

3

Autoria em NCL para TV digital interativa

O editor HyperProp (Coelho & Soares, 2004) foi projetado inicialmente como um ambiente de autoria para criação de aplicações hipermídia em NCL *main profile* (Soares et al., 2006). A fim de se adequar ao contexto de TV digital, essa ferramenta passou por diversas modificações, o que culminou na criação do *Composer*.

Com base nos pontos positivos e negativos de cada uma das ferramentas analisadas, são listados alguns requisitos e/ou aspectos desejáveis para o *Composer* no âmbito de TV digital interativa:

- Necessidade de representação de documentos contendo objetos de duração indeterminada, como conteúdos produzidos ao vivo ou adaptados em função do contexto da apresentação do documento, e eventos de interatividade sobre os objetos de mídia apresentados;
- Exibição das cadeias parciais (trechos de duração previsível) e interações entre elas;
- Suporte à simulação de eventos interativos e adaptação de conteúdo em tempo de criação;
- Exibição das composições com semânticas temporais;
- Suporte à edição espaço/temporal;
- Possibilidade de exibição de um documento a partir de qualquer ponto;
- Suporte à edição ao vivo de documentos NCL;
- Elaboração de uma interface gráfica que facilite e agilize a criação de programas NCL para TV digital.

Este capítulo apresenta novidades do *Composer* para a criação de programas NCL, versão 3.0 (Soares & Rodrigues, 2006), para TV digital interativa.

3.1. Arquitetura do *Composer*

Um ambiente de autoria hipermídia deve oferecer recursos para que os usuários, no caso os autores, editem os hiperdocumentos, especificando, por exemplo, os seus componentes, as propriedades desses componentes, os relacionamentos internos existentes no documento e, possivelmente, relacionamentos externos, isto é, com componentes pertencentes a outros documentos. É importante ainda que no ambiente de autoria o autor tenha ferramentas que permitam especificar as características de exibição para cada um dos componentes (Rodrigues & Soares, 2003).

Da mesma forma que no editor HyperProp, no *Composer* as abstrações são definidas nos diversos tipos de visões que permitem simular um tipo específico de edição. O *Composer* é composto por quatro visões: uma visão gráfica estrutural, uma visão gráfica temporal, uma visão gráfica de leiaute e uma visão textual. As quatro visões funcionam de maneira sincronizada, a fim de oferecer um ambiente integrado de autoria. A Figura 12 apresenta as visões do *Composer*.

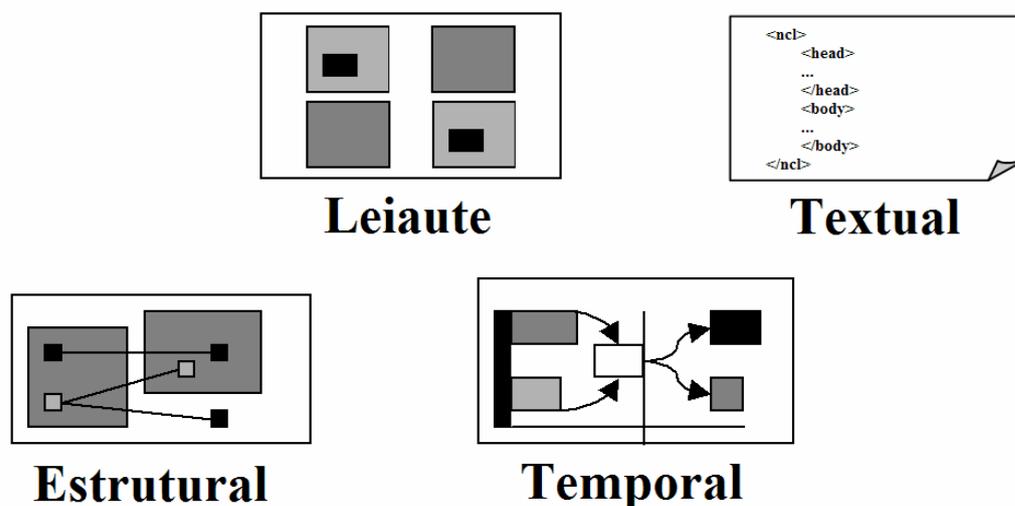


Figura 12 - Visões do *Composer*

A visão estrutural permite ao autor criar a estrutura lógica do documento NCL, ou seja, nela o autor pode criar, editar e apagar composições, objetos de mídia e elos. Na visão estrutural, os nós de contexto e os nós de mídia são representados por vértices de um grafo (compostos e atômicos, respectivamente). Já os elos, são representados por arestas entre vértices do grafo.

