

# **Suporte à Apresentação Adaptativa de Aplicações Multimídia em Sistemas Distribuídos**

Roberta Lima Gomes  
Herli J. de Menezes  
Luiz Fernando Rust da Costa Carmo  
Luci Pirmez

E-mail: {beta,menezes}@novell.nce.ufjr.br, {rust,luci}@nce.ufjr.br

Núcleo de Computação Eletrônica – UFRJ  
Caixa Postal 2324, 20001-970  
Rio de Janeiro – RJ – Brasil

## **Resumo**

Os sistemas multimídia distribuídos empregam uma tecnologia aplicável em vários domínios como, em particular, os ambientes de ensino a distância. Considerando que os documentos multimídia são normalmente compostos de mídias contínuas, a apresentação destes documentos pode demandar uma grande quantidade de recursos devido às dimensões dos dados e às restrições de QoS das mídias. A disponibilidade de recursos em ambientes distribuídos no instante da apresentação é variável e imprevisível. Desta forma, os recursos disponíveis para a apresentação do documento multimídia podem ser insuficientes, resultando em uma degradação arbitrária da qualidade da apresentação. Uma forma de se tratar a escassez de recursos é tornar os documentos multimídia adaptativos. Neste artigo é apresentado um formato flexível para a especificação de documentos multimídia adaptativos. É discutida, também, uma arquitetura para a implementação de dois mecanismos de adaptação propostos.

## **Abstract**

Multimedia presentations are applicable in various domains such as distance learning. Since multimedia documents may comprise continuous medias, the presentation of those documents may require vast system resources due to the huge amount of data and to their QoS restraints. The availability of resources in a distributed environment at presentation time is variable and potentially unpredictable. Hence, it can happen that there are not sufficient resources to render a multimedia document according to its specification, resulting in an arbitrary decrease of presentation's quality. To handle resource scarcity multimedia documents can be made adaptable to different resource availability. In this paper we present a flexible format to the specification of adaptable multimedia documents. An architecture over which are implemented two proposed adaptation mechanisms is discussed.

## **1. INTRODUÇÃO**

Os sistemas multimídia distribuídos empregam uma tecnologia aplicável em vários domínios como, em particular, os ambientes de ensino a distância. Neste tipo de ambiente o especialista ou o professor é o autor dos documentos multimídia e, de acordo com o que é especificado pelo autor, esses documentos representam o conhecimento a ser passado para os alunos (usuários finais). O desejo de um autor é que um documento especificado seja apresentado

integralmente de forma a permitir que os usuários absorvam corretamente os conhecimentos nele contidos. Estes documentos multimídia são compostos por objetos de mídia discreta, como imagens e textos, e contínua, como vídeos e áudios. A apresentação destes documentos em ambientes distribuídos exige, portanto, uma disponibilidade bastante considerável de capacidade de transmissão e processamento.

O processo de apresentação de um determinado documento multimídia não é trivial. Este documento pode ser acessado através de diferentes tipos de redes e apresentado em diferentes tipos de terminais. Conseqüentemente, pode ocorrer a escassez de recursos no instante da apresentação de um documento multimídia. Além disso, mesmo que as redes e os tipos de terminais através dos quais os documentos são acessados sejam similares, as diferentes condições em que a rede pode se encontrar provavelmente ocasionará diferentes disponibilidades de recursos.

Nesses ambientes distribuídos, muitas vezes é necessária uma redução arbitrária na qualidade da apresentação multimídia, ocasionada pela carência de recursos no instante da apresentação. No pior caso esta carência pode forçar a interrupção da apresentação ou nem permitir que a mesma se inicie. Há, portanto, uma necessidade latente de que os sistemas distribuídos apresentem uma natureza adaptativa, de forma a suavizar o impacto da escassez de recursos sobre o usuário final.

Várias propostas e implementações de sistemas levantam a questão da adaptabilidade para a transmissão de mídias em ambientes distribuídos. A maioria dos sistemas considera apenas o instante em que a transmissão da mídia é solicitada, nenhuma adaptação é oferecida uma vez que o fluxo de mídia já tenha sido estabelecido. Um outro problema pouco questionado é a noção de adaptabilidade a nível de apresentação.

O CMIF (*CWI Multimedia Interchange Format*) [Bult91,Bult93] define um modelo para as apresentações multimídia, permitindo a especificação de formatos alternativos de codificação para um mesmo objeto de mídia ou a definição de uma mídia alternativa. Uma vez efetuada a seleção do formato a ser apresentado, não será mais possível a interrupção da mídia devido à violação de seus parâmetros de qualidade de serviço (QoS). A adaptabilidade a nível de apresentação também é considerada em [Wira97], onde é apresentada uma proposta de extensão ao modelo temporal TIEMPO [Wahl95]. Nesse modelo é permitido especificar documentos multimídia flexíveis com alternativas de grupos (trechos da apresentação) de mídia a serem apresentados (informações redundantes). Um algoritmo de escalonamento adaptativo é executado em intervalos predeterminados de tempo, podendo haver uma sobrecarga de processamento. Já em [Wadd98], “Gerenciamento de QoS” (*QoS Management*) é apresentado como um mecanismo adaptativo para a configuração e o controle dos recursos fim-a-fim. Esta adaptação é implementada através de um balanço e uma redistribuição dos recursos locais e de rede, a partir de notificações emitidas por mecanismos de monitoramento de QoS. Os elementos de rede devem fornecer mecanismos como monitoramento de *buffers*.

Esse trabalho apresenta uma proposta que visa contornar, de forma dinâmica, o problema da escassez de recursos disponíveis para a apresentação de documentos multimídia em sistemas distribuídos. É definido um formato flexível para a apresentação desses documentos, permitindo a especificação de dois níveis de adaptabilidade. Esses níveis são implementados através de dois mecanismos que, por sua vez, são aplicados de acordo com a especificação do autor e com os níveis de qualidade obtidos durante a apresentação de um documento. O primeiro mecanismo define variantes de qualidade de serviço para um determinado fluxo de

mídia. A execução de uma determinada variante depende da disponibilidade de recursos no instante da apresentação. O segundo baseia-se em uma especificação flexível do documento multimídia, de forma que o mesmo possa ser adaptado em tempo de execução, quando a apresentação da variante com menores exigências de recursos viola a especificação dos parâmetros de QoS para a respectiva mídia.

