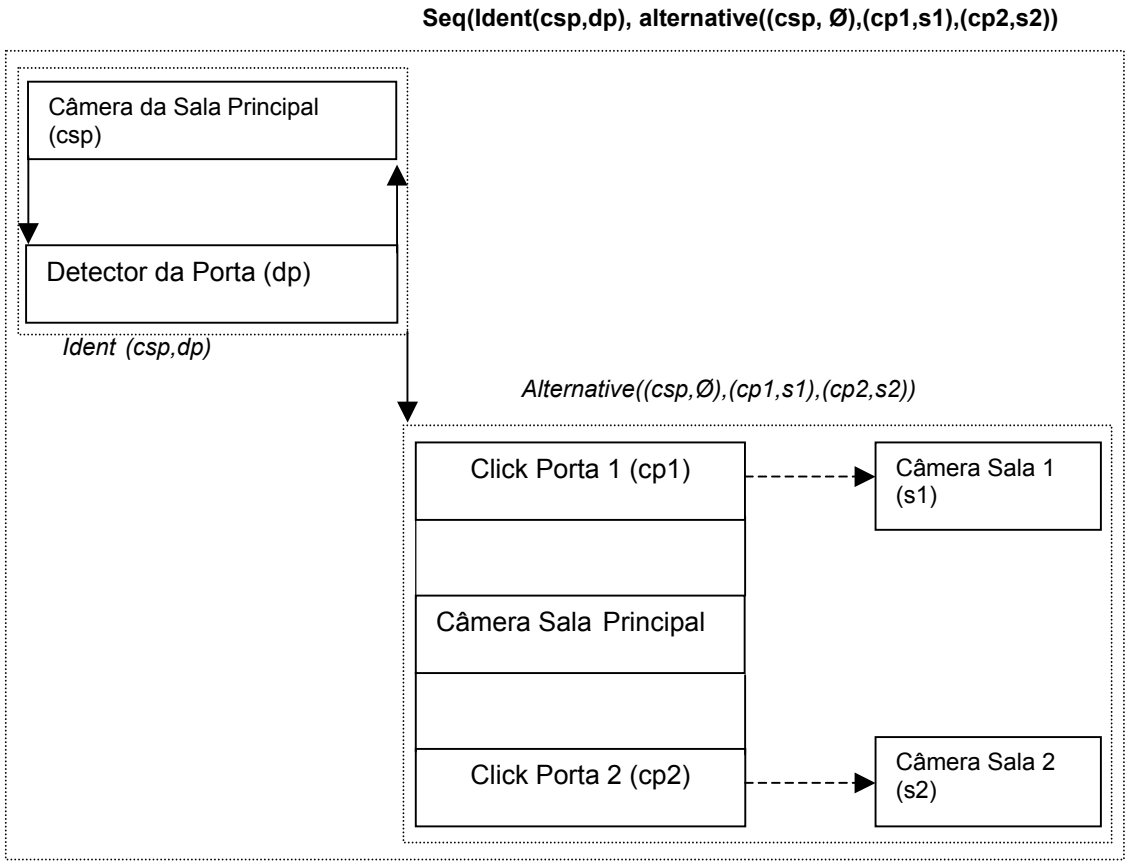


FIGURA 3.10 - Composição temporal de um roteiro guiado a um museu utilizando Expressões Intervalares [KER 97]

Dentre as vantagens desse modelo destacam-se [DUD 95]:

- a estrutura definida por uma Expressão Intervalar é invariante com respeito à duração dos intervalos componentes, ou seja, caso a duração da apresentação de algum objeto seja modificada, o esquema definido para a apresentação permanece o mesmo;
- a consistência da apresentação é garantida, mesmo que ocorram atrasos intramídia e entre-mídias;
- o modelo possibilita a definição de sincronização entre-mídias;
- o modelo apresenta ferramentas para Internet que possibilitam a definição de apresentações de ótima qualidade gerando arquivos pequenos.

O uso de Expressões Intervalares restringe as classes de cenários que podem ser expressas, visando sempre manter a consistência da composição, mas força uma estruturação metodológica e o encapsulamento [KER 95].

Os intervalos que compõem o encapsulamento podem ser especificados em termos de objetos de mídia elementar ou como blocos de mídia. Dessa forma, uma apresentação que usa Expressões Intervalares é composta por blocos de mídia aninhados e sincronizados interna e externamente [KER 95].

Segundo [DUD 95], a estruturação de uma apresentação multimídia utilizando Expressões Intervalares favorece o reuso de sub-componentes, enquanto o encapsulamento favorece a definição de cenários maiores e mais complexos, uma vez que trabalha com expressões intervalares aninhadas.

O capítulo seguinte é responsável por apresentar as Ferramentas de Autoria para Apresentações Multimídia Sincronizadas que se baseiam nos modelos estudados, salientando as principais características, vantagens e desvantagens de sua utilização.

4 Ferramentas de Autoria para Apresentações Multimídia Sincronizadas

A criação de uma apresentação multimídia sincronizada exige do autor habilidades para criar e manipular o conjunto de informações que a compõem, assim como a conveniente definição da sincronização das mídias referenciadas. A implementação desse tipo de aplicação exige do autor o conhecimento técnicas de linguagens de programação, criação e compactação de mídias, ferramentas de autoria[10] disponíveis e modelos que essas utilizam, assim como exige a definição de um projeto que especifique a apresentação e a sincronização das mídias utilizadas.

Segundo [LAD 96], pode-se identificar duas etapas no processo de autoria:

1. Autoria em ponto grande: nessa etapa são definidos a organização geral da aplicação e seu comportamento. Além disso, são identificados os aspectos estruturais, não se levando em consideração detalhes de implementação.

2. Autoria em ponto pequeno: nessa etapa são definidos os detalhes da organização e do comportamento da aplicação. Essa fase apóia o autor na construção da rede completa de nodos, suas ligações e na definição da apresentação dos conteúdos, evidenciando todos os caminhos percorridos.

A sincronização da apresentação multimídia é trabalhada nas duas fases do processo de autoria.

Sendo assim, o projeto da apresentação e a utilização de ferramentas de autoria tornam o trabalho de definição dos sistemas multimídia mais eficaz e simplificado, além de agilizar o processo de criação e, em alguns casos, possibilitar que autores sem profundo conhecimento de informática desenvolvam suas aplicações. Dentre essas ferramentas destacam-se: CMIF, *Macromedia Flash*, *Assimetrix Multimedia Toolbook* e Linguagem SMIL. Tais aplicações ainda podem ser implementadas através do uso de outras linguagens de programação como, por exemplo, C, Java e Pascal, mas que pelas facilidades das ferramentas de autoria não são comumente utilizadas.

No trabalho com Apresentações Multimídia Sincronizadas, sobretudo levando em consideração os modelos estudados (seção 3), o conjunto de sistemas de autoria fica mais restrito, uma vez que cada um deles trabalha com um ou no máximo dois dos modelos citados. Sendo assim, o autor da apresentação multimídia sincronizada deve ter conhecimento do modelo no qual a ferramenta se baseia antes de dar início a sua utilização, pois a adoção de um desses influencia diretamente na escolha da ferramenta empregada para a implementação.

---

[10] Sistemas de Autoria são ambientes onde o usuário pode criar um *software* multimídia sem ter que dispor de um conhecimento aprofundado de programação [MAR 92]. Nesse trabalho utilizamos os termos Sistema de Autoria e Ferramenta de Autoria como sinônimos.

4.1 CMIF (CWI *Multimedia Interchange Format*)

O CMIFed (editor CMIF) foi desenvolvido pelo núcleo de sistemas multimídia do CWI (*Centrum voor Wiskunde en Informatica*), na Holanda, e se fundamenta no Modelo Baseado em Eventos. Esta ferramenta tem por objetivo prover aos autores de sistemas multimídia um ambiente para a visualização e a manipulação estruturada de aplicações. Neste trabalho será apresentado antes o modelo de dados do Editor CMIF para, em seguida, descrever as três visões existentes no processo de autoria.

4.1.1 O Modelo de Dados do CMIF

Um evento do modelo de dados do CMIF é definido como o menor fragmento de mídia que pode ser manipulado pelo sistema como, por exemplo, pequenos fragmentos de vídeo, áudio, imagem ou texto [COS 96].

Os eventos que compõem uma apresentação estão relacionados através de *timing constraints* (relacionamentos temporais), sendo utilizados para sua definição dois mecanismos [COS 96]:

- Composição paralela e seqüencial: representada pela construção de uma árvore de hierarquia da apresentação. O posicionamento dos nós dentro da árvore da apresentação possue relações de tempo, e sua estrutura define o comportamento temporal para exibição.

- Arcos de sincronização (*sinc-arcs*): representados pela relação entre marcadores em dois eventos de uma mesma apresentação. Os arcos de sincronização especificam valores de retardo na apresentação de mídias.

Segundo [GAS 98], a tarefa de autoria é realizada com a utilização de três visões:

- Hierárquica: provê ao autor meios para estruturação da aplicação, permitindo agrupar componentes ou definir estruturas completamente vazias para serem preenchidas posteriormente.

- Canais: permite descrever informações relativas à temporização e à utilização lógica de recursos por meio de uma linha temporal modificada. Informações sobre o fluxo de controle aparecem diretamente nessa vista, através da utilização de arcos que simbolizam descritores de restrição de sincronização.

- Execução: provê ao autor a possibilidade de verificar os resultados de sua especificação da maneira como será apresentada ao usuário final.

4.1.2 A Vista Hierárquica

A vista hierárquica é a janela mais importante no processo de autoria, provendo mecanismos para mostrar e manipular a estrutura da aplicação. Os nodos folha representam componentes, enquanto os nodos não-folha representam composições. A estrutura hierárquica é representada nessa vista como uma estrutura de blocos. Através dos dois tipos de composições suportadas (paralelo e seqüencial), é possível que o autor agrupe componentes a serem apresentados simultaneamente ou um após o outro. Não é necessário que o autor especifique as temporizações neste ponto, uma vez que essas são deduzidas a partir da estrutura hierárquica [GAS 98].

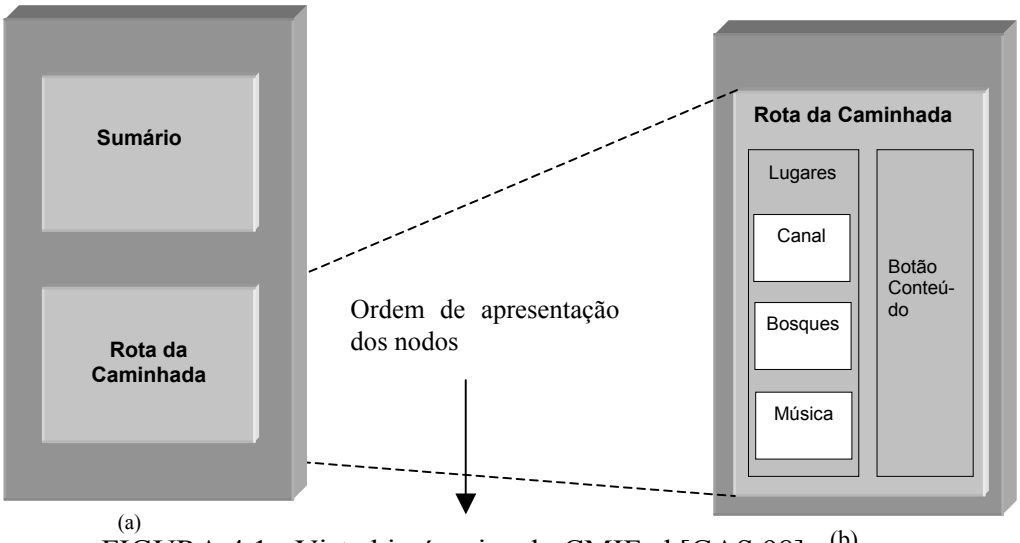


FIGURA 4.1 - Vista hierárquica do CMIFed [GAS 98]

(a) As caixas grandes indicam diferentes níveis de estrutura da aplicação. A caixa referente ao "Sumário" é apresentada antes da "Rota da Caminhada".
(b) Visão detalhada da cena "Rota da Caminhada". O nodo "Lugares" contém três filhos, cada um com uma estrutura aninhada já definida. A caixa à direita ("Botão Conteúdo"), representa uma mídia (nodo folha da estrutura hierárquica).

Na Figura 4.1, são especificadas duas caixas dentro do nodo "*Rota da Caminhada*" (*Lugares* e *Botão Conteúdo*) apresentadas em paralelo; e três caixas aninhadas dentro de "*Lugares*" (*Canal*, *Bosques* e *Música*), apresentadas seqüencialmente. A duração de um nodo composto é derivada pelo sistema. A duração de uma composição serial é a soma das durações dos nodos filhos e a de uma composição paralela é determinada pelo nodo filho de maior duração. Quando um nodo não tem duração explícita, ele é apresentado de acordo com a duração do nodo pai [GAS 98].

4.1.3 A Vista de Canais

A vista de canais fornece ao usuário uma visão baseada nos canais criados pelo autor durante o processo de definição de uma apresentação multimídia. A Figura 4.2 mostra os canais utilizados na apresentação "*time-mgmt*": "*Video*", "*Title Text*", "*Body Text*", "*Sound*" e "*Image*". O posicionamento dos eventos dentro dos canais está

relacionado aos momentos de suas ocorrências. Os eventos estão alinhados seqüencialmente no tempo dentro dessa vista. A ordem de suas disposições dentro dos canais no sentido vertical define o alinhamento temporal (de cima para baixo), ou seja, o evento "*shot 1*" ocorre antes do evento "*shot 2*", que ocorre antes do evento "*shot 3*" [COS 96].

Os relacionamentos temporais definidos na vista de canais são especificados através dos *sinc-arcs*, que podem ser criados e alterados dentro dessa vista. No exemplo da Figura 4.2, foi criado um *sinc-arc* do evento "*still*" do canal "*Image*", para o evento "*title*" do canal "*Title Text*", representado por uma seta conectando os dois eventos [COS 96].

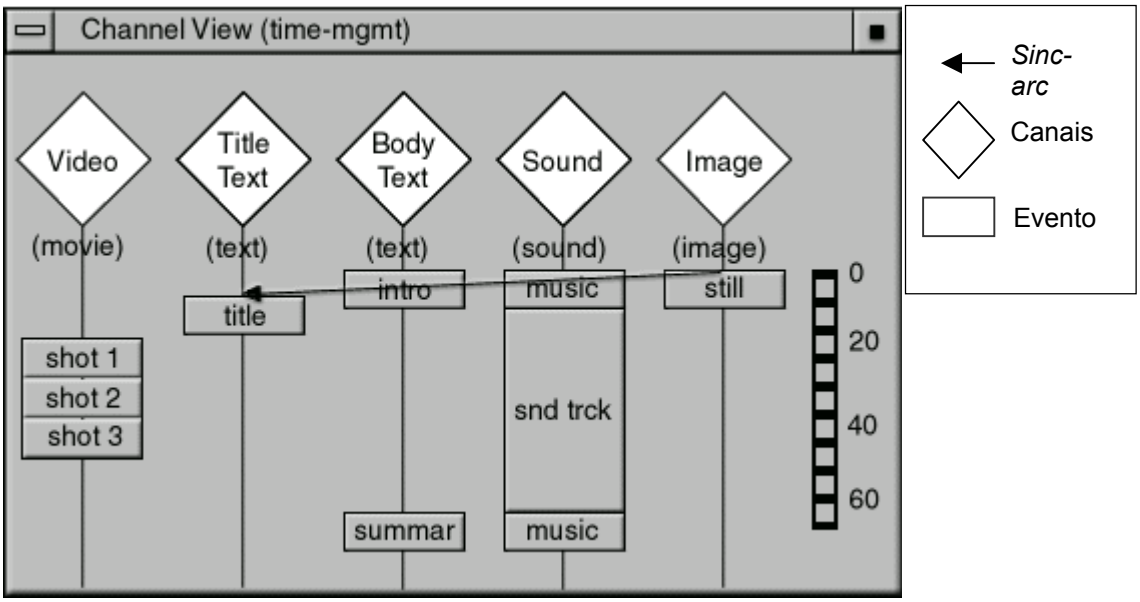


FIGURA 4.2 - A vista de canais do CMIFed [BUL 93]

4.1.4 A Vista da Execução

A vista de execução é fornecida pelo *player*[11] e permite que o autor possa visualizar a apresentação dentro de uma plataforma específica; ou ainda editar aspectos de *layout* da apresentação. Um exemplo desta vista é apresentado na Figura 4.3, onde o *player* possui uma janela de controle (canto inferior esquerdo), cuja função é controlar o tempo de exibição que está sendo mostrado na sua vista. O *layout* das janelas pode ser modificado, e as alterações de comportamento dos nós são armazenadas dentro dos canais das mídias [COS 96].

[11]Exibidor da apresentação.

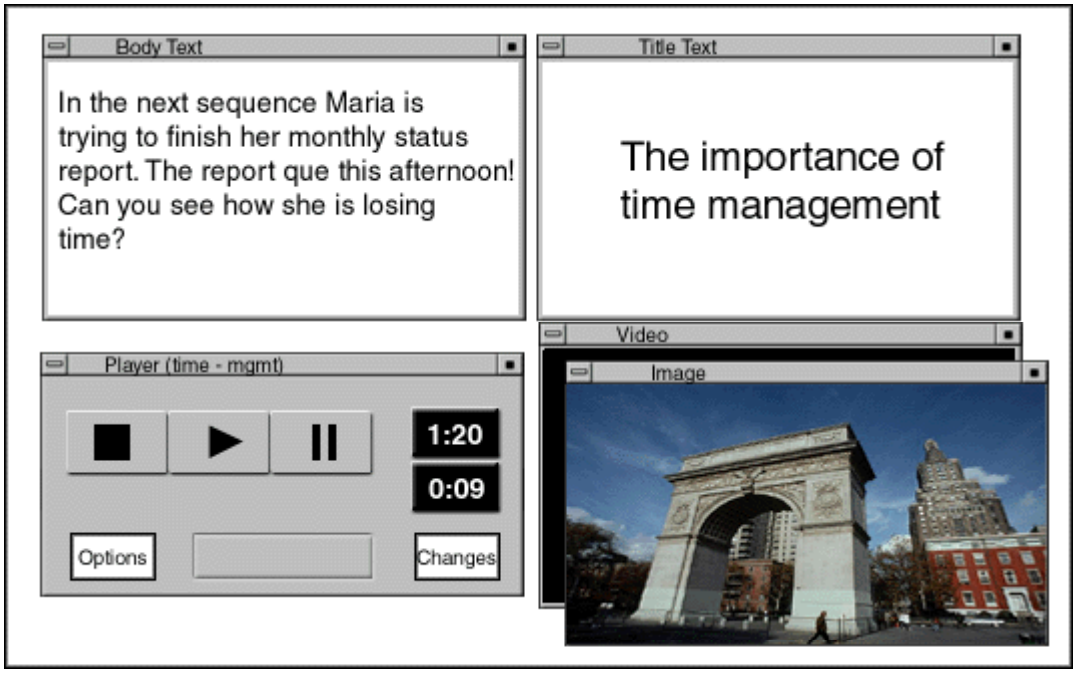


FIGURA 4.3 – Interface do CMIFed [BUL 93]

4.2 *Macromedia Flash*

O *Flash* foi criado pela empresa *Future Wave Technologies* e chamava-se *Future Splash*. Entretanto, no ano de 1997, a empresa foi comprada pela *Macromedia*, que incorporou à ferramenta vários recursos multimídia visando possibilitar o trabalho com imagens, áudio, sincronização e interatividade, passando a se chamar *Macromedia Flash*.

