

6 Conclusões e Trabalhos Futuros

Obter baixo custo nos terminais de acesso é fator crucial para o sucesso da TV digital aberta, principalmente nos países em desenvolvimento. Para fazer com que o baixo custo comprometa o mínimo possível dos recursos dos terminais de acesso, é importante que seja feita uma economia nos custos relativos à propriedade intelectual e a royalties associados ao software presente nesses dispositivos como, por exemplo, o middleware. Os principais middlewares existentes, além de apresentarem esses custos associados, possuem alto requisito de processamento, por estarem atrelados a tecnologias como máquinas virtuais Java. Para resolver essas e outras questões relacionadas ao contexto de TV digital, a implementação de um middleware declarativo, independente do uso de tecnologia Java, fez-se necessária.

Os principais middlewares declarativos existentes privilegiam a interatividade em detrimento da sincronização. Foi discutido que o sincronismo de mídias deve ser o foco da linguagem declarativa a ser utilizada pelo middleware, tratando a interatividade como um caso particular do sincronismo.

Este trabalho apresenta uma análise detalhada das principais questões envolvidas no planejamento, desenvolvimento e implementação de middlewares declarativos para TV digital interativa, podendo servir como uma base para projetos com foco nessa área.

Outra contribuição deste trabalho foi a implementação do middleware *Maestro*, um middleware declarativo que possui foco no sincronismo de mídias, através da linguagem NCL, e faz parte do modelo de referência proposto ao Sistema Brasileiro de TV Digital (SBTVD). Além de demandar baixo poder de processamento das máquinas, todas as tecnologias utilizadas nessa implementação estão livres de encargos adicionais, fazendo com que a implementação do middleware *Maestro* seja uma alternativa interessante, também, para perfis simples de terminais de acesso.

Em terminais com maior poder de processamento, o middleware *Maestro* pode, inclusive, trabalhar em conjunto com middlewares procedurais, existindo também a possibilidade do mesmo ser adicionado em uma plataforma que não suporte aplicações NCL de modo nativo (sem uso do *Maestro*).

Como prova de conceito, o middleware foi integrado no perfil simples de terminal de acesso definido pelo SBTVD, denominado Kalaheo, com a seguinte configuração: processador Celeron 700 MHz e 128 MB de memória RAM. O middleware foi também integrado a um terminal de acesso, denominado Geode, de configuração inferior: processador AMD 233 MHz e 64 MB de RAM. A potencialidade de todos os seus recursos foram verificadas através de sua implementação em um computador pessoal, simulando um terminal de acesso de grande poder computacional (processador Pentium IV de 3.1 GHz e 1 GB de memória RAM). Diversas aplicações NCL foram testadas no middleware *Maestro* em todas as versões implementadas.

Como decorrência da implementação do *Maestro*, outra contribuição deste trabalho foi a implementação de um Formatador NCL, e seus exibidores, na linguagem C++, uma vez que o mesmo é um subconjunto da arquitetura modular do middleware.

A pesquisa realizada nesta dissertação também contribuiu na definição de um mecanismo para possibilitar que modificações no documento, especificadas pelo autor em tempo de exibição, sejam refletidas na apresentação e possam posteriormente ser armazenadas, preservando todos os relacionamentos, incluindo aqueles que definem a estruturação lógica de um documento (Moreno et al, 2005c).

Como trabalho futuro, pretende-se implementar, sobre o middleware *Maestro*, outros exibidores, como, por exemplo, exibidores com capacidade de apresentar arquivos Flash (programa gráfico vetorial utilizado para criar animações interativas, desenvolvido e comercializado pela Macromedia¹⁶), através da integração da biblioteca libflash à biblioteca DirectFB. O objetivo, nesse caso, é explorar os recursos das animações Flash em conjunto com os recursos da linguagem NCL.

Outro trabalho futuro é a implementação de um encapsulador de pacotes IP sobre Fluxo de Transporte MPEG-2 (Moreno, 2005a), bem como a implementação, sobre o middleware *Maestro*, de um módulo desencapsulador de pacotes IP sobre MPEG-2 (Moreno, 2005a). A idéia central é possibilitar o uso do sistema de TV digital como instrumento complementar na inclusão digital. Como consequência, a partir de canais de retorno de baixo custo como, por exemplo, mensagens SMS (*Short Message Service*) por meio de um dispositivo móvel simples, usuários telespectadores poderão solicitar URLs e receber os respectivos conteúdos através da rede de difusão de TV digital (Moreno, 2005a).

Um quarto trabalho importante é a implementação dos módulos Gerenciador de Bases Privadas e Gerenciador DSM-CC, discutidos no Capítulo 4. A implementação de um módulo responsável por tratar fluxos de mídia contínua sobre o protocolo RTP (*Real-Time Transport Protocol*) consiste em outro trabalho futuro. Esse protocolo seria o responsável por realizar o transporte com sincronismo entre diferentes mídias, como áudio, vídeo e dados associados, através do canal de retorno. Além disso, é interessante definir como trabalho futuro o aperfeiçoamento dos testes realizados com o objetivo de medir questões de consumo de recursos e avaliações de desempenho do middleware *Maestro*.

Por fim, espera-se que, com a manipulação dos componentes desenvolvidos no middleware *Maestro*, sejam desenvolvidas novas implementações objetivando o seu funcionamento em conjunto com implementações dos principais middlewares procedurais existentes, bem como uma integração com o middleware procedural FlexTV.

¹⁶ <http://www.macromedia.com>