O artigo está organizado da seguinte forma. Na seção 2 é discutido o processo de autoria de documentos multimídia adaptativos. A seção 3 apresenta uma arquitetura de suporte a sistemas multimídia adaptativos. A seção 4 descreve a apresentação de uma aplicação multimídia utilizando os mecanismos de adaptação propostos. Por fim, a seção 5 apresenta algumas considerações sobre o trabalho proposto e perspectivas futuras.

## 2. AUTORIA DE DOCUMENTOS MULTIMÍDIA ADAPTATIVOS

Em ambientes de ensino a distância, o autor deseja que um determinado conjunto de conceitos seja transferido para o aluno através de um conjunto de documentos multimídia. Este conjunto de conceitos é representado, no modelo mental, sob a forma de uma rede semântica. O processo de mapeamento, ilustrado na figura 1, ou a realização desta rede semântica em um conjunto de documentos multimídia faz parte da etapa de autoria destes documentos. Uma especificação flexível dos documentos permite que usuários com diferentes perfis de sistema possam absorver corretamente o conjunto de conceitos originais. As condições do sistema no momento da restituição dos documentos e de suas respectivas mídias determinam a estrutura lógica a ser apresentada ao usuário.

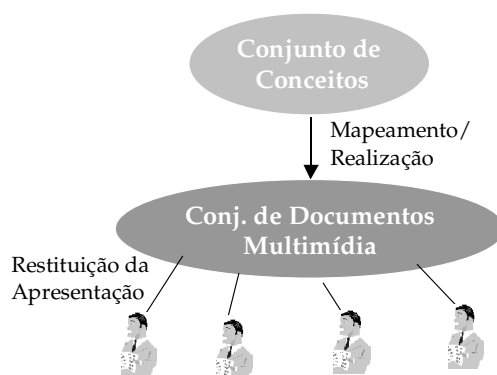


Figura 1- Arquitetura de Restituição da Informação

A especificação flexível de um documento multimídia é obtida através da criação de dois arquivos distintos. No primeiro arquivo, é especificada uma apresentação multimídia convencional. O SMIL 1.0 (*Synchronized Multimedia Integration Language*) [Smil98], uma linguagem declarativa para a especificação de apresentações multimídia, foi escolhida para a representação do documento contendo a apresentação multimídia. Este documento contém, então, a estrutura lógica ideal ou original da apresentação multimídia que seria gerada em um sistema com todos os recursos necessários disponíveis.

O segundo arquivo (arquivo de controle) é composto por informações de controle necessárias para a realização da adaptação da apresentação. Essas informações de adaptabilidade baseiam-se no conceito de causalidade apresentado em [Cunh99a, Cunh99b, Cunh99c]. Este conceito permite que um documento multimídia apresente relacionamentos causais ou condicionais, além dos convencionais relacionamentos temporais e espaciais. Os relacionamentos causais

são estabelecidos entre as mídias que estão relacionadas semanticamente. Desta forma, a apresentação de uma determinada mídia é dependente da apresentação de outras mídias com as quais ela possui relacionamentos causais.

A separação das informações de controle da estrutura original do documento multimídia permite uma interoperabilidade com os sistemas de apresentação multimídia já existentes. Desta forma, usuários que possuem apenas uma ferramenta de apresentação convencional para a linguagem SMIL 1.0 podem visualizar, normalmente, o documento multimídia, perdendo apenas a funcionalidade de adaptabilidade da apresentação.

## 2.1 Mecanismos de Adaptação

Para a implementação dos níveis de adaptabilidade citados inicialmente, dois tipos de mecanismos são implementados. A execução coordenada desses mecanismos visa garantir a consistência semântica desejada pelo autor do documento.

O primeiro mecanismo de invariância semântica utilizado pelo sistema é denominado mecanismo suave (*soft*) e é aplicado sobre as mídias individualmente. Para a execução deste mecanismo, conforme ilustrado na figura 2, variantes de qualidade de serviço para uma determinada mídia devem ser definidas pelo autor de forma que a apresentação da mesma possa ser adaptada em cima destas variantes. Este mecanismo realiza uma adaptação a nível de codificação e consiste de uma forma de seleção entre as diferentes codificações de uma mesma mídia.

No instante de autoria, após definir as mídias que vão compor o documento multimídia, deve-se codificá-las em formatos com resoluções ou qualidades distintas. A degradação do nível da qualidade de uma mídia deve ser tal que, a permutação entre as variantes, durante a apresentação, não ofereça um desconforto maior ao usuário final.

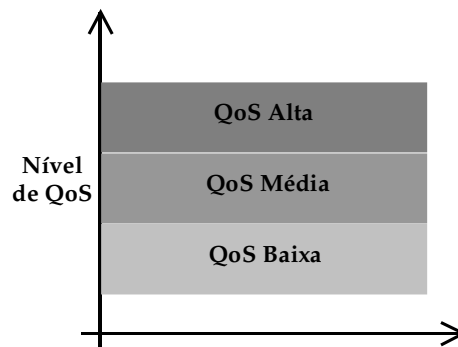


Figura 2- Variantes de QoS

O mecanismo suave de adaptação baseia-se em uma política de monitoramento. Por definição, monitoramento é o processo de observar a utilização de recursos ou de características de QoS no sistema [Wadd98]. O processo de monitoramento é responsável por gerar mensagens que indiquem a ocorrência de violações de contrato de QoS. A adaptação realizada no primeiro nível ocorre sempre quando mensagens indicando falhas de QoS forem geradas durante o monitoramento dos recursos utilizados pelas mídias transmitidas.

A vantagem aparente deste tipo de mecanismo é a transparência com que uma adaptação é executada. O usuário percebe, de forma suave, a alteração do nível de qualidade de serviço.

Além disso, trata-se de um mecanismo reversível, ou seja, o processo de monitoramento, ao detectar uma subutilização dos recursos do sistema, pode gerar mensagens para que o mecanismo de adaptação modifique o fluxo de mídia para uma variante com melhor QoS.

Se o mecanismo suave de adaptação, na ocorrência de escassez de recursos do sistema, ocasiona a apresentação da variante de menor QoS (variante que exige a menor quantidade de recursos) e, mesmo assim, a apresentação da respectiva mídia atinge um nível de qualidade abaixo do especificado pelo autor (de acordo com os parâmetros de QoS definidos para esta mídia no arquivo de controle), o segundo mecanismo de invariância semântica é aplicado. Este mecanismo, denominado mecanismo forte (*hard*), baseia-se na especificação flexível do documento multimídia, contida no arquivo de controle, de forma que o mesmo possa ser adaptado.