O *Macromedia Flash* é um sistema de autoria que integra animação gráfica, som e interatividade em páginas Web, apresentando ferramentas para edição de texto e criação de imagens e possibilitando criar animações de forma fácil e rápida, sem que o tamanho final dos arquivos seja muito grande. A metáfora utilizada por este ambiente é a de um filme dividido em cenas, que são divididas em camadas chamadas *layers* e essas são compostas por *frames*[12].

Assim como outros programas de ilustração gráfica, o *Flash* trabalha com imagens vetoriais, o que possibilita a definição de um desenho ou animação que pode ser visualizado em qualquer resolução sem perda de qualidade. Além disso, o trabalho com esse tipo de imagem faz com que o arquivo gerado seja muito menor, se comparado ao padrão utilizado na Internet (GIF ou BMP). Isso se deve ao fato de que a animação não é construída como uma seqüência de imagens que se sobrepõem, mas sim como a movimentação de objetos geométricos em uma linha de tempo.

A definição da sincronização com o *Macromedia Flash* é realizada baseada no modelo *Timeline* (seção 3.1), onde para cada camada (*layer*) é criada uma trilha através

[12]Representam os quadros da animação, cada quadro representa uma única posição da cena. Uma seqüência de quadros a uma determinada velocidade cria o efeito de uma animação.

da qual o autor definirá os instantes da sincronização. O conjunto destas trilhas vai, por sua vez, representar a sincronização da apresentação multimídia.

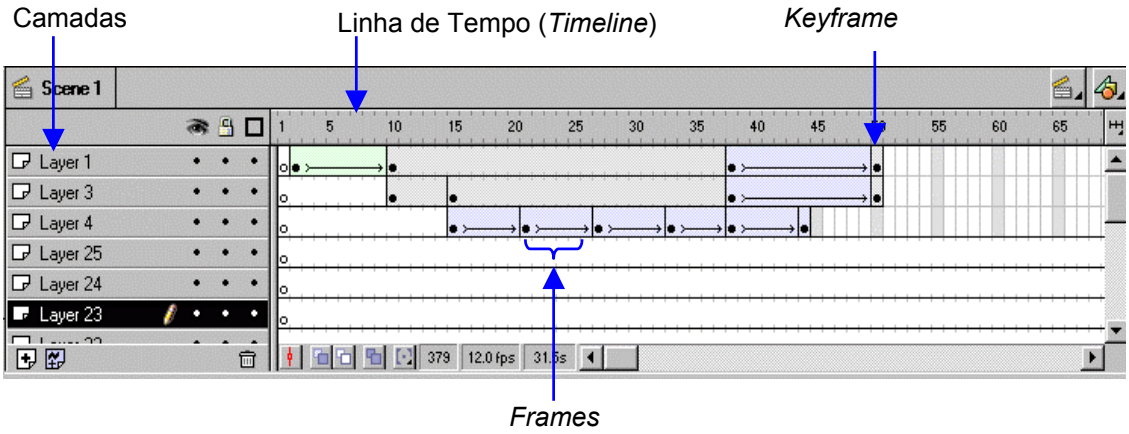


FIGURA 4.4 - *Timeline* do *Macromedia Flash* 4.0

A Figura 4.4 representa a sincronização de uma apresentação utilizando a *timeline* do *Flash*. Os pontos nos *frames* são chamados *keyframes* e representam *frames* modificados como, por exemplo, o final do movimento de uma linha ou a finalização de um som.

As linhas representam os *layers* e têm a função de ajudar a organizar as diferentes partes de uma cena. Cada camada da cena é referenciada por um *layer* para que, posteriormente, seja realizada a sincronização.

A Figura 4.5 apresenta um teste desenvolvido com o *Macromedia Flash* 4.0, onde se pode observar na *timeline* uma referência para cada objeto presente na cena 1 (*layer* 16, *layer* 7 e *layer* 12). Outros testes realizados com o Macromedia Flash são apresentados no Anexo 2.

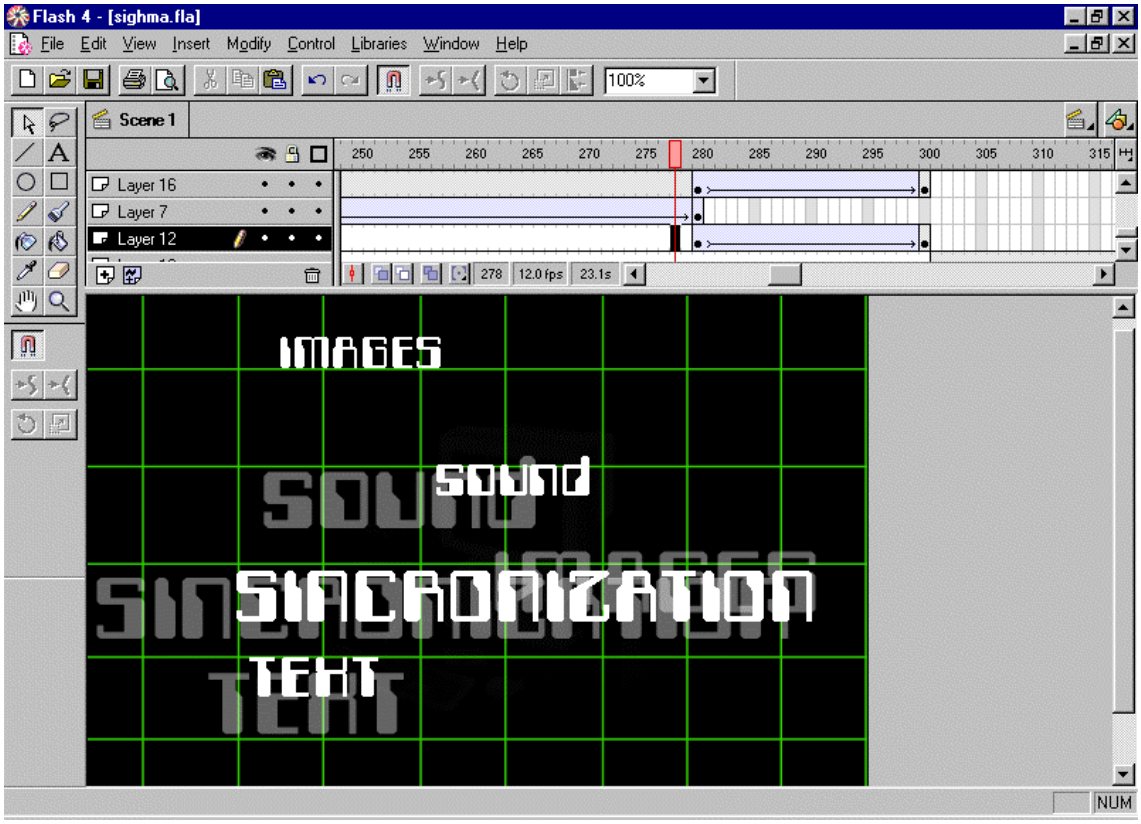


FIGURA 4.5 - Apresentação construída com o *Macromedia Flash* 4.0

Segundo [MAC 2000] e [CAS 98], o *Macromedia Flash* está sendo amplamente utilizado devido principalmente às seguintes características:

- Facilidade de uso: a criação de aplicações multimídia e a definição de sua sincronização podem ser realizadas por autores sem profundo conhecimento de programação.
- Tamanho dos arquivos: os arquivos gerados, mesmo contendo animações, são pequenos se comparados com outras ferramentas que possibilitam este tipo de trabalho.
- *Help* intuitivo: sistema de ajuda fácil, claro e que analisa dificuldades do autor, apresentando automaticamente a melhor solução possível.
- *Loop* dos objetos: possibilita ao autor definir o *loop* de objetos individualmente.
- Possibilidade de definição de aplicações que possibilitem a interação com o usuário.
- Flexibilidade da *timeline*: pode-se desviar a apresentação para qualquer parte da *timeline* sem ter, necessariamente, que passar pelos instantes intermediários.
- Uso em conjunto com outros sistemas de autoria: as aplicações definidas com o *Macromedia Flash* podem ser incorporadas a apresentações desenvolvidas com outras linguagens ou sistemas de autoria, tais como SMIL e HTML.

Dentre as desvantagens no uso do *Macromedia Flash*, podem ser destacadas [MAC 2000] e [CAS 98]:

- Biblioteca de objetos: a biblioteca de objetos é pobre, sendo necessária à utilização de outras ferramentas para a manipulação destes.
- Sincronizações em alguns casos ineficientes: quando o autor deseja, por exemplo, tocar sons de arquivos diferentes (no caso de um único arquivo ser muito grande), a seqüencialização é ineficaz, deixando nítidos os momentos de final de um arquivo e início de outro.
- Verificação de equipamento utilizado pelo usuário: o *Flash* faz uma verificação da máquina do usuário para identificar, por exemplo, a velocidade. Caso a máquina seja lenta, o *anti-aliasing*[13] é retirado, fazendo com que a apresentação seja visualizada mais rapidamente.
- Qualidade da apresentação: em alguns casos, mesmo com menor qualidade a apresentação permanece lenta caso a máquina seja lenta, perdendo assim a sincronização com o tempo, ou seja, uma apresentação que foi construída para ser apresentada em 5 minutos passaria a durar 10 minutos. Sendo assim, a consistência da apresentação pode ficar comprometida.

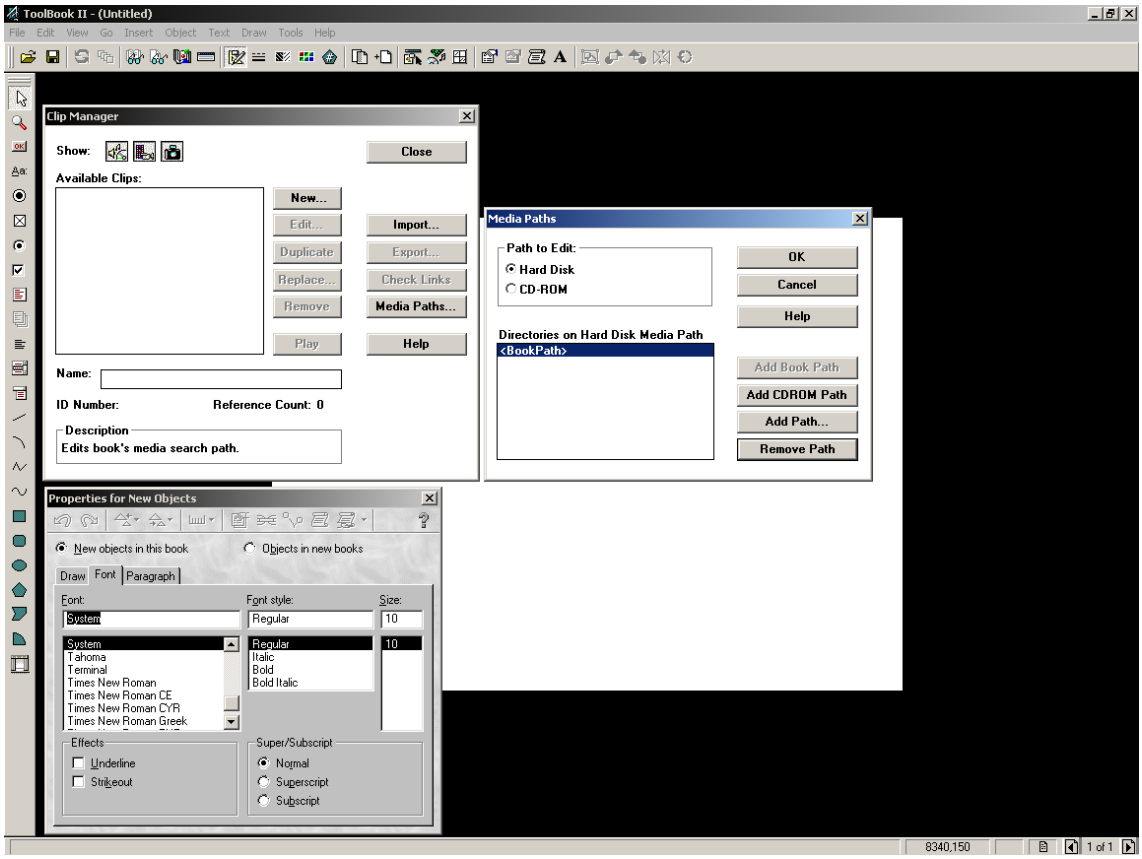
4.3 *Assimetrix Multimedia Toolbook*

O sistema de autoria *Toolbook* foi criado pela empresa *Assimetrix* e baseia-se no Modelo *Script* (seção 3.5). Essa ferramenta de autoria utiliza como metáfora o livro, que é composto por páginas e estas podem, por sua vez, apresentar diversas mídias, tais como: texto, imagens, vídeo e som.

As primeiras versões do *Assimetrix Multimedia Toolbook* foram criadas apenas para trabalho com sistemas multimídia armazenados em máquinas locais, sem o conceito de rede de computadores. Posteriormente, com a disseminação da Internet no meio computacional, seus fabricantes incorporaram funcionalidades que tornaram possível e até mesmo facilitaram o trabalho na Web. Um exemplo disso, foi o desenvolvimento de um *plug-in*[14] para a Internet chamado *Neuron* que pode ser utilizado nos *browsers*[15] *Netscape* e *Internet Explorer*.

---

[13] Suavização do serrilhado. O uso de *anti-aliasing* permite uma mais fácil integração de texto/imagens em uma figura, devido ao fato de misturar (na borda) as cores do fundo com o texto/imagem que inserir.
[14] Extensão do *browser* que facilita a visualização de textos, som, vídeo e maior interação com o usuário [GUI 2000].
[15] Programa de aplicação cliente que permite acessar, por meio de uma interface gráfica, informações diversas contendo textos, imagens, gráficos, sons, etc.



A definição de uma apresentação e sua sincronização através do *Toolbook* exige conhecimentos de programação por parte do autor, uma vez que essa definição será realizada através da linguagem *Openscript*[16].

Alguns problemas relacionados ao tratamento de filmes e áudio foram solucionados na versão do *Toolbook* 7.1, que também direcionou mais o trabalho à Internet possibilitando a geração de arquivos DHTML (*Dynamic HiperText Markup Language*), mas nesse caso restringindo o trabalho ao *Internet Explorer*.

Segundo [WIR 99], podem ser citadas como vantagens no uso desse sistema de autoria:

- interface amigável: o sistema oferece ao autor uma interface de fácil entendimento, apresentando teclas de controle para funções relacionadas à definição da aplicação e até mesmo de sincronizações simples sem a necessidade de utilização da linguagem *Openscript*;
- biblioteca de objetos rica: o conjunto de objetos disponíveis para incorporação em apresentações é grande e de boa qualidade;
- possibilidade de integração com outras linguagens de programação;
- interatividade com o usuário: o *Toolbook* permite a definição de apresentações sujeitas à interações do usuário;
- controle da sincronização: qualquer objeto da apresentação pode ser parâmetro para o controle da apresentação, ou seja, podemos definir uma sincronização baseada na ocorrência ou transição de um objeto da aplicação;

[16] Linguagem de programação baseada em *scripts* que é utilizada no *Toolbook*.

- integração entre texto e imagem: a definição da sincronização entre textos e imagens é fácil e de ótima qualidade;
- execução de controladores de sincronização: podemos ter vários objetos controladores da sincronização sendo executados ao mesmo tempo;
- sincronização em baixo-nível: a utilização da linguagem *Openscript* possibilita definir sincronizações em baixo nível, incorporando algumas características de outros modelos, tais como: controle de tempos das mídias (referente ao Modelo *Timeline*), e definição de sincronizações baseadas em eventos (referente ao Modelo Baseado em Eventos).

Dentre as desvantagens no uso desse sistema de autoria, podem ser destacadas [WIR 99]:

- implementação: a definição de apresentações mais complexas e de suas sincronizações são realizadas por meio de uma linguagem de programação, impossibilitando a utilização por usuários pouco especializados;
- tamanho dos arquivos: os arquivos finais gerados para as apresentações são normalmente grandes, acarretando problemas na disponibilização das mesmas no ambiente Web;
- trabalho com mídias: a ferramenta não apresenta muitas facilidades para manipulação de mídias;
- utilização na Internet: a ferramenta ainda apresenta algumas dificuldades e restrições para a disponibilização na Internet, principalmente relacionadas ao *browser* utilizado para visualização da apresentação.

4.4 A Linguagem SMIL (*Synchronized Multimedia Integration Language*)

A linguagem SMIL é uma *markup language*[17] desenvolvida em 1998 e recomendada pelo W3C (*World Wide Web Consortium*). Ela tem como principal objetivo o desenvolvimento de aplicações multimídia sincronizadas para Web, permitindo integrar um conjunto de objetos multimídia independentes, tais como: texto, áudio, imagens, vídeo e animações numa apresentação multimídia sincronizada, para disponibilização na Internet. Sendo assim, é possível controlar o conteúdo a ser exibido, as transições entre os mesmos e o *layout* de apresentação ao usuário.

As apresentações desenvolvidas com a Linguagem SMIL são, na realidade, documentos XML (*eXtensible Markup Language*) que podem conter uma declaração de tipo de documento (DTD) utilizada na implementação. Os documentos desenvolvidos através dessa linguagem são compostos basicamente por duas partes:

- **Cabeçalho (*Head*)**: contém a informação desassociada do comportamento temporal da apresentação e associada à sincronização espacial como, por exemplo, tamanho da tela e posição onde serão apresentadas as mídias. O cabeçalho contém a descrição da disposição da apresentação num cenário.
- **Corpo (*Body*)**: contém a informação associada ao comportamento temporal e relacional do documento, ou seja, a ordem de apresentação das mídias ao

[17] *Markup language* é um linguagem de marcação. Essa utiliza-se de marcadores chamados *tags* para identificar elementos distintivos em um documento [JAM 99].

usuário. O corpo contém a descrição da sincronização temporal dos objetos que fazem parte da aplicação.