No editor HyperProp, a visão espacial, ainda que não implementada completamente, previa a possibilidade de especificação de leiautes, e a edição de

relacionamentos espaciais (Coelho & Soares, 2004). No *Composer*, a definição das regiões deixou de ser feita na visão espacial, ficando a cargo da visão de leiaute. Através dessa visão, o autor pode criar, editar e apagar regiões, assim como manipulá-las graficamente. Quando uma região é movimentada, todas as suas sub-regiões (regiões filhas) também o são, mantendo suas posições relativas. A visão espacial, ainda que não implementada na atual versão do *Composer*, deve permitir a definição e edição de relacionamentos espaciais entre componentes de um documento. A visão espacial será discutida posteriormente como trabalho futuro (Subseção 5.2).

Já a visão temporal, um dos principais focos deste trabalho, permite a especificação e edição de relacionamentos de sincronização temporal entre entidades de um documento NCL, definindo suas posições relativas no tempo. Por fim, a visão textual (declarativa) permite ao autor editar diretamente o código NCL do documento no qual ele está trabalhando.

Para cada visão gráfica, diferentes elementos da linguagem NCL ganham destaque. Por exemplo, os elementos *region* são os elementos fundamentais utilizados na visão de leiaute. Já na visão temporal, os relacionamentos temporais, juntamente com as âncoras temporais presentes nos nós de mídia (por exemplo, um nó de áudio), são entidades chaves para a visualização do documento no eixo do tempo. No caso da visão estrutural, os nós de composição ganham destaque, pois são responsáveis pelo agrupamento lógico das entidades presentes no documento.

É desejável que o usuário tire proveito de visões distintas em função do contexto de autoria mais relevante. Por exemplo, em um dado momento da edição, pode ser mais útil trabalhar com a visão estrutural dos componentes. Em outras circunstâncias, é possível que seja mais apropriado trabalhar na visão temporal, mais próxima do formato final de apresentação do documento. É importante observar que essas visões não precisam, e muitas vezes não devem, funcionar isoladamente. Na maioria dos casos, as diferentes visões podem ser consideradas como dimensões que se complementam na constituição do documento hipermídia (Rodrigues & Soares, 2003).

Para a sincronização entre as visões do *Composer*, este trabalho propõe a arquitetura de integração apresentada na Figura 13, a qual deriva da arquitetura utilizada em (Coelho & Soares, 2004). A diferença principal fica por conta do

modelo de dados utilizado para integração entre as visões do *Composer* ser o Java NCM (Silva et al., 2004), e não diretamente a árvore DOM (*Document Object Model*) (W3C, 2004c), do editor HyperProp.

O modelo DOM é utilizado para armazenar e manipular árvores XML, no caso, a árvore que descreve a estrutura do documento NCL. Além de lançar mão das vantagens da estruturação do documento NCL através da árvore DOM, o modelo Java NCM fornece uma API de mais alto nível para manipulação das entidades do modelo NCM. Com isso, tem-se uma estrutura de dados mais robusta para integração entre as visões.

