

Mathias H. Weber

**AMBIENTE DE PRODUÇÃO PARA TELEVISÃO
DIGITAL INTERATIVA**

Dissertação submetida à
Universidade Federal
de Santa Catarina como
parte dos requisitos para
a obtenção do grau de
Mestre em Ciência da
Computação.

Prof. Dr. Rogério Cid
Bastos
(Orientador)

Florianópolis, 2010

Catálogo na fonte pela Biblioteca Universitária
da
Universidade Federal de Santa Catarina

W373a Weber, Mathias Henrique

Ambiente de produção para televisão digital interativa
[dissertação] / Mathias Henrique Weber ; orientador,
Rogério Cid Bastos. - Florianópolis, SC, 2010.
84 p.: il., tabs.

Dissertação (mestrado) - Universidade Federal de Santa
Catarina, Centro Tecnológico. Programa de Pós-Graduação em
Ciência da Computação.

Inclui referências

1. Informática. 2. Ciência da computação. 3. Televisão
digital. 4. Ferramenta de autoria. I. Bastos, Rogério Cid.
II. Universidade Federal de Santa Catarina. Programa de
Pós-Graduação em Ciência da Computação. III. Título.

CDU 681

Mathias H. Weber

**AMBIENTE DE PRODUÇÃO PARA TELEVISÃO
DIGITAL INTERATIVA**

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Ciência da Computação, área de concentração Televisão Digital e aprovada em sua forma final pelo Programa de Pós-Graduação em Ciência da Computação.

Florianópolis, 26 de Fevereiro de 2010

Prof. Dr. Coord Mário Antônio Ribeiro Dantas

Banca Examinadora

Prof. Dr. Rogério Cid Bastos
(Orientador)

Prof. Dr. Antônio Augusto Medeiros Fröhlich
(PPGCC/UFSC)

Prof. Dr. Fernando Crocomo
(Jornalismo/UFSC)

Prof. Dr. José Leomar Tedesco
(EGC/UFSC)

Embora ninguém possa voltar atrás e fazer um novo começo, qualquer um pode começar agora e fazer um novo fim. (Chico Xavier)

AGRADECIMENTOS

Agradeço ao meu orientador Rogério Cid Bastos, pela ajuda e por me suportar ao longo destes anos.

Ao professor José Leomar Tedesco, pelas correções, dicas e exigência que possibilitaram a elaboração de um bom documento, bem como ao Airton Zancanaro.

À minha esposa Vanusa Uller, pelo apoio ao longo destes anos, e à minha filha Sofia Emanuelle Weber, que veio no meio do caminho da dissertação para me dar mais ânimo nesta jornada.

Ao meu amigo Eros Comunello, pelas dúvidas sanadas, pela confiança em mim depositada e pelo apoio profissional e pessoal.

À minha grande e eterna amiga Anita Maria da Rocha Fernandes, que está sempre a me ajudar.

Ao Fernando Crocomo, por todo o empenho e colaboração na elaboração deste e de outros projetos, pela parceria em viagens para divulgar o projeto e pela confiança do dia-a-dia.

À secretária do curso de pós-graduação “Verinha”, pelas conversas e problemas resolvidos.

Aos jornalistas Grazielle Pasqual Schneider e João Munhoz, pela presteza e colaboração nos testes com seu trabalho de conclusão de curso, bem como à designer Dafne Arbex, pelos testes realizados com seu trabalho acadêmico.

Ao grupo de pesquisa Cyclops e ao INCoD, pelo apoio financeiro e estrutural concedido.

A todos os meus amigos e amigas, que sempre estiveram presentes, me aconselhando e incentivando com carinho e dedicação.

A todas as pessoas que, direta ou indiretamente, contribuíram para a execução desta dissertação de mestrado.

RESUMO

Esta dissertação apresenta um ferramental desenvolvido para a inserção de interatividade em televisão digital para conteúdos jornalísticos. Para isto é apresentado um levantamento bibliográfico realizado sobre os assuntos decorrentes, bem como o conjunto de ferramentas implementadas. O conjunto proposto e desenvolvido consiste em uma ferramenta de autoria que, aliada a *templates*, permite a criação de projetos baseados em linha de tempo para televisão digital interativa. É também apresentado um servidor capaz de armazenar os projetos gerados com a ferramenta de autoria e receber, através do canal de retorno, informações geradas por aplicativos interativos. Para colher resultados da implementação, foram realizados três projetos com grupos distintos que utilizaram as ferramentas, relatando-se a experiência de cada grupo.

Palavras-Chave: TV Digital, Ferramenta de Autoria, SBTVD-T

ABSTRACT

This dissertation presents a set of tools developed for the inclusion of interactivity in digital television for journalistic content. For this purpose, we present a literature review on the issues that arise, as well as the toolkit implemented. The toolkit proposed and developed consists of an authoring tool which, combined with templates, allows the creation of time-line based designs for interactive digital television. In addition, we present a server capable of storing the projects generated with the authoring tool and receiving, through the return channel, the information generated by interactive applications. In order to gather results from the implementation, three projects were conducted with different groups of people who used the tools, and the experience of each group was described.

Keywords: Digital TV, Authoring Tool, SBTVD-T

LISTA DE FIGURAS

Figura 1	Componentes envolvidos na TV Digital	5
Figura 2	Divisão de banda na TV Digital	6
Figura 3	Mapa de padrões de TV Digital no Mundo . . .	12
Figura 4	Ginga-J: Arquitetura e ambiente de execução .	17
Figura 5	Exemplo de código Ginga-NCL	18
Figura 6	Exemplo da linguagem NCL	20
Figura 7	Exemplo da linguagem LUA	21
Figura 8	Exemplo de código LUA	22
Figura 9	CeltX: Ferramenta para criação de <i>script</i>	23
Figura 10	Visão estrutural da ferramenta Composer . . .	25
Figura 11	XiTV: Exemplos de aplicação com interatividade em XHTML	25
Figura 12	JAME Author: Ferramenta de Autoria	26
Figura 13	XleTVView: Emulador para JavaTV	27
Figura 14	Demonstração na internet de iTV do canal BBC	28
Figura 15	Exemplo de Comércio Eletrônico pela TV . . .	28
Figura 16	Exemplos de aplicação interativa textual desen- volvida	29
Figura 17	Exemplo de pergunta interativa	31
Figura 18	Arquitetura do ambiente proposto	34
Figura 19	Arquitetura da ferramenta de autoria	35
Figura 20	Diagrama de visão geral	39
Figura 21	Tela principal da ferramenta de autoria	40
Figura 22	Biblioteca de recursos	41
Figura 23	Menu multimídia	41
Figura 24	Classificação de templates	42
Figura 25	<i>Timeline</i>	42
Figura 26	Editor de propriedades	43
Figura 27	Exemplo de documento <i>Extensible Markup Lan- guage</i> (XML) descrevendo um <i>template</i> de inte- ratividade	45
Figura 28	Interface de cadastro de programas	47
Figura 29	Lista de execução dos programas no ambiente simulado	48
Figura 30	Gráficos de retorno das aplicações interativas (Ní- vel 2 e 3)	50
Figura 31	Informação textual	52

Figura 32	Vídeo alternativo	53
Figura 33	Quiz	54
Figura 34	Projeto “Mãos”	56
Figura 35	Saiba mais: parte do projeto de dissertação do curso de Design Gráfico	58

LISTA DE QUADROS

Quadro 1	Diferenças das extensões de <i>eXtensible Hyper-Text Markup Language</i> (XHTML) para interatividade em TV Digital	9
Quadro 2	Comparativo entre middlewares	33

LISTA DE SIGLAS

- ACAP** *Advanced Common Application Platform*
- API** *Application Programming Interface*
- ARIB** *Association of Radio Industries and Businesses*
- ATSC** *Advanced Television Systems Committee*
- ATSC-M/H** *ATSC - Mobile/Handheld*
- BML** *Broadcast Markup Language*
- CRT** *Cathode Ray Tube*
- CSS** *Cascading Style Sheets*
- DASE** *Digital television Application Software Environment*
- DMB** *Digital Multimedia Broadcasting*
- DOM** *Document Object Model*
- DTMB** *Digital Terrestrial Multimedia Broadcast*
- DVB** *Digital Video Broadcasting*
- DVB-C** *DVB - Cable*
- DVB-H** *DVB - Handheld*
- DVB-HTML** *Digital Video Broadcasting - HyperText Markup Language*
- DVB-J** *DVB em Java*
- DVB-IPTV** *DVB - IPTelevision*
- DVB-MC** *DVB - Microwave Cable*
- DVB-MS** *DVB - Microwave Satellite*
- DVB-MT** *DVB - Microwave Terrestrial*
- DMB-S** *DMB - Satellite*
- DVB-S** *DVB - Satellite*
- DVB-S2** *DVB - Satellite 2*
- DVB-SH** *DVB - Satellite Handheld*
- DMB-T/H** *DMB - Terrestrial/Handheld*
- DVB-T** *DVB - Terrestrial*
- DVB-T2** *DVB - Terrestrial 2*
- EAD** *Educação a Distância*
- ECMA** *European Computer Manufacturers Association*
- ECMAScript** *European Computer Manufacturers Association Script*
- GEM** *Globally Executable MHP*
- Ginga-NCL** *Ginga - Nested Context Language*
- Ginga-J** *Ginga Java*
- GSM** *Groupe Spécial Mobile*
- HAVI** *Home Audio Video Interoperability*

HDTV *High-Definition Television*
HE-AAC *High-Efficiency Advanced Audio Coding*
IDE *Integrated Development Environment*
INCoD Instituto Nacional de C&T em Convergência Digital
IP *Internet Protocol*
IPTV *Internet Protocol Television*
ISDB *Integrated Services Digital Broadcasting*
ISDB-T *Integrated Services Digital Broadcasting Terrestrial*
ISDB-C ISDB - *Cable*
ISDB-S ISDB - *Satellite*
JavaTV JAVA *Television API*
JMF *Java Media Framework*
LCD *Liquid Crystal Display*
MHP *Multimedia Home Platform*
NCL *Nested Context Language*
NCL-Lua NCL com LUA
NPI *Network Protocol Independence*
NTDI *Núcleo de Televisão Digital Interativa*
NTSC *National Television System Committee*
OCAP *OpenCable Application Platform*
PSP *PlayStation Portable*
SBTVD-T Sistema Brasileiro de Televisão Digital - Terrestre
SMIL *Synchronized Multimedia Integration Language*
STB *Set-Top Box*
TVD Televisão Digital
TVDi Televisão Digital Interativa
UFSC Universidade Federal de Santa Catarina
XDMB *eXtensible DMB*
XHTML *eXtensible HyperText Markup Language*
XML *Extensible Markup Language*
W3C *World Wide Web Consortium*

SUMÁRIO

1 INTRODUÇÃO	1
1.1 OBJETIVOS	2
1.2 JUSTIFICATIVA	2
1.3 ESTRUTURA DO TRABALHO	3
2 A TV DIGITAL	4
2.1 TV DIGITAL	4
2.1.1 Infra-estrutura	4
2.1.2 Interatividade	9
2.2 CENÁRIO ATUAL DA TV DIGITAL NO MUNDO	12
2.3 PADRÃO BRASILEIRO DE TV DIGITAL - TERRESTRE	15
2.3.1 Ginga	16
2.3.2 Ginga-J	16
2.3.3 Ginga-NCL	17
2.4 TRABALHOS CORRELATOS	22
2.4.1 Ferramentas para criação de <i>scripts</i>	23
2.4.2 Ferramentas de Autoria	24
2.4.3 Exemplos de aplicações em Televisão Digital Interativa (TVDi)	27
2.5 TRABALHOS PRELIMINARES	28
2.5.1 Nível 1	29
2.5.2 Nível 2	30
2.5.3 Nível 3	30
2.6 CONCLUSÕES	32
3 PROPOSTA INTEGRADA DAS FERRAMENTAS	34
3.1 DEFINIÇÃO DAS FERRAMENTAS	34
3.1.1 Ferramenta de Autoria - Célula	34
3.1.2 Servidor	36
3.1.3 Emulador	37
3.1.4 Integração	38
4 IMPLEMENTAÇÃO	40
4.1 FERRAMENTA	40
4.1.1 Recursos da ferramenta de autoria	41
4.1.2 <i>Templates</i> disponíveis	43
4.2 SERVIDOR	46
4.2.1 Aplicações	46
4.2.2 Dados do canal de retorno	48

4.3	RESULTADOS	51
4.3.1	Mundo Plano	52
4.3.2	Mãos	55
4.3.3	Design Gráfico	57
5	CONCLUSÃO E TRABALHOS FUTUROS	59
5.1	TRABALHOS FUTUROS	59
	REFERÊNCIAS	61

1. INTRODUÇÃO

A Televisão Digital (TVD) constitui na transmissão de forma digital do sinal responsável pela exibição de um canal de televisão. Para esta transmissão digital são enviados dados na forma de *bits*, de forma similar à já utilizada em redes de computadores. A mudança na transmissão traz novos desafios em todas as etapas do modelo já existente de televisão, como, por exemplo:

- **Parque tecnológico**, com novas antenas e decodificadores para os padrões no formato digital;
- **Telespectador**, com a utilização de equipamentos para decodificação e motores de execução para interatividade; e
- **Etapa de produção**, com novos equipamentos de gravação de imagens, detalhes de figurino, maquiagem e criação de *softwares* para interatividade.

O foco da TVD mais explorado até o momento no Brasil tem sido a utilização da *High-Definition Television* (HDTV), porém outro aspecto inovador deste modelo é a interatividade do telespectador. A vinculação de *softwares* a vídeos possibilita a interatividade do telespectador, e estes são chamados de programas interativos. Estes, juntamente com a TVD, formam o conceito conhecido como Televisão Digital Interativa (TVDi), que vem sendo pouco explorado até o momento, uma reação normal tendo em vista a dependência da transmissão do sinal digital para a interatividade funcionar. Uma das únicas áreas exploradas é a de governo eletrônico (SCHIBELSKY et al., 2008).

Para a transmissão do sinal digital, padrões vêm sendo criados e adotados. Um deles é o Sistema Brasileiro de Televisão Digital - Terrestre (SBTVD-T), criado a partir do padrão *Integrated Services Digital Broadcasting Terrestrial* (ISDB-T), desenvolvido no Japão (MELONI, 2007). A interatividade em TVD pode ser explorada com o auxílio de um *middleware*¹ em todos os padrões hoje disponíveis no mundo. Para a realização de interatividade, o SBTVD-T contém como *middleware* um padrão conhecido como Ginga, o qual utiliza *Nested Context Language* (NCL) e Java como linguagem de implementação (ABNT, 2008c).