4.4.1 Player

Os *players* SMIL são aplicativos-cliente que recebem e exibem as apresentações multimídia sincronizadas. Existem basicamente dois tipos de *players* disponíveis:

- os compatíveis com a linguagem SMIL e os formatos de multimídia padrão utilizados por essas apresentações;
- e, os que também trabalham com extensões de um fabricante específico.

Os *players* utilizados com mais freqüência são: *Realplayer* G2 (Figura 4.7), GriNS (*Graphical iNterface to SMIL*) e o HPAS (*Hypermedia Presentation and Authoring System*). As versões dos *browsers Netscape* (versão superior a 4.6) e *Internet Explorer* (versão superior a 5.0) também permitem a visualização de apresentações sincronizadas implementadas com a Linguagem SMIL [STA 99].

4.4.2 A Sincronização Espacial

A sincronização espacial permite ao usuário definir tamanho e posição da área de visualização, assim como das regiões internas à ela. Essa sincronização será definida no documento SMIL dentro da área de cabeçalho ou *head*. A linguagem de *layout* básica do SMIL funciona como folhas de estilo em cascata nível 2 (CSS2[18]), onde o autor

[18] A CSS2 (*Cascading Style Sheets* 2) é uma recomendação que segue e completa a recomendação CSS1 para suportar *style-sheets* (folhas de estilo) específicas, tais como os *browsers* e outras características de

poderá posicionar precisamente as regiões usando as técnicas do padrão CSS2. É possível ainda nomear *Id's*[19] e índices de ordem *z* para as regiões, permitindo criar regiões que se sobrepõem e reposicionar objetos [STA 99].

```
<layout>
<root-layout background-color="Red" width="250" height="230" />
<region id="RegiaoDeTexto" top="5" left="5" width="240" height="35" />
<region id="RegiaoDeVideo" top="40" left="30" width="180" height="180"
/>
</layout>
```

FIGURA 4.8 - Exemplo de sincronização espacial utilizando SMIL [STA 99]

No exemplo da Figura 4.8, a área de visualização principal (comando *root-layout*) é dimensionada em 250x230 *pixels*. São criadas duas regiões dentro da área de visualização, chamadas "*RegiaoDeTexto*" e "*RegiaoDeVideo*", sendo posicionadas respectivamente, a 5 *pixels* à direita da extremidade esquerda e 5 *pixels* abaixo da parte superior da área de visualização principal; e 30 *pixels* à direita da extremidade esquerda e 40 *pixels* abaixo da parte superior da área de visualização principal respectivamente.

4.4.3 A Sincronização Temporal

A sincronização temporal é uma das principais características da linguagem SMIL, sendo que a definição dessa sincronização se dá no corpo do documento e é coordenada, fundamentalmente, com o uso de duas *tags*[20]:

- **Paralelo**: representado pela *tag* "*par*", determina a apresentação simultânea dos objetos de mídia especificados. Por *default*[21] o tempo final da apresentação é igual à duração máxima dos elementos constituintes da mesma apresentação. Mas, essa semântica pode ser mudada utilizando-se os atributos definidos na linguagem (Tabela 4.1).

```
<par>
<img src="sunrise.jpg" begin="0"  dur="8,5s"/>
<video id="video1" begin="0"  end="30,1s"/>
<audio src="song1.ra"  begin="30,1"  clip-begin="5s" clip-end="15s"/>
<img src="sunset.jpg" begin="30,1"  dur="15s"/>
</par>
```

FIGURA 4.9 - Exemplo de sincronização temporal com o operador "*par*" utilizando SMIL [STA 99]

---

formatação em alto nível (como posicionamento do conteúdo, *layout* de tabelas e contadores automáticos) [ROI 99].

[19] Identificadores.

[20] Identificador que representa uma função de uma linguagem de programação para Web.

[21] Neste trabalho, é utilizado como sinônimo de padrão.

No exemplo da Figura 4.9, quando a apresentação começa a ser executada, a imagem "*sunrise.jpg*" é exibida durante 8,5 segundos juntamente com um vídeo "*video1*", que terminará sua exibição aos 30,1 segundos. Em seguida, um clipe de áudio "*song1.ra*" é iniciado e durante sua execução a imagem "*sunset.jpg*" é exibida durante 15 segundos. No segmento de áudio, utilizam-se os atributos "*begin*" e "*end*" para ignorar os primeiros 5 segundos e executar os próximos 15 segundos do clipe.

- **Seqüencial**: representado pela *tag* "*seq*", determina a apresentação dos objetos filhos um após o outro, constituindo uma seqüência temporal.

```
<seq>
<img src="sunrise.jpg" begin="0"  dur="8,5s"/>
<video id="video1" begin="0"  end="30,1s"/>
<audio src="song1.ra"  begin="30,1"  clip-begin="5s" clip-end="15s"/>
<img src="sunset.jpg" begin="30,1"  dur="15s"/>
</seq>
```

FIGURA 4.10 - Exemplo de sincronização temporal com o operador "*seq*" utilizando SMIL [STA 99]

No exemplo da Figura 4.10, quando a apresentação começa a ser executada, a imagem "*sunrise.jpg*" é exibida durante 8,5 segundos. Após o seu término, é apresentado o vídeo "*video1*" que executa até o 30,1 segundos da apresentação. Após a finalização do vídeo "*video1*", o clipe de áudio "*song1.ra*" é iniciado. Quando for finalizada a execução do áudio, é exibida a imagem "*sunset.jpg*" durante 15 segundos.

Segundo [W3C 98], a linguagem SMIL possibilita basicamente a definição de três tipos de tempo para um elemento de sincronização: início, duração e final. Sendo que em cada um desses tipos de tempo, o autor poderá definir os parâmetros como: implícito, desejado, explícito e concreto.

Os tempos de início, duração e final concreto especificam a percepção do leitor do documento. Já os valores implícito, explícito e desejado são valores auxiliares utilizados para definir os valores concretos.

4.4.4 Largura de Banda

A largura de banda, ou *bandwidth*, se refere ao limite máximo que pode ser alcançado na transmissão de dados por uma conexão de rede e, normalmente, é expresso em *Kilobits* por segundo (Kbps) [GUT 2000]. Assim como existem especificações para uma máquina local, começam a surgir requisitos para aplicações em rede como, por exemplo, a definição da largura de banda adequada para uma apresentação.

Devido a problemas de congestionamento na rede e de leitura do servidor, o limite de largura de banda pode variar em instantes de tempo utilizando um mesmo modem. Por exemplo, um modem de 28.8 Kbps pode receber 11 Kbps de dados em um momento e 22 Kbps de dados em outro momento. Sendo assim, o desenvolvedor não tem controle

sobre os problemas que podem surgir durante a apresentação, mas pode determinar que a transmissão de dados não exceda a largura de banda do usuário.

A linguagem SMIL apresenta um atributo para testar a capacidade da largura de banda, em termos de bits por segundo disponíveis para o sistema. Por exemplo, o autor define um valor para largura de banda (os valores mais comuns para os utilizadores de modem são: 14400, 28800, 56000 bits/s, etc.), caso a taxa de bits disponível no sistema seja igual ou superior ao valor especificado, o atributo é avaliado como "*true*". Se a taxa de bits disponível para o sistema for inferior ao valor especificado, o atributo é avaliado como *"false*". Isso auxilia o processo de sincronização, uma vez que o *player* SMIL seleciona o tipo de informação que pode ser transmitido na largura de banda disponível.

4.4.5 Vantagens no Uso do SMIL

A utilização da linguagem SMIL no desenvolvimento de apresentações multimídia sincronizadas provê ao autor importantes funcionalidades, dentre elas destacam-se [BRO 98]:

- simplicidade de autoria: a implementação da aplicação pode ser feita em um simples editor de texto como, por exemplo, o Bloco de Notas do *Windows*;
- simplicidade na definição da sincronização: a sincronização temporal pode ser definida se o usuário dominar apenas as *tags*: "*seq*" e "*par*";
- sincronização espacial: a definição da sincronização espacial pode ser estabelecida diretamente com o uso da linguagem;
- largura de banda: a linguagem oferece suporte a múltiplas larguras de banda, que podem ser definidas pelo próprio autor da apresentação;
- ajuste da apresentação ao usuário: por ser um arquivo texto, a apresentação SMIL pode ser gerada automaticamente para cada leitor de acordo com as preferências definidas no *browser* de sua máquina;
- utilização na Web: os *players* SMIL são gratuitos e as versões do *browser Netscape* (versão superior a 4.6) já vem com ele incluído.

4.4.6 Desvantagens no Uso do SMIL

O trabalho de definição de uma apresentação multimídia sincronizada através da Linguagem SMIL pode apresentar algumas desvantagens, tais como [ROU 98]:

- operadores para composição: a linguagem provê somente dois operadores "*par*" e "*seq*", através dos quais deverá ser definida a sincronização da apresentação. Esses operadores podem, em alguns casos, ser insuficientes para especificar a sincronização adequada da apresentação;
- definição de *loop*: a linguagem não permite a definição de um *loop* para a apresentação de mídias;
- *layout* do documento: o *layout* de um documento SMIL é estático, ou seja, ele não muda com o tempo. O *layout* estático limita as possibilidades de autoria, porque os objetos temporais podem exigir uma apresentação que varie no tempo.

4.4.7 O Modelo de Expressões Intervalares e a Linguagem SMIL

A implementação de apresentações multimídia sincronizadas baseadas no Modelo de Expressões Intervalares pode ser realizada fazendo-se uso da Linguagem SMIL. Entretanto, essa implementação ainda não agrupa funcionalidades para a utilização de todos os operadores definidos no modelo. Sendo assim, a Tabela 4.1 apresenta os operadores do Modelo de Expressões Intervalares e a *tag* correspondente na Linguagem SMIL:

TABELA 4.1 - Comparação entre os operadores das Expressões Intervalares e a Linguagem SMIL

Operador do Modelo de Expressões Intervalares	Tags correspondentes na linguagem SMIL
Seq	<seq> </seq>
Par-begin	Atributo begin= "id" do primeiro segmento de mídia
Par-end	<par> </par>
Par-Min	Atributo endsync= "first"
Par-Max	Atributo endsync= "last"
Follow	Não é implementado
Equal	Atributo begin= "id" do primeiro segmento e atributo end= "id" do primeiro segmento
Ident	Atributo begin= "id" do primeiro segmento e atributo end= "id" do segundo segmento
Alternative	<switch> </switch>
Loop	Não é implementado

Os operadores do Modelo de Expressões Intervalares são implementados pela linguagem SMIL através de *tags* de controle, ou ainda de atributos que podem ser definidos pelo autor da apresentação, visando seguir a funcionamento descrito em [DUD 95].

Recentemente, o W3C lançou o primeiro material de trabalho da nova versão do SMIL, chamada SMIL *Boston*. Essa nova versão acrescenta importantes extensões

como módulos reutilizáveis, animação genérica, interatividade aprimorada, integração com TV e possibilidade de integração com outras linguagens baseadas em XML.

O capítulo seguinte objetiva traçar um paralelo entre o trabalho com sincronização multimídia na Web e a utilização do Modelo de Expressões Intervalares, assim como apresentar uma proposta de extensão para esse visando incorporar funcionalidades não previstas na especificação descrita em [DUD 95].

5 Proposta de Extensão do Modelo de Expressões Intervalares

Segundo [INF 98], "*a Internet é considerada por muitos um dos mais importantes e revolucionários desenvolvimentos da história da humanidade. Pela primeira vez no mundo um cidadão comum ou uma pequena empresa pode (facilmente e a um custo muito baixo) não só ter acesso à informações localizadas nos mais distantes pontos do globo como também e é isso que torna a coisa mais revolucionária – criar, gerenciar e distribuir informações em larga escala, no âmbito mundial, algo que somente uma grande organização poderia fazer usando os meios de comunicação convencionais. Isso com certeza afetará substancialmente toda a estrutura de disseminação de informações existente no mundo, a qual é controlada primariamente por grandes empresas*".

Os avanços noticiados em relação à Internet também foram aplicados aos documentos disponibilizados nesse eficaz meio de comunicação. Mas, juntamente com essas melhorias surgiram algumas dificuldades relacionadas ao desenvolvimento e disponibilização desses documentos na rede mundial de computadores. Aplicações multimídia que antigamente podiam contar somente com as mídias tradicionais, como áudio, vídeo, texto e imagens, muitas vezes com problemas na apresentação e transferência de dados, hoje abrangem fluxos de mídia por demanda e aplicações em tempo real. Atualmente, é possível que usuários assistam conferências realizadas em diferentes partes do mundo, ou até mesmo seu canal de TV preferido, utilizando apenas seu microcomputador conectado à Internet.

Segundo [RUB 96], a transmissão de várias mídias em redes de computadores requer a utilização de mecanismos especiais que possibilitem apresentações multimídia coerentes. Baseados nessa observação e nos requisitos desejáveis para Modelos de Tempo definidos em [COS 96], podem ser citadas como dificuldades no trabalho com Sincronização Temporal Multimídia na Web:

- **Modelos de Tempo**: grande parte dos modelos de tempo utilizados no desenvolvimento de apresentações multimídia sincronizadas não são direcionados à Web, possibilitando somente a implementação de aplicações para uso local ou ineficientes aplicações em rede.

- **Redes de Computadores**: as redes de transmissão de dados normalmente são congestionadas. Isso é ocasionado por diversos fatores, dentre eles podem ser destacados: meio físico para transmissão de dados de baixa eficiência, número de usuários e projeto mal feito para disponibilização dos dados multimídia em servidores. A sincronização é prejudicada uma vez que essa variação afeta o envio e recebimento de dados correspondentes às mídias sincronizadas.

- **Diversidade de Configuração de Máquina**: o usuário provavelmente, utilizará uma máquina com características diferentes daquela na qual foi implementada a apresentação. Essas diferenças de configuração entre as máquinas de autor e leitor poderão acarretar alterações na apresentação das mídias sincronizadas.

- **Interação com o Usuário**: o autor não tem conhecimento do tempo utilizado pelo usuário para interagir com a aplicação. Mas, em alguns casos essa duração é estimada, fator que desfavorece os usuários que por algum motivo ultrapassem esses limites, mas em outras aplicações esta estimativa é necessária para a continuidade da apresentação.

- **Consistência da Apresentação**: é desejável que o modelo utilizado garanta a consistência das apresentações, mas a grande maioria dos modelos existentes não favorece esse tipo de implementação.

Sendo assim, o presente estudo sobre os modelos utilizados no desenvolvimento de aplicações multimídia sincronizadas para Internet, caracteriza um esforço adicional para auxiliar os desenvolvedores desse tipo de aplicação. Tendo em vista os problemas identificados nos modelos anteriormente apresentados (seção 3), observa-se que o de Expressões Intervalares é um dos mais indicados para o trabalho de sincronização na Web, principalmente devido às seguintes características:

- Consistência da Apresentação: o modelo assegura a consistência das apresentações implementadas, pois mesmo ocorrendo variações nas mídias sincronizadas, a expressão intervalar definida permanece inalterada.

- Possibilidade de teste no ambiente Web: através da Linguagem SMIL é possível definir apresentações baseadas no modelo a serem disponibilizadas na Web.

- Interação com o Usuário: o modelo permite a implementação de aplicações em que ocorram interações do usuário.

- Definição da Sincronização: o modelo é de média complexidade, uma vez que o autor deve ter conhecimento dos operadores disponíveis e da formalização utilizada para a definição de uma apresentação multimídia sincronizada.

O Modelo de Expressões Intervalares é adequado ao ambiente Web, embora poderia ser auxiliado por mecanismos que fornecessem ao autor maiores possibilidades de manipular a apresentação. Visando assim, permitir definições de sincronização não possíveis com o modelo, tais como: determinação de um período para aguardar a interação com o usuário ou o início de uma mídia; e permitir ao autor a definição de operações de sincronização do tipo "*overlaps*", onde uma mídia é disparada durante a apresentação de outra.

Diante da constatação das deficiências do Modelo de Expressões Intervalares, serão apresentadas duas sugestões de solução: a definição do segmento de mídia Conjunto V e a criação do Ponto de Sincronização. Essas soluções caracterizam a Proposta de Extensão do Modelo de Expressões Intervalares e serão descritas a seguir.

5.1 Definição do Segmento de Mídia Conjunto V

O Modelo de Expressões Intervalares não apresenta uma funcionalidade que possibilite ao autor da apresentação multimídia sincronizada determinar um intervalo para a ocorrência, por exemplo, da interação do usuário ou ainda para a apresentação de uma mídia. Visando solucionar tal deficiência, propõe-se a criação de um novo segmento de mídia chamado Conjunto V, que deverá ser incorporado aos demais conjuntos criados por Duda e Kéramane [DUD 95].

O Conjunto V será caracterizado pela inclusão de um controlador chamado "**mídia invisível**", cuja função é determinar um intervalo para ocorrência de algum evento de sincronização, mas que continue garantindo a consistência da apresentação. O exemplo do roteiro guiado ao museu, descrito no Capítulo 4 e representado na Figura 5.1, apresenta uma circunstância onde seria adequada a utilização desse controlador.