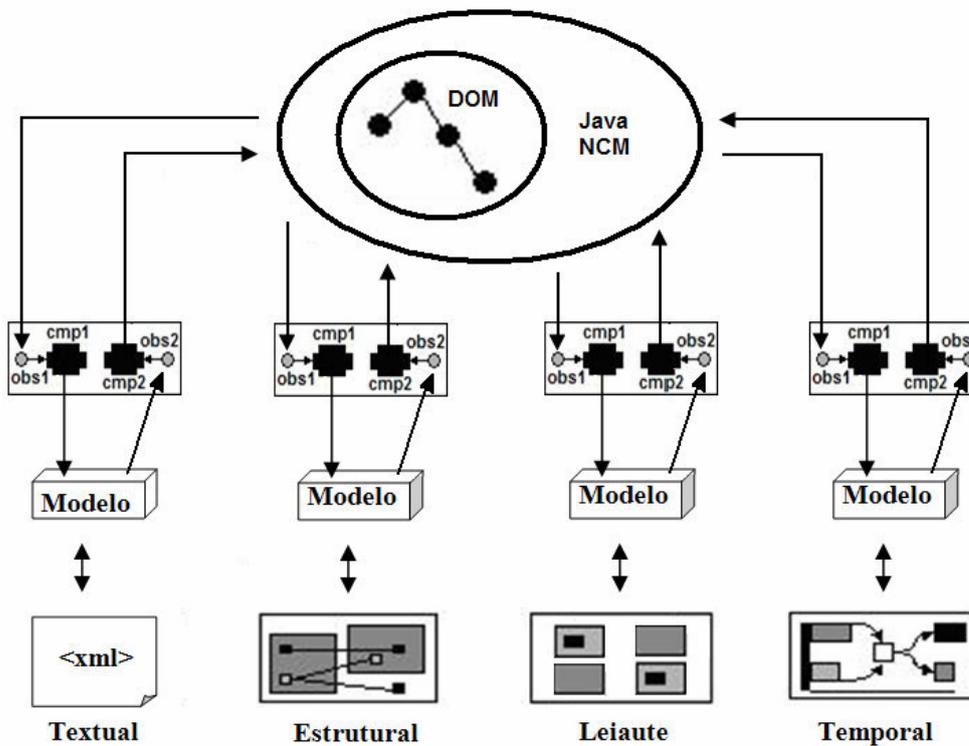


Figura 13 - Integração entre visões no *Composer*

Para cada visão, existe um par de compiladores responsáveis, respectivamente, por traduzir a representação de um documento NCL no modelo Java NCM para estrutura de dados utilizada na visão, e por converter essa estrutura de dados na representação do modelo de integração, no caso o Java NCM.

Além dos compiladores, cada visão tem um par de observadores (um para cada compilador). Sempre que o modelo Java NCM é modificado, os observadores *obs1* (vide Figura 13) são notificados e acionam os respectivos compiladores para que a partir do modelo Java NCM seja gerado um novo modelo para cada uma das visões.

Por sua vez, os observadores *obs2* (Figura 13) são notificados caso haja alterações no modelo da visão em questão (resultantes de ações do usuário sobre a ferramenta de edição) e acionam os compiladores correspondentes para que reflitam essas mudanças no modelo de integração.

Apesar de possuir uma arquitetura de visões muito próxima da já existente no editor HyperProp, o *Composer* teve grande parte da sua interface gráfica remodelada, a fim de facilitar e agilizar a criação de aplicações voltadas para TV digital. Além de tornar essa ferramenta mais atrativa, a nova interface visa ajudar o autor, permitindo que ele possa se concentrar, principalmente, no processo de criação. Para isso foi realizada uma avaliação formativa, por inspeção, por uma equipe de IHC (Interface Homem-Computador) de toda a interface do editor HyperProp e do *Composer*.

O editor HyperProp é um ambiente de autoria que está muito aquém de atender a demanda dos mais variados tipos de usuários. Isso se deve em grande parte à falta de diálogo do ambiente com seu parceiro mais importante, o autor. Se por um lado essa ferramenta atende satisfatoriamente a usuários que conhecem profundamente a linguagem, por outro, não dá suporte a usuários leigos em NCL. O *Composer* visa ajudar o autor no processo de autoria de aplicações voltadas para TV digital abstraindo toda, ou pelo menos parte da complexidade de se programar em NCL. Para isso, além da abstração de visões, o *Composer* fornece outros mecanismos de suporte ao usuário, como por exemplo, mecanismos de ajuda e mensagens de erro, conforme será apresentado no próximo capítulo.

3.2. Sincronismo Temporal

Segundo (Bulterman & Hardman, 2005), a autoria gráfica através de uma visão temporal é a melhor maneira de mostrar quando os objetos de mídia são exibidos, bem como de visualizar os relacionamentos entre eles. No entanto, devido às restrições da representação no tempo, geralmente, a visão temporal é aplicável somente na autoria de documentos não interativos e não adaptativos, por restrições do paradigma utilizado para exibição da sincronização.

Surge assim a necessidade de alternativas para contornar problemas de representação de eventos imprevisíveis, de forma a tornar a autoria através da

visão temporal, e todo ambiente *Composer*, mais atrativa para desenvolvedores de programas para TV digital.

Uma forma de solucionar o problema levantado é através da definição de uma estrutura de dados que permita a representação de documentos contendo objetos de duração indeterminada, como conteúdos produzidos ao vivo ou adaptados em função do contexto da apresentação do documento, e eventos de interatividade sobre os objetos de mídia apresentados.

Essa estrutura de dados deve ser capaz de preservar a representatividade do modelo NCM, e ainda favorecer a representação da apresentação na escala do tempo, permitindo que a visão temporal seja capaz de facilitar o processo de edição de hiperdocumentos.

A próxima subseção apresenta detalhes da estrutura de dados adotada como base para a visão temporal do *Composer*. Ao final, são feitas considerações sobre a utilização dessa estrutura nos ambientes HyperProp e *Composer*.

