

1 Introdução

O mundo em que vivemos é tridimensional. Logo, é natural que o ser humano busque reproduzi-lo em ambientes virtuais. Isso explica a crescente demanda por aplicações tridimensionais (aplicações 3D) nas diversas formas de mídias existentes, entre as quais estão a Web e a TV Digital Interativa (TVDi). Somado a isso, o atual desenvolvimento de *hardware* para processamento gráfico e o surgimento de métodos e tecnologias mais efetivas de interação humano-computador tornam cada vez mais iminente a popularização dessas aplicações 3D. Com o desenvolvimento e o barateamento das placas de aceleração gráfica, diversas classes de aplicações, entre as quais estão as aplicações hipermídia, podem ser beneficiadas, permitindo-as reproduzir novos tipos de mídia, mais realistas.

Hipermídia é uma aplicação multimídia (composta por vários tipos de mídia tais como, áudio, vídeo, imagem etc.) que usa relacionamentos associativos entre informações contidas nessas múltiplas mídias para o propósito de facilitar o acesso e a manipulação dessa informação (LOWE, 1999). Utilizar-se de sistemas hipermídia para descrever aplicações 3D permite que mecanismos poderosos desses sistemas (tais como estruturação e navegação em uma grande quantidade de informações) também sejam utilizados para descrição e busca de informação de aplicações 3D (ROMERO e CORREIA, 2003).

Os modelos hipermídia geralmente baseiam-se nos conceitos de nós e relacionamentos entre nós. *Nós* representam informações. Relacionamentos entre essas informações são usualmente expressos através de *elos* ou *nós* de composição. Como exemplo de modelo conceitual hipermídia que se baseia nos conceitos de nós e elos temos o NCM (*Nested Context Model*) (SOARES e RODRIGUES, 2005). No NCM, nós de conteúdo (também denominados de nós de mídia ou objetos de mídia em NCM) representam os objetos com conteúdos de mídia usuais (texto, áudio, vídeo etc.). Nós de composição são usados em NCM para agrupar diversos nós de mídia ou de composição e elos entre esses nós,

permitindo organização e encapsulamento. Em especial, os modelos hipermídia servem de base para o desenvolvimento de linguagens de programação denominadas linguagens de autoria hipermídia (ou simplesmente linguagens hipermídia).

As linguagens de programação podem ser classificadas de diversas formas diferentes (SAMMET e HEMMENDINGER, 2003) (WATT, 1990) (MELO e SILVA, 2003). Uma dessas classificações, em particular, divide as linguagens em dois grandes grupos: *declarativas* e *imperativas*. Em uma abordagem declarativa, o autor deve especificar o que intenta como sendo o resultado final, diferente de uma abordagem imperativa, na qual o autor deve especificar os passos necessários para se chegar ao resultado, de forma algorítmica. A especificação declarativa de uma aplicação deve ser interpretada e executada por meio de exibidores (ou *players*). O exibidor da linguagem declarativa também é responsável por executar essa especificação da melhor forma possível, visando chegar ao resultado final de forma eficiente. São exemplos de linguagens declarativas: XHTML (W3C, 2002), SMIL (W3C, 2008), NCL (SOARES e RODRIGUES, 2005). C, C++, Java e Pascal são alguns exemplos de linguagens imperativas.

De uma forma geral, as linguagens hipermídia permitem estruturar e relacionar informações de forma *declarativa*. As linguagens declarativas tipicamente definem um modelo específico para desenvolver aplicações em um domínio também específico. Essas especificações declarativas costumam estar mais perto de uma especificação em alto nível, quando comparado com as especificações em linguagens de programação imperativas (SOARES, RODRIGUES, *et al.*, 2009).

NCL (*Nested Context Language*) é um exemplo de linguagem hipermídia declarativa. Ela é uma aplicação XML (W3C, 2008) que mapeia os principais conceitos do NCM. NCL é a linguagem declarativa padrão para o sistema de TV digital terrestre ISDB-T_B (ABNT, 2007) e da *International Telecommunication Union* para serviços de IPTV (ITU, 2009). Assim como o NCM, a linguagem NCL não está focada em descrever os diversos conteúdos dos nós de mídia que fazem parte de um documento hipermídia, mas sim como eles são estruturados e estão relacionados. Tal característica permite se mencionar NCL como uma linguagem de cola.

Como NCL não restringe o conteúdo individual dos objetos de mídia, torna-se natural cogitar também o uso de objetos de mídia 3D (cujo conteúdo deve ser representado em formatos específicos para esse fim) em aplicações NCL. Entretanto, na versão atual, NCL 3.0, não há outra forma de apresentar objetos de mídia que não seja em regiões bidimensionais. Um objeto de mídia, em NCL 3.0, é sempre apresentado em uma região definida pelos parâmetros: *top* (ou *bottom*), *left* (ou *right*), *width* e *height*.

Quando o foco do desenvolvimento é aplicações 3D, hoje já existem linguagens que dão suporte ao desenvolvimento de forma declarativa, os quais se baseiam principalmente nos modelos de grafo de cena e grafo de rotas. Os grafos de cena são estrutura de dados formadas por nós e arcos entre os nós, compondo uma árvore ou um DAG (*Directed Acyclic Graph*). Os arcos entre os nós representam relações hierárquicas, ou seja, o grafo de cenas representa a estrutura espacial da cena. O uso de grafo de cena, em especial, permite que diversas otimizações sejam realizadas pelo exibidor no momento da renderização, tais como descartes por nível de detalhes, volume de visão etc. – a Subseção 2.1.1 discute algumas dessas otimizações. O grafo de rotas, por outro lado, relaciona os nós, definidos no grafo de cena, informando qual o comportamento da cena. Isso é feito informando qual evento em um nó deve ser disparado quando outro evento, possivelmente em outros nós, acontece.