A adaptação a nível de aplicação, na qual se baseia o mecanismo forte, é uma técnica pouco referenciada na literatura. A maioria das propostas de adaptabilidade trata os objetos de uma mesma apresentação de forma independente. Essas adaptações a nível de objeto, ou a nível de codificação, são limitadas à estrutura lógica estática do documento e proporcionam uma funcionalidade restrita.

O mecanismo forte de adaptação manipula a estrutura lógica do documento, de acordo com premissas previamente estipuladas pelo autor, de forma que a estrutura semântica do documento seja preservada. Quando esse mecanismo é ativado, uma nova estrutura lógica é automaticamente criada, gerando uma nova apresentação semanticamente equivalente à anterior. Portanto, a ativação do mecanismo forte de adaptação implica na reestruturação do documento multimídia. É importante observar que o “novo” documento é reapresentado apenas a partir do instante em que a adaptação foi realizada.

## 2.2 Definição do Arquivo de Controle

O objetivo principal do arquivo de controle é especificar formas alternativas para a apresentação de um documento multimídia e as condições que definem os momentos de ativá-las, ou seja, definir mídias alternativas que podem ser apresentadas na ocorrência de possíveis falhas de QoS durante a apresentação do documento original. A definição destas condições de ativação, bem como das ações que devem ser tomadas, é feita através da especificação de um conjunto de relações de causalidade entre as mídias do documento original e as mídias alternativas disponibilizadas pelo autor. As relações de causalidade indicam como deverá ocorrer o processo de adaptação, expressando a consistência desejada para um documento multimídia.

As relações de causalidade são conceitualmente definidas através de um objeto *Link*, que pode ser de dois tipos: *startlink* e *stoplink*. O tipo do *Link* indica a ação executada quando o mesmo é ativado (apresentar ou interromper uma mídia). O objeto *Link* é composto de um *LinkSource* e um *LinkTarget*. No *LinkSource* são definidas as condições associadas às apresentações das diferentes mídias: início (*started*), fim (*concluded*) e interrupção (*stopped*), que podem ser combinadas através de operadores booleanos. Quando as condições estipuladas no *LinkSource* tornam-se verdadeiras, a mídia indicada pelo *LinkTarget* deve sofrer uma determinada ação. Por exemplo, a expressão *startlink*=“(x:stopped)” associada a uma mídia y define que a apresentação de y deve ser ativada caso ocorra uma interrupção na apresentação de x.

Além das informações de causalidade, o arquivo de controle contém as informações relativas aos parâmetros de QoS de cada mídia contínua. Três parâmetros, sendo eles *delay*, *jitter*, e *lossrt*, foram definidos. A escolha desses parâmetros baseou-se nos estudos realizados em [Hafi98]. Sobre estes parâmetros deve ser implementado um mecanismo de monitoramento que irá notificar uma violação de seus valores durante a recepção dos fluxos das respectivas mídias.

Para a especificação do arquivo de controle, foi definida uma nova linguagem, *SMIL Control Language* (SCL), a qual permite a coerente estruturação das informações de causalidade entre as mídias de uma apresentação. Para definir esta linguagem, foi utilizada a meta-linguagem *eXtensible Markup Language* (XML) [Bray98]. O XML é uma linguagem reconhecida e suportada publicamente para o desenvolvimento de linguagens. Muitas bibliotecas de classes e até mesmo ferramentas vêm sendo desenvolvidas de forma a dar suporte à definição de linguagens em XML (*parser, generators, ...*).

Na figura 3, um trecho de um arquivo contendo a especificação em SMIL 1.0 de uma aplicação multimídia é apresentado. Na mesma figura, é ilustrada a especificação dos relacionamentos de causalidade para algumas mídias pertencentes à apresentação feita através do arquivo de controle. O trecho exibido da aplicação SMIL contém uma apresentação sequencial de dois conjuntos de mídias. O primeiro conjunto, “exp1”, possui o áudio “a1” e o vídeo “v1”, que devem ser apresentados em paralelo.



Figura 3- Arquitetura de Restituição da Informação

No arquivo de controle ilustrado na figura 3, o elemento “med” faz referência aos objetos de mídia declarados no arquivo SMIL correspondente. Dentro deste são definidos os parâmetros de QoS e os relacionamentos causais referentes a cada mídia. O parâmetro “stoplink” estabelece um link condicional para que se interrompa a apresentação da mídia. No caso do elemento de áudio “a1”, o “stoplink” estipula que sua exibição seja cancelada caso a apresentação de “v1” seja interrompida. Um relacionamento condicional recíproco para “v1” também foi definido.

A tag “newmed” é utilizada na declaração de um novo elemento. O elemento “newmid” também apresenta parâmetros de QoS e relacionamentos causais. Na definição de um novo elemento composto “expls” o parâmetro “startlink” estabelece um link condicional para o início da apresentação deste objeto. O valor “(expl:stoped)” indica que, caso seja detectado que o elemento “par” “expl” foi interrompido (este fato ocorre quando todas as mídias contidas neste elemento são interrompidas), a apresentação do elemento “expls” deve ser iniciada.

### 3. ARQUITETURA DO SISTEMA

A arquitetura do sistema descrita neste trabalho é baseada no paradigma cliente-servidor. Os elementos dessa arquitetura, conforme a figura 4, são: (i) os servidores de mídia (*Media Servers*) – responsáveis pelo armazenamento das mídias e dos documentos que fazem referência às mesmas, e (ii) os clientes (*Media Client*) - responsáveis pela apresentação do documento multimídia, podendo apresentar diferentes tipos de características de sistema (terminal, acesso à rede, etc.).

Nesta arquitetura, os elementos de rede intermediários, presentes no caminho de transmissão entre o cliente e os servidores, são desconsiderados. Todo o controle dos parâmetros de QoS é efetuado nos *edges* (pontas) do sistema. Esta política foi adotada devido à sua simplicidade, se comparada às arquiteturas que dependem dos nós intermediários do sistema distribuído.