3.2.1. Modelo de Grafo de Cadeias Temporais Hiperímia

O modelo de grafo de cadeias temporais hiperímia (Costa & Soares, 2005) é formado por um conjunto de nós e arcos unidirecionais dirigidos, onde os nós são especializados em nós de início, nós de término e nós conectores. Vale ressaltar que toda a discussão sobre a representação de documentos NCL no modelo de cadeias temporais hiperímia é feita seguindo a máquina de estados de eventos definida na especificação da NCL versão 3.0 (Soares & Rodrigues, 2006).

Os nós de início podem, por exemplo, representar uma transição de início (*start*) ou de recomeço (*resume*) de um evento, enquanto os nós de término podem representar uma transição de término (*stop*) ou de pausa (*pause*). Já os nós conectores são utilizados como pontos de união em relacionamentos multiponto, conforme será apresentado em um exemplo mais adiante⁴.

Nesse modelo, os arcos podem ser utilizados, por exemplo, para representar o estado de ocorrendo (*occurring*) ou de pausado (*paused*) de um evento de

⁴ Nós conectores poderiam ser usados também em relacionamentos ponto-a-ponto, mas isso não é relevante para o modelo, uma vez que a não utilização desse tipo de nó para esses relacionamentos torna o modelo mais simples.

apresentação (*presentation*). O arco que representa o estado de ocorrendo tem origem em um nó de início (transição de começo ou recomeço) e destino em um nó de término (transição de término ou pausa). O estado de pausado, por sua vez, é representado por um arco com origem em um nó de término (transição de pausa), e destino em um nó de início (transição de recomeço).

Além de representarem o estado de ocorrendo e de pausado de eventos, os arcos podem ser utilizados para definir relacionamentos de causalidade. Os arcos causais unem nós (transições de eventos) no grafo. A transição origem do arco causal corresponde a uma condição que deve ser satisfeita. Já a transição destino corresponde a uma ação disparada em função de a condição definida na origem ter sido alcançada. Além dos relacionamentos causais, relacionamentos de restrição também podem ser representados. Não obstante, no contexto de TV digital, os relacionamentos de restrição não serão considerados⁵.

Outra característica importante é que, no modelo de cadeias temporais hipermídia, os arcos definem sempre intervalos de tempo, que podem ser referentes, por exemplo, à duração da apresentação de um objeto de mídia. Diferentemente dos arcos, o valor do tempo de execução de um nó é sempre absoluto. Na realidade, o valor do tempo de cada nó no modelo não é especificado, e sim calculado a partir da soma dos tempos do conjunto de arcos utilizados para alcançar o nó, a partir de um ponto inicial qualquer no grafo, conhecido como ponto de entrada.

Cabe ressaltar que a apresentação de um documento (Rodrigues & Soares, 2003) pode ser iniciada a partir de qualquer nó do grafo (um mesmo documento pode ser visto por diferentes perspectivas). Considerando que, em um grafo, um nó pode ser alcançado por vários caminhos, formados pela combinação de arcos, o valor do tempo atribuído a um nó será sempre o menor valor calculado. Esse valor corresponde ao caminho mais curto até o nó considerado, a partir de um ponto de entrada pré-definido.

Nesse modelo, os pontos de indeterminação são os limites entre as cadeias temporais hipermídia que formam o grafo. Esses pontos podem, por exemplo, indicar a existência de relacionamentos interativos, onde os tempos das

⁵ Para tornar a especificação dos relacionamentos mais simples, sem perder muito em expressividade, elos de restrição não são considerados no contexto de TV digital.

ocorrências não podem ser previstos antes da apresentação do documento. Os nós do grafo que só possuem arestas de entrada (incidentes) com origem em pontos de indeterminação não podem ter os tempos de suas ocorrências calculados. Em (Buchanan & Zelweger, 1992), uma cadeia temporal é definida como uma seqüência de exibições previsíveis, sendo que a cadeia formada a partir do início da execução do documento é definida como cadeia principal. No modelo de cadeias temporais hipermídia, a cadeia principal é aquela formada pelos nós onde são conhecidos os tempos de execução a partir do ponto de entrada escolhido.

Ainda em (Buchanan & Zelweger, 1992), as cadeias iniciadas por transições imprevisíveis na máquina de estados de algum evento são denominadas cadeias auxiliares. No modelo de cadeias temporais hipermídia, as cadeias auxiliares são aquelas onde os pontos de indeterminação, gerados por arcos cujo valor não é determinístico, impedem o cálculo dos tempos de execução dos nós. Quando os pontos de indeterminação em uma cadeia temporal auxiliar são resolvidos, essa cadeia deixa de existir e passa a integrar a cadeia principal.

A conversão de especificações temporais do modelo NCM para o modelo de cadeias temporais hipermídia, e vice-versa, é exemplificada na Figura 14. Essa figura apresenta um trecho de um documento NCL onde o objeto de mídia *audio1*, contendo duas âncoras temporais, é especificado e também representado no modelo de cadeias temporais hipermídia.