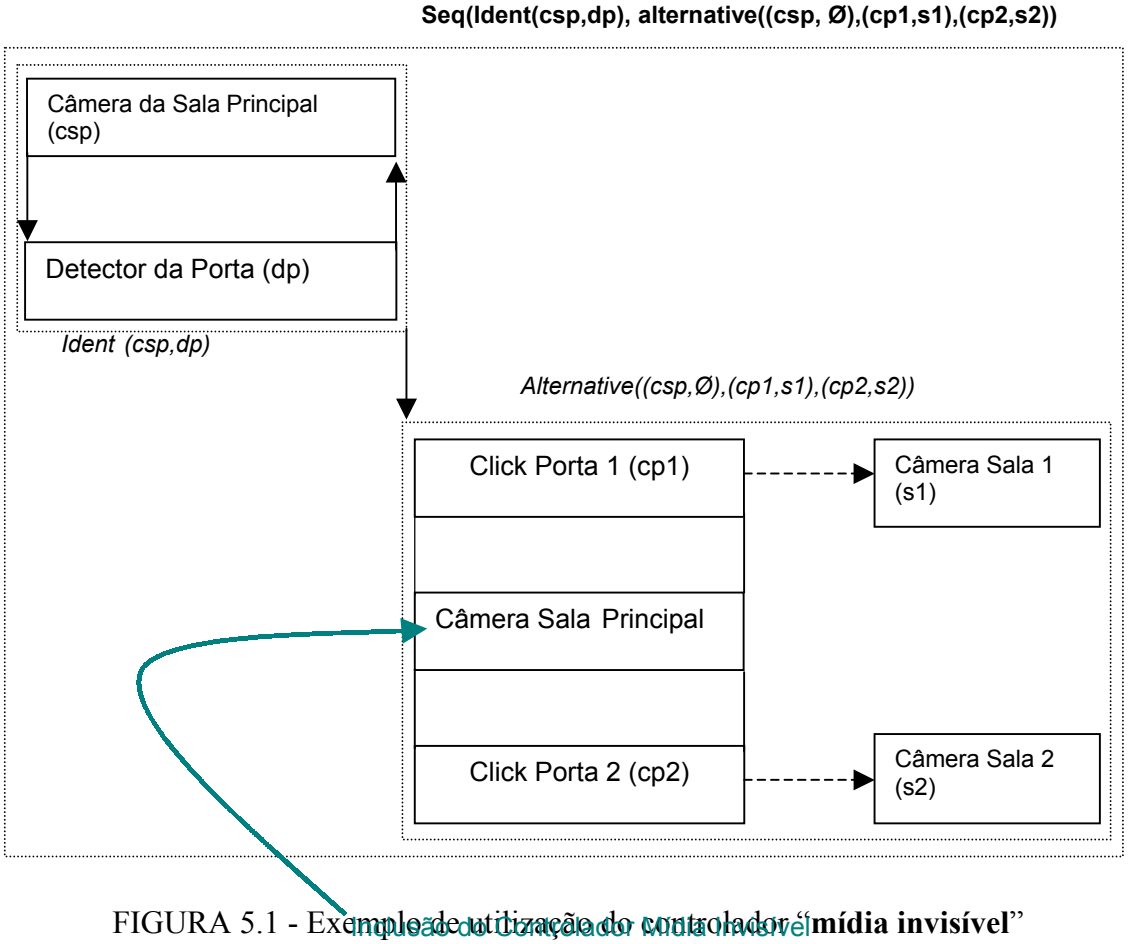


FIGURA 5.1 - Exemplo de utilização do controlador "**mídia invisível**"

Na Figura 5.1 está representada uma composição temporal onde a não-interação do usuário não é prevista, uma vez que somente será visualizada uma sala diferente da principal quando o usuário clicar na porta correspondente. Com a inclusão do controlador **mídia invisível**, poderá ser determinado um intervalo aceitável para a ocorrência da interação (Figura 5.2).

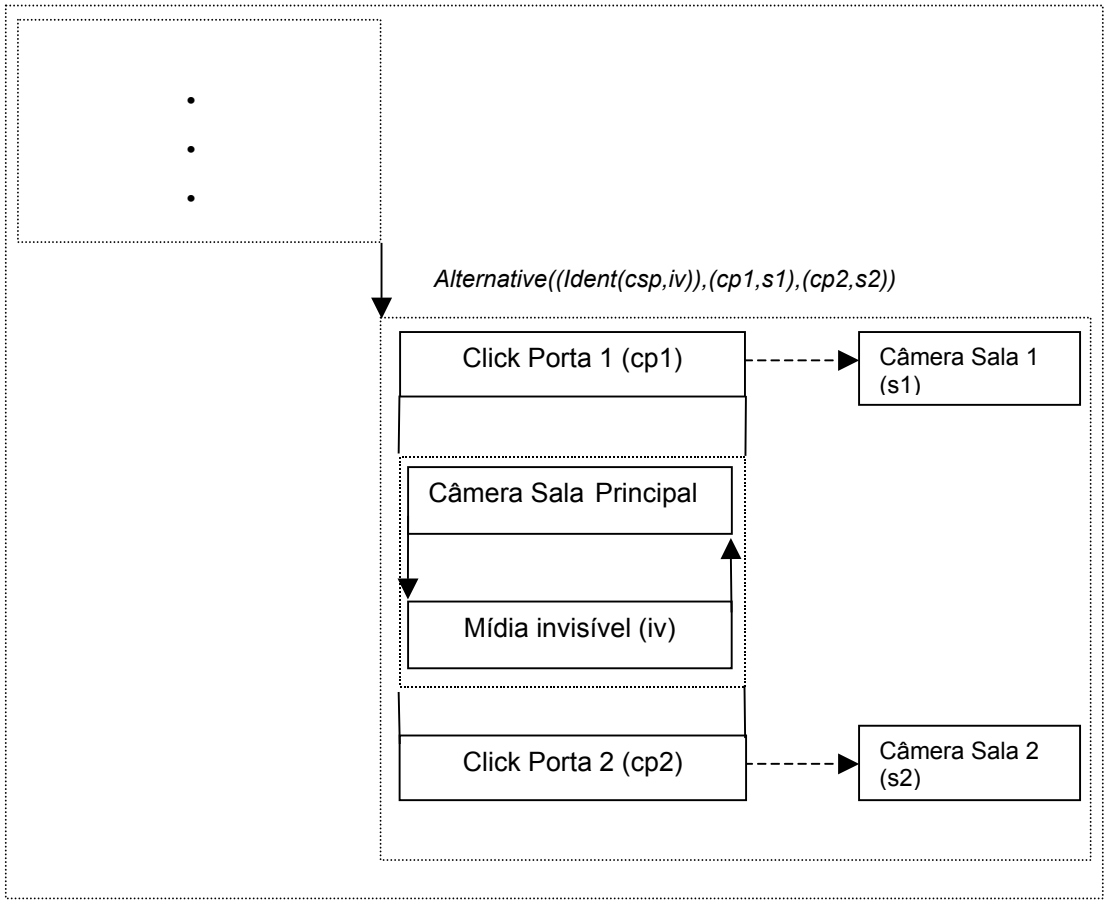


FIGURA 5.2 - Composição temporal de um roteiro guiado a um museu com a utilização de um controlador de tempo

A expressão intervalar correspondente à Figura 5.2 é a seguinte:

Seq (...(Alternative((Ident(csp,iv)),(cp1,s1),(cp2,s2))))

Observa-se que a inclusão do controlador ocasiona poucas modificações, tanto na composição temporal quanto na expressão intervalar, mas proporciona maior flexibilidade à apresentação.

Dessa forma, o controlador "**mídia invisível**" teria a finalidade de determinar um intervalo para a ocorrência da interação do usuário. Caso esse tempo seja ultrapassado, os nodos da aplicação seriam apresentados segundo uma ordem pré-definida pelo autor. Logo, a apresentação teria duas possibilidades de execução: com a interação do usuário e sem interação do usuário. Salienta-se que a não-interação do usuário em um determinado ponto da apresentação, não significa que todo o restante da aplicação deva seguir a ordem pré-determinada, ou seja, o leitor pode interagir em determinados momentos e em outros não. Isso é possível, pois o controlador "**mídia invisível**" estará presente sempre que seja necessário determinar um intervalo para a ocorrência da interação do usuário e funcionará independentemente do restante da aplicação.

5.1.1 Funcionamento do Controlador "Mídia Invisível"

O controlador "**mídia invisível**" faz referência a uma mídia cuja apresentação é transparente ao usuário, mas que possui um intervalo de duração associado, semelhante aos demais conjuntos do Modelo de Expressões Intervalares. Essa mídia controladora não é visualizada pelo usuário, entretanto o autor da aplicação a utiliza para realizar a sincronização.

O funcionamento do controlador dar-se-á de duas formas:

- No controle de interação: a apresentação dispara o controlador. Caso a interação do usuário ocorra antes de finalizar o intervalo do controlador, a apresentação segue de acordo com a escolha do usuário. No caso da não-interação, o controlador determina o próximo nodo a ser apresentado.

- No controle de mídias: a apresentação dispara o controlador e, ao final do seu intervalo, o próximo nodo da apresentação é disparado. Salienta-se que esse nodo pode pertencer a qualquer um dos segmentos de mídia definidos em [DUD 95].

A utilização do controlador "**mídia invisível**" é viabilizada fazendo-se uso dos operadores "*seq*" e "*ident*" definidos em [DUD 95], pois eles possibilitam ao autor definir o controlador e garantir que esse seja executado na sua totalidade, ou então até que ocorra uma interação do usuário. Quaisquer outros operadores que sejam utilizados, não garantirão que o controlador "**mídia invisível**" seja executado corretamente, podendo ocasionar a definição de apresentações inconsistentes.

A Tabela 5.1 apresenta a relação de operadores definidos por Duda e Kéramane [DUD 95] e a respectiva justificativa da impossibilidade de uso com o controlador "**mídia invisível**":

TABELA 5.1 - Relação entre operadores do modelo e o controlador "mídia invisível"

Operador	Justificativa
Follow	O início do segundo intervalo é ativado externamente. Dessa forma, o controlador teria necessariamente que ser ativado pelo usuário.
Par-begin	O primeiro intervalo finaliza a execução do segundo, não garantindo a execução adequada do controlador.
Par-end	O final do primeiro intervalo determina o final do segundo, o que não garante que o controlador seja executado na sua totalidade. Além disso, o segundo intervalo deveria ser ativado externamente.
Par-min	O final da apresentação ocorre quando o menor intervalo for finalizado, o que não garante que o "**mídia invisível**" seja executado na sua totalidade.
Equal	O primeiro intervalo finaliza o segundo, o que não garante que o controlador seja executado na sua totalidade.
Alternative	O primeiro intervalo que terminar finaliza os demais, o que não garante a execução adequada do "**mídia invisível**"
Loop	O final dos intervalos não pode ser determinado por ativações externas, o que impossibilitaria o controle de interação do usuário.

Na Figura 5.3 observa-se a representação gráfica dos operadores "*seq*" e "*ident*" para a sincronização de duas mídias, sendo que a mídia "a" pertence a qualquer conjunto especificado em [DUD 95] e a "**mídia invisível**" pertence ao Conjunto V.

seq(a,mídia invisível) ***ident(a,mídia invisível)***

FIGURA 5.3 - Representação gráfica dos operadores "*seq*" e "*ident"* utilizando mídia pertencente ao Conjunto V

Os operadores definidos por Duda e Kéramane para a definição de uma apresentação multimídia sincronizada, permanecem inalterados em relação ao novo segmento (Conjunto V). Sendo assim, um segmento de mídia seria definido pela união dos conjuntos definidos em [DUD 95] e o novo conjunto especificado:

$$I_{base} = M \cup S \cup P \cup L \cup \mathrm{V} \cup \{\varnothing\}$$

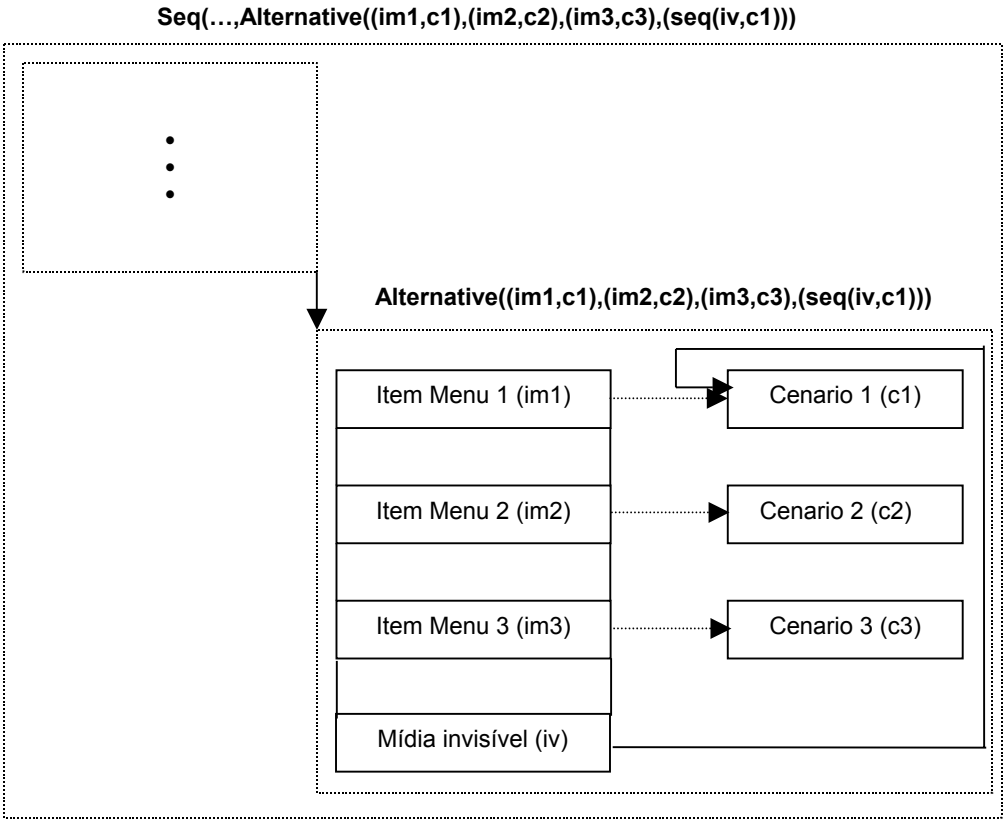


FIGURA 5.4 - Representação gráfica de uma apresentação utilizando o Conjunto V

A Figura 5.4 apresenta outro exemplo de composição temporal utilizando o Conjunto V, cujo funcionamento dar-se-á da seguinte forma: são apresentadas ao usuário três possibilidades de escolha de menu: "**Item Menu 1**" (im1), "**Item Menu 2**"

(im2) e "**Item Menu 3**" (im3). Juntamente com a apresentação desses itens, o controlador "**mídia invisível**", representado pela mídia "iv" é disparado sem que o usuário tome conhecimento. Caso este selecione algum dos itens durante a apresentação do "**Mídia Invisível**", o nodo correspondente ao item será executado. Se nenhum dos itens for selecionado durante a execução do "**Mídia Invisível**", a aplicação segue uma ordem de execução pré-definida pelo autor, apresentando o nodo "**Cenario 1**" (c1), que corresponde à mídia associada ao primeiro item do menu.

O usuário poderá interagir com o sistema em qualquer momento da apresentação. Sendo assim, teremos uma mídia controladora a cada novo conjunto de nodos apresentados e sobre os quais se possibilite a interação. Logo, para cada mídia controladora teremos uma seqüência pré-definida de nodos a serem apresentados no caso da não-interação do usuário.

A expressão intervalar correspondente à apresentação dos itens do menu na Figura 5.4 é a seguinte:

Alternative((im1,c1),(im2,c2),(im3,c3),(seq(iv,c1)))

5.1.2 O Conjunto V e o Conjunto Vazio

O Conjunto V se difere do Conjunto Vazio proposto em [DUD 95], pois no segundo caso a referência é a uma sincronização onde um dos intervalos sobre o qual se aplica um operador é inexistente. A Figura 5.5 demonstra graficamente essa situação:

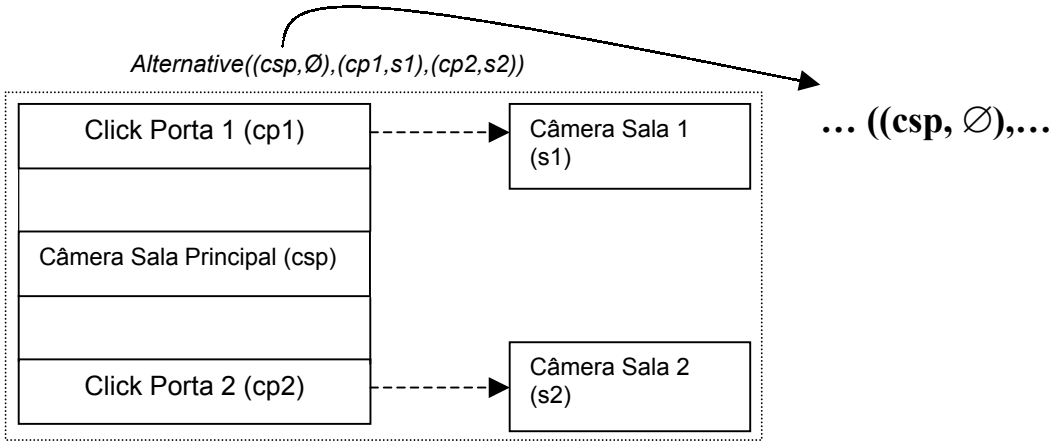


FIGURA 5.5 - Exemplo de composição temporal utilizando o Conjunto ∅ [KER 97]

Na implementação descrita na Figura 5.5, quando o usuário estiver visualizando a sala principal e clicar em qualquer lugar que não seja as portas 1 ou 2, continuará visualizando a mesma sala principal. Já no momento em que ele clicar nas portas 1 ou 2, os dois controladores de interação do usuário, "*Click Porta 1*" e "*Click Porta 2*", possibilitarão a visualização da sala correspondente. Dessa forma, o Conjunto Vazio proposto em [DUD 95], é utilizado para definir um mecanismo que implemente situações como, por exemplo, aquela em que a interação com o usuário não vai ocasionar qualquer modificação na execução da apresentação.

5.1.3 Implementação do Conjunto V

O controlador "**mídia invisível**" proposto nesta seção, apesar de não ter sido definido no Modelo de Expressões Intervalares, pode ser implementado através da Linguagem SMIL. Isso se deve ao fato de os operadores utilizados na inclusão do controlador serem os mesmos definidos em [DUD 95] e estarem presentes na linguagem. Dessa forma, serão necessárias apenas modificações na expressão intervalar definida para a apresentação, devido à inclusão de uma nova mídia na sincronização.

Tendo em vista testar a aplicabilidade do controlador "**mídia invisível**", desenvolveu-se um protótipo denominado SIGHMA (**S**istema de **G**erência de **H**iperdocumentos **A**tivos), apresentado na seção 6, onde a implementação do controlador foi baseada na utilização de uma mídia que é transmitida juntamente com o restante da aplicação. Essa mídia tem o mínimo de qualidade possível, visando gerar um menor arquivo, mas apresenta um intervalo de exibição adequado ao controle da apresentação.

Salienta-se que outra forma de implementar o controlador seria enviar juntamente com a apresentação a indicação do intervalo da mídia controladora, mas por inexistência de ferramentas que implementem esta funcionalidade não foi possível usar no protótipo desenvolvido.

5.2 Ponto de Sincronização

Os operadores disponíveis no Modelo de Expressões Intervalares permitem ao autor definir apresentações com as mais diversas características. Apesar disso, nota-se que a impossibilidade de criação de sincronizações baseadas no transcorrer de uma mídia, ainda é uma deficiência a ser corrigida. Esse tipo de operação proporciona ao autor maior flexibilidade, uma vez que permite criar sincronizações do tipo "*overlaps*" (Figura 5.6), presente em outros modelos, mas inexistente no de Expressões Intervalares.