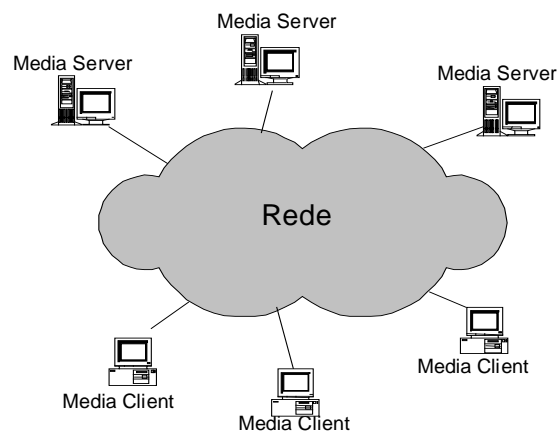


Figura 4- Ambiente Cliente-Servidor

Na figura 5 são ilustrados os principais módulos do sistema, presentes no *Media Client* e no *Media Server*. Os tipos de mensagens geradas pela comunicação entre esses módulos estão descritas na figura 6.

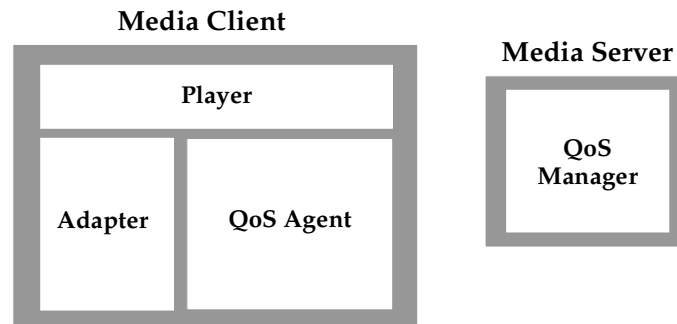


Figura 5- Principais módulos do sistema

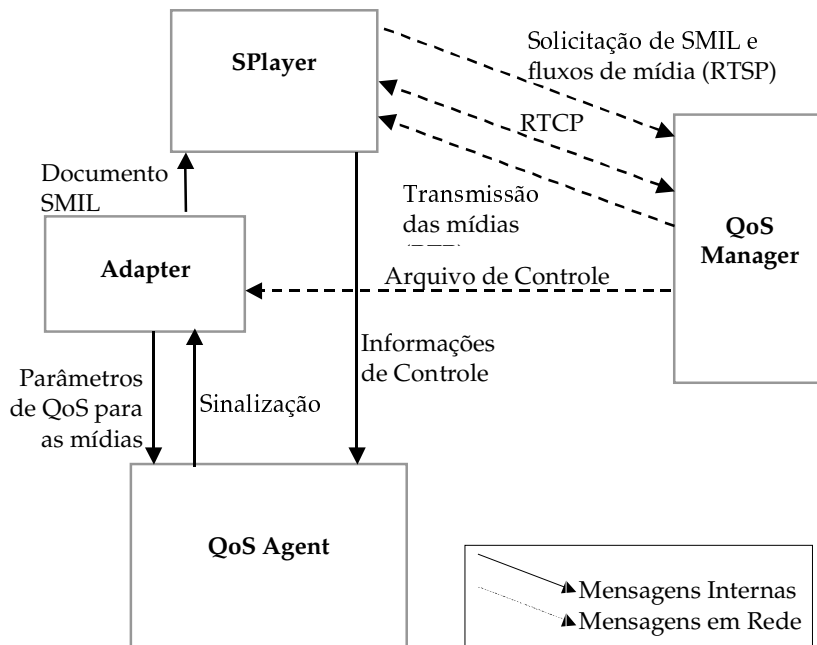


Figura 6- Esquema de comunicação entre os módulos

### 3.1 SPlayer

O *SPlayer* é uma aplicação Java para a apresentação de documentos SMIL 1.0. Para o desenvolvimento de um protótipo, os seguintes *packages*, ou bibliotecas de classes, estão sendo utilizados: *Java Media Framework (JMF)* [JMF98], uma *application programming interface (API)* desenvolvida pela *Sun Microsystems*; e o *JMF for RealSystem G2*, que adiciona ao JMF da *JavaSoft* a habilidade de tratar tipos de dados proprietários da *RealNetwork* e o próprio SMIL.

Em um instante inicial, o usuário solicita através do *SPlayer* a apresentação de um documento multimídia localizado em um dos servidores. Sequencialmente, após receber o documento SMIL, o *SPlayer* faz a solicitação aos servidores (via RTSP [Schu98]) da transmissão (via RTP [Schu98]) das mídias referenciadas pelo documento SMIL. Estas solicitações são realizadas sob-demanda, de acordo com o escalonamento da apresentação.

O *SPlayer* é uma ferramenta que comunica-se tanto com o *QoS Manager* quanto com qualquer servidor *RealServer* (ou outros servidores de mídia que implementem os protocolos



RTSP e RTP/RTCP). Desta forma, no instante da autoria do documento, o autor não fica restrito às mídias armazenadas nos servidores do sistema.

### 3.2 QoS Manager

O primeiro papel deste módulo é transmitir para o *Adapter* o arquivo de controle associado ao documento SMIL. Quando um documento SMIL é solicitado por algum *player* (qualquer ferramenta de apresentação multimídia), o *QoS Manager* verifica se existe algum documento SCL associado ao mesmo (a primeira linha de comentário do documento SMIL faz esta referência). Em caso positivo, este arquivo de controle é enviado (utilizando o protocolo TCP) através de uma porta conhecida. Caso o cliente seja um *SPlayer*, o *Adapter* irá receber o arquivo, caso contrário este último será descartado.

O *QoS Manager* também é responsável por todas as sessões RTP/RTCP referentes aos fluxos de mídias transmitidos aos clientes. Ele realiza um monitoramento sobre esses fluxos de forma a controlar o chaveamento entre as variantes de QoS para um mesmo fluxo (figura 2), executando a adaptação *soft*. Supondo, por exemplo, que existam três variantes de QoS para um mesmo vídeo (30 frames/s, 20 frames/s e 10 frames/s), o *QoS Manager* transmite, inicialmente, a variante de melhor qualidade. Se for detectado que os requisitos de QoS para a respectiva variante estão sendo violados, a variante de qualidade média passa a ser transmitida (a situação inversa pode ocorrer quando as condições do sistema melhorarem).

O controle do chaveamento entre as variantes de QoS é realizado com base nas informações intercambiadas nas sessões RTCP de cada fluxo de mídia. Em uma sessão RTP/RTCP, diferentes parâmetros de QoS relevantes são informados ao sistema. Nesta sessão, o fluxo de dados RTP é recebido por uma porta  $n$  e as informações concernentes ao controle da sessão são recebidas pela porta  $n+1$ . A RFC 1889 [Schu96] define cinco tipos de pacotes RTCP: Receiver Report (RR); Sender Report (SR); Source DEscription; BYE, indicando o fim da participação na sessão; e, finalmente, um pacote especial APP previsto para outras aplicações. Uma política de gerência de QoS pode ser implementada com base nos parâmetros observáveis (definidos nos campos do protocolo RTCP) que permitem o monitoramento do jitter, da perda e do atraso ponto a ponto.