Para a utilização do padrão SBTVD-T de forma interativa, é necessário conhecimento sobre programação para o desenvolvimento de

¹ *Middleware* é uma camada de *software* instalada em um sistema operacional que serve como comunicação entre as aplicações e o SO.

programas. Este trabalho disponibiliza um conjunto de ferramentas voltadas para jornalistas, permitindo que a produção de conteúdo interativo em TVDi seja realizada por este tipo de profissional, sem a necessidade de intervenção de programadores. O ferramental, atualmente, está voltado para a produção de conteúdo interativo para o padrão SBTVD-T.

Observa-se que este tema está sendo pouco explorado, tanto no Brasil quanto no exterior. Isto ocorre devido à inovação e ao pouco conhecimento da tecnologia de TVDi. A falta de um método ou ferramenta que possibilite a produção de conteúdo para TVDi torna sua utilização quase inviável devido ao alto custo de desenvolvimento e à necessidade de capacitação diferenciada das equipes jornalísticas geradoras dos conteúdos/vídeos televisivos.

1.1 OBJETIVOS

Objetivo Geral

Disponibilizar um conjunto de ferramentas para a criação e o gerenciamento de aplicações com interatividade em vídeos para TVDi, com foco na produção por jornalistas.

Objetivos Específicos

- Elaborar um documento com revisão bibliográfica de conceitos relevantes acerca de TVDi;
- Propor uma ferramenta de autoria para SBTVD-T capaz de inserir elementos de interatividade;
- Criar *templates* de interatividade integrados com a ferramenta de autoria; e
- Construir um servidor com a capacidade de receber e armazenar informações enviadas por uma aplicação de TVDi e que simule uma emissora de televisão para a execução e verificação dos programas interativos no padrão SBTVD-T.

1.2 JUSTIFICATIVA

A produção de conteúdo interativo é uma rotina diária em ambientes computacionais, principalmente em ambientes distribuídos como a internet. Muitos destes conteúdos são desenvolvidos por programadores de sistemas, mas uma pequena parte é desenvolvida por profissionais detentores do conhecimento, como, por exemplo, jornalistas ou docentes

de Educação a Distância (EAD). Este desenvolvimento é feito com ferramentas específicas, que propiciam um desenvolvimento rápido e sem a necessidade de conhecimento em linguagens de programação (SILVEIRA, 2007).

Na criação de conteúdo que envolve interação na TVDi, o conhecimento pertence hoje aos programadores. Contudo, a produção de conteúdo por programadores torna-se inviável pelo fato de que é indispensável o conhecimento adquirido ao longo dos anos por profissionais de jornalismo. Neste ponto surge a necessidade de produção de conteúdo diretamente por profissionais como jornalistas, produtores e docentes de EAD, assim como já acontece com a produção de conteúdo na web (SILVEIRA, 2007).

Pesquisas e projetos vêm sendo desenvolvidos neste contexto, tentando unir a necessidade e o conhecimento de jornalistas à capacidade de programadores de gerar aplicações. O cenário ideal, entretanto, seria uma ferramenta que auxiliasse estes profissionais na produção independente de conteúdos interativos; dentre alguns trabalhos nesta área estão (MIRANDA; PICCOLO; BARANAUSKAS, 2008), (GOMES; LIMA; NEVADO, 2008), (CATTELAN et al., 2008) e (BORGES et al., 2008).

Um projeto pioneiro neste sentido foi o SBTVD (CAMPAGNOLO et al., 2004), que desenvolveu aplicações interativas para televisão digital a fim de testar os possíveis recursos desta nova tecnologia, que estava surgindo na época. Uma das conclusões dos pesquisadores do projeto é que o tempo de desenvolvimento dos programas interativos demandava um porção considerável do processo de produção. Este cenário, de dificuldade no desenvolvimento de aplicações para TVDi, gera uma necessidade de ferramentas de autoria voltadas para TVD.

1.3 ESTRUTURA DO TRABALHO

Esta dissertação está dividida em quatro capítulos, iniciando por este, a introdução, que dá uma visão geral sobre o trabalho. No capítulo dois é apresentada a revisão bibliográfica referente aos temas de televisão digital, qual seu cenário atual no mundo, detalhando o padrão utilizado no Brasil, trabalhos realizados nesta área, alguns trabalhos desenvolvidos para testar a tecnologia Ginga-NCL, a definição das ferramentas e, por fim, a implementação. No terceiro capítulo, é apresentada a avaliação do modelo, explicando três projetos realizados utilizando o método proposto. E por último são apresentadas as conclusões sobre o trabalho e sugeridos trabalhos futuros.

2. A TV DIGITAL

Neste capítulo será apresentada a revisão bibliográfica que auxiliou na elaboração do ferramental. Serão vistas características sobre televisão digital, o cenário atual da televisão digital no mundo, especificamente o padrão utilizado no Brasil e um levantamento de trabalhos correlatos à produção de conteúdo de televisão digital interativa. Ao final são apresentados trabalhos preliminares desenvolvidos e a implementação do ferramental para produção de conteúdo.

2.1 TV DIGITAL

A televisão vem evoluindo ao longo do tempo na qualidade de transmissão da imagem. Os televisores precisaram ser melhorados para suportar esta nova evolução, passando de *Cathode Ray Tube* (CRT) para *Liquid Crystal Display* (LCD) ou plasma. Os aparelhos de LCD e plasma proporcionam uma melhora de qualidade que está diretamente ligada à capacidade de envio e codificação das informações para os aparelhos de TV.

Na televisão convencional, o sinal analógico é recebido e apresentado para o telespectador diretamente, sem nenhuma forma de processamento. Com o sinal digital, surge a necessidade de um processamento prévio à apresentação dos áudios e vídeos recebidos. Este processo extra é conhecido como decodificação do sinal. Novas formas de transmissão permitem melhorias na qualidade de imagem, ocasionando alta qualidade de definição, também chamada de HDTV (BENOIT, 2008).

Como outra possibilidade desta nova forma de transmissão, a interatividade é reestruturada, criando-se a possibilidade de execução de programas interativos (TEIXEIRA, 2009; GAWLINSKI, 2003). Além da melhoria na qualidade das imagens, outra vantagem da TVD é a capacidade de receber o sinal em movimento, possibilitando que aparelhos móveis recebam e apresentem a programação de emissoras de televisão (BENOIT, 2008). Uma ilustração dos componentes envolvidos nesta nova tecnologia é apresentada na Figura 1.

Nas seções a seguir são apresentadas as três infra-estruturas necessárias para o funcionamento deste novo conceito de televisão, que são infra-estrutura de *hardware*, sistema operacional e *middleware*.

2.1.1 Infra-estrutura

- **Hardware:** a TVD difere da televisão analógica na forma de transmissão. Informações de áudio, vídeo e arquivos são enviados na forma de dados codificados. Para a recepção eficiente

destes dados codificados, é necessário um equipamento que realize a decodificação; neste caso, o equipamento pode ser integrado à televisão ou a um dispositivo externo específico para este fim. A decodificação também pode ser realizada em um computador normal através de um programa (*software*). Entretanto, para realizar a decodificação das informações e transformá-las em áudio, vídeo e dados de forma eficiente, normalmente é utilizado um *hardware* para realizar a decodificação (YE; LI; SU, 2006).

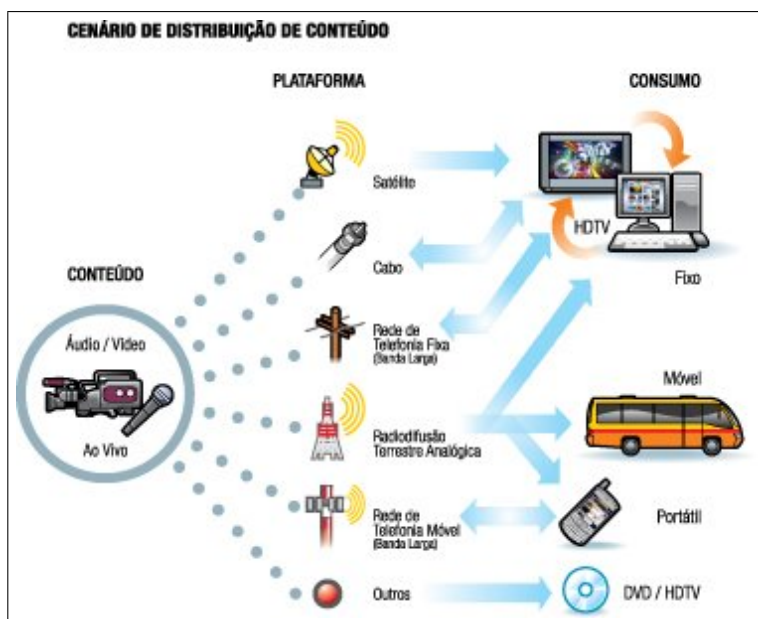


Figura 1: Componentes envolvidos na TV Digital
Fonte: <http://tvdigitalcuritiba.blogspot.com>

No Brasil, este equipamento de conversão dos dados possui a capacidade de receber pacotes no padrão MPEG2², decodificar no codec³ H.264 e permitir a execução de *softwares* desenvolvidos no padrão Ginga.

A transmissão do sinal para estes equipamentos se dá em uma

²**MPEG2:** padrão de codec desenvolvido pelo Moving Picture Experts Group (MPEG).

³**Codec:** Codificador/Decodificador, dispositivo de *hardware* ou *software* que codifica/decodifica sinais.

única banda, dividida em três partes que servirão para o equipamento distinguir qual informação diz respeito a cada parte da banda de transmissão. Uma ilustração desta divisão da banda de transmissão é apresentada na Figura 2.

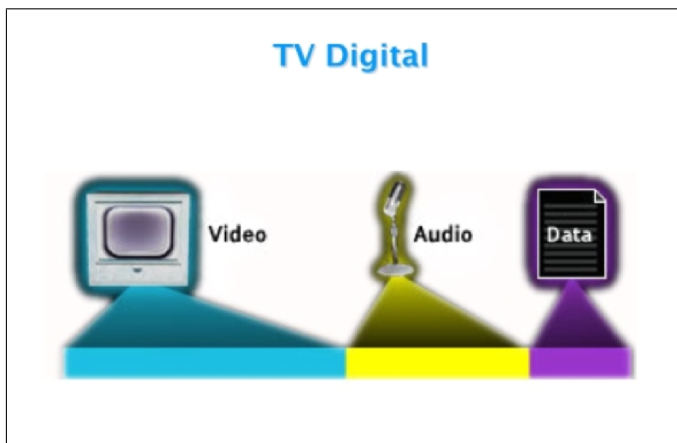


Figura 2: Divisão de banda na TV Digital

Fonte: <http://www.telemidia.puc-rio.br>

- **Sistema Operacional:** os *hardwares* têm a necessidade de serem controlados por um *software* (RODRIGUES; MORENO; SOARES, 2006) conhecido como sistema operacional. Estes, na TVD, têm capacidade de receber *streams*⁴ de áudio e vídeo, repassados pelo *hardware*, e de processar eventuais aplicativos de interatividade (documento hipermédia) disponibilizados por um programa que esteja no ar em um canal de TVD.

Segundo Rodrigues, Moreno e Soares (2006), nos padrões europeu *Digital Video Broadcasting - HyperText Markup Language* (DVB-HTML), americano (ACAP-X ou XXML) e japonês *Broadcast Markup Language* (BML), que serão vistos na Seção 2.2, os módulos são interpretadores de documentos no formato XHTML com suporte a *Cascading Style Sheets* (CSS), *Document Object Model* (DOM) e *European Computer Manufacturers Association Script* (ECMAScript). Estes módulos serão desenvolvidos desta forma para facilitar a interatividade, baseando-se no conceito já

⁴*Streaming* é o método de transmitir dados na internet diretamente por usuários de computador, sem a necessidade de fazer *download*.

bem difundido de hipertextos da internet, conhecido também como HTML.

No Brasil, uma equipe formada por dois laboratórios distintos, Telemídia ⁵ da PUC-Rio ⁶ e LAViD ⁷ da UFPB ⁸, desenvolve atualmente o *software* referência necessário para prover a infraestrutura adequada para que aplicativos interativos funcionem no padrão brasileiro de televisão digital (Fórum SBTVD, 2009).

Além do *software* responsável por controlar o *hardware*, é necessária uma camada (*middleware*) responsável por controlar o fluxo e as aplicações interativas recebidas junto como o vídeo/áudio decodificados.

- ***Middleware***: é a camada de *software* que fica entre a aplicação e a camada de infra-estrutura de execução, composta pelo *hardware* e o sistema operacional. No contexto de TV Digital o *middleware* consiste na máquina que permite a execução das linguagens oferecidas, bem como nas bibliotecas de funcionalidades, que permitem a execução das aplicações. Esta camada de *software*, quando identificada na área de TV Digital, pode ser observada em dois formatos: estruturado e declarativo.

Atualmente os *middlewares* declarativos são mais difundidos; dentre eles podem-se encontrar DVB-HTML (europeu), *Digital television Application Software Environment* (DASE) (americano), BML (japonês) (MORENO, 2006) e nos últimos anos o padrão brasileiro SBTVD-T, com adesão de muitos países na América Latina.

O DVB-HTML faz parte de uma especificação maior chamada *Multimedia Home Platform* (MHP) (DVB Project Office, 2009), que foi desenvolvida para estar em conformidade com o padrão europeu *Digital Video Broadcasting* (DVB). O *middleware* MHP é baseado no uso de uma máquina virtual Java, tendo assim sua implementação nos terminais chamada de DVB-J. Entretanto, como alternativa mais utilizada atualmente, existe um modelo capaz de interpretar documentos hipermídia (DVB-HTML), que usa como linguagem declarativa o XHTML.

O padrão DVB-HTML, além do XHTML, permite também o suporte de outras linguagens, como: CSS, utilizada para desenvolvi-

⁵<http://www.telemidia.puc-rio.br/>

⁶<http://www.puc-rio.br/>

⁷<http://www.lavid.ufpb.br/>

⁸<http://www.ufpb.br/>

mento de *layout* de documentos hipermídia; ECMAScript, responsável por adicionar recursos lógicos de programação a documentos XHTML; e DOM, utilizado para permitir a inclusão de *scripts* ECMAScript e aplicativos DVB em Java (DVB-J) (códigos procedurais). Todos estes recursos seguem normas *World Wide Web Consortium* (W3C) (W3C, 2009).

Outro padrão que se destaca é o DASE que foi criado em 2002 pelo grupo *Advanced Television Systems Committee* (ATSC), por ser pioneiro (ATSC, 2009a). Hoje o DASE foi fundido com o padrão *OpenCable Application Platform* (OCAP), formando o padrão *Advanced Common Application Platform* (ACAP) (SALEMI; BJÖRKQVIST; LILIUS, 2008). Assim como o padrão MHP, este padrão também permite a geração de aplicações em XHTML com interpretação de CSS e ECMAScript, e em linguagem Java.

O padrão japonês inicialmente foi concebido apenas como declarativo (BML), e posteriormente foi implantado o *Association of Radio Industries and Businesses* (ARIB), que é um *middleware* procedural (MATSUMURA et al., 2003; MORENO, 2006).