1.1. Objetivos e Contribuições

Dada a importância de NCL no cenário de TV Digital, este trabalho foca, em um primeiro momento, em como embutir objetos de mídia 3D (um polígono, malha de triângulos, ou mesmo objetos declarativos compostos, como cenas) em aplicações NCL, apresentando-os em uma região (2D) de NCL. O trabalho discute como é possível adicionar objetos de mídia 3D à NCL e relacioná-los espacial e temporalmente a outros objetos de mídia (2D ou 3D). Para isso, é realizada uma análise de como é possível:

- definir âncoras de conteúdo (porções do conteúdo do objeto de mídia) em objetos 3D compostos, em especial, os grafos de cena;
- tratar os eventos que possivelmente sejam gerados por esses objetos 3D, estendendo os tipos de eventos de NCL; e

- como utilizar a linguagem NCL para especificar o controle do comportamento desses objetos 3D.

No que se refere a objetos 3D compostos representados por grafos de cena, é proposto o uso da linguagem NCL para controlar o seu comportamento, em detrimento ao uso do grafo de rotas. O uso de NCL para controlar o comportamento de grafos de cena permite, dentre outras coisas: a especificação de relações de restrição entre os nós; o uso de múltiplos dispositivos de exibição; e a adaptação do conteúdo a ser exibido, utilizando informações contextuais do usuário. Adicionalmente, diferente das linguagens baseadas em grafo de cena e grafo de rotas, NCL trata a interatividade apenas como um caso particular do sincronismo intermídia, o que a torna bem mais expressiva.

Em um segundo momento, ainda no escopo deste trabalho, são propostas extensões à linguagem NCL que permitem a exibição de objetos de mídia bidimensionais (vídeo, imagem, ou mesmo objetos declarativos e imperativos) em regiões que são superfícies de formas geométricas tridimensionais. Isso é similar à aplicação de texturas dinâmicas (tais como vídeos) em objetos geométricos, comuns em ambientes tridimensionais. Esta proposta consiste em modificar a área funcional *Layout* de NCL para permitir que as regiões sejam definidas com forma de cubos, esferas etc. Como um passo intermediário para se chegar a esse objetivo, são analisadas diversas transformações geométricas 2D e 3D que podem aperfeiçoar as regiões NCL. Nesse momento, é apresentado como é possível definir essas transformações na linguagem, propondo duas notações diferentes e analisando suas vantagens e desvantagens.

1.2. Organização da Dissertação

Esta dissertação está dividida como segue:

O Capítulo 2 discute os principais conceitos que servirão de base para os capítulos seguintes. Em especial, serão abordados os conceitos de grafo de cena, grafo de rotas e os conceitos pertinentes ao modelo NCM, em que se baseia a NCL.

O Capítulo 3 apresenta os principais trabalhos relacionados, com destaque para: X3D (WEB3D CONSORTIUM, 2009a), linguagem padrão W3C para a especificação de aplicações virtuais em 3D, baseada em grafo de cena e grafo de

rotas; e XMT (KIM, WOOD e CHEOK, Novembro de 2000), linguagem padrão MPEG-4 (PEREIRA e EBRAHIMI, 2002), baseada em X3D e que embute conceitos de multimídia herdados da linguagem SMIL (W3C, 2005). Ainda no Capítulo 3, é apresentada a arquitetura do *middleware* Ginga (ABNT, 2007), que será utilizado como base para a implementação das extensões à NCL propostas por este trabalho.

O Capítulo 4 discute como NCL pode embutir objetos de mídia 3D atômicos (como polígonos) e compostos (como grafos de cena). Em relação a objetos compostos, também é abordado como definir âncora de conteúdo (porções do conteúdo) e de propriedade nesses objetos, em documentos NCL. Além da definição de âncoras, esse capítulo também propõe novos eventos à NCL, específicos de ambientes 3D. Ao definir âncoras em objetos 3D e eventos específicos desses objetos será possível sincronizá-los com os objetos 2D de NCL já existentes. Como resultado natural do sincronismo entre eventos de NCL, neste capítulo também é demonstrado como NCL pode controlar o comportamento de objetos 3D compostos, em detrimento, por exemplo, ao grafo de rotas. Por fim, é apresentado um quadro comparativo entre o modelo NCM, a linguagem XMT e a programação por meio de grafo de rotas.

No Capítulo 5 é apresentado como objetos de mídia 2D podem ser apresentados em superfícies de objetos 3D, estendendo o módulo *Layout* da linguagem NCL. Como passo intermediário visando esse objetivo, são propostas diversas transformações, tanto 2D como 3D, que podem aperfeiçoar como os objetos de mídia NCL são exibidos. Esse capítulo também apresenta como tais transformações devem ser mapeadas em descritores e propriedades de objetos de mídia em NCL.

A implementação das propostas desta dissertação toma como base principal a Implementação de Referência do Ginga-NCL. O Capítulo 6 discute como o presente trabalho modificou a Implementação de Referência para que as propostas desta dissertação fossem possíveis de serem implementadas. Discute também, os principais pontos relacionados à implementação destas propostas.

Por fim, o Capítulo 7 apresenta as conclusões e possíveis trabalhos futuros.