O *QoS Manager* apresenta uma base de informações composta pelas variantes de QoS pertencentes a cada mídia. Nesta base de informações, estão presentes os limites aceitáveis de *delay*, *jitter* e *loss* para cada variante. Durante a transmissão de um determinado fluxo RTP, há um controle dos parâmetros informados pelo receptor deste fluxo através dos pacotes RTCP. Este controle é feito com base nos limites definidos para a variante de QoS que está sendo transmitida. A violação de algum desses limites implica na substituição da variante corrente por uma variante com requisitos de QoS menos exigentes, caso esta outra exista. Deve-se observar que se a variante transmitida for a mais “pobre” e, mesmo assim, houver uma violação de algum de seus parâmetros, o *QoS Management* não realiza mais nenhuma ação.

### 3.3 QoS Agent

O *QoS Agent* tem como propósito o monitoramento dos fluxos de mídia pertencentes a uma apresentação, mantendo para cada fluxo um estado sobre o qual é efetuado este monitoramento. Este módulo é responsável por controlar os estados de cada mídia ativa de uma apresentação. Para isso utiliza-se uma tabela com uma entrada para cada fluxo recebido

pelo *player*. Em cada entrada gera-se um par de triplas  $(jitter_t, delay_t, lossrt_t)$   $(jitter_s, delay_s, lossrt_s)$ : a primeira tripla indica os limites superiores para os parâmetros de QoS determinados pelo autor do documento multimídia (dados informados pelo *Adapter*); a segunda tripla informa o estado corrente dos parâmetros do respectivo fluxo de mídia.

Os dois conjuntos de parâmetros mantidos pelo *QoS Agent* podem ser representados em um espaço multidimensional (figura 7), onde cada eixo corresponde a um parâmetro. O conjunto que indica os limites para os parâmetros define um paralelepípedo neste espaço, dentro do qual o vetor de estado (o conjunto de parâmetros que informa o estado corrente do fluxo) pode mover-se. A extrapolação dos parâmetros de QoS é indicada pela presença desse vetor fora dos limites estabelecidos pela superfície do paralelepípedo.

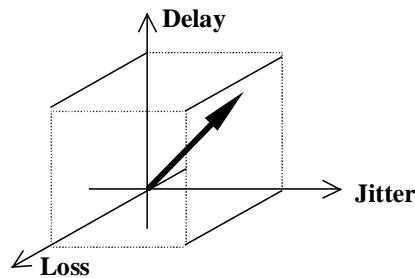


Figura 7- Conjunto de parâmetros

A atualização do estado do fluxo de cada mídia, mantido pelo *QoS Agent*, é realizada sempre que um pacote RR é gerado. O *SPlayer*, no instante em que são calculados os valores dos parâmetros de QoS, informa-os ao *QoS Agent*, de forma que o mesmo possa atualizar o estado do respectivo fluxo de mídia.

### 3.4 Adapter

Este módulo apresenta duas funcionalidades importantes dentro da arquitetura do sistema:

- (i) receber e interpretar o arquivo de controle associado ao respectivo documento multimídia, no instante inicial de uma apresentação; as informações de QoS obtidas deste arquivo são repassadas ao *QoS Agent*, que irá utilizá-las para o monitoramento dos fluxos de mídias recebidos pelo *Player*;
- (ii) reestruturar um novo documento SMIL, ao ser sinalizado pelo *QoS Agent*, efetivando o segundo mecanismo de adaptação (adaptação *hard*); no instante em que o *QoS Agent* detecta a violação dos requisitos de QoS definidos para uma determinada mídia, ele informa ao *Adapter* o “id” e o instante de tempo em que ocorre essa violação (valor relativo ao início da apresentação desta mídia); baseado nas informações obtidas da especificação SCL, a estrutura lógica do documento SMIL é reformulada, gerando-se *on-the-fly* um novo documento SMIL referente a uma nova apresentação multimídia.

A construção do novo arquivo SMIL parte de alterações realizadas sobre o documento multimídia original. Essas alterações são definidas de acordo com os dados sinalizados pelo *QoS Agent* e com os relacionamentos condicionais especificados no arquivo SCL.

A figura 8 ilustra o processo de adaptação no momento em que ocorre uma *QoS Failure* na apresentação do vídeo "V". Esta falha é detectada após 80s decorridos do início da apresentação desta mídia. O mecanismo de adaptação é ativado quando o *QoS Agent* informa ao *Adapter* que a mídia "V" sofreu uma *QoS Failure* no instante 80s (100s – 20s). O novo documento SMIL é construído e passado ao *Player*, que reinicia a apresentação partir do mesmo instante em que ocorreu a falha.