Assim como as linguagens DVB-HTML e *eXtensible DMB* (XDMB), o MHP possui uma linguagem baseada em XHTML chamada de BML que suporta as linguagens CSS2, ECMAScript e DOM (MATSUMURA et al., 2003).

Todos os padrões apresentados utilizam tecnologias desenvolvidas para a *web* como suporte para interatividade de conteúdos em TVD, entretanto todos adotam diferentes funcionalidades para XHTML, e estas diferenças podem ser percebidas na Tabela 1.

Recursos que são disponibilizados por todos os padrões e não são apresentados nesta tabela são: “Structure”, “Hypertext”, “List”, “Presentation”, “Bidirecional Text”, “Form”, “Tables”, “Client-side image map”, “Object”, “Frames”, “Targets”, “Meta-Information”, “Scripting”, “Style sheet” e “Style attribute”. O elemento “applet” existente na linguagem XHTML não é implementado por nenhuma das linguagens apresentadas.

No Brasil, o *middleware* utilizado é dividido em duas implementações distintas chamadas de Ginga-NCL, que é responsável por interpretar documentos na linguagem NCL, e Ginga-J que possui a capacidade de interpretar aplicativos em *bytecode*⁹ na linguagem

⁹ **Bytecode:** código em *bytes*, é um código intermediário de execução, necessitando de uma máquina virtual para interpretá-lo.

Java (ABNT, 2008c).

Quadro 1: Diferenças das extensões de XHTML para interatividade em TV Digital

XHTML	DVB-HTML	XDMB	BML
Text		X	X
Edit			X
Basic forms			X
Basic tables			X
Image	X		X
Server-side image map			X
IFrame	X		X
Intrinsic events		X	X
Link	X		X
Base	X		X
Name identification		X	

Fonte: Adaptada de (PAES, 2006)

Na tentativa de unificar os padrões de *middleware*, existe ainda uma especificação chamada de *Globally Executable MHP* (GEM), que possui características do *middleware* MHP. Foi proposta pelo grupo DVB e adotada pelos *middlewares* japonês e americano (FILHO; LEITE; BATISTA, 2007). Esta especificação pretende unificar as plataformas de *middleware*, formando assim aplicações fáceis de serem migradas de um padrão para outros. Também permite o uso do mesmo aplicativo de interação nos mais diversos padrões de *middleware* desenvolvidos.

2.1.2 Interatividade

Interatividade significa a capacidade de um dispositivo interagir ou permitir interação com o seu respectivo utilizador (GELEIJNSE; ALLAKSEYEU; SARROUKH, 2009). Na televisão, a interatividade não é um novo elemento inserido pela tecnologia de TVD, e sim uma readequação da forma de interação com o telespectador (TEIXEIRA, 2009). Segundo Lugmayr, Niiranen e Kalli (2004), a interatividade na televisão digital cria uma maior diversidade de serviços interativos.

Vários autores descrevem a classificação de interatividade. Uma destas, defendida por Reisman (2009), é agrupada em três níveis de abrangência:

- **Interatividade com o conjunto televisivo:** limitada ao uso do controle remoto, permitindo a troca de canais e o avanço, o retrocesso e a pausa de imagens no videocassete. O telespectador, neste nível, não pode alterar o conteúdo, apenas a forma como ele é visto.
- **Interatividade com o conteúdo do programa da televisão:** nesse nível a interatividade é total e representa o maior desafio para os produtores. Nesta visão, o telespectador pode controlar o conteúdo do programa que está assistindo, assim como é capaz de controlar a programação que gostaria de assistir.
- **Interatividade com o conteúdo que se encontra na televisão:** este nível contém as mesmas características que o nível anterior e, ainda, funcionalidades que mudarão nas próximas décadas a forma como a televisão é assistida. Ele permite obter informações a qualquer momento sobre as condições climáticas, esportes, a programação das emissoras, notícias, etc., assim como obter informações detalhadas acerca dos produtos anunciados e poder comprá-los.

Outra classificação é apresentada por Montez e Becker (2005) e é discutida em sete níveis:

- **Nível 0:** caracteriza-se pela exposição de imagens em preto e branco, dispondo de poucos canais. Neste caso, a interatividade está restrita a ações simples como ligar e desligar a televisão, trocar de canal e regular o brilho, o contraste e o volume.
- **Nível 1:** a televisão ganha cores, um número maior de canais e o controle remoto, reduzindo o esforço físico necessário para o controle do aparelho de televisão e, conseqüentemente, aumentando o conforto do usuário.
- **Nível 2:** surgem periféricos como o videocassete, as câmeras portáteis e os aparelhos de jogos eletrônicos, que, acoplados à televisão, permitem que os programas das emissoras sejam gravados e assistidos posteriormente, assim também possibilitando ver filmes e o lazer com jogos eletrônicos.
- **Nível 3:** as emissoras permitem um nível de interatividade através de ligações telefônicas, mensagens de correio eletrônico ou envio de fax.

- **Nível 4:** o telespectador tem a possibilidade de escolher, em tempo real, ângulos de câmeras ou dar diferentes encaminhamentos às informações.
- **Nível 5:** o próprio telespectador pode participar da programação, enviando vídeos de baixa qualidade produzidos através de *webcam* ou filmadoras analógicas.
- **Nível 6:** semelhante ao nível 5, mas agora permitindo a transmissão de vídeos de alta qualidade.
- **Nível 7:** telespectador alcança a interatividade plena, gerando conteúdo da mesma forma que a emissora. Neste modelo, o telespectador rompe o monopólio de produção e veiculação das redes de televisão.

Outra visão é apresentada por Crocomo (2007), em que existem três níveis de interação:

- **Nível 1:** também chamado de “interação local”, é a situação em que o telespectador interage com informações armazenadas no terminal ou *set-top box*. Neste nível, é possível citar interações como sinopse sobre um filme, letra de uma música em um *clip*, escalação em uma partida de futebol ou perguntas educativas sem retorno de informações para a emissora.
- **Nível 2:** onde existe um canal de retorno para a emissora. Neste caso, o envio de informações para a emissora poderá ser feito de forma síncrona ou não. Já como exemplos deste nível é possível citar os questionários sobre uma pergunta de um locutor, perguntas sobre satisfação dos telespectadores ou votações em programas de *reality show*, onde a emissora recebe através do canal de retorno informações passadas pelo telespectador.
- **Nível 3:** onde a interação envia e recebe informações em tempo real (ZALETELJ; SAVIC; MEZA, 2009). Neste caso, pode-se citar como exemplos aplicações como salas de bate-papo (*chat*) e jogos *multiplayer*.

Neste trabalho é adotada a definição apresentada por Crocomo (2007). A escolha deve-se à possibilidade de melhor definir interatividade do ponto de vista tecnológico e de estruturar os níveis de tal forma a possibilitar a classificação quanto aos recursos computacionais.

A interatividade na TVD pode ir muito além da interação entre o controle remoto e um programa (VIANA; LUCENA JR., 2009; OBRIST et al., 2009).

Uma vez apresentados os elementos envolvidos no conceito de TVD, será descrito na próxima seção quais são os principais padrões utilizados no mundo.

2.2 CENÁRIO ATUAL DA TV DIGITAL NO MUNDO

Nesta seção veremos os padrões existentes, suas características e como são utilizados ao redor do mundo. Atualmente existem quatro padrões de TVD adotados no mundo: ATSC, também chamado de padrão americano (adotado nos Estados Unidos, Canadá, México, Guatemala, Honduras e Coréia do Sul); DVB, chamado também de padrão europeu (utilizado em cerca de 60 países, incluindo União Europeia, Rússia, Índia, Filipinas, Malásia, Arábia Saudita, Turquia, Vietnã, Nova Zelândia, Austrália e Irã); *Digital Multimedia Broadcasting* (DMB), (chinês); e *Integrated Services Digital Broadcasting* (ISDB) (empregado no Japão), chamado de padrão japonês. Um quinto padrão surge no Brasil e é conhecido como SBTVD-T, que é baseado no ISDB com uma variação de *middleware* (LEE, 2008; ginga.org.br, 2009). Este padrão já mostra potencial, sendo exportado para países como Argentina, Peru, Chile e Venezuela (Fórum SBTVD, 2009). Uma visão completa sobre os padrões utilizados no mundo é apresentada na Figura 3, e os padrões são mais detalhados nas seções seguintes.

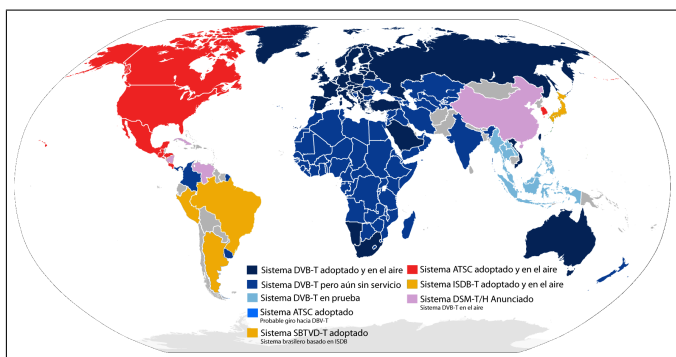


Figura 3: Mapa de padrões de TV Digital no Mundo

Fonte: <http://en.wikipedia.org/wiki/>

- **Padrão americano (ATSC):**

Nos Estados Unidos, o padrão analógico *National Television System Committee* (NTSC) estava previsto para ser substituído completamente pelo digital em fevereiro de 2009. Contudo, o padrão ATSC, desenvolvido pelo comitê Advanced Television Systems Committee, teve sua consolidação apenas em junho do mesmo ano do encerramento da transmissão analógica no país (ATSC, 2009b). Este padrão define imagens de qualidade de até 1920×1080 *pixels* em formato *wide screen* (16:9), diferenciado do formato atual de televisão analógica (4:3) e se aproximando mais dos padrões de cinema. Contudo, esta resolução não é a única suportada por este padrão. Assim como o formato de mídia utilizado (A/53 Part 4 e A/63), este padrão consegue transmitir em diversas resoluções e nos dois formatos: normal (4:3) e *wide screen* (16:9).

Juntamente com o padrão ATSC existe o ATSC - *Mobile/Handheld* (ATSC-M/H), que é candidato a ser o padrão de TVD para aplicativos móveis, o qual utiliza uma parte da banda disponível para a transmissão de *stream* de TVD para os mais diversos aparelhos móveis. Este tipo de processo é usado em todos os padrões em que é separada uma parte da banda para estes dispositivos. Uma pequena parte desta banda garante uma qualidade inferior, porém suficiente para dispositivos móveis, devido ao seu tamanho de *display* reduzido (MORENO, 2006 apud ARIB, 2004).

O padrão ATSC apresenta deficiências tecnológicas importantes em relação à radiodifusão quando comparado a outros padrões, provocando problemas na recepção em aparelhos fixos e portáteis com antena interna localizados em áreas com presença de interferência. Na época do desenvolvimento da modulação deste padrão, a telefonia móvel ainda não havia adquirido a importância dos dias de hoje. O objetivo principal a ser alcançado era oferta de vídeos em alta definição. Embora tenha ocorrido uma evolução dos receptores que suportam este padrão, a utilização da modulação deste padrão em aparelhos móveis ainda permanece prejudicada (BELAÑO; VIEIRA, 2004) (FERNANDES; LEMON; ELIAS, 2004).

- **Padrão europeu (DVB):**

O DVB está dividido em quatro partes, de acordo com as diversas formas de transmissão possíveis:

- satélite, sendo adotado DVB - *Satellite* (DVB-S), DVB - *Satellite 2* (DVB-S2) e DVB - *Satellite Handheld* (DVB-SH);

- cabo (DVB - *Cable* (DVB-C));
- televisão terrestre com DVB - *Terrestrial* (DVB-T) e DVB - *Terrestrial 2* (DVB-T2); e
- televisão terrestre para dispositivos móveis com DVB-H e DVB-SH;

Assim como os demais padrões, o DVB prevê um espaço entre o áudio e o vídeo para a transmissão de dados (DVB-DATA) e especifica um canal de retorno com as mais diversas formas de transmissão (satélite, *Grupos Especial Mobile* (GSM), internet, etc.) com os protocolos *Network Protocol Independence* (NPI), *Internet Protocol* (IP), DVB - IPTelevision (DVB-IPTV) e outras formas de internet independentes (ATSC A/53 Part 1, 2009). O MHP e o GEM são os *middlewares* criados para o desenvolvimento de interatividade na TVD europeia (DVB Project Office, 2009). Baseados em Java, estes *middlewares* têm a capacidade de trabalhar com *streams* de áudio e vídeo, além de ter a capacidade de interpretar aplicativos em documentos hipermídia (DVB-HTML com capacidade de interpretação de XHTML, CSS, DOM e ECMAScript) e aplicativos em *bytecode* da linguagem Java (DVB-J), que obedecem o padrão de Xlets. O padrão Xlet é um conceito similar aos *applets* utilizados em páginas *web*, que foi introduzido no padrão JavaTV pela empresa Sun Microsystems.

● Padrão Chinês (DMB):

Na China o DMB - *Terrestrial/Handheld* (DMB-T/H), também chamado de *Digital Terrestrial Multimedia Broadcast* (DTMB) (CHESTERFIELD; RODRIGUEZ, 2005), é um padrão direcionado a dispositivos móveis, que foi desenvolvido para transmissão terrestre mas contém também seu padrão DMB - *Satellite* (DMB-S) para transmissão através de satélites. Foi desenvolvido para ser utilizado em dispositivos móveis com serviços inovadores, TV móvel, tráfego e informações seguras. Já está em uso na Coréia do Sul, disponibilizando serviços de multimídia em *broadcasting* para dispositivos móveis (YIM; CHOY; LIM, 2009). Ainda segundo Yim, Choy e Lim (2009), a adoção do sistema MPEG-4 BIFS permite uma pseudo-interatividade contendo serviços.

O padrão DMB vem sendo especificado desde 2001 para atender especificamente a China, não sendo utilizado por nenhum outro país até então (BELAÑO; VIEIRA, 2004). Para ressaltar a impor-

tância deste padrão, ele já é utilizado também na França para serviços específicos de rádio (PELLAN; CONCOLATO, 2008).

- **Padrão japonês (ISDB):**

O padrão japonês utiliza a tecnologia chamada BML e é mantido pelo grupo ARIB. Assim como os demais padrões pelo mundo, o padrão ISDB é dividido em subcategorias diferenciadas pelo meio de transmissão: satélite (ISDB-S), terrestre (ISDB-T) e cabo (ISDB-C). Os padrões ISDB e DVB diferenciam-se dos demais por implementar outras formas de compressão, incluindo JPEG e MPEG-4 (CHESTERFIELD; RODRIGUEZ, 2005; MATSUMURA et al., 2003). O ISDB utiliza o MPEG-2 como compressão para transporte, pois este é capaz de transmitir uma qualidade de alta definição (HDTV) de áudio e vídeo.

