

## 6 Conclusões e Trabalhos Futuros

A mudança no perfil dos usuários, agora produtores de conteúdo, e a evolução das tecnologias de captura e edição multimídia contribuíram para o surgimento de soluções voltadas para o enriquecimento de páginas Web. Por outro lado, no âmbito da TV Digital, a restrição a autoria por parte de profissionais especializados e o difícil acesso as tecnologias envolvidas limitam o emprego da tecnologia as grandes empresas de comunicação.

Como visto no na Seção 3.1, existem outras abordagens para o provimento de um *plugin* reprodutor hipermídia para Web, como as soluções Ambulant WebKit Plugin e o Flash Plugin. Assim, considerando os aspectos apresentados na referida seção é possível concluir que, se o objetivo é inclusão de aplicações hipermídia adaptativas que especifiquem alguma forma de sincronismo temporal e espacial entre objetos dos dois ambientes, as soluções baseadas em NCL e SMIL são as mais adequadas. Ademais, se a adaptação a envolver a edição em tempo de exibição e o suporte a exibição em múltiplos dispositivos como parte do processo, a solução baseada em NCL é a mais indicada. Por outro lado, se o objetivo é embutir aplicações que, sobre tudo, ofereçam um extenso suporte a animações e a efeitos visuais de uma forma geral, a solução baseada em Flash é a mais indicada.

Desse modo, considerando o foco dado neste trabalho, utilizar-se de uma linguagem hipermídia, a NCL, voltada para especificação de aplicações interativas para a TV Digital como forma de beneficiar uma página Web é visto como uma oportunidade de oferecer aos usuários da Web mais uma ferramenta para a especificação de conteúdo multimídia e, em contrapartida, popularizar a linguagem em questão. Com essa finalidade, a proposta apresentada neste documento defende o uso da NCL (*Nested Context Language*) como linguagem para a especificação de conteúdo multimídia em páginas Web. Para isso, é discutida a adaptação do *middleware* Ginga-NCL como *plugin* para o navegador Mozilla Firefox e também a integração entre as máquinas de apresentação NCL e HTML.

## 6.1. Contribuições da Dissertação

Esta dissertação apresenta como é possível embutir aplicações NCL em páginas Web, habilitando-as a controlar a apresentação, modificar o conteúdo embutido em tempo de exibição e promover a sincronização entre os objetos de mídia dos ambientes envolvidos. Com isso, o trabalho contribui diretamente para a popularização do uso da NCL como linguagem de autoria de aplicações hipermídia visando o enriquecimento interativo e visual de páginas Web. Além disso, como efeito colateral é esperado a migração dos novos programadores NCL também para o ambiente da TV Digital no país e a evolução de soluções para Televisão social e Internet TV.

Uma das maiores contribuições deste trabalho decorre do suporte ao sincronismo entre os objetos de mídia dos dois ambientes, atingido pela especificação de elos a partir de *âncoras de propriedades* e *conteúdo*, definidas pelo código JavaScript. Esse sincronismo permite que uma página Web defina ou compartilhe propriedades com o documento NCL embutido e promova o sincronismo temporal e espacial entre os objetos de mídia, além de permitir a especificação de momentos de interação que envolva ambos os contextos.

Outra contribuição de grande importância envolve a edição de aplicações NCL embutidas em tempo de exibição, obtida pela submissão de comandos de edição através de código JavaScript. Aliado ao recurso de sincronização, esse recurso confere a página Web um alto poder adaptação, permitindo que ela responda as mudanças internas ao documento NCL através de operações de edição.

Não menos importante, no que concerne ao trabalho de implementação, a adaptação do *middleware* a nova plataforma (Windows) e a consequente substituição do *backend* gráfico são vistas como contribuições secundárias, porém permitem vislumbrar pesquisas direcionadas a nova plataforma e ao desenvolvimento de extensões para linguagem NCL que contemplem as três dimensões.

## 6.2. Trabalhos Futuros

Em um trabalho futuro poderá ser considerado a migração do Ambiente de Apresentação (Ginga-NCL) para a linguagem JavaScript e em contrapartida a migração de alguns componentes do Ginga-CC, no que diz respeito ao *backend*

gráfico. Dessa forma, pode-se vislumbrar, por exemplo, a adaptação do módulo *Gerente Gráfico* ao elemento de desenho <canvas> (W3C, 2010), o que permitiria a abstração de regiões NCL como superfícies do *canvas* HTML. Por outro lado, os exibidores poderiam utilizar os reprodutores de mídia disponíveis ou embutidos como *plugins* para realizar a renderização nessas superfícies. Em particular, caso fosse realizado, esse trabalho conferiria ao *Plugin Ginga-NCL* a independência de plataforma de boa parte da arquitetura.

Neste trabalho, o suporte ao *streaming* se resume ao fornecimento de exibidores capazes de obter um fluxo RTSP (IETF, 1998), o que permite a inserção de objetos de mídia localizados em repositórios *streaming*. A fim de ampliar esse suporte, em um trabalho futuro pode-se propor o encapsulamento e multiplexação das aplicações NCL e dos objetos de mídia junto ao fluxo RTSP. Mais ainda, pode-se pensar em estratégias de pré-busca para os objetos de mídia baseados na gerência de sincronização das aplicações através de mecanismos como o HTG (*Hypermedia Temporal Graph*) (COSTA, 2008).

Neste trabalho, o suporte ao recebimento de fluxo de transporte pelo sintonizador se resume a demultiplexação do conteúdo áudio visual, como destaca a Subseção 5.1.3. Assim, outro importante trabalho futuro é estender o suporte ao modelo de Carrossel de objetos a nova plataforma. O que, em segunda instância, permitirá a criação de um ambiente híbrido, onde a página Web poderá referenciar tanto aplicações NCL contidas no fluxo de transporte com aplicações embutida nela mesma.