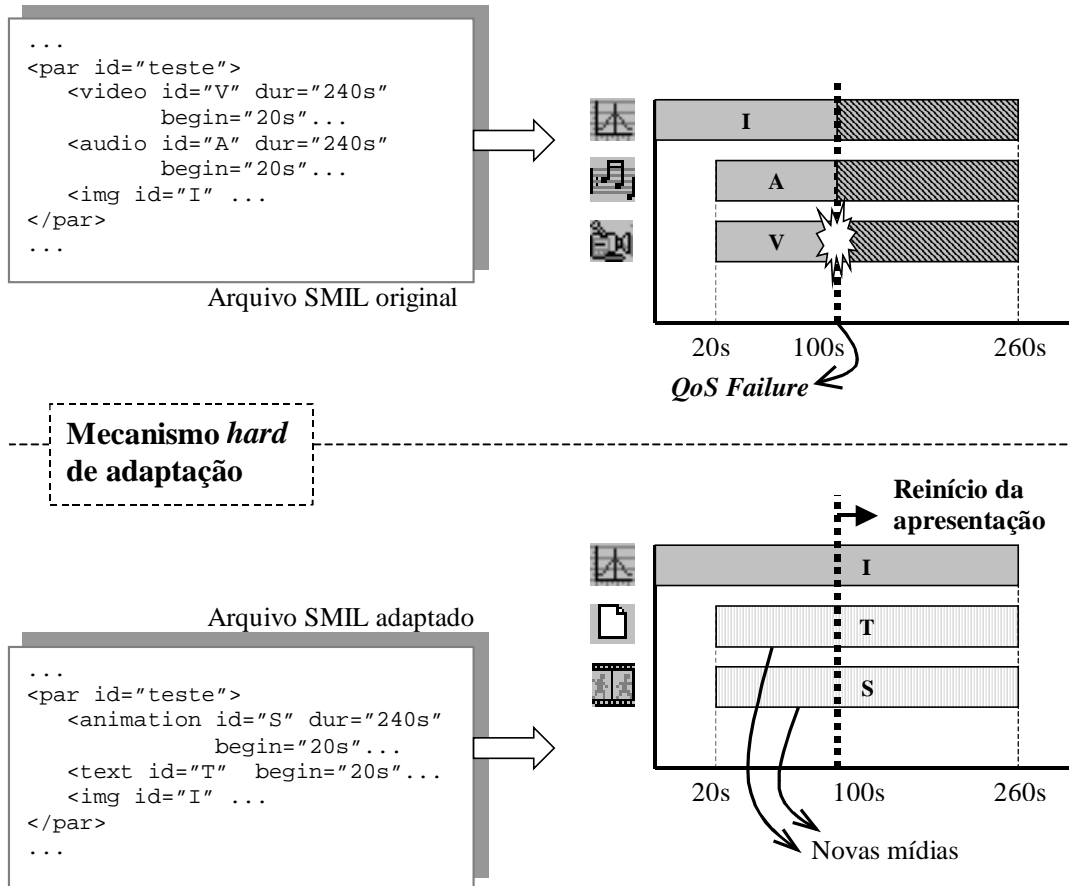


Figura 8- Adaptação *hard*

O novo documento SMIL é repassado ao *Player*, que substitui a apresentação corrente pela nova apresentação. Primeiramente, o usuário é informado de que uma falha de QoS ocorreu e que uma nova apresentação multimídia está sendo preparada. Enquanto isso, os fluxos pertencentes à "antiga" apresentação são cancelados e os novos fluxos são solicitados aos servidores. O SMIL permite que se especifique a partir de que instante uma apresentação deve ser iniciada, com isso, a nova apresentação só é executada a partir do ponto em que ocorreu a falha de QoS. O cálculo deste instante é realizado com base em parâmetros de escala definidos pelo autor no arquivo de controle.

A biblioteca (*package*) Java XML4J [Xml4j], disponibilizada pela IBM, está sendo utilizada na implementação deste módulo. Este *package* oferece ferramentas de interpretação (*parsers*) e geração de saídas XML. A estrutura deste módulo é dividida, basicamente, em dois sub-módulos: um *parser* XML e um gerenciador. O *parser* XML é responsável pela interpretação das *tags* descritas no arquivo de controle, bem como pela geração dos documentos SMIL. O sub-módulo gerenciador mantém os dados de controle extraídos destas *tags*. Este

gerenciamento consiste na identificação das informações (associação dessas informações de controle com as mídias definidas no arquivo SMIL original e com as novas mídias definidas apenas no arquivo de controle), formatação das informações em parâmetros que são transferidos ao *QoS Agent*, e tomada de decisão quanto à estrutura lógica que o novo documento SMIL, gerado no momento da adaptação, deve apresentar.

#### 4. UMA APLICAÇÃO EM ENSINO À DISTÂNCIA

Para avaliar o uso de sistemas multimídia adaptativos, está sendo desenvolvida uma aplicação dentro do contexto de ensino à distância. Esta aplicação consiste de uma demonstração elementar do fenômeno de eletrização sob a forma de um experimento físico.

O processo de autoria deste *testcase* envolve três diferentes etapas: (i) definição da estrutura lógica e dos relacionamentos semânticos relativos às mídias pertencentes à apresentação; (ii) criação dos arquivos referentes às variantes de QoS de cada mídia; e (iii) especificação dos arquivos SMIL e SCL (arquivo de controle).

A aplicação consiste de uma apresentação sequencial de dois conjuntos de mídias, ilustradas na figura 9. Cada conjunto corresponde a um experimento físico. A execução do primeiro experimento é efetuada através da apresentação de um áudio e um vídeo em paralelo. Com a finalização do vídeo, uma imagem é apresentada até que o áudio seja interrompido. O segundo experimento é representado por um áudio e um vídeo em paralelo.

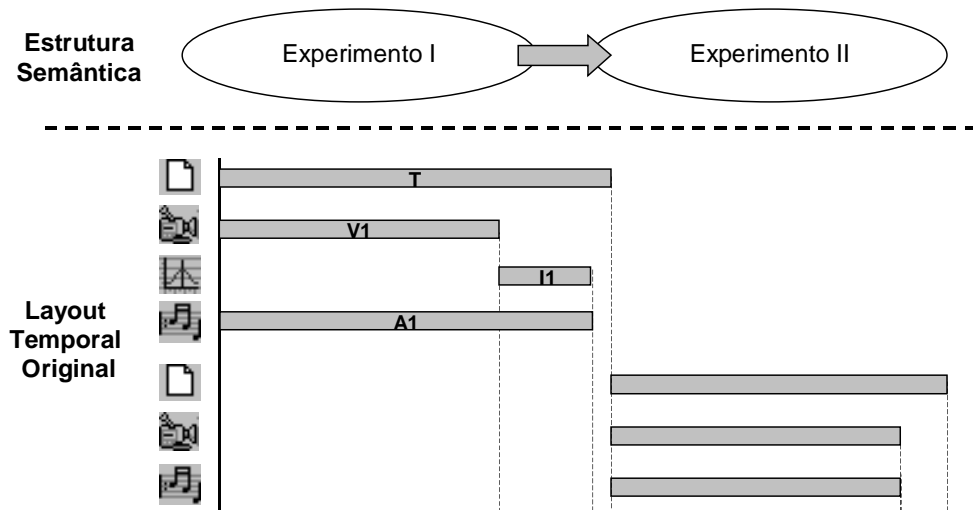


Figura 9- Estrutura da Aplicação Teste

O layout temporal apresentado na figura 9 refere-se à apresentação original definida no documento SMIL. No arquivo SCL são definidas as mídias redundantes, os relacionamentos condicionais e os parâmetros de QoS necessários para a execução dos mecanismos de adaptabilidade durante a exibição da aplicação. As novas mídias “S1” e “T1”, *slideshow* e texto respectivamente, são declaradas no arquivo de controle. Os relacionamentos condicionais envolvendo essas novas mídias e as mídias já referenciadas no arquivo SMIL devem ser especificados obedecendo-se as seguintes condições:

- (i) se o vídeo “V1” é interrompido, ele deve ser substituído pelo *slideshow* “S1”;

- (ii) se o áudio “A1” é interrompido, a apresentação de “V1” também deve ser interrompida; “A1” e “V1” devem ser substituídos por “T1” e “S1”, respectivamente.