O ISDB, assim como os demais padrões, prevê um espaço na banda de transmissão para o envio de dados para aplicações interativas e um canal de retorno através de muitas formas de transmissão como linhas telefônicas, redes de computadores por cabo, telefonia celular, *wireless*, entre outras (ABNT, 2008a).

Os receptores adotados para este padrão atualmente são televisões digitais e *Set-Top Box* (STB). STB é conhecido por facilitar a inclusão de pessoas menos favorecidas financeiramente, possibilitando transformar o sinal digital de forma a ser transmitido a um televisor analógico. (ABNT, 2008a)

- **Padrão brasileiro (SBTVD-T):**

No Brasil é utilizada uma variação do padrão ISDB-T, que possui compressão em MPEG-4 AVC, também conhecida como H-264, e áudio com *High-Efficiency Advanced Audio Coding* (HE-AAC). Esta variação diferencia-se dos demais padrões por utilizarem em geral o padrão de compressão MPEG-2 (RBSTV, 2009).

2.3 PADRÃO BRASILEIRO DE TV DIGITAL - TERRESTRE

No Brasil, o SBTVD-T é baseado no padrão ISDB-T com compressão de dados em MPEG-4 AVC, também conhecido como H-264, e áudio com HE-AAC. O padrão brasileiro diferencia-se dos demais padrões por utilizar o MPEG-4 AVC no lugar de MPEG-2, como geralmente ocorre nos demais padrões (RBSTV, 2009). Nesta seção será apresentado o *middleware* Ginga em suas duas formas: Ginga-J e Ginga-NCL.

2.3.1 Ginga

Ginga, o nome do *middleware* específico para ISDB-T e que está sendo adotado oficialmente como padrão brasileiro do sistema de TVD Terrestre (FILHO; LEITE; BATISTA, 2007).

A origem da palavra, Ginga, vem da forma de caminhar, falar, dançar e relacionar-se. Esta expressão é mais conhecida no Brasil por ser o movimento básico para a capoeira. Segundo o site Ginga.org.br (ginga.org.br, 2009), o nome Ginga foi escolhido em reconhecimento à cultura, à arte e à contínua luta por liberdade e igualdade do povo brasileiro.

O decreto N° 4.901, de 26 de novembro de 2003, estipula, dentre outras coisas, que a TV Digital no Brasil deve promover a inclusão digital e a diversidade cultural, visando a democratização da informação e, por este motivo, o *middleware* Ginga é desenvolvido de forma a ser distribuído gratuitamente.

Aplicações inicialmente desenvolvidas para testar o potencial de aplicações para TVDi comprovam as intenções básicas do decreto. O projeto SBTVD-T (CROCOMO et al., 2009) desenvolveu aplicações voltadas à população, como atendimento em postos de saúde e programas interativos para passar conhecimentos sobre doenças e animais peçonhentos.

O padrão Ginga é dividido em duas partes: (a) parte responsável pela execução de programas procedurais chamados de Ginga Java (Ginga-J); e (b) parte responsável pela execução de programas declarativos, chamados também de Ginga - *Nested Context Language* (Ginga-NCL) (MORENO, 2006; ABNT, 2008b).

2.3.2 Ginga-J

Ginga-J é um subsistema do *middleware* Ginga que define todas as *Application Programming Interface* (API)s em Java que englobam os formatos de dados e protocolos até o nível de aplicação (Figura 4). Não são necessários *scripts*, pois a própria linguagem Java, na qual o Ginga-J é fundamentado, já é uma linguagem procedural e não uma linguagem declarativa. A proposta de arquitetura formulada e apresentada por (FILHO; LEITE; BATISTA, 2007) serve como referência da estrutura do padrão brasileiro Ginga, chamada de Ginga-J, e apresenta os recursos e ilustra as interações entre as diversas camadas necessárias para a execução deste tipo de linguagem pelo padrão SBTVD-T. A arquitetura proposta Ginga-J é compatível com a especificação que tenta padronizar os *middlewares* de TV Digital chamada GEM. Contudo, o Ginga-J não

possui até o momento uma implementação referência. Muitos desenvolvedores utilizam o emulador XleTView, o qual utiliza muitas bibliotecas padrão, porém este emula aplicações no padrão JAVATV, que é compatível com MHP (SVEDEN, 2009).

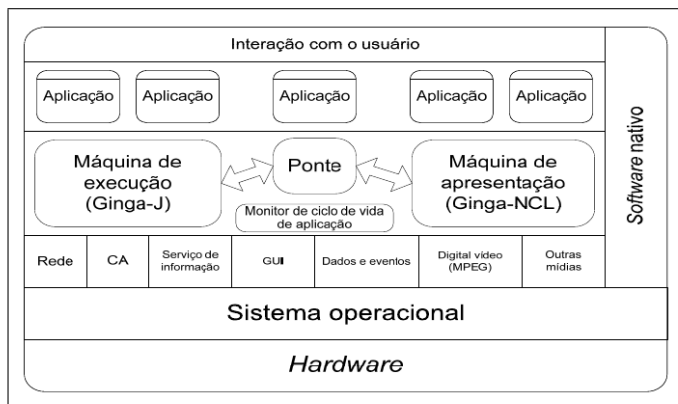


Figura 4: Ginga-J: Arquitetura e ambiente de execução

Fonte: (ABNT, 2008b)

2.3.3 Ginga-NCL

O Ginga-NCL é parte integrante do padrão Ginga. Desenvolvido para servir como *middleware* responsável por execução de aplicações interativas declarativas, tem a capacidade de interpretar documentos na linguagem NCL. Diferentemente dos padrões europeu e americano, que têm a capacidade de interpretar documentos hipermídia no formato XHTML, este *middleware* também interpreta documentos em NCL, CSS, ECMAScript¹⁰ e Lua (ABNT, 2008c).

Diferentemente do Ginga-J, no Ginga-NCL já existem sistemas capazes de auxiliar no desenvolvimento de programas interativos utilizando a linguagem NCL, como pode ser visto na Seção 2.4, juntamente com ferramentas capazes de auxiliar outras linguagens utilizadas em todo o mundo.

Exemplos que demonstram funcionalidades de documentos Ginga-NCL, bem como o formato do documento NCL, podem ser baixados no

¹⁰ECMAScript é uma linguagem de *script* (procedural) padronizada pela ECMA International, muito utilizada em tecnologias para a internet (ECMA International, 2009).

portal do *software* público, mantido pelo governo federal, mais precisamente no repositório do grupo Ginga (Portal de Software Publico, 2009). Parte de um dos exemplos disponibilizados pelo grupo é apresentado na Figura 5. Este exemplo apresenta um vídeo em alta definição e, ao iniciar este vídeo, apresenta o título dele, que é um documento HTML.

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<ncl id="exemplo" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
  <head>
    <regionBase>
      <region id="rgTV" width="1920" height="1080">
        <region id="rgTitulo1" left="448" top="116" width="1024" heig
        <region id="rgVideo1" left="448" top="156" width="1024" heigh
      </region>
    </regionBase>
    <descriptorBase>
      <descriptor id="dTitulo1" region="rgTitulo1">
        <descriptorParam name="border" value="none" />
      </descriptor>
      <descriptor id="dVideo1" region="rgVideo1">
        <descriptorParam name="soundLevel" value="1" />
      </descriptor>
    </descriptorBase>
    <connectorBase>
      <importBase alias="connectors" documentURI="connectorBase.ncl"
    </connectorBase>
  </head>
  <body>
    <port id="pInicio" component="video1" />
    <media type="text/html" id="titulo1" src="media/titulo1.html" de
    <media type="video/mpeg" id="video1" src="media/video1.mpg" desc
    <link id="lVideo1Titulo1Start" xconnector="connectors#onBegin1St
      <bind component="video1" role="onBegin" />
      <bind component="titulo1" role="start" />
    </link>
    <link id="lVideo1Titulo1Stop" xconnector="connectors#onEnd1StopN
      <bind component="video1" role="onEnd" />
      <bind component="titulo1" role="stop" />
    </link>
  </body>
</ncl>

```

Figura 5: Exemplo de código Ginga-NCL

O padrão Ginga-NCL limita a utilização de mídias em documentos aos seguintes formatos: arquivos CSS, XML, HTML ou texto; imagens no formato PNG, JPEG, GIF e BMP; vídeos MPEG e MPEG4; e aplicativos NCL-Lua e NCLet (para objetos Java Xlet) (ABNT, 2008c).

Dentre os padrões de mídias disponíveis na estrutura do documento NCL, o NCL-Lua merece ser destacado, pois ele permite que sejam referenciados aplicativos procedurais desenvolvidos na linguagem LUA.

- **NCL** (*Nested Context Language*)

A NCL é uma linguagem de aplicação em XML para a criação de documentos hipermídia (SOARES; RODRIGUES, 2009), utilizada no SBTVD-T. Segundo Rodrigues e Soares (2006), a NCL tem como base o sistema japonês ISDB-T. Com uma linguagem declarativa como o NCL, é possível a criação de programas televisivos não lineares, possibilitando que ocorram interações do telespectador junto ao fluxo do programa NCL. Um documento hipermídia no padrão NCL é estruturado conforme apresenta a Figura 6

A estrutura de um documento no padrão NCL está dividida em duas partes (NETO et al., 2009):

- **O cabeçalho:** é obrigatório e é responsável por definir regiões (**region**, responsável por definir uma região com uma área onde será apresentado um nó de mídia), descritores (**descriptor**, responsável pela definição de como um nó de conteúdo irá se apresentar, incluindo a associação com uma região) e conectores com regras (**connectorBase**, são responsáveis por definir as regras entre os nós de origem e destino); e
- **O corpo:** também é obrigatório e é responsável por definir os nós de conteúdo (**media**, pode ser composto por um nó de mídia ou um contexto responsável por representar e organizar objetos de mídia), elos (**link**, realiza a associação entre nós através dos conectores), portas (**port**, responsável por especificar o nó principal de um documento ou o nó de entrada de um contexto), âncoras de conteúdo (**area**, definido dentro de um nó de mídia, utilizado como ponto de ativação dos elos) e âncora de propriedade (responsável por definir a propriedade de um nó de origem ou destino, que pode ser manipulado por elos, como, por exemplo, volume de áudio de um nó).

Ainda segundo Neto et al. (2009), os passos para a criação de aplicações em NCL são:

- 1 Os cabeçalhos básicos do arquivo NCL e do programa;
- 2 As regiões da tela onde aparecerão os elementos visuais (**regionBase**);
- 3 Como e onde os nós de mídia serão exibidos, através de descritores (**descriptorBase**);

cabeçalho do arquivo NCL	1: <?xml version="1.0" encoding="ISO-8859-1"?> 2: <ncl id="exemplo01" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ncl.org.br/NCL3.0/EDTVProfile http://www.ncl.org.br/NCL3.0/profiles/NCL30EDTV.xsd">
cabeçalho do programa	3: <head>
base de regiões	4: <regionBase> 5: <!-- regiões da tela onde as mídias são apresentadas --> 6: </regionBase>
base de descritores	7: <descriptorBase> 8: <!-- descritores que definem como as mídias são apresentadas --> 9: </descriptorBase>
base de conectores	10: <connectorBase> 11: <!-- conectores que definem como os elos são ativados e o que eles disparam --> 12: </connectorBase>
	13: </head>
corpo do programa	14: <body>
ponto de entrada no programa	15: <port id="plnicio" component="ncPrincipal" interface="ilnicio"/>
conteúdo do programa	16: <!-- contextos, nós de mídia e suas âncoras, elos e outros elementos -->
	17: </body>
término	18: </ncl>

Figura 6: Exemplo da linguagem NCL

Fonte: Adaptado de (NETO et al., 2009)

- O conteúdo (nós de mídia - **media**) e a estrutura (contexto - **context**) do documento (seção **body**), associando-os aos descritores;
- A porta de entrada do programa, apontando para o primeiro nó a ser exibido, bem como as portas para contextos, visando a construção dos elos entre os contextos e nós de mídia (**port**);
- Âncora para os nós de mídia, visando a construção dos elos entre nós (**area** e **attribute**);
- Elos para sincronismo e interatividade entre os nós de mídia e contextos (**link**); e

8 Os conectores que especificam o comportamento dos elos do documento (**connectorBase**).

Assim como nas linguagens BML, DVB-HTML e XDMB, que utilizam ECMAScript (RODRIGUES; SOARES, 2006), a NCL também necessita de uma linguagem de script para superar limitações. No caso da NCL, esta linguagem é LUA.

- **LUA - Linguagem de programação**

LUA é uma linguagem de script criada em 1993 pela Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio). LUA é largamente utilizada para o desenvolvimento de jogos (IERUSALIMSKY; FIGUEIREDO; CELES, 2007) e para tornar a interatividade em TV Digital, no padrão SBTVD-T, extremamente dinâmica e vantajosa. Os dois principais motivos para que a linguagem LUA seja utilizada para este contexto de infra-estrutura são sua velocidade de execução e seu tamanho (Satori, 2009).

Um exemplo da linguagem LUA pode ser observado no trecho de código apresentado na Figura 7, disponibilizado no manual do usuário da linguagem.

```
1 x = 10      -- global variable
2 do         -- new block
3   local x = x  -- new 'x', with value 10
4   print(x)    --> 10
5   x = x+1
6   do        -- another block
7     local x = x+1 -- another 'x'
8     print(x)    --> 12
9   end
10  print(x)    --> 11
11 end
12 print(x)    --> 10 (the global one)
```

Figura 7: Exemplo da linguagem LUA

- **Integração NCL/LUA**

Em virtude da falta de integração até o momento entre a linguagem NCL e Java (Ginga-J) e também pela facilidade proporcionada pela linguagem de *script* LUA, optou-se por utilizar esta última para implementar as aplicações de interatividade, bem como o envio das informações a um servidor de dados através do canal de

retorno dos *set-top box*. Porém, para realizar aplicações que proporcionem interatividade, é necessária uma forte integração entre NCL e LUA.

Esta integração é realizada através de uma biblioteca de componentes: um visual, chamado de “canvas”; um de registro de eventos, chamado de “event”; e outro de manipulação do documento NCL, chamado de “ncledit”. O resultado desta integração são aplicações na linguagem LUA, os quais serão executados pelo *set-top box* e interpretados em tempo de execução, fazendo com que o telespectador consiga interagir com uma aplicação.

O acesso a bibliotecas no NCL acontece através do nome da biblioteca. Um exemplo de acesso à biblioteca “canvas” é apresentado na Figura 8. Esta figura mostra uma função na linguagem LUA responsável por desenhar um retângulo na tela.

