

1 Introdução

Com o advento da adoção de um padrão de TV digital interativa pelo Brasil, tem crescido o interesse pela análise das possíveis alternativas nas mais diversas áreas que compõem um sistema de TV digital.

Este capítulo descreve as motivações e os objetivos desta dissertação, a qual é focada em um ambiente de autoria de aplicações para TV digital. O capítulo também apresenta a estrutura do restante deste documento.

1.1. Motivação

Aplicações para TV digital são casos particulares de aplicações multimídia/hipermídia e devem lidar com a sincronização espacial e temporal de objetos de diferentes tipos de mídia, além dos objetos de vídeo e áudio que compõem o fluxo principal.

Em um sistema de TV digital, a capacidade de difusão de informação em várias mídias diferentes, tais como o áudio principal, o vídeo principal e outros dados (como texto, imagens e outros vídeos e áudios), faz com que esse ambiente possa oferecer diversos tipos de serviços, desde a simples distribuição de conteúdo áudio-visual da programação atual da TV analógica por difusão, até serviços antes não disponíveis, com o destaque para os serviços interativos.

Muitas vezes, o sincronismo existente nesses documentos (aplicações) é baseado em eventos onde o tempo e local de suas ocorrências não são exatamente previstos. A interatividade e a edição ao vivo são exemplos de eventos imprevisíveis, onde a interação do usuário telespectador ou do autor, respectivamente, pode dar início a um outro evento. Documentos multimídia onde é possível estabelecer o sincronismo entre a interação do usuário e outros objetos que compõem o documento são usualmente denominados documentos hipermídia. Mais recentemente, no contexto de TV digital, a mesma analogia pode ser

postergada para a edição ao vivo, neste caso devido à interação do autor em tempo de apresentação.

Os principais padrões de TV digital (Soares et al., 2004) oferecem linguagens para a especificação da interatividade e do sincronismo nas aplicações. Essas aplicações, usualmente classificadas em procedurais e de marcação, são concebidas, em sua maioria, por linguagens baseadas em Java (Sun, 1994) e HTML (*HyperText Markup Language*) (W3C, 1999a), respectivamente¹.

Aplicações desenvolvidas em Java são criadas a partir de um conjunto de classes (pacotes) oferecidas pelo *middleware* (elemento capaz de fornecer uma abstração do sistema para as aplicações e os usuários, escondendo toda a complexidade dos mecanismos definidos pelos padrões, protocolos de comunicação e até mesmo do sistema operacional instalado no terminal de acesso - *set-top box*). Para desenvolver essas aplicações, é necessário que o autor conheça as funcionalidades das classes do *middleware*. Além disso, também é necessário que o autor possua conhecimentos de programação e orientação a objeto, como encapsulamento, herança, polimorfismo, etc.

Complementarmente, as aplicações desenvolvidas em HTML (na verdade XHTML - *eXtensible HyperText Markup Language* (Pemberton et al., 2002), a princípio, favorecem a autoria ao permitir a especificação das aplicações através de marcações XML (*eXtensible Markup Language*) (W3C, 2004a). No entanto, a aplicabilidade do HTML restringe-se à formatação da apresentação e à definição de relações de referência entre documentos. Para aplicações mais elaboradas, incluindo o sincronismo e outros tipos de interatividade, são utilizadas adicionalmente linguagens baseadas em script. Essas linguagens definem estruturas de controle, variáveis e procedimentos que podem, eventualmente, acessar funções oferecidas pelo *middleware*.

Além das linguagens Java e HTML, as especificações de relacionamentos entre eventos de objetos de mídia podem ser feitas por linguagens declarativas, as quais têm como foco principal a descrição de tais relacionamentos. Nessas linguagens, as relações entre eventos podem ser especificadas sem que seja

¹ Nesse caso, as nomenclaturas das aplicações não correspondem às classes das linguagens utilizadas para a concepção. A linguagem Java é destinada à criação de aplicações orientadas a objeto, enquanto HTML é voltada para a formatação de documentos.

necessário definir estruturas complexas, como estruturas de controle, de repetição, variáveis, observadores etc., favorecendo assim o processo de autoria.

Atualmente, linguagens declarativas, que não as linguagens só de marcação, não são encontradas nos padrões de TV digital. No entanto, essas linguagens são amplamente difundidas para a modelagem de aplicações multimídia/hipermídia, com destaque para as linguagens SMIL (Bulterman & Rutledge, 2004) e NCL (Muchaluat-Saade et al., 2003).

A linguagem SMIL (*Synchronized Multimedia Integration Language*) é uma linguagem bastante intuitiva para o desenvolvimento de apresentações, pois nela é possível definir de forma simples como os objetos de mídia estarão dispostos no tempo através de um conjunto básico de composições que possuem semântica de sincronização.

A NCL (*Nested Context Language*) é uma linguagem para autoria de documentos hipermídia baseada no modelo conceitual NCM (*Nested Context Model*) (Soares et al., 2003). No modelo NCM um documento hipermídia é representado por um nó de composição, podendo este conter um conjunto de nós, que podem ser objetos de mídia ou outros nós de composição, recursivamente, e ainda elos relacionando esses nós.