A figura 10 ilustra os relacionamentos causais estabelecidos entre as mídias que compõem o primeiro experimento. O *startlink* entre “V1” e “S1” indica que, caso “V1” seja interrompido, apenas “S1” é ativado. O *stoplink* entre “A1” e “Seq1” descreve que a apresentação deste último deve ser finalizada se “A1” for interrompido. O *startlink* entre “Seq1” e “Par1” define que caso o primeiro seja interrompido o segundo é iniciado.

Para a execução do mecanismo de adaptação soft, o vídeo “V1” é codificado com três taxas distintas: 30 frames/s, 20frames/s, 10frames/s. Da mesma forma, “A1” é codificado com taxas 14400Hz e 8000Hz.

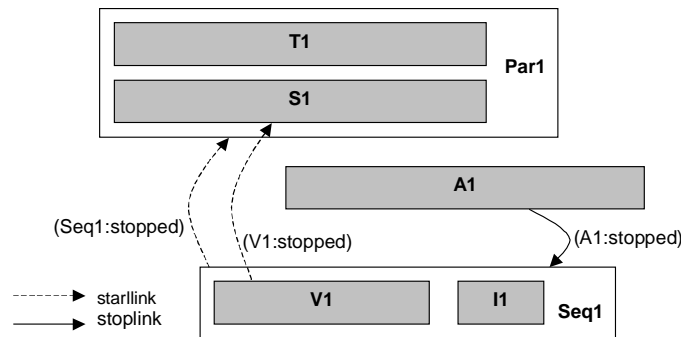


Figura 10- Relacionamentos causais da aplicação

As diferentes condições do sistema durante a apresentação desta aplicação pode levar a três cenários de execução distintos. No primeiro cenário, apenas o mecanismo *soft* de adaptação é efetuado. Inicialmente, as variantes do áudio e do vídeo com maiores níveis de qualidade são transmitidos pelo *QoS Manager*. Na ocorrência de alguma violação dos parâmetros de QoS especificados para as variantes das mídias, o *Media Server* faz uma *QoS Degradation* alternando a transmissão para uma variante com qualidade menor do que a corrente. Neste caso, a apresentação, do ponto de vista do *Media Client*, não sofre nenhuma alteração.

Em um segundo cenário, o mecanismo soft não é suficiente para garantir que o estado da mídia “V1” permaneça entre os limites estipulados no arquivo de controle. Isto significa que a tripla ( $jitter_s, delay_s, lossrt_s$ ) ultrapassa, em algum campo, a tripla ( $jitter_l, delay_l, lossrt_l$ ) na entrada da tabela referente ao vídeo. Com isso, o *QoS Agent* comunica o ocorrido ao *Adapter* informando o “id” do vídeo e o instante  $t_f$  em que ocorreu a falha de QoS. Este último adapta o documento SMIL, que é exibido pelo *Player* a partir de  $t_f$  (figura 11).

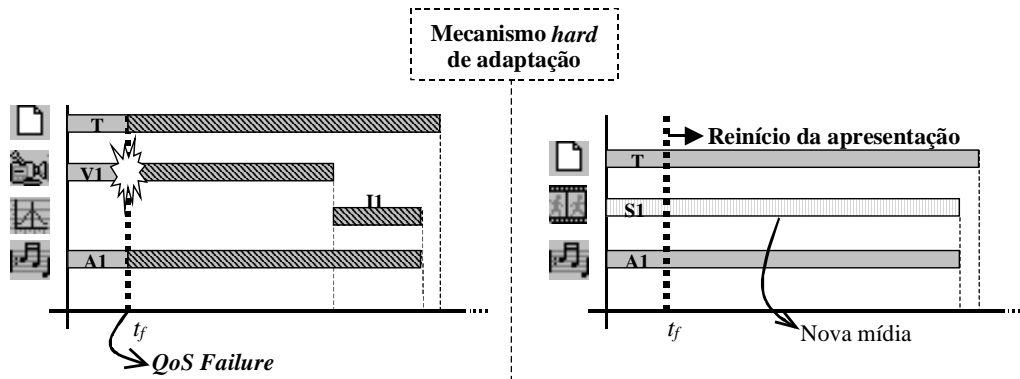


Figura 11- Segundo cenário de adaptação.

No terceiro cenário, a escassez de recursos apresenta-se de tal forma que os parâmetros da mídia “A1” ultrapassem os limites estipulados no arquivo de controle. A interrupção da apresentação de “A1” é detectada pelo *QoS Agent*, que sinaliza ao *Adapter* o “id” do áudio e o instante  $t_f$  de falha. Da mesma forma que no cenário anterior, um novo documento é estruturado para que o *Player* mude a apresentação multimídia (figura 12).

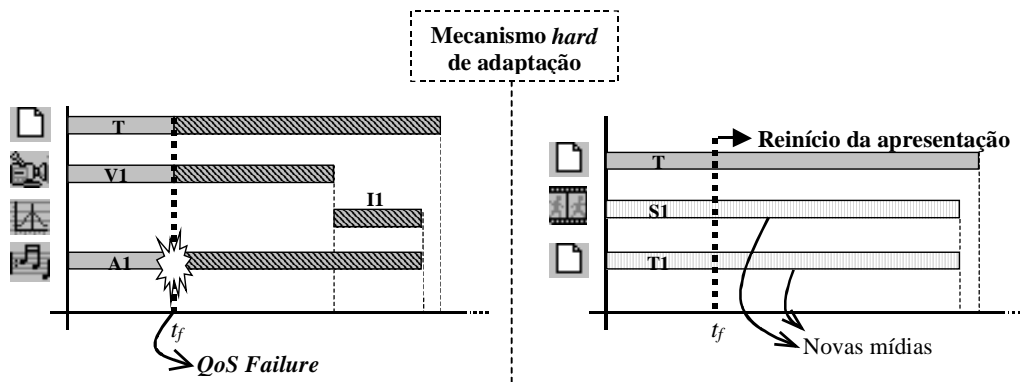


Figura 12- Terceiro cenário de adaptação.