```
1 local DX, DY = canvas:attrSize() -- dimensoes da tela
2 function drawRect (self) -- funcao para desenho de retangulos
3   canvas:attrColor(self.color)
4   canvas:drawRect('fill', self.x, self.y, self.dx, self.dy)
5 end
6
```

Figura 8: Exemplo de código LUA

A documentação completa sobre estes componentes, juntamente com o restante da linguagem LUA, pode ser encontrada em (IERUSALIMSKY; FIGUEIREDO; CELES, 2006).

Outro recurso proveniente desta integração entre as duas linguagens é a possibilidade da utilização de um canal de retorno, que permite que sejam enviadas informações para um servidor de dados.

O canal de retorno é a forma utilizada para uma TV enviar informações, proporcionando interatividades de nível dois e três (CROCOMO, 2007). Este canal de retorno pode estar disponível através de um cabo telefônico ou internet banda larga, por exemplo. É através deste canal de retorno que é possível enviar informações para o servidor remoto de dados sobre as interações realizadas pelos telespectadores.

2.4 TRABALHOS CORRELATOS

Nesta seção são apresentados trabalhos de alguma forma relacionados ao conjunto de ferramentas proposto. São analisadas ferramentas

de criação de *script*, comumente ligadas a TVDi, por serem muito utilizadas por programadores, e ferramentas de autoria utilizadas para criar aplicações. Outro tema apresentado é o estudo sobre exemplos de programas interativos desenvolvidos por pesquisadores.

2.4.1 Ferramentas para criação de *scripts*

Ferramentas que auxiliam produtores de vídeo a desenvolver *scripts* de gravação são utilizadas para facilitar e reduzir o trabalho para a organização das gravações. Existem algumas ferramentas disponíveis como, por exemplo, a ferramenta CeltX, apresentada na Figura 9, que é encontrada gratuitamente em <http://celtx.com>. Esta ferramenta disponibiliza um editor de texto onde são apenas digitados os textos desejados e a ferramenta utiliza modelos disponibilizados na internet para formatar o documento no formato de *script*, que servirá para a gravação dos vídeos pela equipe de produção.

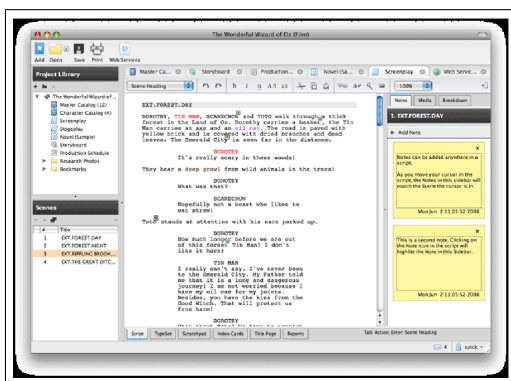


Figura 9: CeltX: Ferramenta para criação de *script*

Fonte: <http://celtx.com/screens.html>

Outra ferramenta disponível também gratuitamente é a “easyscriptbasic”. Esta ferramenta está na língua Alemã e sem a possibilidade de tradução. Um ponto fraco das duas ferramentas e também de outras disponíveis no mercado é a falta de elementos de interatividade, impossibilitando assim a geração de conteúdo para TVDi. Como ferramenta com mais recursos podemos citar a NewScript, produzida pela empresa STEP Software, que pode ser encontrada em <http://www.stepsoftware.com.br>. Apesar de esta ferramenta ser mais robusta, atraente e de fácil manipulação, ela também não possibilita a produção de conteúdo interativo para TVDi.

Quando, no padrão brasileiro, citamos a produção de interatividade, podemos encontrar dois *plugins* para o IDE Eclipse: NCLEclipse, utilizado para a criação de documentos NCL; e LuaEclipse, utilizado por desenvolvedores para a criação de aplicações na linguagem LUA e na interação entre LUA e NCL (AZEVEDO; TEIXEIRA; NETO, 2009).

Juntando estes dois *plugins* é possível criar interatividade, porém sem a possibilidade de integração com *scripts* de gravação de telejornalismo.

2.4.2 Ferramentas de Autoria

Uma ferramenta de autoria para TVD é um aplicativo multimídia que permite ao desenvolvedor reunir diferentes elementos de mídia em uma linha de tempo, com a opção de adicionar interatividade entre os elementos (VILJOEN; CALITZ; COWLEY, 2003).

Para a criação de aplicações interativas e programas televisivos no padrão brasileiro existe uma implementação desenvolvida em Java chamada Composer (GUIMARÃES, 2007), a qual é voltada para o desenvolvimento de aplicações NCL. Esta ferramenta, por sua vez, apresenta uma forma gráfica de controle de fluxo do programa de televisão, bem como as regiões visíveis de cada multimídia. A ferramenta possui quatro formas de visualizar o projeto que está em desenvolvimento:

- **Estrutural:** apresenta os componentes e o fluxo, como pode ser observado na Figura 10;
- **Textual:** onde é permitido alterar diretamente o conteúdo do arquivo XML na linguagem NCL;
- **Temporal:** onde é apresentado parte do fluxo e interatividade de forma temporal; e
- **Layout:** apresenta e possibilita a edição das mais diversas regiões das multimídias a serem apresentadas no programa televisivo.

O Composer é uma ferramenta disponibilizada pelo grupo Ginga como o editor de aplicativos interativos do *framework*. Contudo, esta ferramenta apresenta pontos negativos quanto à curva de aprendizado, pois é necessário, muitas vezes, compreender componentes específicos da linguagem NCL para descobrir erros notificados pela aplicação. Outro ponto a ser observado é o tempo de desenvolvimento necessário para a produção de aplicações interativas. E, por último, segundo o *site* dos desenvolvedores, a ferramenta se encontra descontinuada.

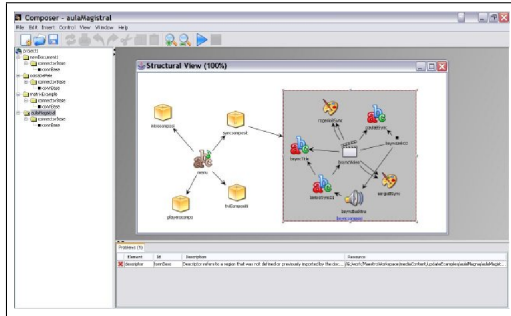


Figura 10: Visão estrutural da ferramenta Composer

Ainda seguindo o conceito de *software* livre como o Composer, pode ser encontrado o *framework* XiTV, que é desenvolvido em ECMAScript para uso em documento XHTML dentro dos padrões DVB e ACAP. Este sistema ainda está em desenvolvimento, porém apresenta uma aplicação exemplo (Figura 11).

O Cardinal Studio Professional 4 (Cardinal Home Page, 2009) é uma ferramenta de autoria desenvolvida pela empresa Cardinal. Esta ferramenta roda em ambiente *desktop* de forma gráfica e possibilita a criação de aplicativos interativos especificamente para o padrão MHP. A Cardinal trabalha com tecnologia aberta e configura seus projetos com arquivos na linguagem XML.

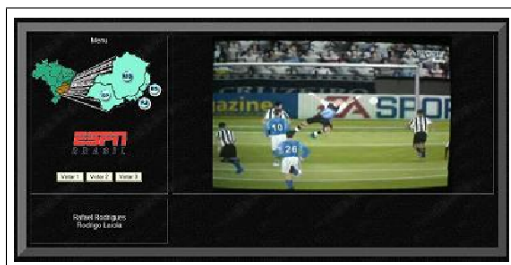


Figura 11: Exemplos de aplicação com interatividade em XHTML
Fonte: <http://sourceforge.net/projects/xitv/>

O AltiComposer (Alticast Home Page, 2009), desenvolvido pela empresa Alticast, também é uma ferramenta de autoria. Entretanto, ela é focada principalmente em desenvolvedores que não possuam co-

nhecimento em Java.

O iTVProject, apresentado em (OLIVEIRA; FILHO; FER, 2008), é um projeto que gera aplicações interativas no padrão GEM, sendo focado principalmente no desenvolvimento genérico de aplicações para TVDi. Aponta principalmente duas linguagens geradas pela ferramenta, MHP e JavaTV.

Estes três projetos podem ser considerados de certa forma obsoletos, já que seus desenvolvimentos estão aparentemente interrompidos. Contudo, existem outros projetos no mercado para a criação de aplicações interativas, como a ferramenta JAME Author ¹¹, que permite a criação de aplicações no padrão MHP. Uma tela desta ferramenta está apresentada na Figura 12.

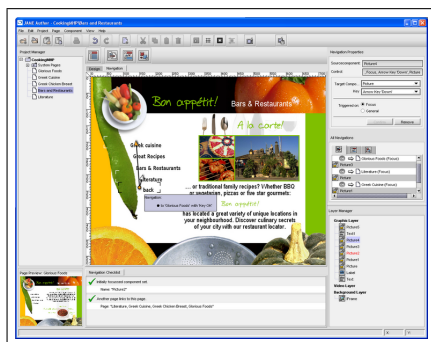


Figura 12: JAME Author: Ferramenta de Autoria

Fonte: <http://www.iais.fraunhofer.de/jame.html>

Até aqui foram apresentadas aplicações nos modelos brasileiro e europeu. Já no padrão americano, foram encontrados dois projetos:

- GRiNS (BULTERMAN1 et al., 1998): Ferramenta usada para criar e visualizar aplicações no padrão *Synchronized Multimedia Integration Language* (SMIL); e
- LimSee2 ¹²: Ferramenta de autoria disponibilizada de forma livre para a criação de programas interativos.

Para o desenvolvimento de aplicativos interativos na linguagem Java com o padrão JavaTV é necessário o uso também de um emulador para a execução das aplicações desenvolvidas. Para este fim, sugere-se

¹¹<http://jame.tv>

¹²<http://limsee2.gforge.inria.fr/>

o uso do emulador chamado de XleTView, que pode ser observado na Figura 13.

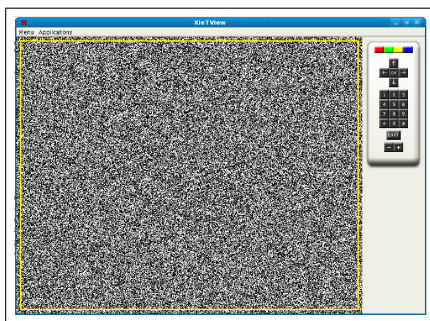


Figura 13: XleTView: Emulador para JavaTV

O XleTView é desenvolvido para emular aplicações interativas do tipo “Xlet” que tenham sido desenvolvidas de acordo com a definição da API JavaTV. Em decorrência da falta de uma implementação de referência para o padrão Ginga-J, O XleTView, muitas vezes, é utilizado para emular aplicações que teoricamente estão neste padrão. Com ele também é possível uma integração de ferramentas de desenvolvimento de código Java.

Este emulador permite que aplicativos do tipo “Xlet” sejam cadastrados e acessados através de um menu, criando-se uma biblioteca de aplicativos para execução. Possui ainda a capacidade de emular um fluxo contínuo de dados através de um arquivo de vídeo ou uma conexão a um stream.

2.4.3 Exemplos de aplicações em Televisão Digital Interativa (TVDi)

Alguns exemplos podem ser resgatados de grandes emissoras de televisão, como a BBC, que disponibiliza mesmo para pessoas que não tenham a TV digital, através da internet, uma prévia dos seus serviços disponíveis, como pode ser observado na Figura 14.

Já aplicações para um público mais seletivo, como os telespectadores de um jogo de futebol, muitas possibilidades podem ser citadas. Exemplos destas possibilidades são mencionadas em (BORGES et al., 2008), o qual demonstra aplicativos de informações sobre a partida corrente, a tabela do campeonato e a tabela de jogos da rodada. Também

são descritos aplicativos que possibilitam uma interatividade de nível 2, como perguntas e itens sobre os times de futebol que podem ser comprados pela TV. Este último pode ser observado na Figura 15.

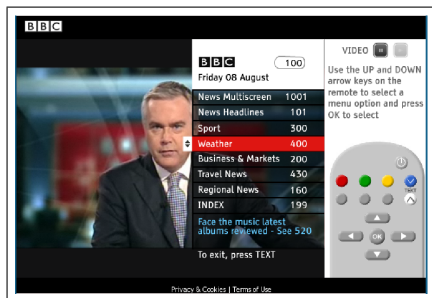


Figura 14: Demonstração na internet de iTV do canal BBC

Fonte: <http://www.bbc.co.uk/iplayer>



Figura 15: Exemplo de Comércio Eletrônico pela TV

Fonte: (BORGES et al., 2008)

2.5 TRABALHOS PRELIMINARES

Como trabalhos preliminares, foram realizados experimentos a fim de confirmar a tecnologia e direcionar o desenvolvimento das ferramentas. Os experimentos consistiram em testar recursos disponíveis, como apresentação de textos, desenho de imagens, interação com controle remoto e conexão com internet. Seguindo a classificação de interatividade em três níveis (CROCOMO, 2007), sugerem-se algumas aplicações básicas a serem disponibilizadas no ambiente a ser desenvolvido. Uma breve explicação sobre cada nível e as sugestões de aplicações interativas são apresentadas a seguir.

2.5.1 Nível 1

Com este nível de interatividade foram inicialmente estudados e desenvolvidos aplicativos que possibilitam representar informações visuais, como imagens e texto. Outro recurso testado com este tipo de aplicação foi a interação do controle remoto, possibilitando ao usuário manipular a visualização de um texto.

Aplicativos de nível 1 são aqueles onde não existe um canal de retorno e a aplicação precisa funcionar inteiramente no equipamento de interpretação Ginga-NCL (CROCOMO, 2007). Como sugestão para este nível, são propostos três tipos de interatividade:

- **Informação textual:** podendo ser utilizada em qualquer tipo de programa televisivo, pode ser amplamente explorada em programas de reportagens ou em telejornais. Tem o objetivo de apresentar informações em forma de texto, podendo ou não serem acompanhadas de uma imagem ou vídeo, como, por exemplo, na Figura 16, que apresenta uma informação textual sobre um equipamento utilizado e abordado no vídeo apresentado (Filme 16 mm);



Figura 16: Exemplos de aplicação interativa textual desenvolvida

- **Programação televisiva:** onde se permite que o usuário acesse a programação do canal a qualquer instante que desejar;

- **Aplicativos de entretenimento para telespectadores:** principalmente jogos que possam ser inseridos em programas. Este tipo de entretenimento, unido com a televisão digital, pode tornar-se uma ferramenta de grande poder aliado à educação infantil, por exemplo. Jogos de interatividade nível 1 podem ser citados como jogos de memória, caça-palavras, palavras cruzadas, quebra-cabeças, entre outros.

2.5.2 Nível 2

Este nível é caracterizado pela inserção do canal de retorno, possibilitando o envio de informações para as emissoras (CROCOMO, 2007).

Com aplicações neste nível é possível uma maior interação com um servidor de dados, que armazena as informações decorrentes da interatividade. Para enviar os dados para um servidor, foram testadas e implementadas conexões com a internet, acabando por adotar a conexão do tipo HTTP/GET. Foi possível com isto desenvolver questionários e jogos, permitindo uma avaliação melhor do canal de retorno e da interatividade do controle remoto.

Entre as aplicações para este nível destacam-se principalmente aplicativos onde os telespectadores possam dar sua opinião para uma provável mudança no comportamento do programa que está sendo apresentado. Exemplos são jogos de perguntas e respostas, enquetes, programas de auditório e votações em programa de competições estilo *reality show*.

Neste tipo de aplicação deve-se sempre observar a forma com que o telespectador dará sua opinião. Para isso podem ser disponibilizadas opções prontas, parecidas com uma prova de múltiplas alternativas, ou uma forma de escrita, utilizando, por exemplo, teclados virtuais de sistemas operacionais para pessoas com necessidades especiais. Um exemplo de aplicação, do tipo perguntas e respostas, pode ser observado em duas etapas, onde é apresentado a interatividade sendo visualizada, Figura 17.

2.5.3 Nível 3

Este nível exige que o telespectador tenha, além de um canal de retorno para a emissora, a infra-estrutura necessária para o recebimento de informações através de um canal paralelo de dados, normalmente uma conexão de internet do tipo banda larga (CROCOMO, 2007).

Neste nível foram testados exemplos de conexões permanentes que possibilitassem tanto o envio de informações quanto o recebimento.

Implementando-se um *chat*, foi possível realizar um experimento das dificuldades encontradas neste tipo de aplicação. Este nível não foi muito explorado devido a uma decisão baseada na escassez de dispositivos mais elaborados do que um controle remoto.

Neste nível é possível sugerir a disponibilização de aplicativos como salas de bate papo (*chat*), que normalmente são ofertadas por emissoras em páginas de internet, onde elas disponibilizam atores e pessoas ilustres para responder a perguntas elaboradas por telespectadores de forma paralela ao programa de televisão.

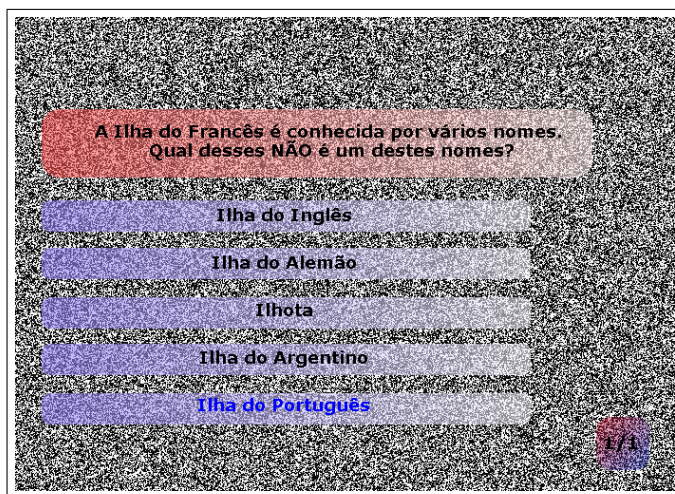


Figura 17: Exemplo de pergunta interativa

Outro serviço bastante utilizado por usuários de internet que pode ser migrado para aplicações interativas em TVD desenvolvidas neste nível é o recebimento de notícias através de *web services* no padrão RSS, que possibilita ao telespectador, a qualquer momento, receber as notícias nas quais ele mais tem interesse.

Exemplos adicionais são jogos em que mais de uma pessoa pode participar, fato que torna o jogo mais interessante e estimulante, como é o caso de damas ou xadrez. Com estes tipos de jogos, adultos e crianças podem ser estimulados mentalmente e socialmente com seus amigos ou conhecidos com o conforto de estar em suas residências.

2.6 CONCLUSÕES

Na primeira seção deste capítulo foi possível perceber que existem inúmeros *middlewares*, cada um com suas particularidades. Um resumo destas características pode ser observado na Tabela 2.

Observando esta tabela, percebe-se que o *middleware* adotado pelo Brasil se assemelha em muitos dos pontos com os demais *middlewares* pelo mundo. Este capítulo permitiu a comparação dos padrões de televisão digital existentes e com isso permitir que o ferramental a ser proposto seja utilizado no padrão SBTVD-T. Esta decisão foi necessária uma vez que dois padrões são utilizados no Brasil hoje, o SBTVD-T para transmissão terrestre e o DVB para transmissão via cabo e satélite.

Na segunda seção foi possível, através do levantamento bibliográfico sobre o padrão brasileiro e as duas versões de *middleware* disponíveis, realizar uma comparação e comprovar que, por limitações técnicas do Ginga-J, deverá ser utilizado o Ginga-NCL com integração da linguagem LUA. Esta integração entre o Ginga-NCL e LUA permite que o recurso de interatividade seja utilizado sem problemas para a produção de programas televisivos interativos.

Os exemplos de aplicações e ferramentas de autoria apresentados na última seção deste capítulo foram de fundamental importância para o entendimento, a elaboração e a implementação do conjunto de ferramentas proposto que é apresentado no próximo capítulo.

Quadro 2: Comparativo entre middlewares

Característica	DASE	ARIB	MHP	GINGA
Padrão	ATSC	ISDB	DVB	SBTVD-T
Segurança	Sim	Sim	Sim	Sim
Decodificação de conteúdo comum (PNG, JPEG, ZIP, etc)	Sim	Sim	Sim	Sim