Nós de composição no modelo NCM não têm nenhuma semântica embutida, diferente das composições SMIL que, por exemplo, têm semântica temporal. A semântica dos relacionamentos entre os componentes de um documento é feita através dos elos NCM, que podem especificar, por exemplo, relações de sincronização. Elos NCM são definidos fazendo referência a um conector hipermídia que pode representar qualquer tipo de relação. Além da referência a um conector, um elo define um conjunto de *binds* (associações) mapeando papéis do conector a nós do documento.

Considerando a complexidade das linguagens baseadas em Java e em HTML + script para a especificação de aplicações para TV digital, o uso de ferramentas para autoria gráfica faz-se muito importante. Como a linguagem Java e as linguagens de script possuem várias características de linguagens de propósito geral, as ferramentas de autoria para essas linguagens devem suportar os recursos oferecidos pelas linguagens e, simultaneamente, criar abstrações para o desenvolvimento de aplicações voltadas para TV digital. Nesse aspecto, nota-se uma tendência em definir estruturas auxiliares para ajudar no desenvolvimento

mental do autor na criação, abstraindo as estruturas originais das linguagens. A mesma observação pode ser considerada quando o que está em foco são as linguagens declarativas.

Existem diversas ferramentas de autoria para TV digital. Alguns exemplos são: JAME Author (Fraunhofer, 2007a), Cardinal Studio (Cardinal, 2007) e AltiComposer (Alticast, 2007a) para criação de aplicações procedurais para o *middleware* MHP (*Multimedia Home Platform* - ES 201 812 (ETSI, 2003)); Dreamweaver (Adobe, 2007a) e Microsoft FrontPage (Microsoft, 2007a) para criação de documentos HTML. Já para as linguagens declarativas, pode-se destacar o ambiente GRiNS (Bulterman et al., 1998) para SMIL, e o editor HyperProp (Coelho & Soares, 2004) para criação de aplicações hipermídia em NCL *main profile* (Soares et al., 2006), um perfil mais geral dessa linguagem.

No caso do Brasil, NCL é a linguagem declarativa adotada para modelagem de aplicações interativas no Sistema Brasileiro de TV Digital Terrestre (ISDTV-T – *International System for Digital TV*). Nesse contexto, surge a demanda por uma ferramenta de autoria de documentos NCL nos seus perfis (*profiles*) para TV digital (Soares & Rodrigues, 2006).

1.2. Objetivos

Esta dissertação apresenta a ferramenta *Composer*, um ambiente de autoria voltado para a criação de documentos NCL para TV digital interativa. Da mesma forma que no editor HyperProp, no qual é baseado, no *Composer* as abstrações são definidas nos diversos tipos de visões que permitem simular um tipo específico de edição (estrutural, temporal, leiaute e textual). Essas visões funcionam de maneira sincronizada, a fim de oferecer um ambiente integrado de autoria.

O objetivo deste trabalho é remodelar o editor HyperProp, tendo a edição temporal como principal foco. A intenção é solucionar problemas de representação e edição de objetos de mídia, relacionamentos de sincronismo entre objetos, dentre eles os relacionamentos interativos, e edição ao vivo. Este trabalho contempla também a sincronização da visão temporal com as demais visões do

ambiente de autoria de modo a fazer que alterações realizadas em uma determinada visão também sejam refletidas na visão temporal, e vice-versa.

Em resumo, o sistema proposto visa facilitar e agilizar a criação de aplicações voltadas para TV digital abstraindo do autor toda, ou pelo menos parte da complexidade de se programar em NCL, através desse ambiente de autoria.

1.3. Estrutura da Dissertação

Esta dissertação encontra-se organizada como a seguir. O Capítulo 2 apresenta algumas ferramentas de autoria pertinentes ao escopo deste trabalho. Algumas dessas ferramentas são voltadas à produção de conteúdo multimídia/hipermídia para TV digital interativa, outras não. Nesse capítulo são analisadas as principais características de cada uma delas.

O Capítulo 3 destaca melhorias do ambiente de autoria *Composer* com relação ao editor HyperProp para adequação ao contexto de TV digital interativa. São apresentados detalhes da estrutura de dados utilizada para representar as cadeias temporais da visão temporal. Esse capítulo contempla ainda características da edição ao vivo através do *Composer*.

O Capítulo 4 foca na implementação dos pontos levantados no Capítulo 3, a fim de tornar o *Composer* mais atraente para criação de programas NCL para TV digital interativa.

Já o Capítulo 5 tece as conclusões, fazendo ainda uma análise comparativa do *Composer* com as ferramentas analisadas no Capítulo 2 e o editor HyperProp, e descrevendo trabalhos futuros. O Capítulo 6 sumariza as referências bibliográficas utilizadas na elaboração desta dissertação.

Não menos importante, o Apêndice A faz uma discussão sobre os mecanismos de sincronização e interatividade nos padrões de TV digital, com destaque para as linguagens utilizadas na especificação desses eventos. Uma parte específica trata de linguagens declarativas e da possibilidade de uso dessas linguagens pelos padrões.