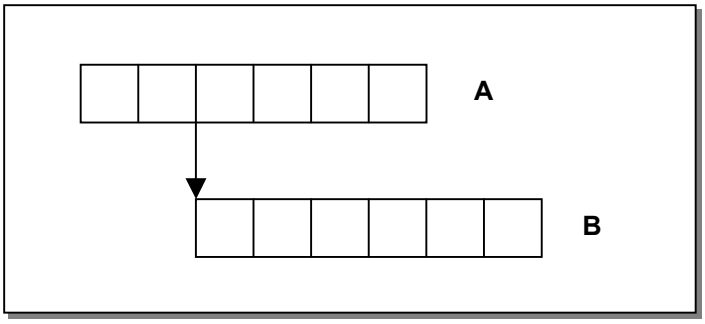


FIGURA 5.6 - Representação da operação "*overlaps*"

Diante da deficiência observada, uma possível solução seria incorporar um mecanismo que possibilite ao autor determinar pontos da mídia que serão utilizados como base para a sincronização. Inicialmente, ponderou-se a hipótese de determinar um tempo específico, que após ser ultrapassado dispararia outra mídia. Entretanto, a

inclusão de um "temporizador" traria os problemas observados nos modelos estudados como, por exemplo, a inconsistência da apresentação.

Então foram analisadas outras duas possibilidades para solução do problema:

- **Criação de um *Link* na mídia**: caracterizada pela inclusão de um *link* em uma parte da mídia e que seria responsável por disparar a segunda mídia. Essa solução foi descartada, pois caso a parte da mídia que contivesse o *link* fosse perdida na rede, resultaria também a perda da sincronização.
- **Criação de um ponto de sincronização**: caracterizada pela determinação de um ponto da mídia que seria a base para a sincronização. Essa solução foi adotada no presente trabalho e chamar-se-á **Ponto de Sincronização** ou ***Sinc-Point***.

O **Ponto de Sincronização** deverá ser estabelecido pelo autor e consiste em um valor a ser contabilizado pelo *browser* para o disparo da segunda mídia da sincronização. A determinação desse ponto vai depender do tipo de mídia a que pertence e será definida em termos de suas sub-unidades, da seguinte forma:

- **Animações e vídeo**: o ponto de sincronização será representado pelo número de *frames* transcorridos.
- **Áudio**: o ponto de sincronização será determinado pelo número de *samples*[22] transcorridos.
- **Texto**: o ponto de sincronização será determinado pelo número de caracteres exibidos. Ainda poderia ser representado pelo número de parágrafos.

Dessa forma, caberá ao autor trabalhar com uma ferramenta de manipulação de mídia que possibilite a visualização dos *frames* de um vídeo e/ou animação, dos *samples* de um áudio ou ainda dos caracteres de um texto. Visando assim, especificar ao *browser* o ponto de sincronização desejado para o início da apresentação da segunda mídia sincronizada.

A utilização do **Ponto de Sincronização** será possível através de um operador chamado "SINC-POINT", cuja representação será baseada no formato descrito a seguir:

SINC-POINT(α,α_p,β);

onde α = mídia referência da sincronização

α_p = ponto da mídia α em que se deseja iniciar a apresentação da segunda mídia (ponto de sincronização)

β = unidade a ser contabilizada pelo *browser* (*frame*, *sample*, caracteres)

Sendo assim, a Figura 5.7 representa uma composição temporal que faz uso do operador "SINC-POINT". A execução ocorre da seguinte forma: é apresentado um texto, ao seu final inicia-se um vídeo, e ao final do terceiro *frame* deste inicia-se a apresentação de um áudio:

[22] Unidade de amostragem de som [RID 95].

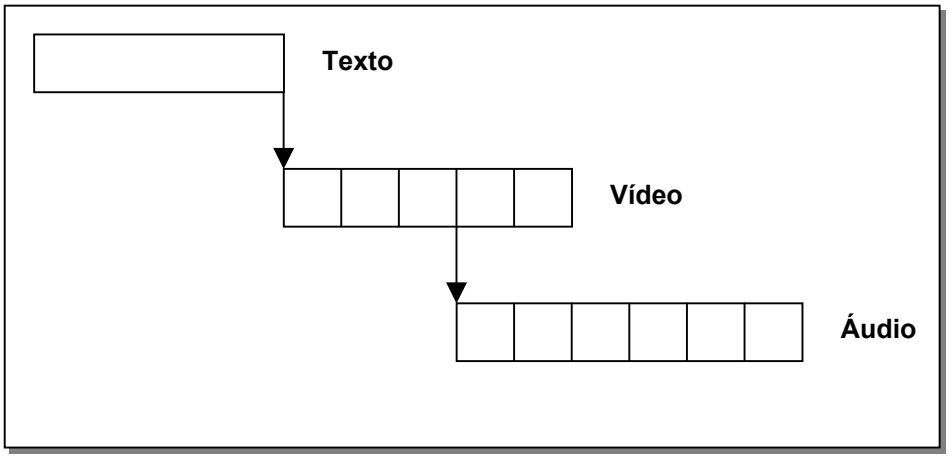


FIGURA 5.7 – Representação gráfica de uma composição temporal utilizando o operador SINC-POINT

A expressão intervalar que representa a Figura 5.7 é apresentada a seguir:

Seq(Seq(Texto,Vídeo),Seq(Sinc-Point(Vídeo,3,frame),Áudio))

Na implementação de uma aplicação que faz uso do operador "SINC-POINT", caso a duração da mídia seja modificada, o parâmetro base para a sincronização possivelmente deverá ser modificado. Mas, essa implementação ocasiona somente a modificação em uma operação da expressão intervalar definida para a sincronização. Isso se deve ao fato de o Modelo de Expressões Intervalares trabalhar com operadores binários, que não ocasionam uma propagação da alteração às outras mídias que fazem parte da apresentação.

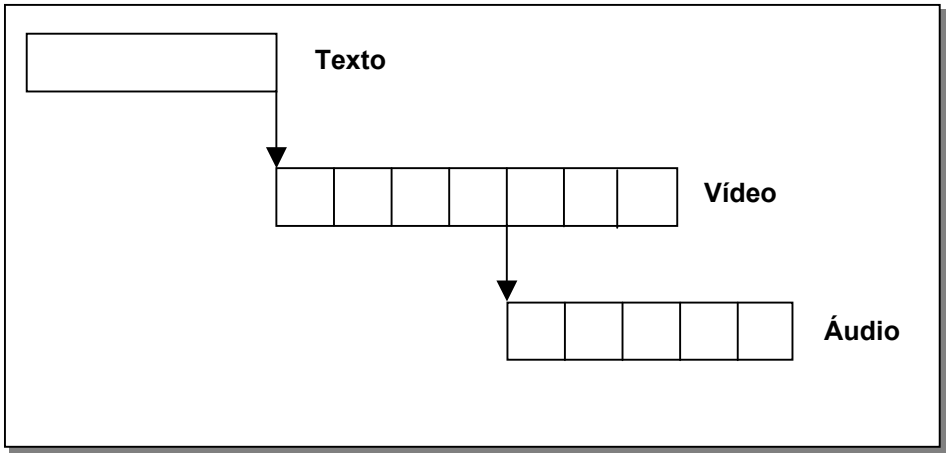


FIGURA 5.8 – Representação gráfica da modificação de composição temporal utilizando o operador SINC-POINT

A Figura 5.8 apresenta uma modificação da composição temporal descrita anteriormente na Figura 5.7, onde o intervalo correspondente à mídia "*Vídeo*" foi aumentado em dois *frames* e o correspondente à mídia "*Áudio*" foi reduzido em um *sample*. Dessa forma, a composição temporal correspondente à Figura 5.8 é a seguinte:

Seq(Seq(Texto,Vídeo),Seq(Sinc-Point(Vídeo,4,frame),Áudio))

Observa-se que a modificação ocorreu apenas no número do *frame* que dispara a execução da mídia "*Áudio*", enquanto os operadores definidos para a sincronização permanecem inalterados.

O próximo capítulo apresenta a descrição do protótipo denominado SIGHMA, que foi desenvolvido com a Linguagem SMIL e teve por objetivo testar aplicabilidade do Conjunto V incorporado ao Modelo de Expressões Intervalares.

6 Estudo de Caso: SIGHMA

A implementação do estudo de caso denominado **SIGHMA** (**S**istema de **G**erência de **H**iperdocumentos **M**ultimídia **A**tivos) constituiu uma possibilidade de testar a aplicabilidade do Modelo de Expressões Intervalares aliado à inclusão do controlador **"mídia invisível**", proposto neste trabalho. Esse estudo de caso foi desenvolvido através da Linguagem SMIL e teve como principais objetivos:

- Comprovar a adequação do Modelo de Expressões Intervalares para o desenvolvimento de aplicações multimídia temporalmente sincronizadas para a Web.
- Identificar os possíveis problemas surgidos nas fases de projeto e implementação de uma apresentação multimídia sincronizada para Web, assim como as deficiências apresentadas pelo Modelo de Expressões Intervalares.
- Testar a aplicabilidade do "**mídia invisível**" no controle da interação do usuário.

A definição do estudo de caso SIGHMA se na utilização de seis tipos de mídias: áudio, animação, imagem, texto, *links* e programas executáveis. Dessa forma, temos a representação de três conjuntos do modelo proposto por Duda e Kéramane [DUD 95]:

- **Conjunto M**: representado pelos segmentos de mídia de texto, áudio, imagem e animações.
- **Conjunto L**: representado pelos *links*.
- **Conjunto P**: representado pelos programas executáveis.

Além dos conjuntos anteriormente citados, utilizamos na implementação o Conjunto V proposto neste trabalho (seção 5). A definição e sincronização das mídias pertencentes a estes quatro conjuntos de segmentos de mídia serão responsáveis pela composição do SIGHMA.

6.1 A Estrutura do SIGHMA

A estrutura do SIGHMA é baseada em um conjunto de blocos, que correspondem a cada um dos seis temas principais relacionados ao Projeto SIGHMA[23]. São eles:

⇒***Main***: responsável pela apresentação dos demais blocos, servindo como elo de ligação entre eles. Representa o menu de controle ou menu principal.

⇒**SIGHMA**: responsável pela apresentação geral do Projeto SIGHMA.

⇒***People***: responsável pela apresentação dos participantes do grupo, assim como pela disponibilização de pequenas demonstrações dos protótipos desenvolvidos.

⇒***Projects***: responsável por apresentar os sub-projetos e suas responsabilidades dentro do Projeto SIGHMA.

[23] Projeto ao qual está vinculado esta pesquisa.

⇒***Papers***: responsável por disponibilizar ao usuário os artigos submetidos à eventos de pesquisa e possibilitar o *download*[24] dos mesmos.

⇒***Prizes***: responsável por apresentar os prêmios recebidos pelo projeto e os *links* para os eventos correspondentes.

O principal bloco do SIGHMA é o "*Main*", pois este faz a relação entre todos os demais blocos. Dessa forma, a estrutura definida para a apresentação pode ser representada pela Figura 6.1:

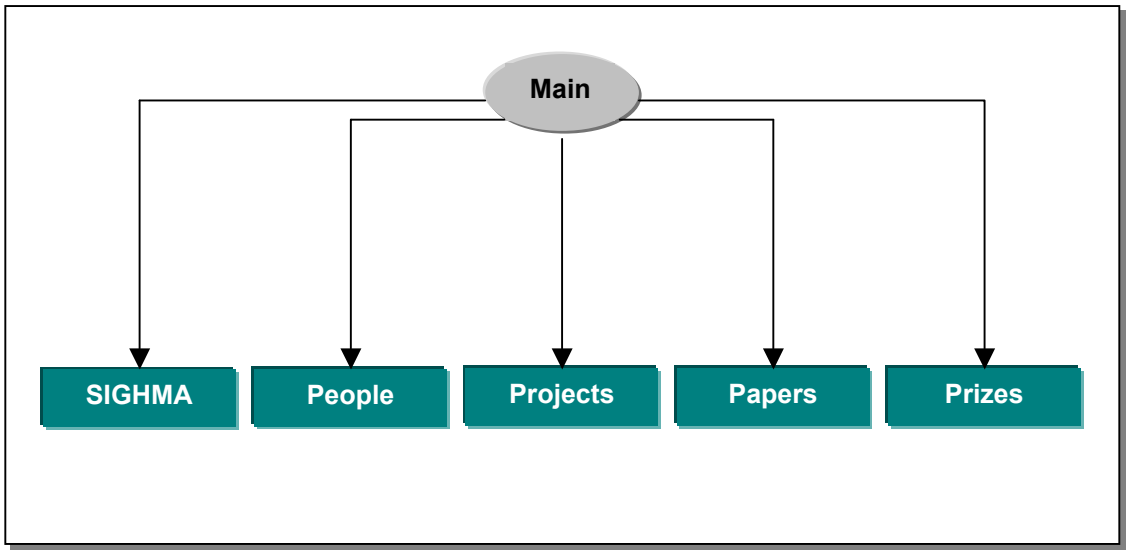


FIGURA 6.1 - Representação gráfica da relação dos blocos no SIGHMA

6.2 A Implementação do SIGHMA

A implementação do estudo de caso passou por duas versões: a primeira foi caracterizada pelo desenvolvimento e criação dos nodos do SIGHMA de forma independente, e a segunda foi uma reestruturação que teve por objetivos unificar o código (ver Anexo 1) e deixá-lo mais claro e coeso.

Em um primeiro momento, os blocos que formam o SIGHMA eram tratados em arquivos SMIL separados, pois estes funcionavam de maneira diferente dependendo do último tópico visitado pelo usuário. Por exemplo, se o usuário inicialmente houvesse visitado o bloco "SIGHMA", e em outra utilização, houvesse iniciado a visita a partir do bloco "*Papers*", a apresentação se comportaria de maneiras diferentes. No primeiro caso, o próximo bloco a ser apresentado seria o "*People*", enquanto que no segundo caso, o bloco conseguinte seria o "*Prizes*". Nesta primeira fase da implementação, foram utilizados arquivos criados com o *Macromedia Flash* (seção 4.2), pois na linguagem SMIL inexistia uma função que permitisse um *link* automático para a definição do próximo bloco a ser apresentado. Dessa forma, cada bloco era composto por três arquivos:

[24] Nome dado ao processo de transferência dos dados de um computador para outro.

1. arquivo HTML: utilizado para indicar ao *browser* o arquivo correspondente à apresentação SMIL que deveria ser interpretado pelo *plug-in*;
2. arquivo SMIL: utilizado para a definição da estrutura da apresentação;
3. arquivo SWF[25]: utilizado para definir os *links* entre os blocos que constituíam o estudo de caso.

Sendo assim, a estrutura dos blocos do SIGHMA seria representada da seguinte forma:

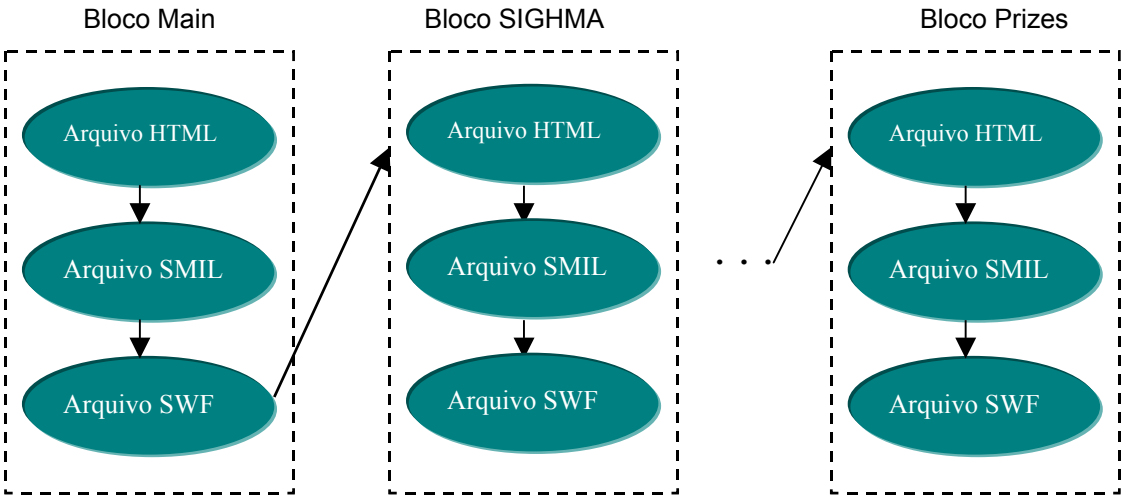


FIGURA 6.2 - Representação gráfica da primeira implementação do SIGHMA

6.3 A reestruturação do SIGHMA

A complexidade no gerenciamento do grande número de arquivos produzidos pela definição do estudo de caso em seis blocos, juntamente com a impossibilidade de fazer uma ligação entre estes blocos através da Linguagem SMIL, ocasionou a necessidade de reestruturação do SIGHMA. A solução empregada consiste na definição de apenas um bloco que ficará responsável pela implementação, reduzindo significativamente a quantidade de arquivos utilizados, assim como os problemas de desempenho na passagem de um arquivo para outro via rede.

A nova estrutura corresponde a apenas um bloco principal "*Main*", que seqüencializa os demais nodos constituintes do estudo de caso, ou seja, que determina a ordem de apresentação dos nodos sem interação do usuário. A Figura 6.3 representa a ordem de apresentação dos nodos principais do SIGHMA definida pelo autor após a reestruturação:

[25] Arquivo gerado com o *Macromedia Flash*.

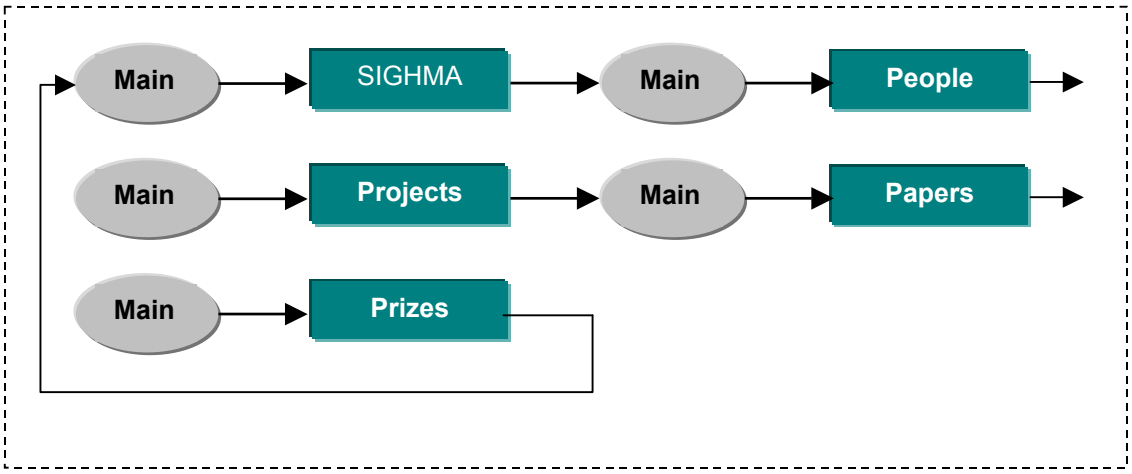


FIGURA 6.3 - Representação gráfica da ordem da apresentação após a reestruturação

A sincronização dos nodos intermediários também deve ser definida pelo autor, pois o controle do intervalo para interação também é realizado nestes. Por exemplo, o nodo SIGHMA é subdivido em "*Introdução*", "*Objetivos*" e "*Participantes*" que são apresentados seqüencialmente. O controle da interação do usuário é feito através de um áudio "*audio.rm*", que não é audível pelo usuário. Sendo assim, a expressão intervalar correspondente a esta sincronização é a seguinte:

Ident (Seq (*Introdução*, Seq (*Objetivos*, *Participantes*)), *audio.rm*)

Neste caso, foi utilizado o operador "*ident*" para que o controle de interação comece a ser realizado juntamente com a apresentação do primeiro interno do bloco SIGHMA.