Tipos de aplicativos	XHTML, CSS, EXMA Script, JavaTV	HTML, XML, JavaTV	HTML, JavaTV	HTML, XML, LUA script
Distinção entre aplicações declarativas procedurais	Sim	Sim	Sim	Sim
Interação com usuário	Sim	Sim	Sim	Sim
Capacidade de Áudio	(audio/basic), (Dolby AC-3)	MPEG-2 AAC	MPEG BC	MPEG-2 AAC
Capacidade de Vídeo	(Multiple Network Graphics), (MPEG-2)	MPEG 2	MPEG 2	MPEG 4
Capacidade Gráfica	1920X1080, 1280X720, 960X540, 640X480	HD: 1920X1080, 1280X720 e 960X540 SD: 620X480	320X240, 640X480, 720X480, 1920X1080	1920X1080, 1280X720, 960X540, 620X480

Fonte: Adaptado de (PAES, 2006)

3. PROPOSTA INTEGRADA DAS FERRAMENTAS

Para a descrição do ferramental proposto, é necessário apresentar a integração entre o conjunto de ferramentas. Esta integração e as definições necessárias para cada ferramenta funcionar de maneira apropriada são apresentadas a seguir.

3.1 DEFINIÇÃO DAS FERRAMENTAS

O conjunto proposto possui três ferramentas distintas e interligadas, que são ilustradas na Figura 18 e descritas a seguir.

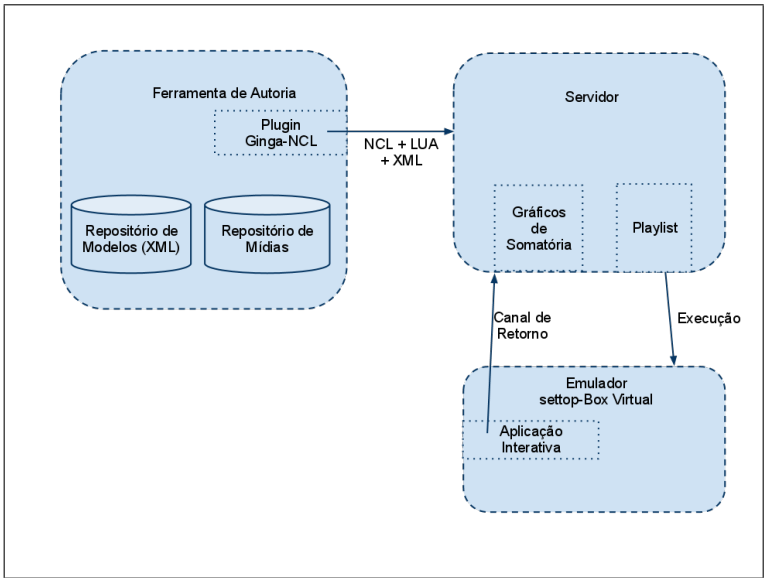


Figura 18: Arquitetura do ambiente proposto

No ambiente apresentado existem três elementos distintos, discutidos nas seções a seguir.

3.1.1 Ferramenta de Autoria - Célula

Ferramenta de autoria capaz de produzir conteúdo baseado em uma linha de tempo, adicionando *templates* de modelos unidos a mídias. Possibilita uma conexão com o servidor para envio de programas interativos no formato Ginga-NCL, compondo arquivos nos formatos NCL, XML e LUA.

Esta etapa do processo é fundamental e é o foco do projeto. Por este motivo, é necessária a elaboração de uma arquitetura melhor detalhada sobre a proposta de funcionamento da ferramenta. Esta, por sua vez, necessita ter as capacidades de: a) reconhecer um repositório de *templates* de aplicações interativas; b) possibilitar a união de documentos de mídia, como vídeos, imagens, sons, textos e estilos; e c) produzir aplicativos que possam ser executados no ambiente de simulação e em futuros ambientes reais que obedeçam ao padrão Ginga-NCL. A arquitetura proposta da ferramenta de autoria é apresentada na Figura 19.

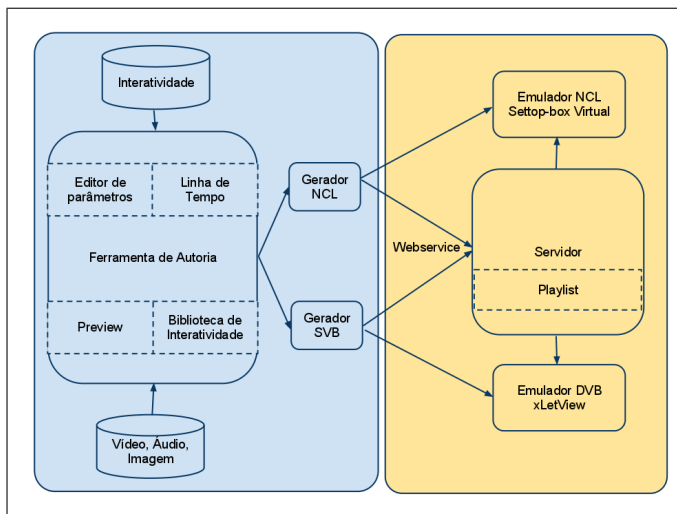


Figura 19: Arquitetura da ferramenta de autoria

3.1.1.1 Requisitos

- **Funcionais:**

- **RF¹³1:** Gerenciar projetos, podendo serem salvos e abertos localmente;
- **RF2:** Gerar um projeto no padrão Ginga-NCL;
- **RF3:** Gerenciar a lógica de apresentação baseada em linha de tempo;

¹³RF: Requisito Funcional.

- **RF4:** Permitir a utilização de mídias locais (vídeos, imagem, áudio);
- **RF5:** Permitir adicionar a um projeto *templates* de interatividade;
- **RF6:** Gerenciar dinamicamente as propriedades dos *templates*;
- **RF7:** Permitir a pré-visualização de um *frame* em um determinado instante de tempo;
- **RF8:** Listar *templates* instalados de forma classificada;
- **RF9:** Conexão direta com o servidor para envio de projetos gerados no padrão Ginga-NCL; e
- **RF10:** Permitir a inserção de recursos em uma biblioteca para seu reuso no mesmo projeto.

- **Não Funcionais:**

- **RNF¹⁴1:** Ser multiplataforma;
- **RNF2:** Utilizar XML para armazenar templates e projetos salvos; e
- **RNF3:** Utilizar recursos de vídeo em memória.

3.1.2 Servidor

Esta ferramenta é dividida em duas partes. A primeira é um recurso que realiza o papel de simular uma emissora de televisão, responsável por enviar o conteúdo de mídia e os programas interativos para os telespectadores; este recurso pode ser utilizado como uma *playlist* para um exibidor de uma transmissora. A segunda parte consiste em um recurso de armazenamento de dados que tem o objetivo de receber as informações enviadas pelas interações de níveis 2 e 3 e apresentar os acumulados de forma gráfica.

3.1.2.1 Requisitos

- **Funcionais:**

- **RF1:** Receber o projeto de forma integrada com a ferramenta de autoria, através de *web service*;
- **RF2:** Permitir o cadastro de projetos;

¹⁴RNF: Requisito Não Funcional.

- **RF3:** Permitir o envio do projeto compactado;
- **RF4:** Disponibilizar um *web service* para receber informações de aplicações interativas através do canal de retorno;
- **RF5:** Gerar gráficos acumulativos dos dados capturados pelo canal de retorno;
- **RF6:** Permitir a criação de *playlist* com os programas cadastrados; e
- **RF7:** Permitir o download do programa atual da *playlist*.

- **Não Funcionais:**

- **RNF1:** Executar sobre o servidor JBoss 4.2 ou superior;
- **RNF2:** Utilizar a linguagem Java com o *framework* JSF (RichFaces); e
- **RNF3:** Utilizar *servlets* para comunicação.

3.1.3 Emulador

O envio das aplicações para o emulador (TV) é feito através da internet, pois o telespectador, neste caso, conta com um *set-top box* emulado em um computador pessoal. Para emular o *set-top box* no padrão Ginga-NCL, é utilizada a implementação modelo do padrão Ginga, desenvolvida pelo laboratório de tele-mídia (SOARES, 2009) e disponibilizada no repositório de *software* público (Portal de Software Publico, 2009). Uma televisão acoplada a este emulador pode tornar o ambiente mais real.

3.1.3.1 Requisitos

- **Funcionais:**

- **RF1:** Conectar com o servidor para baixar e executar o programa atual da *playlist*;

- **Não Funcionais:**

- **RNF1:** Executar utilizando a implementação de referencia do Ginga-NCL;
- **RNF2:** Emular uma máquina virtual utilizando o VMPlayer; e
- **RNF3:** Utilizar uma linguagem “batch” para comunicação com servidor.

3.1.4 Integração

A integração das ferramentas consiste em utilizar um ambiente para produção e teste de conteúdo interativo integrado com o armazenamento de aplicações e a comunicação de aplicações interativas com a transmissora de televisão. Os requisitos das ferramentas são discutidos a seguir.

O processo de produção utilizando o conjunto de ferramentas proposto segue o fluxo:

- **Passo um:** desenvolvimento de aplicações interativas a partir de uma ferramenta capaz de reconhecer *templates* de aplicações em um repositório, unindo-os com documentos de mídia e transformando este conjunto de elementos em programas interativos para TVDi;
- **Passo dois:** na ferramenta de autoria os programas interativos são gerados;
- **Passo três:** envio dos programas para um repositório através de uma comunicação direta da ferramenta com um servidor, deixando estes programas disponíveis para a emissora de TV;
- **Passo quatro:** adicionam-se estes programas diretamente na fila de execução de um exibidor¹⁵;
- **Passo cinco:** o programa é assistido no emulador, que é capaz de executar as aplicações; e
- **Passo seis:** o indivíduo que criou o programa tem a possibilidade de visualizar os dados retornados pelas aplicações através de um *browser*.

Estes passos podem ser observados no diagrama de visão geral do sistema apresentado na Figura 20. Este diagrama apresenta o fluxo de funcionamento do processo de produção de programas interativos utilizando o conjunto de ferramentas proposto.

Para apresentar o fluxo da informação e a integração entre as ferramentas propostas é apresentado na Figura 20 um diagrama de visão geral. Este diagrama apresenta o fluxo de informações de alto nível no sistema.

¹⁵Exibidor: nome comumente utilizado para o equipamento responsável pela reprodução dos programas em transmissoras.

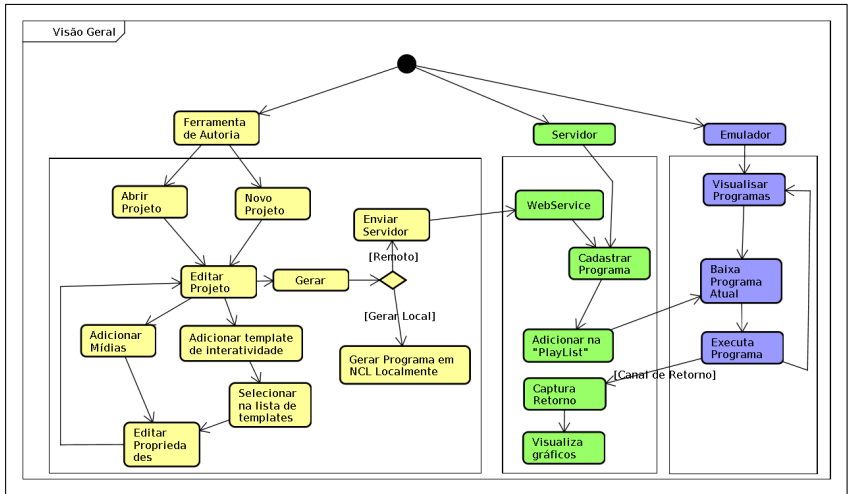


Figura 20: Diagrama de visão geral

No diagrama apresentado podemos destacar o item “Enviar para o Servidor”, que realiza o envio de informação da ferramenta de autoria para o servidor, e os elementos “Adicionar na Playlist” e “Executar Programa”, que realizam a integração do servidor com o emulador.

4. IMPLEMENTAÇÃO

O ambiente foi desenvolvido em três partes, em que a primeira é a implementação de uma ferramenta de autoria, a segunda é um servidor que possa receber, armazenar e organizar os programas interativos desenvolvidos e a terceira é um anexo ao servidor para possibilitar o retorno de informações de aplicações em TVDi.

4.1 FERRAMENTA

A ferramenta consiste em um ambiente baseado em uma linha temporal (*timeline*) criada a partir de uma mídia principal (fluxo principal). A partir desta linha de tempo o jornalista responsável pela edição do programa pode efetuar a inserção de elementos de mídia e aplicativos interativos. Os aplicativos interativos podem ser classificados, editados e complementados a partir de *templates* capturados em um repositório. Uma visualização da ferramenta de autoria pode ser observada na Figura 21

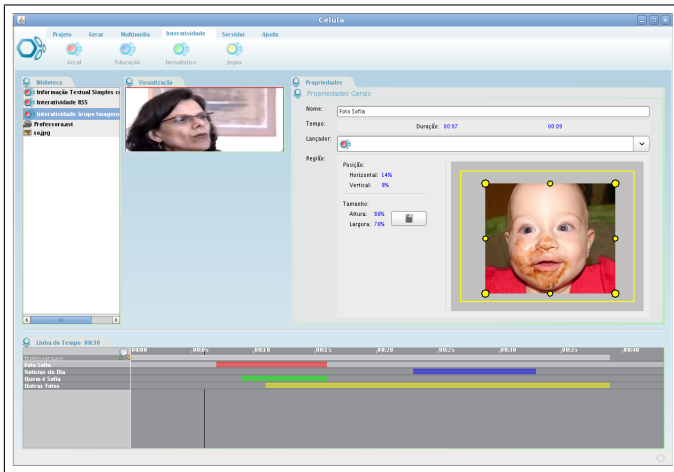


Figura 21: Tela principal da ferramenta de autoria

O *middleware* Ginga-NCL foi escolhido para servir como modelo de comprovação das ferramentas por conter sua definição completa e uma implementação de modelo que permite a implantação do ambiente de simulação. As ferramentas podem, sem muita dificuldade, serem adaptadas para outros padrões que permitam interatividade.

A etapa de geração do conteúdo no padrão Ginga-NCL é realizada por um componente modular na forma de *plugin*, o que permite que a geração de aplicações em novos padrões possa ser incorporada facilmente por um desenvolvedor. Dentre estes padrões de TVDi, existe o padrão brasileiro Ginga-J, que está em etapa de implementação. A seguir, serão apresentados maiores detalhes sobre os recursos da ferramenta de autoria.

4.1.1 Recursos da ferramenta de autoria

Os recursos disponíveis na ferramenta tentam facilitar a criação de programas interativos. São eles:

- **Biblioteca de recursos:** uma lista à qual podem ser adicionadas mídias (Figura 22). Dentre as mídias disponíveis que podem ser adicionadas nestas listas estão imagens, vídeos, áudios e aplicativos (*templates*);

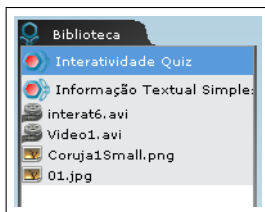


Figura 22: Biblioteca de recursos

- **Multimídias:** o menu multimídias (Figura 23) permite que vídeos, imagens e áudios sejam escolhidos e enviados para a biblioteca de recursos;

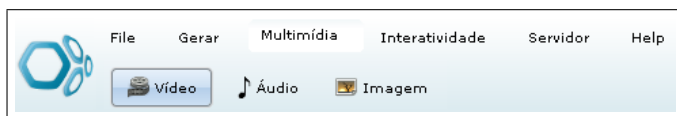


Figura 23: Menu multimídia

- **Classificação e seleção de *templates*:** outro recurso que pode ser adicionado à biblioteca de recursos. Classificados em “Geral”, “Educação”, “Jornalismo” e “Jogos”, os *templates* (Figura 24) são lidos de um diretório e disponibilizados de forma gráfica para que o usuário os utilize como aplicativos interativos;

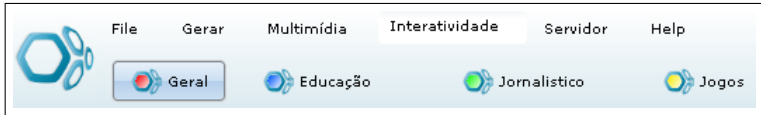


Figura 24: Classificação de templates

- **Timeline:** depois que os recursos são adicionados à biblioteca de recursos, eles são “arrastados” para a *timeline* (Figura 25). Esta, por sua vez, permite que os recursos nela adicionados possam ser selecionados e “arrastados” ao longo da linha do tempo, com a possibilidade de aumentar o tempo do recurso;

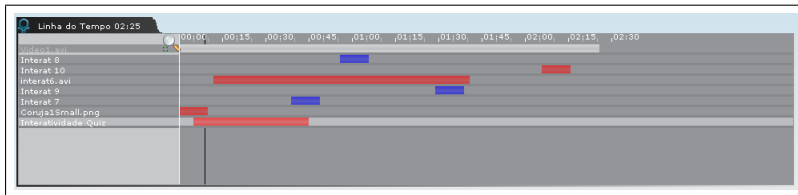


Figura 25: *Timeline*

- **Editor de propriedades:** sempre que um recurso é selecionado na *timeline*, as propriedades deste recurso são apresentadas no editor de propriedades, o que permite que o recurso seja personalizado. O painel do editor de propriedades é gerado dinamicamente conforme a mídia selecionada, podendo variar o tipo e a quantidade de campos a serem informados. Existem campos base que são disponibilizados para todas as mídias (Figura 26).
- **Gerar:** a opção de gerar no menu permite que o projeto criado pelo usuário seja gerado localmente na máquina. Esta opção no modo NCL gera o documento no padrão XML e anexa os arquivos necessários para esta interatividade funcionar em um emulador; e



Figura 26: Editor de propriedades

- **Conexão para o servidor:** a conexão com o servidor permite que o usuário informe um servidor de destino para onde deseja enviar o projeto gerado. A geração e o envio ocorre de forma automática, requerendo apenas que o usuário indique o nome do projeto que ele deseja enviar.

Outros recursos, como criar, salvar, carregar e pré-visualizar um projeto também estão disponíveis na ferramenta.

4.1.2 *Templates* disponíveis

A ferramenta de autoria permite que *templates* sejam personalizados, disponibilizando aplicações complexas desenvolvidas na linguagem LUA, que podem então ser anexadas a projetos de interatividade. O usuário ainda tem à sua disposição, através da ferramenta, a opção de escolher atributos para todos os *templates*, como posição, tamanho e tempo para que a interatividade apareça para o telespectador.

Seis *templates* foram disponibilizados até o momento para validar a ferramenta de autoria. O foco do desenvolvimento dos *templates* iniciais tem a intenção de disponibilizar os três tipos de mídias visuais conhecidas, que são vídeos alternativos, imagens ilustrativas e textos.

- 1 **Informação textual:** com o objetivo de apresentar texto, foram desenvolvidos dois *templates*. O primeiro apenas para apresentar textos pequenos e um segundo *template* para apresentar textos maiores, possibilitando “rolar” o texto com o auxílio das teclas para cima e para baixo do controle remoto. Para personalizar o visual deste *template* é disponibilizada a alteração de propriedades de imagens de *background* e da barra de rolagem;

- 2 **Informação textual com imagem:** pensando em agregar valor às informações textuais e facilitar a criação destas aplicações, foi desenvolvido um *template*, baseado no *template* de informação textual com a barra de rolagem, agregando uma imagem ilustrativa;
- 3 **Imagem ilustrativa:** opção criada apenas para apresentar imagens, que podem ser utilizadas para apresentar os mais diversos conteúdos, como, por exemplo, mapas, tabelas, ilustrações, etc.;
- 4 **Vídeos alternativos:** para inserção de vídeos alternativos durante a programação corrente de um canal;
- 5 **QUIZ:** desenvolvido para realizar perguntas em modo texto com até cinco alternativas, possibilitando a criação de quantas perguntas o desenvolvedor desejar e de forma somatória gerar regras para apresentação dos resultados, também em modo texto; e
- 6 **Jogo infantil:** para validar o potencial educacional, foi desenvolvido um jogo onde uma criança precisa responder a questões para percorrer um caminho em um tabuleiro, podendo escolher caminhos a serem percorridos.

4.1.2.1 Classificação dos *templates* Os *templates* de interatividade são classificados de acordo com áreas de utilização e conforme os níveis apresentados por Crocomo (2007). Com isso, permite-se que novos *templates* desenvolvidos sejam classificados conforme seus desenvolvedores desejarem. Esta classificação de *templates* permite que seja encontrada de forma fácil e simples a interatividade que melhor contextualiza com o programa em desenvolvimento. Esta classificação é realizada através do valor informado no parâmetro “tipo”, como apresentado na seção seguinte.

4.1.2.2 Estrutura de *templates* Os *templates* são estruturados através de documentos XML. Este documento estrutura variáveis que indicam à ferramenta como apresentar o *template* e disponibilizar as variáveis necessárias para a criação da aplicação interativa. Um exemplo de documento que estrutura um *template* é apresentado na Figura 27, onde é descrito o *template* implementado de informação textual simples.

Neste documento são apresentadas todas as *tags* que especificam o funcionamento de um *template*, as quais são explicadas a seguir.

- AplicativoTVDI, que indica que o documento XML é um *template* de aplicativo interativo. Um de seus atributos é o tipo utilizado para descrever para quais *frameworks* este *template* pode ser

gerado, tendo como opções atualmente disponíveis Ginga-NCL e MHP. Outro atributo desta *tag* é “preview”, que conterà o caminho de uma imagem que servirá para apresentar uma ilustração da interatividade pronta;

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<aplicativoTVDI preview="Imagens/previewInfoText.png" tipo="Ginga-NCL,MHP">
  <tipo>informacaoTextualSimples</tipo>
  <nome>Informação Textual Simples</nome>
  <inicio>5</inicio>
  <duracao>40</duracao>
  <regiao>0,0,100%,100%</regiao>
  <atributos>
    <item nome="texto" default="Texto Exemplo" tipo="textarea"></item>
    <item nome="imagem" default="Imagens/0.png" tipo="arquivo"></item>
    <item nome="fundo" default="Imagens/fundoInfoTxt.jpg" tipo="arquivo"></item>
  </atributos>
  <arquivos>
    <arquivo nome="lua/TextInfo.lua" tipo="Ginga-NCL"></arquivo>
    <arquivo nome="libs/ltn12.lua" tipo="Ginga-NCL"></arquivo>
    <arquivo nome="libs/io.o" tipo="Ginga-NCL"></arquivo>
    <arquivo nome="libs/io.h" tipo="Ginga-NCL"></arquivo>
    <arquivo nome="libs/vera.ttf" tipo="Ginga-NCL"></arquivo>

    <arquivo nome="mhp/SampleXlet.class" tipo="MHP"></arquivo>
  </arquivos>
</aplicativoTVDI>
```