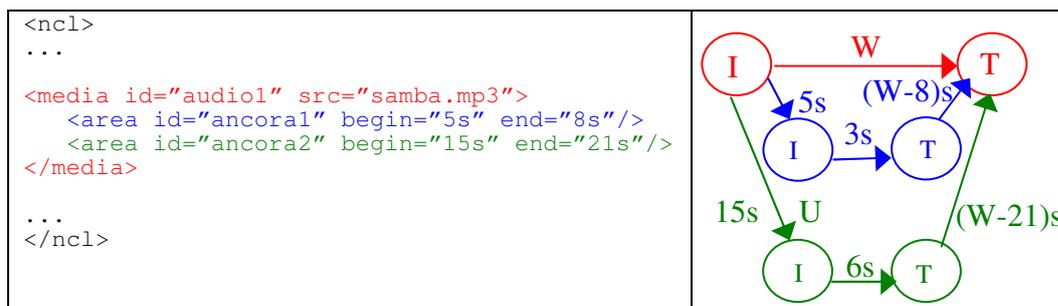


Figura 14 - Evento de apresentação no modelo de cadeias temporais hipermídia

Para facilitar o entendimento da explanação que segue, os arcos, que representam o estado de ocorrendo e o estado de pausado do evento de apresentação, serão chamados simplesmente de (arco de) apresentação e (arco de) pausa, respectivamente. Nesse e nos próximos exemplos, os nós *I* e os nós *T* representam nós de início e de término, respectivamente.

Na Figura 14, a apresentação do objeto de áudio é representada pelo arco *W*, onde o valor de *W* corresponde à duração da exibição desse objeto. Ademais, tem-

se que o objeto *audio1* possui duas âncoras temporais, denominadas *ancora1* e *ancora2*. No modelo de cadeias temporais hipermídia, as âncoras associadas aos objetos também são modeladas através de arcos de apresentação, que são iniciados e terminados pelo início e pelo fim, respectivamente, da apresentação do objeto no qual essas âncoras foram definidas. Como exemplo, as arestas (arcos) de apresentação que representam as âncoras definidas sobre o objeto *audio1* têm início no nó de início e término no nó de término do arco de apresentação relativo a esse objeto.

Para especificar a duração da apresentação das âncoras, além da definição do tempo no arco que representa a sua apresentação, é necessário definir os tempos nas arestas que unem o arco de apresentação do objeto de mídia ao arco de apresentação da âncora (como exemplo, a aresta *U*). Na Figura 14, a aresta de apresentação da âncora *ancora1* possui duração de 3 (três) segundos. A aresta que une o nó de início da apresentação do objeto de mídia ao nó de início da apresentação da âncora possui duração de 5 (cinco) segundos. Complementarmente, os arcos que unem os nós de término da apresentação das âncoras ao nó de término da apresentação do objeto de mídia possuem duração de (*W*-8) segundos e (*W*-21) segundos.

A Figura 15 apresenta um trecho de um documento NCL que estende o exemplo da Figura 14. Nesse documento são definidos dois relacionamentos causais entre o objeto *audio1* e outro objeto, denominado *video1*. Os relacionamentos são definidos através de dois elos, associados à âncora temporal *ancora2*, do objeto *audio1*. O primeiro elo, denominado *link1*, relaciona o início da apresentação da âncora *ancora2* à ação de início da apresentação do objeto *video1*. O segundo elo, denominado *link2*, relaciona o término da apresentação da âncora *ancora2* à ação de término da apresentação do objeto *video1*.