6.4 Funcionamento do SIGHMA

A exibição dos nodos do estudo de caso pode se dar com a ocorrência da interação do usuário e/ou sem esta interação. Tendo em vista proporcionar esta flexibilidade à apresentação, utilizou-se o controlador "**mídia invisível**" (seção 5.1). Esse foi implementado através da inclusão de uma mídia que o usuário não pode visualizar, mas que é utilizada para realizar o controle do intervalo para realização da interação.

Durante a visualização da apresentação, o usuário poderá interagir com o sistema, sendo necessário para tanto se estabelecer um intervalo máximo para a ocorrência da interação. Quando este intervalo chegar ao seu limite, a aplicação seguirá a ordem pré-definida pelo autor. O usuário poderá interagir independentemente em cada nodo do SIGHMA, ou seja, em um determinado momento ele pode definir a informação a ser visualizada, e em outro pode seguir a ordem pré-determinada. Esta característica é importante para resolver casos onde o usuário fique desorientado, visto que o sistema vai verificar se o intervalo definido para o "**mídia invisível**" foi ultrapassado e, automaticamente, apresentará o nodo conseguinte àquele que está sendo visualizado.

O usuário poderá intervir ou não na apresentação, proporcionando então dois tipos diferentes de visualização para cada nodo controlado:

- Sem interação do usuário: a aplicação vai seguir a ordem pré-definida pelo autor da apresentação.
- Com interação do usuário: o usuário vai definir a ordem de visitação dos nodos da aplicação. Observando-se que, ao final da apresentação do último nodo, um operador "*Loop*" atua deslocando automaticamente a execução para o nodo "*Main*".

A apresentação ainda disponibiliza ao usuário os botões "*Back*", "*Forward*" e "*Stop*" (Figura 6.4), que possibilitam, respectivamente, o deslocamento para os nodos anterior e posterior ao que está sendo visitado, ou determinam a finalização da apresentação.

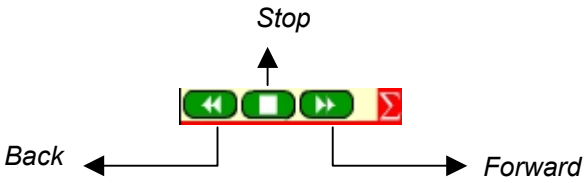


FIGURA 6.4 - Botões para navegação no SIGHMA

A Figura 6.5 apresenta a implementação dos botões de navegação do SIGHMA:

```
<seq id="prizes">
  <animation src="chttp://www.inf.ufrgs.br/~nick/prizes.swf" region="full
screen"  fill="remove"/>
<img src="chttp://www.inf.ufrgs.br/~nick/bar.jpg" region="bar"  fill="remove"/>
<a href=#main4" show="new">
<img src="chttp://www.inf.ufrgs.br/~nick/back.jpg" region="back"
fill="remove"/></a>
<a href=#main0" show="new">
<img src="chttp://www.inf.ufrgs.br/~nick/stop.jpg" region="stop"
fill="remove"/></a>
<a href=#main0" show="new">
<img src="chttp://www.inf.ufrgs.br/~nick/forward.jpg" region="forward"
fill="remove"/></a>
<audio src="chttp://www.inf.ufrgs.br/~nick/music/fire.rm"/>
</seq>
```

FIGURA 6.5 - Implementação do funcionamento dos botões de navegação

De acordo com a Figura 6.5, o usuário poderá interagir com a apresentação através dos botões "*Back*", "*Forward*" e "*Stop*", durante o período determinado pelo intervalo de duração do áudio "*fire.rm*". Caso este período seja ultrapassado, a apresentação segue conforme a ordem pré-estabelecida pelo autor. A expressão intervalar correspondente a esta implementação é a seguinte:

Seq(Seq (prizes.swf, Seq (*bar.jpg*, Seq (*back.jpg*, Seq (*forward.jpg*, *stop.jpg)*))), *fire.rm*)

A Figura 6.6 apresenta o de controle da interação do usuário realizado no bloco "*Main*":

```
<seq>
<par>
  <animation src="chttp://www.inf.ufrgs.br/~nick/main.swf" region="full
screen"  fill="remove"/>
      <anchor href="#sighma"   coords="0%, 2%, 48%, 20%"/>
      <anchor href="#people"   coords="0%, 2%, 48%, 20%"/>
      <anchor href="#projects"   coords="0%, 2%, 48%, 20%"/>
      <anchor href="#papers"   coords="0%, 2%, 48%, 20%"/>
      <anchor href="#prizes"   coords="0%, 2%, 48%, 20%"/>
  </animation>
</par>
 <audio src="chttp://www.inf.ufrgs.br/~nick/music/star.rm"/>
</seq>
```

FIGURA 6.6 - Implementação da reestruturação do SIGHMA

De acordo com a Figura 6.6, são apresentados em paralelo os nodos que compõem o menu principal do SIGHMA (*SIGHMA, People, Projects, Papers, Prizes*) através de uma animação. Essa ocorre seqüencialmente a um áudio "*star.rm*", cujo som não é audível pelo usuário. Caso não tenha ocorrido uma interação até o término da apresentação, a aplicação seguirá a ordem definida pelo autor.

A expressão intervalar correspondente à apresentação descrita na Figura 6.6 é a seguinte:

Seq (Par(*SIGHMA*, Par(*People*, Par(*Projects*, Par(*Papers*,*Prizes*)))), "*star.rm*")

A Figura 6.7 apresenta a janela de menu do protótipo SIGHMA:

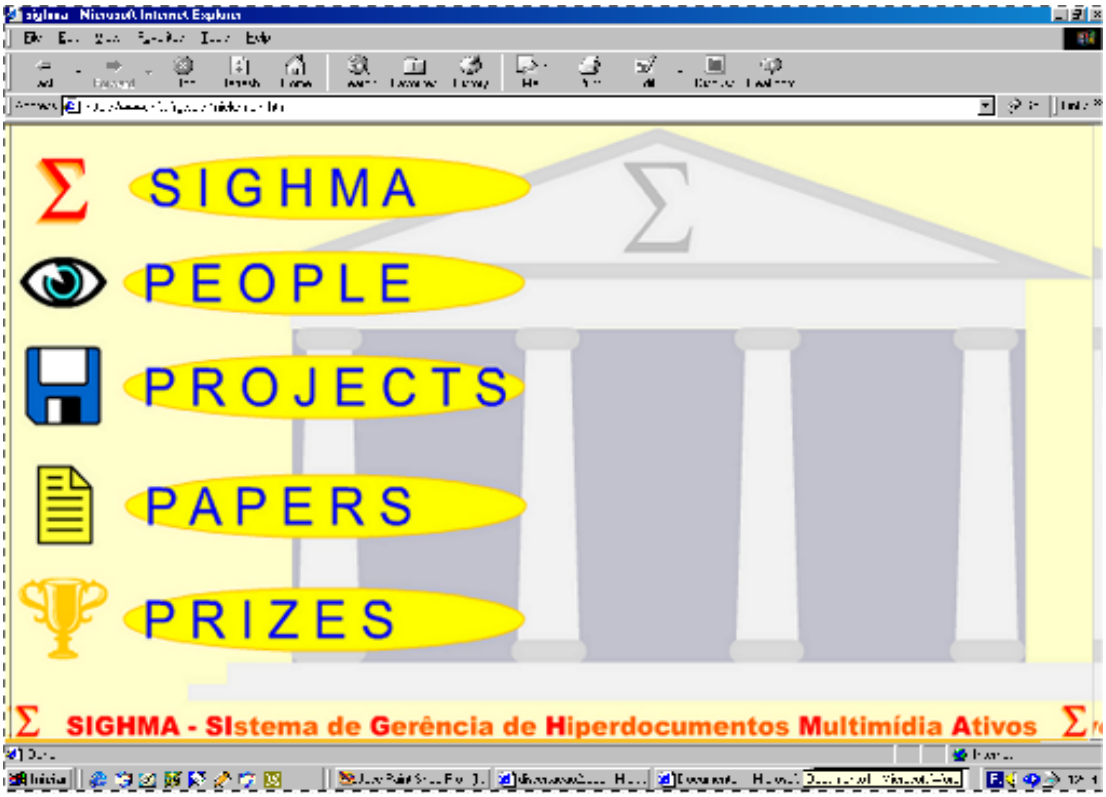


Figura 6.7 - Janela de menu do SIGHMA

O SIGHMA foi implementado fazendo uso dos operadores definidos em [DUD 95], do controlador "**mídia invisível**", proposto no presente trabalho e de algumas funções comuns em aplicações disponibilizadas na Web, tais como: *back, forward* e *stop*. Estas funções influenciaram diretamente na sincronização da aplicação, sendo importantes auxiliares na análise da usabilidade do Modelo de Expressões Intervalares através da Linguagem SMIL, principalmente na determinação de problemas, vantagens e desvantagens de sua utilização na Web.

As Figuras 6.8 e 6.9 apresentam algumas janelas do SIGHMA compostas pela sincronização de animações, imagens e áudio.

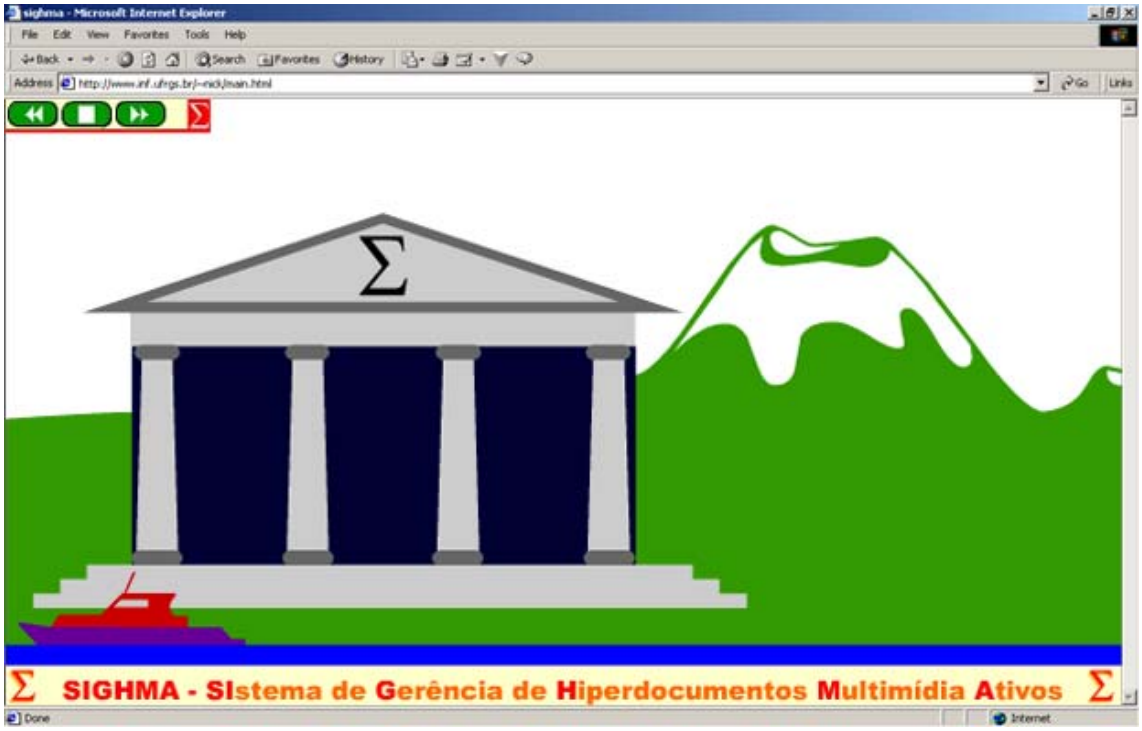


FIGURA 6.8 - Janela de apresentação do SIGHMA

FIGURA 6.9 - Janela de animação do SIGHMA

A Figura 6.10 apresenta a janela do módulo "*Papers*" onde o usuário poderá clicar sobre o nome do artigo desejado e fazer *download* do mesmo. No canto esquerdo superior da janela ainda são apresentados os botões "*Back*", "*Forward*" e "*Stop*".

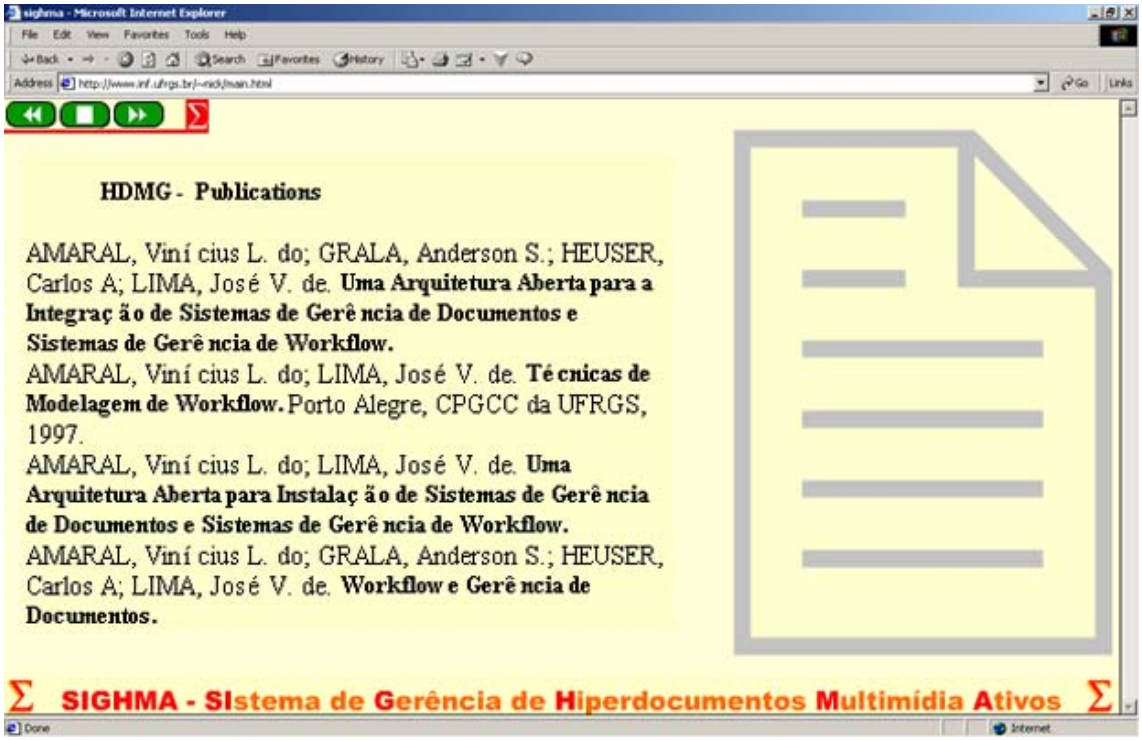


FIGURA 6.10 – Exemplo de Janela do módulo "*Prizes*"

No capítulo seguinte serão apresentadas as conclusões obtidas após o estudo realizado, assim como as próximas etapas da implementação do SIGHMA e as perspectivas de trabalhos futuros.

7 Conclusões e Trabalhos Futuros

Este trabalho apresentou um estudo sobre os modelos utilizados na definição de apresentações multimídia sincronizadas para a Web, visando especificar vantagens e desvantagens de seus usos, fornecendo assim subsídios para a determinação de um modelo a ser utilizado neste tipo de implementação.

O estudo dos Modelos de Sincronização possibilitou a definição de um conjunto de requisitos desejáveis para uma apresentação multimídia sincronizada:

- Consistência da apresentação.
- Abstração do conteúdo.
- Possibilidade de interação com o usuário.
- Definição da sincronização de forma fácil.
- Apresentação de ferramentas que possibilitem a implementação de apresentações baseadas no modelo.
- Utilização adequada na Web.

No estudo das vantagens e desvantagens dos Modelos de Sincronização, assim como na análise segundo os requisitos especificados, observou-se que todos os modelos ainda apresentam deficiências, que vão desde a definição da sincronização até o suporte a apresentações para disponibilização na Web. Baseando-se nestas conclusões, optou-se por utilizar o Modelo de Expressões Intervalares, que agrega importantes características, dentre as quais destacam-se a manutenção da consistência da apresentação, a possibilidade de interação com o usuário e a possibilidade de disponibilização da apresentação na Web.

Visando solucionar algumas deficiências do Modelo de Expressões Intervalares, assim adequando-o mais eficazmente ao trabalho a que se propõe, sugeriu-se a incorporação de duas funcionalidades: a criação do controlador "**mídia invisível**" e a definição do operador "**SINC-POINT**", que compõem a Proposta de Extensão do Modelo de Expressões Intervalares. Essa apresenta uma solução para o problema dos controles da interatividade do leitor e do disparo de mídias; bem como para a definição de um operador que possibilite ao autor da apresentação determinar a sincronização baseada no transcorrer de mídias.

A utilização do "**mídia invisível**" representa uma solução para o controle da interatividade do leitor e do disparo de mídias e foi testada através da implementação do SIGHMA. Observou-se que o "**mídia invisível**" possibilita este controle de forma simples, com a vantagem de não exigir a modificação dos operadores do modelo original, nem da ferramenta utilizada na definição da sincronização.

A criação do operador "**SINC-POINT**" para o controle de mídias disparadas durante o transcorrer de outras, seria uma importante funcionalidade a ser incorporada ao Modelo de Expressões Intervalares, necessitando para tanto a inclusão de um novo operador no modelo original, mas que por sua vez aumentaria significativamente a aplicabilidade do mesmo, possibilitando assim a definição de sincronizações mais finas.