## 5. CONCLUSÕES

Neste artigo foram apresentados dois mecanismos de adaptação para a apresentação de documentos multimídia em sistemas distribuídos e a respectiva arquitetura de implementação: (i) o mecanismo suave, que efetua uma adaptação a nível de codificação de mídias independentes de forma transparente ao usuário; e (ii) o mecanismo forte, que leva o conceito de adaptabilidade ao nível do documento, gerando uma nova estrutura lógica e temporal em tempo de execução.

Para garantir a interoperabilidade com sistemas multimídia já existentes, as informações de controle de adaptabilidade (formato SCL) foram mantidas externas ao documento multimídia original (formato SMIL 1.0). Este tipo de abordagem simplifica muito o desenvolvimento de um *player* para documentos adaptativos (*SPlayer*) e implica naturalmente em três grandes vantagens: (i) a apresentação do documento multimídia pode ser efetuada por um *player* SMIL, mesmo que este não seja capaz de considerar as informações de controle; (ii) não existe a necessidade do desenvolvimento integral de uma nova ferramenta de apresentação, já

que módulos de outras ferramentas já existentes podem ser aproveitados no desenvolvimento do *SPlayer*; e (iii) torna o sistema independente de novas versões do SMIL.

O emprego dos protocolos RTSP e RTP/RTCP foi um grande facilitador no desenvolvimento desta arquitetura adaptável pois permite que se realize o monitoramento em tempo-real da comunicação através dos parâmetros de qualidade de entrega das mídias existentes nas unidades de controle destes protocolos.

A realização de uma adaptação em dois níveis permite que a maioria das aplicações não apresentem tempo de chaveamento significativo. O primeiro nível de adaptação, mecanismo *soft*, possui uma latência muito baixa. O mecanismo de adaptação *hard* só deverá ser ativado nas situações em que o primeiro mecanismo não conseguir manter os requisitos de QoS especificados. Apesar do tempo de resposta da adaptação *hard* ser implicitamente superior ao tempo de adaptação do mecanismo suave, em um ambiente de ensino à distância este atraso é bastante aceitável pois as dimensões temporais das mídias acessadas neste tipo de aplicação são grandes (muitas vezes superior ao atraso). Além disso, quando lidamos com aplicações voltadas ao ensino é melhor a convivência com um pequeno atraso de chaveamento do que a possibilidade de uma interpretação equivocada das informações.

Como trabalho futuro, pretende-se estudar o uso de mecanismos para o monitoramento dos recursos locais, como taxa de utilização de processador e memória. Desta forma, novas opções para a determinação dos parâmetros de QoS podem ser definidas. No que diz respeito aos mecanismos de adaptação, uma proposta de refinamento na sincronização do reinício do documento adaptado também está sendo estudada. Uma consequência direta deste trabalho será a posterior incorporação na ferramenta de autoria desenvolvida no projeto Servimídia [Cunh99a, Cunh99b, Cunh99c] de toda a metodologia de criação de documentos adaptativos que está sendo gradualmente amadurecida. O objetivo é tornar esta difícil tarefa do ponto de vista do autor o mais transparente possível.

## 6. REFERÊNCIAS

- [Bray98] Bray, T., Paoli, J., Sperberg-McQueen, C. M. "Extensible Markup Language (XML) 1.0". *W3C Recommendation, REC-xml-19980210*, Feb. 1998.
- [Bult91] Bulterman, D.C.A., van Liere, R., van Rossum, G., "A Structure for Transportable Dynamic Multimedia Documents", *Proc. of 1991 Usenix Spring Conference on Multimedia Systems*, pp. 137-155, 1991.
- [Bult93] Bulterman, D.C.A., "Synchronization of Multi-Sourced Multimedia Data for Heterogeneous Target Systems", *Network and Operating Systems Support for Digital Audio and Video, LNCS-712, Springer-Verlag*, pp.119-129, 1993.
- [Cunh99a] Cunha, E.C.C., Carmo, L.F.R.C., Pirmez, L., "Design of an Integrated Environment for Adaptive Multimedia Document", *6th International Workshop on Interactive Distributed Multimedia Systems and Telecommunication Services, LNCS-1718, Springer-Verlag*, pp. 191-204, 1999.
- [Cunh99b] Cunha, E.C.C., Carmo, L.F.R.C., Pirmez, L., "A Multimedia Document Authoring Strategy for Adaptive Retrieval in a Distributed System", *Proc. of 6th International Conference on Distributed Multimedia Systems*, University of Aizu-Japan, Jul. 1999.

- [Cunh99c] Cunha, E.C.C., Carmo, L.F.R.C., Pirmez, L., "A Stream Relationship Monitor for Adaptive Multimedia Document Retrieval", *Globecom'99*, Rio de Janeiro, Dezembro 1999.
- [Hafi98] Hafid, A., Bochman, G. "Quality of Service adaptation in distributed multimedia applications", *Multimedia Systems*, pp.299-315, 1998.
- [JMF98] Java Media Framework API,  
<[http://java.sun.com/marketing/collateral/jmf\\_ds.html](http://java.sun.com/marketing/collateral/jmf_ds.html)>
- [Schu96] Schulzrinne, H., Casner, S., Frederick, R., Jacobson, V. "RTP: a transport protocol for real-time applications," RFC 1889, Internet Engineering Task Force, Jan. 1996.
- [Schu98] Schulzrinne, H., Rao, A., Lanphier, R. "Real Time Streaming Protocol (RTSP)", RFC 2326, Apr. 1998.
- [Smil98] W3C, "Synchronized Multimedia Integration Language (SMIL) 1.0 Specification", W3C Recommendation, Jun. 1998.
- [Wadd98] Waddington, D.G., Hutchison, D., "End-to-end QoS Provisioning through Resource Adaptation", *Proc. of IFIP Conference on High Performance Networking (HPN'98)*, Sep. 1998.
- [Wahl95] Wahl, T., Wirag, S., Rothermel, K., "TIEMPO: Temporal Modeling and Authoring of Interactive Multimedia", *Proc. of IEEE 2nd Intl. Conference on Multimedia Computing and Systems*, pp. 274-277, May 1995.
- [Wira97] Wirag, S., "Modeling of Adaptable Multimedia Documents", *Proc. Interactive Distributed Multimedia Systems and Telecommunication Services (IDMS'97)*, pp. 420-429, Sep. 1997.
- [Xml4j] XML Parser for Java, < <http://www.alphaworks.ibm.com/formula/xml>>