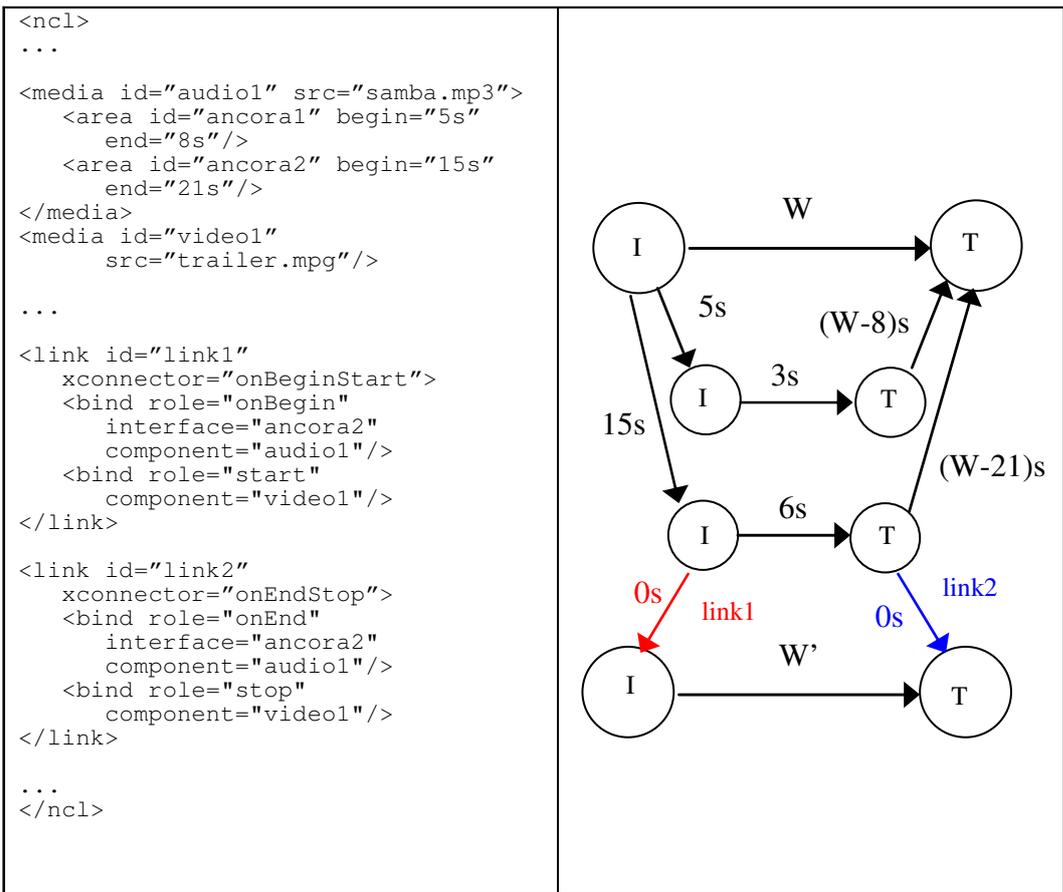


Figura 15 - Relacionamentos causais no modelo de cadeias temporais hipermídia

Na Figura 15, o objeto de mídia *video1* é representado pela aresta *W'*, onde o valor de *W'* corresponde à duração da exibição desse objeto. Nessa mesma figura, as arestas que representam os elos *link1* e *link2* são indicadas. Neste exemplo, os elos causais têm duração igual a zero, no entanto, caso o relacionamento definido no documento especificasse um tempo de duração para a ocorrência da ação ou para avaliação da condição, esse tempo seria atribuído a essas arestas.

O modelo de cadeias temporais hipermídia permite ainda a representação de relacionamentos interativos. A Figura 16 exibe o mesmo trecho de documento NCL apresentado nos exemplos anteriores, complementado de um elo de interatividade.