Observa-se ainda, a importância na análise das características dos modelos utilizados na sincronização multimídia, devendo-se levar em consideração os requisitos necessários para a apresentação (interatividade, utilização de mídias, tamanho dos arquivos gerados, etc.), sua adaptabilidade à Web e a existência de ferramentas que possibilitem a definição da composição temporal. Além disso, é importante salientar que as apresentações definidas estarão também sujeitas aos problemas normais de rede.

Sendo assim, a definição de uma apresentação sincronizada ainda é uma tarefa que, além de trabalhosa, está sujeita a fatores inerentes à vontade do usuário ou à definição estabelecida pelo autor. Fatores estes que devem ser considerados na escolha não só dos modelos, mas na definição de mídias de boa qualidade, nas exigências de hardware e de software utilizados para implementação e visualização da apresentação.

As principais contribuições desta dissertação foram: levantamento dos principais modelos utilizados no trabalho de sincronização multimídia, considerando suas características e identificando vantagens e desvantagens de seu uso, principalmente relacionados ao ambiente Web; sugestão de utilização de um modelo para uso na Web, o de Expressões Intervalares; desenvolvimento de uma Proposta de Extensão para o Modelo de Expressões Intervalares, visando incorporar algumas funcionalidades que proporcionem ao autor um controle maior da apresentação e permitam a definição de sincronizações mais flexíveis.

A continuidade do trabalho prevê as seguintes atividades:

- Reestruturação do SIGHMA e mudança para nova versão da Linguagem SMIL.
- Estudo da possibilidade de inclusão de um controlador do tipo "*timeout*", que determine um intervalo de aceitável para a exibição da mídia levando-se em consideração os atrasos na transmissão dos dados multimídia.
- Estudo de outros modelos utilizados na sincronização de apresentações multimídia, tais como: Modelo Hierárquico e Modelo Baseado em Pontos de Referência.
- Envio de artigos a eventos na área.

Anexo 1 Código do SIGHMA

```
<smil>
---------------------- Cabeçalho do Código SMIL ---------------------------
   <head>
      <meta name="Title" content="SIGHMA" />
      <meta name="Author" content="Projeto SIGHMA" />
      <meta name="Copyright" content = "(c) 2000, SIstema Gerência
Hiperdocumentos Multimídia Ativos" />

---------------------- Definição do Layout da Apresentação ---------------
      <layout>
         <root-layout height="325" width="600" background-color="black" />
         <region id="full_screen" left="0" top="0" height="325" width="600" fit="fill"
            z-index="1" />
         <region id="bar" left="0" top="0" height="18" width="110" fit="fill" z-
            index="2" />
         <region id="back" left="0" top="0" height="16" width="29" fit="fill" z-
            index="3" />
         <region id="stop" left="29" top="0" height="16" width="29" fit="fill" z-
            index="4" />
         <region id="forward" left="58" top="0" height="16" width="29" fit="fill" z-
            index="5" />
         <region id="main" left="10" top="30" height="250" width="350" fit="fill" z-
            index="6" />
      </layout>
   </head>

---------------------- Definição do Corpo do Código SMIL ----------------
   <body>
      <par>
         <seq>
            ------------------- Definição do Bloco Main -----------------
            <par endsync="last" id="main0">
               <animation src="chttp://www.inf.ufrgs.br/~nick/main.swf"
                  region="full_screen" fill="remove" >
                  <anchor href="#sighma" coords="0%,2%,48%,20%" />
                  <anchor href="#people" coords="0%,20%,48%,33%" />
                  <anchor href="#projects" coords="0%,33%,48%,53%" />
                  <anchor href="#papers" coords="0%,53%,48%,71%" />
                  <anchor href="#prizes" coords="0%,71%,48%,90%" />
               </animation>
               <audio src="chttp://www.inf.ufrgs.br/~nick/music/star.rm" />
</par>
           ------------------- Definição do Bloco Sighma -----------------

            <par endsync="last" id="sighma">
               <animation src="chttp://www.inf.ufrgs.br/~nick/sighma.swf"
                  region="full_screen" fill="remove" />
               <img src="chttp://www.inf.ufrgs.br/~nick/bar.jpg" region="bar"
                  fill="remove" />
```

```
    <a href="#main0" show="new"> <img
        src="chttp://www.inf.ufrgs.br/~nick/back.jpg" region="back"
        fill="remove" /></a>
    <a href="#main1" show="new"><img
        src="chttp://www.inf.ufrgs.br/~nick/stop.jpg" region="stop"
        fill="remove" /></a>
    <a href="#main1" show="new"><img
        src="chttp://www.inf.ufrgs.br/~nick/forward.jpg" region="forward"
        fill="remove" /></a>
    <audio src="chttp://www.inf.ufrgs.br/~nick/music/term.rm" />
</par>

<par endsync="last" id="main1">
    <animation src="chttp://www.inf.ufrgs.br/~nick/main.swf"
        region="full_screen" fill="remove" >
        <anchor href="#sighma" coords="0%,2%,48%,20%" />
        <anchor href="#people" coords="0%,20%,48%,33%" />
        <anchor href="#projects" coords="0%,33%,48%,53%" />
        <anchor href="#papers" coords="0%,53%,48%,71%" />
        <anchor href="#prizes" coords="0%,71%,48%,90%" />
    </animation>
    <audio src="chttp://www.inf.ufrgs.br/~nick/music/star.rm" />
</par>
```

---------- Definição do Bloco People ----------

```
<par endsync="last" id="people">
  <animation src="chttp://www.inf.ufrgs.br/~nick/people.swf"
    region="full_screen" fill="remove" />
        <img src="chttp://www.inf.ufrgs.br/~nick/bar.jpg" region="bar"
            fill="remove" />
        <a href="#main2" show="new"><img
            src="chttp://www.inf.ufrgs.br/~nick/back.jpg" region="back"
            fill="remove" /></a>
        <a href="#main3" show="new"><img
            src="chttp://www.inf.ufrgs.br/~nick/stop.jpg" region="stop"
            fill="remove" /></a>
        <a href="#main3" show="new"><img
            src="chttp://www.inf.ufrgs.br/~nick/forward.jpg"
            region="forward" fill="remove" /></a>
    </animation>
    <audio src="chttp://www.inf.ufrgs.br/~nick/music/bond.rm" />
</par>

<par endsync="last" id="main2">
    <animation src="chttp://www.inf.ufrgs.br/~nick/main.swf"
      region="full_screen" fill="remove" >
        <anchor href="#sighma" coords="0%,2%,48%,20%" />
        <anchor href="#people" coords="0%,20%,48%,33%" />
        <anchor href="#projects" coords="0%,33%,48%,53%" />
        <anchor href="#papers" coords="0%,53%,48%,71%" />
        <anchor href="#prizes" coords="0%,71%,48%,90%" />
```

```
    </animation>
    <audio src="chttp://www.inf.ufrgs.br/~nick/music/star.rm" />
</par>
```

---------- Definição do Bloco Project ----------

```
<par endsync="last" id="projects">
  <animation src="chttp://www.inf.ufrgs.br/~nick/projects.swf"
    region="full_screen" fill="remove" />
    <img src="chttp://www.inf.ufrgs.br/~nick/bar.jpg" region="bar"
    fill="remove" />
    <a href="#main2" show="new"><img
    src="chttp://www.inf.ufrgs.br/~nick/back.jpg" region="back"
    fill="remove" /></a>
    <a href="#main3" show="new"><img
    src="chttp://www.inf.ufrgs.br/~nick/stop.jpg" region="stop"
    fill="remove" /></a>
    <a href="#main3" show="new"><img
    src="chttp://www.inf.ufrgs.br/~nick/forward.jpg" region="forward"
    fill="remove" /></a>
  <audio src="chttp://www.inf.ufrgs.br/~nick/music/mission.rm" />
</par>

 <par endsync="last" id="main3">
      <animation src="chttp://www.inf.ufrgs.br/~nick/main.swf"
     region="full_screen" fill="remove" >
          <anchor href="#sighma" coords="0%,2%,48%,20%" />
          <anchor href="#people" coords="0%,20%,48%,33%" />
          <anchor href="#projects" coords="0%,33%,48%,53%" />
          <anchor href="#papers" coords="0%,53%,48%,71%" />
          <anchor href="#prizes" coords="0%,71%,48%,90%" />
      </animation>
      <audio src="chttp://www.inf.ufrgs.br/~nick/music/star.rm" />
 </par>
```

---------- Definição do Bloco Papers ----------

```
<par endsync="last" id="papers">
    <animation src="chttp://www.inf.ufrgs.br/~nick/papers.swf"
      region="full_screen" fill="remove" />
        <img src="chttp://www.inf.ufrgs.br/~nick/bar.jpg" region="bar"
          fill="remove" />
        <a href="#main3" show="new"><img
          src="chttp://www.inf.ufrgs.br/~nick/back.jpg" region="back"
          fill="remove" /></a>
        <a href="#main4" show="new"><img
          src="chttp://www.inf.ufrgs.br/~nick/stop.jpg" region="stop"
          fill="remove" /></a>
        <a href="#main4" show="new"><img
          src="chttp://www.inf.ufrgs.br/~nick/forward.jpg" region="forward"
          fill="remove" /></a>
    <audio src="chttp://www.inf.ufrgs.br/~nick/music/yeah.rm" />
```

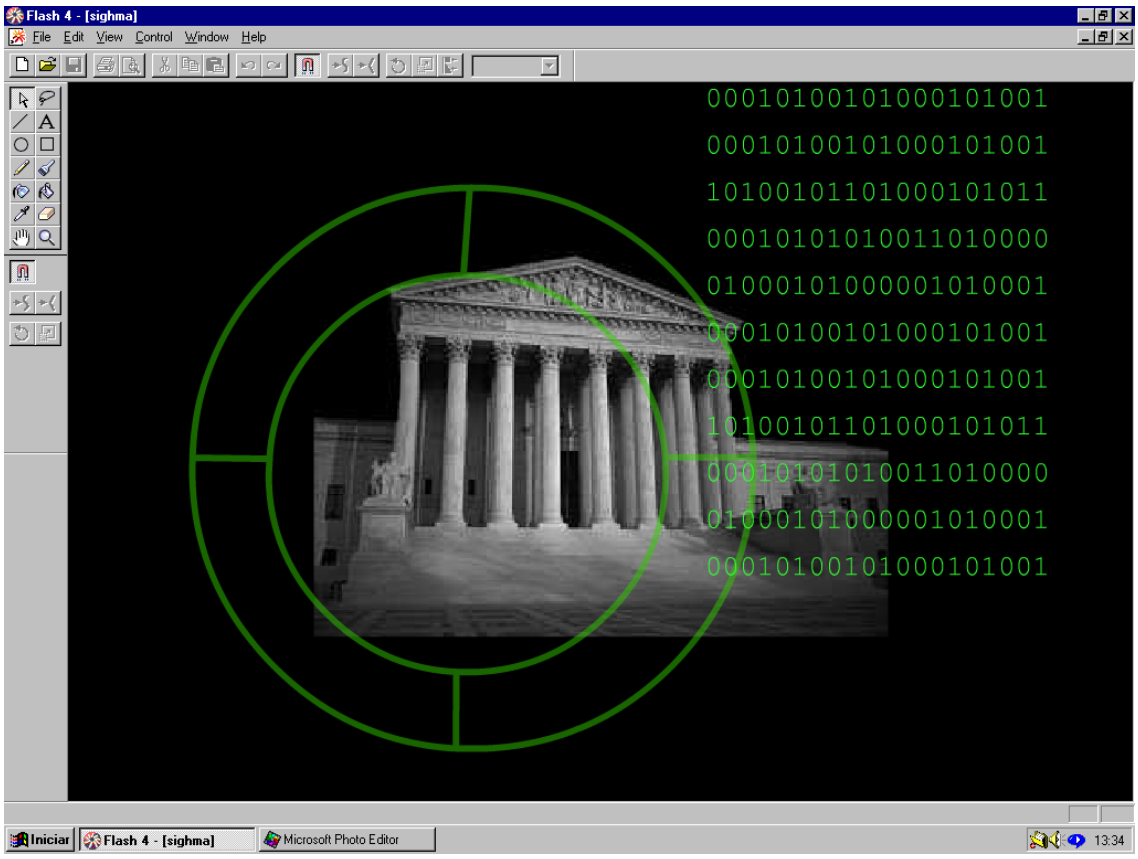
```
            <textstream src="chttp://www.inf.ufrgs.br/~nick/texto01.rt"
              region="main" fill="remove" />
          </par>

          <par endsync="last" id="main4">
            <animation src="chttp://www.inf.ufrgs.br/~nick/main.swf"
            region="full_screen" fill="remove" >
                <anchor href="#sighma" coords="0%,2%,48%,20%" />
                <anchor href="#people" coords="0%,20%,48%,33%" />
                <anchor href="#projects" coords="0%,33%,48%,53%" />
                <anchor href="#papers" coords="0%,53%,48%,71%" />
                <anchor href="#prizes" coords="0%,71%,48%,90%" />
            </animation>
            <audio src="chttp://www.inf.ufrgs.br/~nick/music/star.rm" />
          </par>
```

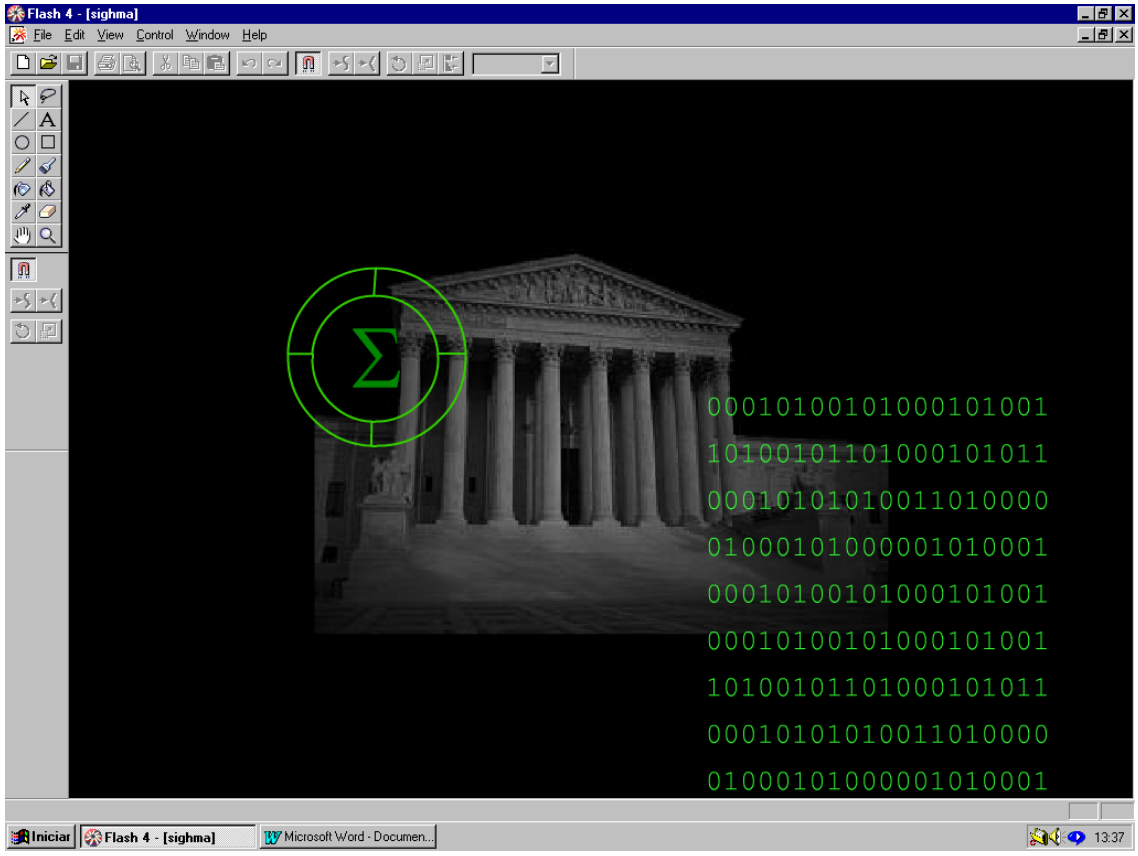
---------- Definição do Bloco Prizes ----------

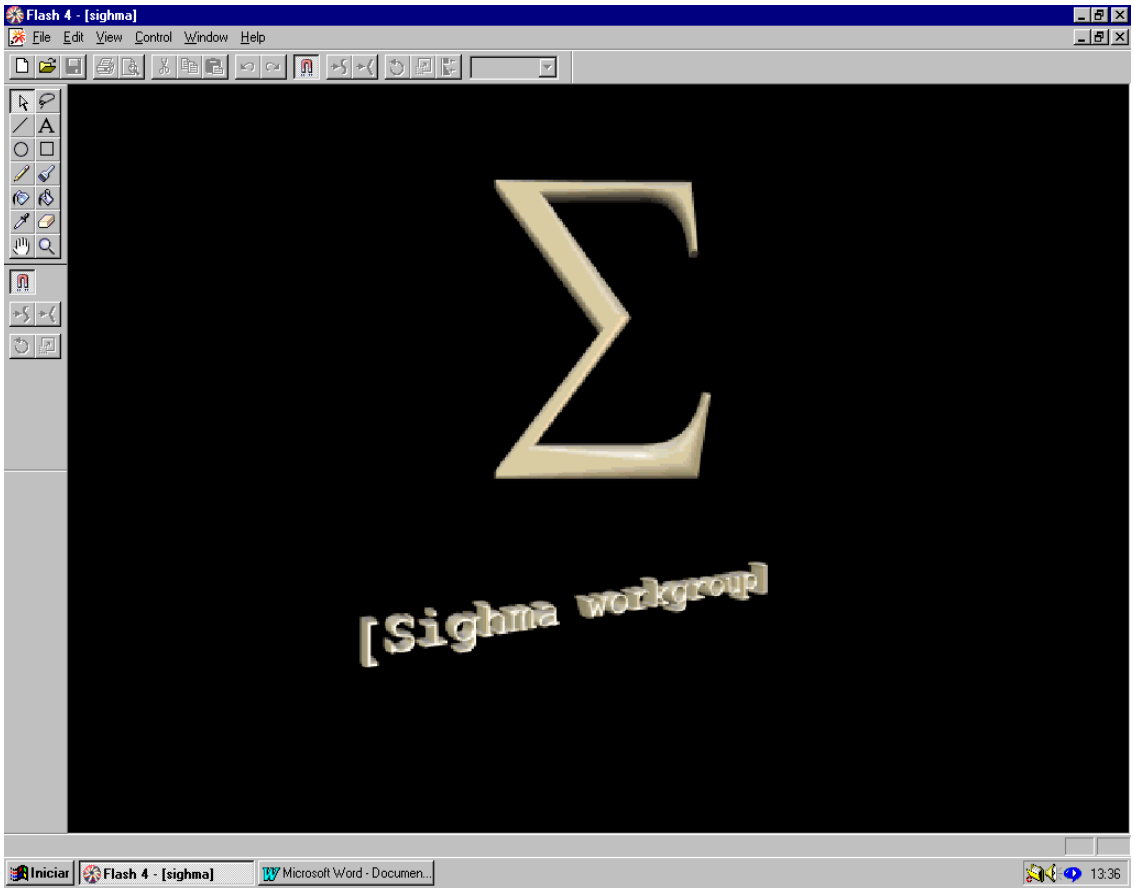
```
          <par endsync="last" id="prizes">
            <animation src="chttp://www.inf.ufrgs.br/~nick/prizes.swf"
            region="full_screen" fill="remove" />
              <img src="chttp://www.inf.ufrgs.br/~nick/bar.jpg" region="bar"
              fill="remove" />
              <a href="#main4" show="new"><img
              src="chttp://www.inf.ufrgs.br/~nick/back.jpg" region="back"
              fill="remove" /></a>
              <a href="#main0" show="new"><img
              src="chttp://www.inf.ufrgs.br/~nick/stop.jpg" region="stop"
              fill="remove" /></a>
              <a href="#main0" show="new"><img
              src="chttp://www.inf.ufrgs.br/~nick/forward.jpg" region="forward"
              fill="remove" /></a>
            <audio src="chttp://www.inf.ufrgs.br/~nick/music/fire.rm" />
          </par>

          <par endsync="last" begin="4s">
            <animation src="chttp://www.inf.ufrgs.br/~nick/loop.swf"
            region="full_screen" fill="remove" />
          </par>
        </seq>
      </par>
    </body>
</smil>
```

Anexo 2 Testes realizados com o Macromedia Flash

As Figuras seguintes apresentam alguns testes realizados com o Macromedia Flash, que tiveram por finalidade fazer um levantamento das vantagens e desvantagens de seu uso.