Figura 27: Exemplo de documento XML descrevendo um *template* de interatividade

- Tipo, descreve o tipo da aplicação utilizada pela ferramenta para diferenciar um *template* do outro;
- Nome, descreve o nome a ser apresentado para o usuário da ferramenta;
- Início, valor padrão para o início da interatividade;
- Duração, valor padrão para a duração da abertura da interatividade;
- Região, valor padrão para a região de apresentação da interatividade;
- Atributos, grupo de *tags* que descreverão quais atributos poderão ser informados pelo usuário;
 - Item, descreve cada variável que pode ser descrita pelo usuário e contém os atributos:

- 1 Nome, nome da variável a ser utilizada no código fonte;
 - 2 *Default*, valor padrão a ser apresentado; e
 - 3 Tipo, tipo de campo preenchido, podendo assumir o valor de “textarea”, “arquivo” ou “texto”, que representa um campo de entrada de texto com varias linhas, seleção de arquivo ou campo de entrada de texto com apenas uma linha.
- Arquivos, grupo de *tags* que descreverão quais arquivos devem ser copiados para o correto funcionamento do *template*:
 - arquivo, descreve o arquivo a ser copiado, tendo como atributo “nome”, que é o local do arquivo a ser copiado e “tipo”, que indica se o arquivo deve ser copiado quando a ferramenta gerar para determinado *framework*, tendo as mesmas opções do atributo correspondente descrito na *tag* “aplicativoTVDI”.

4.2 SERVIDOR

O servidor foi elaborado em duas partes, uma para a armazenagem e outra para a organização de aplicações interativas.