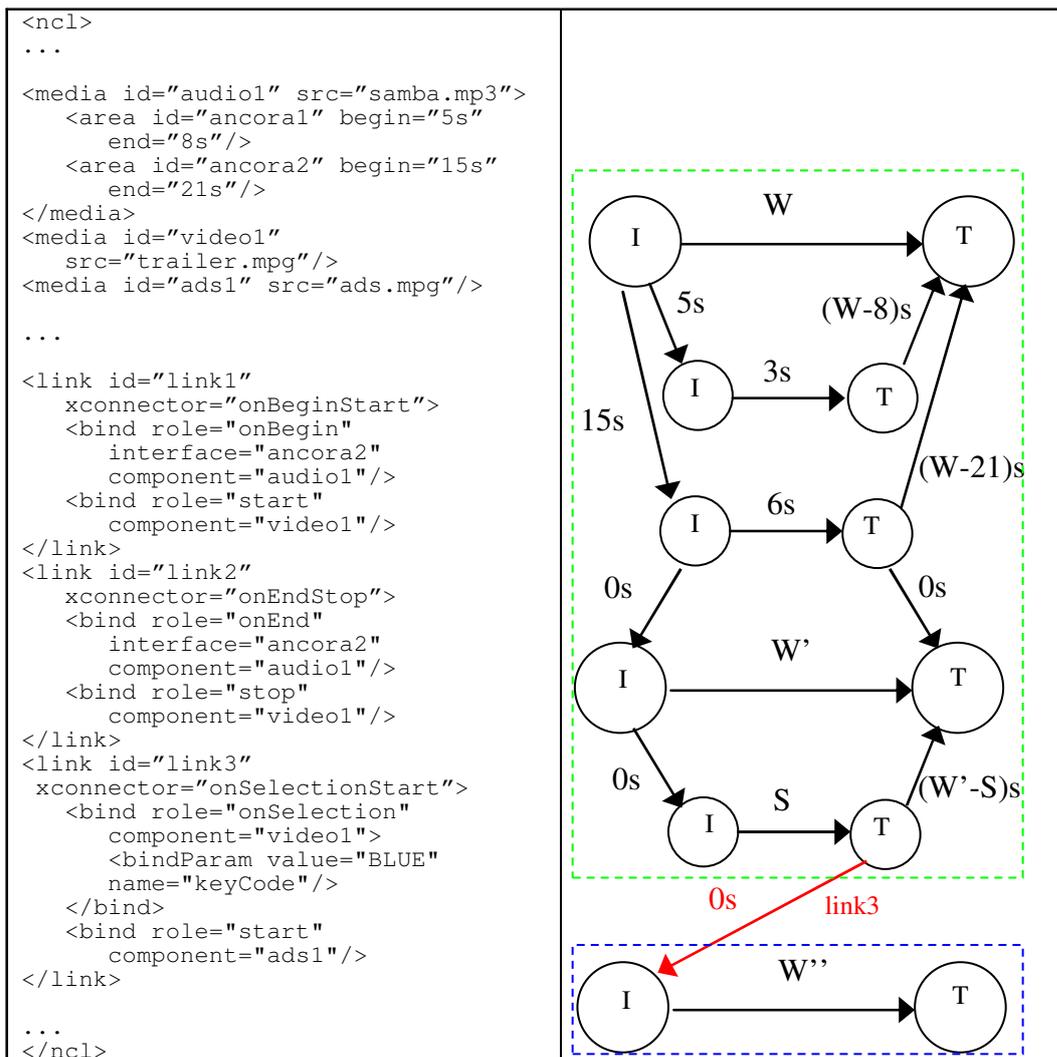


Figura 16 - Interatividade no modelo de cadeias temporais hipermedia

Neste exemplo, o elo *link3* representa um relacionamento onde o acionamento do botão azul (*BLUE*) do controle remoto⁶, durante a apresentação do vídeo *video1*, dispara a exibição do objeto de vídeo *ads1*. No modelo da Figura 16, a apresentação do objeto *ads1* é representada pela aresta *W''*, onde o valor de *W''* corresponde à duração da exibição desse objeto.

Ainda no exemplo da Figura 16, a aresta *S* é chamada de aresta de interatividade. Em virtude de representar um ponto de interatividade, o intervalo de tempo da aresta *S* é indeterminado. Se o evento relativo a essa aresta ocorrer, o tempo destinado a ela será no mínimo zero e no máximo igual ao tempo da aresta *W'*. Como o tempo da aresta *S* é desconhecido, o nó de término dessa aresta é um ponto de indeterminação. Nesse exemplo, duas cadeias temporais podem ser

⁶ A menção ao botão azul é uma mera ilustração, que será utilizada ao longo desta dissertação.

identificadas. Considerando o ponto de entrada do modelo como sendo o nó de início da aresta W , a partir desse nó até o nó de término da aresta S é definida a cadeia principal. A partir do ponto de término da aresta S até o ponto de término da aresta W' é definida uma cadeia auxiliar. Na Figura 16, a cadeia principal é indicada pelo retângulo tracejado verde, enquanto a cadeia auxiliar é indicada pelo retângulo tracejado azul.

Os tempos dos nós na cadeia auxiliar do exemplo da Figura 16 somente podem ser exatamente definidos durante a execução do documento. Caso o usuário não realize a ação de interatividade, os tempos dos nós na cadeia auxiliar não serão definidos (terão valor indeterminado), e esta cadeia não será incorporada à cadeia principal. Caso o usuário realize a ação de interatividade, o valor da aresta S será definido com o tempo transcorrido desde o início da apresentação da aresta W' . Complementarmente, o tempo da aresta que une o nó de término da aresta S ao nó de término da aresta W' deverá ser devidamente atualizado.

A Figura 17 apresenta um trecho de um documento NCL contendo um relacionamento multiponto, representado através do elo *link3*. Esse elo define um relacionamento, onde a apresentação do objeto *ads1* e a pausa do objeto *audio1* são disparadas através de uma ação de interatividade. Já o elo *link4* define um relacionamento onde o término da execução do objeto *ads1* dispara o recomeço do objeto *audio1*.

Figura 17, a apresentação desse objeto é representada por duas arestas complementares $W1$ e $W2$ ⁷.