Flash 4 - [sighma]
File Edit View Control Window Help
[Sighma workgroup]
Iniciar
Flash 4 - [sighma]
Microsoft Word - Documen...
13:36

Flash 4 - [sighma]
File Edit View Control Window Help
IMAGES
SINCRONIZATION
TEXT
Iniciar
Flash 4 - [sighma]
Microsoft Word - Documen...
13:36

Bibliografia

[ABR 97] ABRÃO, Iran Calixto. **Ambiente para o Intercâmbio e Apresentação de Objetos Multimídia em Sistemas Distribuídos e Heterogêneos**. São Carlos: UFSCAR, 1997. Dissertação de Mestrado.

[ALL 83] ALLEN, J. F. Maintaining Knowledge about Temporal Intervals. **Communications of the ACM**, New York, v.26, n.11, p.832-843, Nov. 1983.

[BAR 98] BARRÉRE, Eduardo *et al*. Ferramenta para Elaboração e Composição de Material Didático Multimídia com Sincronização Intermídia. In: SIMPÓSIO BRASILEIRO DE INFORMÁTICA NA EDUCAÇÃO (SBIE), 9., 1998, Fortaleza. **Anais...** Fortaleza: [s.n.], 1998. p.71-78.

[BEI 96] BEILER, Adriana. **Proposta de um Ambiente Hipermídia que Permita o Leitor Desempenhar o Papel de Criador de Siginificado**. Porto Alegre: PUCRS, 1996. Dissertação de Mestrado.

[BOU 98] BOUTHILLIER, Larry. Synchronized Multimedia on the Web. **Web Techniques Magazine**, [s.l.], v.3. Sept. 1998.

[BRO 98] BROWN, Kim. **Synchronizing the Web.** Disponível em: <http://webbreview.com/wr/pub/98/01/99/feature/smil.htm>. Acesso em: 23 nov. 1998.

[BUC 93] BUCHANAN, Maria Cecelia; ZELLWEGER, Polle T. Automatically Generating Consistent Schedules for Multimedia Documents. In: ACM INTERNATIONAL CONFERENCE ON MULTIMEDIA, 1., 1993, Anaheim. **Proceedings …** Anaheim: [s.n.], 1993. p.341-350.

[BUF 94] BUFORD, J. F. K. **Multimedia Systems**. New York: ACM Press, 1994.

[BUL 93] BULTERMAN, Dick; MULLENDER, K.; JANSEN, Jack; ROSSUM, Guido. CMIFed: A Presentation Environment for Portable Hypermedia Documents. In: ACM MULTIMEDIA, 1993, Anaheim. **Anais...** Anaheim: [s.n.], 1993. p.183-188.

[CAR 96] CARMO, L. F.; COURTIAT, J. P.; OLIVEIRA, R. C. A General Purpose Multimedia Synchronization Mechanism Based on Causal Relations. **IEEE Journal on Selected Areas in Communications.** New York, v.14, n.1, p.185-195, Jan. 1996.

[CAS 98] CASTELLANI, Alexandre Moreno. Multimídia na Internet. **Revista Informática Médica**, [s.l.], v.1, n.3. Disponível em: <http://www.epub.org.br/informaticamedica/n0103/castellani.html>. Acesso em: 23 nov. 1998.

[CHO 97] CHORFI, Marcos Alexandre *et al*. Projeto e Desenvolvimento de Ferramentas para Manipulação de Hiperdocumentos em Ambientes Abertos Distribuídos. In: WORKSHOP DE SISTEMAS MULTIMÍDIA E HIPERMÍDIA, 3., 1997, São Carlos. **Anais...** São Carlos: FAPESP, 1997. p. 227-238.

[WIR 99] WIRED WORLD COMPANY. **Neuron:** The Toolbook Plug-in. Disponível em: <http://www.Asymetrix.com/products/neuron.html>. Acesso em: 12 dez. 1999.

[COS 96] COSTA, Fábio Rodrigues. **Um Editor Gráfico para Definição e Exibição do Sincronismo de Documentos Multimídia/Hipermídia.** Rio de Janeiro: PUC-RJ, 1996. Dissertação de Mestrado.

[COU 96] COURTIAT, J. P.; OLIVEIRA, R. C. Proving Temporal Consistency in a New Multimedia Syncronization Model. In: ACM MULTIMEDIA, 1996, Boston. **Anais...** Boston: [s.n.], 1996. Também disponível em: <http://www.laas.fr/~courtiat/publications.html>.

[COU 98] COURTIAT, Jean-Pierre *et al*. **Design Methodology and Formal Validation of Hypermedia Documents**. Disponível em: <http://www.laas.fr/~courtiat/PAPERS/ACM98/ACM98.html>. Acesso em: 04 dez. 1998. Trabalho apresentado no ACM Multimedia.

[DIE 96] DIETRICH, Elton. **Projeto de um Sistema de Suporte à Autoria Cooperativa de Hiperdocumentos**. Porto Alegre: CPGCC da UFRGS, 1996. Dissertação de Mestrado.

[DUD 95] DUDA, Andrzej; KERAMANE, Chérif. Structured Temporal Composition of Multimedia Data. Disponível em: <http://www.ri.silicomp.fr/adv-dvt/java/old/mmedia/references.htm >. Acesso em: 20 dez. 1995. Trabalho apresentado no IEEE INTERNATIONAL WORKSHOP ON MULTI-MEDIA DATA BASE MANAGEMENT SYSTEMS.

[ESC 95] ESCOBAR, J. Multimedia Synchronization over Wide Area Networks. In: IFIP CONFERENCE ON HIGH PERFORMANCE NETWORKING., 9., 1995, Espanha. **Proceedings...** Espanha: [s.n.], 1995.

[GAS 98] GASPARY, Luciano Paschoal. **Muse :** Um Ambiente para Modelagem de Aplicações Multimídia Interativas com Tradutor para E-LOTOS. Porto Alegre: CPGCC da UFRGS, 1998. Dissertação de Mestrado.

[GON 98] GONÇALVES, Paulo André da Silva. **Aplicativo em Tcl/TK para Composição de Objetos Multimídia.** Rio de Janeiro: UFRJ, 1998. Trabalho de Conclusão de Curso.

[GUT 2000] **GUIA de Termos Usados na Web**. Disponível em: <http://www.guia2000.com.br/clickdavez/glossario.html>. Acesso em: 12 jan. 2000.

[HAR 93] HARDMAN, L.; ROSSUM, G.; BULTERMAN, D. C. A. Structured Multimedia Authoring. **ACM MULTIMEDIA**, Anaheim, p.283-289, Dec. 1993.

[HIR 95] HIRZALLA, Naél; FALCHUK, Bem; KARMOUCH, Ahmed. A Temporal Model for Interactive Scenarios. **IEEE Multimedia**, New York, v.2, n.3, p.24-31, Fall 1995.

[HUA 98] HUANG, Chung-Ming; WANG, Chian. Synchronization for Interactive Multimedia Presentations. **IEEE Multimedia**, New York, v.5, n.4, p.44-62, Oct. 1998.

[INF 98] INFORMATIVO Técnico. Disponível em: <http://www.ccuec.unicamp.br/revista/navegacao/index4.html> Acesso em: 3 set. 1998.

[JAM 99] JAMSA, Kris; LALANI, Suleiman; WEAKLEY, Steve. **Programando para a World Wide Web**. São Paulo: Makron Books, 1999.

[KER 95] KERAMANE, Chérif; LAYAÏDA, Nabil. Maintaining Temporal Consistency of Multimedia Documents. Disponível em: <http://www.cs.tufts.edu/~isabel/Layaida/ACM95-1.html>. Acesso em: 4 nov. 1995. Trabalho apresentado no ACM WORKSHOP ON EFFECTIVE ABSTRACTIONS IN MULTIMEDIA.

[KER 97] KERAMANE, Chérif; DUDA, Andrzej. Operator Based Composition of Structured Multimedia Presentations. In: WORKSHOP FROM MULTIMEDIA SERVICES TO NETWORK SERVICES, 1997, Lisboa. **Proceedings...** Lisboa: [s.n.], 1997.

[KER 97a] KERAMANE, Chérif. **Spécification de Présentations Multimédia Structures Interactives**. Grenoble: L'Institut National Polytechnique de Grenoble, 1997. Tese de Doutorado.

[KUT 97] KUTOVA, Marcos A. S.; MACEDO, Alessandra A. **Intercâmbio de Objetos Multimídia.** São Paulo: USP – Instituto de Ciências Matemáticas de São Carlos, 1997.

[LAD 96] LADEIRA, Marcelo *et al*. HSM – Modelagem Semântica para Aplicação Hipertexto. In: CONFERÊNCIA LATINO AMERICANA DE INFORMÁTICA, 1996, Bogotá. **Anais...** Bogotá: [s.n.], 1996. p.894-906.

[LEL 2000] LEITE, Leticia Lopes; LIMA, José Valdeni de. Apresentações Multimídia Sincronizadas para Web e Implicações da Interatividade com as Mídias. In: WORKSHOP IBEROAMERICANO DE ENGENHARIA DE REQUISITOS E AMBIENTES DE SOFTWARE, 2000, Cancún. **Anais...** Cancún: [s.n.], 2000. 514p. p.499-500.

[LEI 96] LEIVA, Willie Dresler. **Suporte ao Desenvolvimento de Ferramentas para Manipulação de Hiperdocumentos em Ambientes Abertos Distribuídos.** São Paulo: USP, 1996. Relatório Final de bolsa DTI.

[LEL 99] LEITE, Leticia Lopes. **Apresentações Complexas do Tipo Multimídia**. Porto Alegre: CPGCC – UFRGS, 1999. (TI – 770).

[LEL 99a] LEITE, Leticia Lopes. Modelos de Tempo para Apresentação Multimídia Complexas. In: SIMPÓSIO NACIONAL DE INFORMÁTICA, 1999, Santa Maria. **Anais...** Santa Maria: SNI, 1999. 148p. p.62-66.

[LIM 97] LIMA, Ronaldo Mussauer; DUARTE, Otto Carlos Muniz. Implementação de um Serviço de Sincronização Multimídia para Comunicação de Dados Pré-Orquestrados. In: SIMPÓSIO DE REDES DE COMPUTADORES, 1997, São Carlos. **Anais...** São Carlos, 1997. 554p. p.19-35.

[LIT 90] LITTLE, T.D.C; GHAFOOR, A. Synchronization and Storage Models for Multimedia Objects. **IEEE Journal on Selected Areas in Communications**, New York, v.8, n.3, p.413-426, Apr. 1990.

[LIT 91] LITTLE, T. D. C. *et al*. Multimedia Synchronization. **IEEE Data Engineering Bulletin**, New York, v.14, n.3, p.26-35, Sept. 1991.

[LUQ 2000] LUQUE, M. J. Pérez; LITTLE, T. D. C. **A Temporal Framework for Multimedia Synchronization**. Disponível em: <http://hulk.bu.edu/sync95/sync95.html>. Acesso em: 17 ago. 2000.

[MAC 2000] REVISTA MACMANIA. **Macromedia Flash**. Disponível em: <http://www.zaz.com.br/macmania/atmac/064anime.html>. Acesso em: 25 jan. 2000.

[MAR 92] MARTIN, James. **Hiperdocumentos e como criá-los**. São Paulo: Makron Books, 1992.

[MOU 99] MOURA, César Olavo de; OLIVEIRA, Mauro. **Videoconferência em Educação a Distância**. Ceará: ETFCE, 1999.

[NIC 90] NICOLAOU, C. An Architecture for Real-Time Multimedia Communication Systems. **IEEE Journal on Selected Areas in Communications**, New York, v.8, n.3, p.391-400, Apr. 1990.

[PER 99] PEREIRA, Marco Antonio Carvalho. **World Wide Web – WWW**. Disponível em: <http://www.marco.eng.br/web.html>. Acesso em: 12 nov. 1999.

[REA 98] RealSystem G2 Reviewer's Guide. **Clip Synchronization and Layout with SMIL.** Disponível em: <http://prognet.com/company/pressroom/g2guide/clipsync.html>. Acesso em: 12 dez. 1998.

[RID 95] RIDGE, Peter *et al*. **Sound Blaster – O Livro Oficial**. São Paulo: Makron Books, 1995.

[ROD 97] RODRIGUES, Rogério Ferreira; SOARES, Luiz Fernando Gomes. Autoria e Formatação Estruturada de Documentos Hipermídia com Restrições Temporais. In: WORKSHOP DE SISTEMAS MULTIMÍDIA E HIPERMÍDIA, 3., 1997, São Carlos. **Anais...** São Carlos: FAPESP, 1997. p. 183-197.

[ROI 99] ROISIN, Cécile. **Authoring Structured Multimedia Documents**. In: CONFERENCE ON CURRENT TRENDS IN THEORY AND PRACTICE OF INFORMATICS, 25. Disponível em: <http://www.inrialpes.fr/opera/BibOpera.html>. Acesso em 22: dez. 1999.

[ROU 98] ROUSSEAU, Franck; DUDA, Andrzej. Synchronized Multimedia for the WWW. In: INTERNATIONAL WORLD-WIDE WEB CONFERENCE, 7., 1998, Australie. **Proceedings …** Australie : [s.n.], 1998. p.417-429.

[ROU 99] ROUSSEAU, Franck; GARCIA-MACÍAS, J. Antonio, LIMA, José Valdeni; DUDA, Andrzej. **User Adaptable Multimedia Presentations for the WWW.** Disponível em: <http://www8.org/w8-papers/2b-customizing/user/user.html>. Trabalho apresentado na WWW8 – THE EIGHTH INTERNATIONAL WORLD WIDE WEB CONFERENCE.
Acesso em: 10 jun. 1999.

[ROU 99a] ROUSSEAU, Franck. **Présentations Multimédia Synchronisées pour le WWW**. Grenoble: L'Institut National Polytechnique de Grenoble, 1999. Tese de Doutorado.

[RUB 96] RUBINSTEIN, Marcelo Gonçalves. **Um Serviço de Sincronização Contínua para Aplicações Multimídias**. Rio de Janeiro: UFRJ, 1996. Dissertação de Mestrado.

[RUB 97] RUBINSTEIN, Marcelo Gonçalves; DUARTE, Otto Carlos Bandeira. Implementação de um Serviço de Sincronização Intra e Intermídia Considerando Relações Temporais Implícitas. In: SEMINÁRIO FRANCO-BRASILEIRO EM SISTEMAS INFORMÁTICOS DISTRIBUÍDOS, 11., 1997, Fortaleza. **Anais...** Fortaleza: [s.n.], 1997.

[SOA 92] SOARES, Luiz Fernando Gomes; TUCHERMAN, L.; CASANOVA, M. A.; NUNES, P. L. Fundamentos de Sistemas Multimídia. In: ESCOLA DE COMPUTAÇÃO, 8., 1992, Gramado. **Anais ...** Gramado: [s.n.], 1992. 186p. p.80-85.

[SOA 95] SOARES, Luiz Fernando Gomes; LEMOS, Guido; COLCHER, Sérgio. **Das LANs, MANs e WANs às Redes ATM**. Rio de Janeiro: Campus, 1995.

[STA 99] STANEK, William. SMIL : O Novo Formato Web para Multimídia. **PC Magazine Brasil**, São Paulo, v.9, n.3, p.95-99, Mar. 1999.

[STE 90] STEINMETZ, Ralph. Synchronization Properties in Multimedia Systems. **IEEE Journal on Selected Areas in Communications**, New York, v.8, n.3, p.401-402, Apr. 1990.

[STE 95] STEINMETZ, Ralph; NAHRSTEDT, Klara. **Multimedia:** Computing, Communications & Applications. [s.l.]: Prentice-Hall, 1995.

[STE 96] STEINMETZ, Ralph. Human Perception of Jitter and Media Synchronization. **IEEE Journal on Selected Areas in Communications**, New York, v.14, n.1, p.61-72, Jan. 1996.

[W3C 98] W3C CORPORATION. **Synchronized Multimedia.** Disponível em: <http://www.w3.org/TR/1998/REC_smil_1980.html>. Acesso em: 15 nov. 1998.

[WAH 94] WAHL, Thomas; ROTHERMEL, Kurt. Representing Time in Multimedia Systems. In: INTERNATIONAL CONFERENCE ON MULTIMEDIA COMPUTING AND SYSTEMS, 1994, Boston. **Proceedings...** Boston: [s.n.], p.538-543.

[XIA 97] XIANG, Yuanyuan; YU, Jin. **Hypermedia Presentation and Authoring System.** Src Technical Note. Disponível em: <http://gatekeeper.dec.com/pub/DEC/SRC/technical-notes/SRC-997-003.html>. Acesso em: 6 maio 1997.