Ainda com relação à Figura 17, o ponto de indeterminação no nó de término da aresta S não permite o cálculo do tempo exato da apresentação dos nós de início relativos às arestas W' , P e $W2$. No modelo de cadeias temporais hipermídia, as arestas relativas à pausa não são utilizadas para o cálculo do tempo dos nós. Na realidade, essas arestas são atualizadas com a diferença dos valores dos tempos dos nós que elas unem, a fim de manter a consistência do modelo. Dessa forma, se o evento de interatividade representado pela aresta S não for disparado pelo usuário, o valor da aresta P permanecerá zero e o evento de pausa não ocorrerá. Caso contrário, os valores do nó de término de $W1$ e do nó de início de $W2$ serão atualizados e o valor de P será calculado a partir da diferença do valor do tempo entre esses dois nós.

Como pode ser notado, o modelo de cadeias temporais hipermídia define uma estrutura de dados bastante robusta para representação temporal de documentos NCL. Esse modelo é adotado como base da visão temporal tanto do editor HyperProp quanto do *Composer*. Entretanto, na visão temporal do HyperProp, não é possível nem a manipulação de objetos de mídia no tempo, nem a especificação de relacionamentos temporais entre objetos. Em outras palavras, a visão temporal do editor HyperProp só serve para visualização de como os objetos de mídia estarão dispostos no tempo.

A visão temporal do *Composer* contempla a criação e manipulação de objetos de mídia no tempo, bem como a especificação de relacionamentos entre eles. A sincronização da visão temporal permite ainda que as modificações feitas nessa visão possam ser refletidas no documento e em todas as outras visões.

Além disso, a visão temporal do *Composer* dá suporte à simulação de eventos interativos. Deve ser ressaltado que tal funcionalidade permite ao autor visualizar como se dará a apresentação em tempo de criação, sem a necessidade

⁷ O estado de ocorrendo (*occurring*) do evento de apresentação (*presentation*) de um objeto de mídia, no modelo de cadeias hipermídia, é representado por um ou mais arcos de apresentação. Quando o objeto é exibido sem pausas, um único arco de apresentação corresponde ao tempo total de apresentação do objeto. No entanto, quando o objeto é pausado, são necessários dois arcos de apresentação, o primeiro para o intervalo antes da mudança do estado do objeto e o segundo após o recomeço da apresentação do objeto.

de recorrer ao formatador (ferramenta de apresentação de documentos NCL) (Rodrigues & Soares, 2003) para apresentar o documento. A simulação faz com que pontos de indeterminação do modelo possam assumir valores determinísticos mediante interação do autor.

Detalhes da implementação da visão temporal do ambiente *Composer* serão apresentados no próximo capítulo.

3.3. Edição ao Vivo

No contexto de TV digital, a autoria e a apresentação de documentos hipermídia são usualmente tratadas em fases distintas. Entretanto, muitas vezes essas fases precisam ser consideradas em conjunto. Esse é o caso, por exemplo, dos programas editados ao vivo, onde não só alguns conteúdos são desconhecidos a priori, como também alguns relacionamentos espaço/temporais entre objetos de mídia podem não ter ainda sido definidos (Moreno et al., 2005).

Muitos ambientes de autoria dispõem de um emulador, que permite ao autor visualizar a apresentação de um documento durante o processo de criação. Contudo, quando o documento é modificado, a apresentação geralmente precisa ser reiniciada para que o resultado possa ser constatado. Diferente desse cenário, a edição ao vivo demanda outros requisitos, e para atendê-los, um sistema de TV digital interativa, e conseqüentemente um ambiente de autoria, deve prover meios para se definir uma especificação inicial do documento, e meios para modificar essa especificação em tempo de apresentação, no ambiente do cliente telespectador.

Em (Costa et al., 2006) é proposta e implementada uma solução para edição textual ao vivo, que se aplica a documentos NCL. Nela é apresentado um mecanismo para edição de documentos hipermídia declarativos, que visa preservar a semântica da estrutura lógica definida pelo autor enquanto modifica a apresentação do documento.

Essa proposta (Costa et al., 2006) ainda integra algumas funcionalidades dos ambientes de autoria e de execução, abstraindo do autor a criação e a semântica dos mecanismos necessários (Apêndice A, Subseção A.2.1) para o envio dos comandos de edição, cujo controle fica completamente a cargo do ambiente de autoria e do exibidor. Com isso, o autor precisa apenas se preocupar com a

semântica do documento e não com a forma com que o sincronismo será implementado.

Para dar suporte à edição ao vivo, o *Composer*, além de monitorar os eventos de edição através da sua interface gráfica, principalmente de suas visões, utiliza a infraestrutura implementada em (Costa et al., 2006). Detalhes de como isso é feito são apresentados no próximo capítulo.