4.2.1 Aplicações

O servidor de aplicação é desenvolvido em Java e disponibilizado em um servidor JBoss¹⁶. Ele permite que sejam cadastrados e organizados os programas interativos para simular um exibidor de uma transmissora de sinal de televisão. Este servidor possui a funcionalidade de cadastrar um programa televisivo e organizar a *playlist* de execução.

- **Cadastro**

O cadastro de programas pode ser feito de duas maneiras:

- 1 **Upload de arquivo compactado:** através de uma página *web*, conforme mostra a Figura 28, o usuário informa um nome para o programa, o arquivo compactado, a hora de início/fim e arquivos de execução do aplicativo de interatividade. Esta opção é utilizada principalmente para enviar programas que possam de alguma forma terem sido desenvolvidos sem a utilização da ferramenta de autoria; e

¹⁶www.jboss.org



Figura 28: Interface de cadastro de programas

2 **Web service:** a outra opção é utilizada diretamente pela ferramenta, possibilitando que, ao finalizar a aplicação, o usuário submeta o cadastro diretamente ao servidor. Para isto, é necessário apenas que a ferramenta esteja configurada para encontrar o *web service* e que o usuário informe um nome para o projeto que deseja cadastrar; a ferramenta captura o restante das informações a partir do projeto em desenvolvimento para enviar para o servidor.

- **Playlist**

O usuário tem a possibilidade de criar uma lista de programas a serem executados. O usuário também pode informar o horário em que o programa deverá executar. Este cadastro é realizado através de uma página *web*, como é apresentado na Figura 29. Além destes, outros recursos disponíveis nesta página são: criação da *playlist*, ordenação dos itens na lista, cálculo automático ou alteração manual do horário dos programas e visualização da programação através de um gráfico semelhante a um gráfico de Gantt.

Os tempos na *playlist*, quando calculados automaticamente, são baseados sempre no horário de início mais a duração do programa anterior, com exceção do primeiro programa listado, que é definido

sempre manualmente pelo usuário.

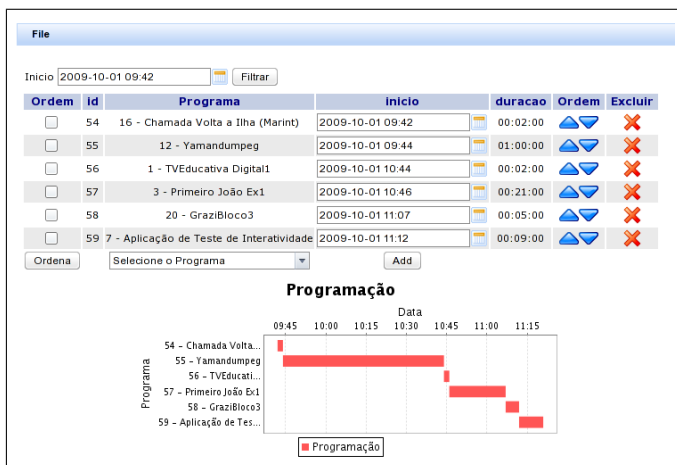


Figura 29: Lista de execução dos programas no ambiente simulado

4.2.2 Dados do canal de retorno

O serviço para canal de retorno serve para receber as informações enviadas pelas aplicações interativas quando executadas por um telespectador. Com um banco de dados simples, este servidor armazena cada registro recebido das aplicações. Permite que os dados armazenados sejam recuperados e quantificados. Para que este processo possa ser realizado, os *templates* disponíveis precisam ter esta capacidade. O serviço é desenvolvido em Java e disponibilizado como parte do servidor. O recebimento dos dados dá-se através de um *servlet*. Em seguida, apresenta-se como acontece o recebimento dos dados, o que os *templates* precisam implementar para terem a capacidade de enviar dados para este servidor e qual a forma gráfica utilizada para visualizar as somatórias dos resultados.

• Recebimento

O recebimento das informações se dá através de um *servlet* disponibilizado no padrão HTTP/GET¹⁷. As variáveis que são enviadas para este *servlet* por um aplicativo interativo são:

¹⁷Forma de execução de conteúdo *web* no protocolo HTTP, passando parâmetros de forma não segura.

- 1 **IP:** identificador do *set-top box* que está enviando as informações;
- 2 **Programa:** nome do programa que esta sendo apresentado;
- 3 **Aplicativo:** nome do aplicativo que está sendo executado (interatividade);
- 4 **Tipo:** descritivo para diferenciar e classificar o recebimento de informações; e
- 5 **Interação:** informações que são desejáveis de se armazenar.

Um exemplo de envio de informações de um aplicativo de *quiz* seria:

- 1 **IP:** “150.162.202.54”;
- 2 **Programa:** “Grazi Bloco 2”;
- 3 **Aplicativo:** “Quiz”;
- 4 **Tipo:** “Resposta da Pergunta 1”;
- 5 **Interação:** “3”;

Este envio é realizado pela aplicação interativa. Para realizar este envio é necessário que o *template* desenvolvido e disponibilizado na ferramenta de autoria seja implementado.

- **Implementação de envio de informações em *templates***

Cada *template* é responsável por disponibilizar ao usuário um parâmetro para realizar o envio de informações para o servidor de canal de retorno. Pensando em futuras implementações de *templates* por desenvolvedores de TVDi, foi disponibilizado uma função para fácil envio destas informações, que esta disponível juntamente com outras funções em uma biblioteca no diretório de *templates* da ferramenta de autoria. Um exemplo da utilização desta função, desenvolvida em LUA, seria: **send(“Grazi Bloco 2”, “QUIZ”, “ESCOLHA”, “CERTA”)**. Esta informação, uma vez enviada pela aplicação interativa, recebida e armazenada pelo servidor de canal de retorno, precisa ser apresentada. A forma escolhida neste trabalho para apresentá-la foi a somatória dos resultados através de gráficos de “pizza” e “linear”.

- **Gráficos**

Foram elaborados gráficos simples para a exibição das informações armazenadas. Um operador da emissora interessada pode escolher

para qual programa e para qual aplicativo deseja gerar os gráficos. Um gráfico linear é apresentado para representar a somatória das interações ao longo dos meses. Cada parâmetro enviado como “tipo” pela aplicação interativa é agrupado e é gerado um gráfico da somatória dos valores informados como interação. Ambos os gráficos são apresentados de uma só vez para o usuário. Um exemplo desta apresentação pode ser observado na Figura 30.

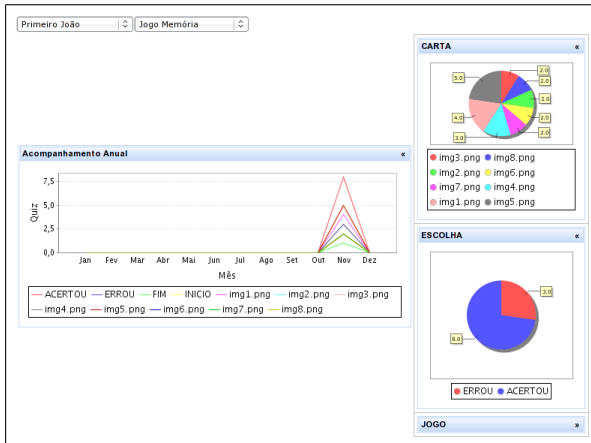


Figura 30: Gráficos de retorno das aplicações interativas (Nível 2 e 3)

- **Emulador:**

O emulador consiste em uma pequena alteração do *set-top box* virtual existente com o objetivo de emular aplicações em GINGA-NCL. Esta alteração é a inserção de um *script* do tipo *batch* que conecta ao servidor e verifica o programa atual em execução. Uma vez que o programa atual no servidor for diferente do programa em execução, este é baixado para o emulador e é executado automaticamente.

4.3 RESULTADOS

A avaliação do conjunto de ferramentas foi conduzida da seguinte forma: cada grupo de interessados desenvolveu um projeto, elaborado por eles mesmos, e ao longo do processo de uso do ambiente foi observado o comportamento de cada grupo, verificando-se as dificuldades enfrentadas. Ao final, solicitou-se que falassem um pouco sobre suas experiências com o uso da ferramenta.

Para desenvolver os projetos, foram escolhidas pessoas que, sem o uso do ambiente desenvolvido, não teriam o conhecimento para a criação de aplicativos interativos. Este modelo de avaliação é apresentado também por pesquisas recentes nas mais diversas áreas (CONTEJEAN et al., 2010; RUDOLPH; THIEMANN, 2010). Com o uso do ambiente, estas pessoas desenvolveram três projetos com pessoas distintas, sendo eles:

- Um trabalho de conclusão do curso de Jornalismo da UFSC, elaborado e defendido pelos acadêmicos Grazielle Pasqual Schneider e João Gustavo Munhoz, intitulado “Mundo Plano”;
- Um vídeo interativo desenvolvido em parceria com o professor de Jornalismo Fernando Antônio Crocomo, intitulado “Mãos”, com o apoio da FAPESC; e
- Uma dissertação do curso de Mestrado em Design Gráfico da UFSC, elaborado e defendido pela acadêmica Dafne F. Arbex.

Já para auxiliar o grupo de pessoas a falar sobre a sua experiência no uso das ferramentas, foram feitas algumas perguntas:

- Por que você precisou da ferramenta?
- Quais foram suas principais dificuldades em usar a ferramenta de autoria?
- A similaridade com outras ferramentas de produção de vídeo ajuda na localização dos recursos?
- Quais recursos você sentiu falta na ferramenta e sugere como uma possível implementação?
- Você utilizaria esta ferramenta de autoria para desenvolver conteúdo interativo no seus trabalhos não acadêmicos? Por quê?

Eles foram orientados a colocar críticas e sugestões de forma livre, sem seguir este roteiro. O grupo de pessoas está dividido em três equipes, uma em cada projeto, sendo que uma equipe é composta por dois indivíduos.

Os projetos implementados foram realizados pelos próprios usuários, com acompanhamento para que eventuais dúvidas fossem esclarecidas. Os detalhes são apresentados a seguir.

4.3.1 Mundo Plano

Este trabalho foi desenvolvido na forma de um documentário sobre a economia da cidade de Blumenau. A interatividade foi inserida no contexto ao longo dos três blocos apresentados. Foram utilizados três *templates* de interatividade ao longo do programa desenvolvido pelos estudantes de jornalismo:

- 1 Informação textual com barra de rolagem, vista na Figura 31, apresentando textos complementares sobre informações culturais e econômicas da região;

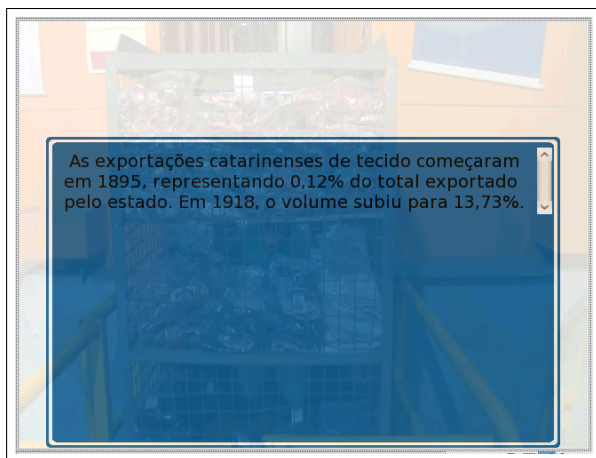


Figura 31: Informação textual

- 2 Vídeo alternativo ao fluxo principal (Figura 32), com dois trechos apresentando imagens sobre enchentes recentes acontecidas na cidade; e



Figura 32: Vídeo alternativo

3 Quiz (Figura 33), perguntando sobre os hábitos que esta pessoa pretende adotar em sua viagem.

Neste projeto os alunos conseguiram utilizar a ferramenta, não necessitando maiores explicações. O projeto desenvolvido foi enviado pela ferramenta para o servidor de aplicações e visualizado em um emulador conectado a este servidor.

Este grupo é composto pelos dois acadêmicos de Jornalismo e por isso foi solicitado que ambos escrevessem seus comentários de forma individual, para tentar apresentar a visão de cada um sobre a sua experiência. Os comentários a seguir foram obtidos.

- **Por que você precisou da ferramenta?**

João Munhoz: “Precisei da ferramenta para fazer meu Trabalho de Conclusão de Curso, o projeto editorial de Mundo Plano, um programa de economia das cidades para a TVDigital.”;

Grazielle Pasqual Schneider: “Meu Trabalho de Conclusão de Curso, feito em conjunto com o João Gustavo Munhoz, foi o projeto editorial para um programa de televisão com interatividade e alta definição, voltado para a TV Digital. Como parte da proposta, elaboramos um programa piloto. A ferramenta foi necessária para simular a interatividade nesse programa piloto, apresentado para a banca avaliadora em julho de 2009.”

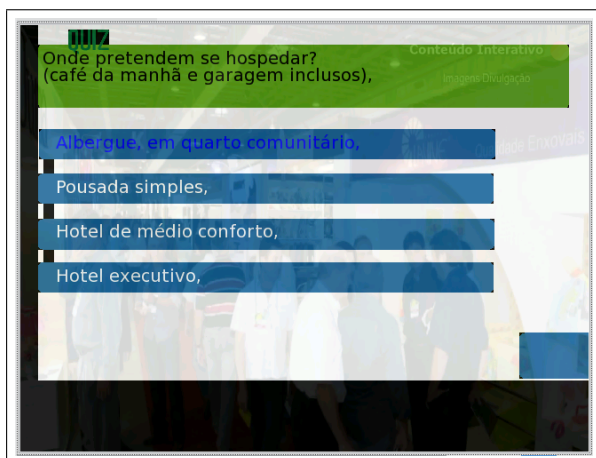


Figura 33: Quiz

- **Quais foram as suas principais dificuldades em usar a ferramenta da autoria?**

João Munhoz: “A principal dificuldade foi não ter um *preview* daquilo que foi programado.”;

Grazielle Pasqual Schneider: “Na época, a conversão dos vídeos para um formato que a ferramenta aceitasse foi o que deu mais trabalho, pois eles haviam sido gravados em alta definição e estavam muito pesados. Fora isso, tivemos dificuldades apenas com a sua operação, pois a interface ainda estava em processo de desenvolvimento.”

- **A similaridade com outras ferramentas de produção de vídeo ajuda na localização dos recursos?**

João Munhoz: “Ajuda, principalmente na visualização e organização da *timeline*.”

Grazielle Pasqual Schneider: “Ajuda, sim. É necessário praticar para aprender, mas torna-se intuitivo.”

- **Quais recursos você sentiu falta na ferramenta e sugere como uma possível implementação?**

João Munhoz: “Conversor de formatos de vídeos e *preview*.”

Grazielle Pasqual Schneider: “Infelizmente não consigo lembrar agora. De bate-pronto, acho que seria necessário implementar

diferentes formatos de vídeo na ferramenta, para que não fosse necessária a conversão. Se não me engano, também não era possível visualizar a interação posta no vídeo em tempo real, no momento da ‘edição’ do material. Para rodar uma prévia, era preciso jogar no simulador e marcar um horário de exibição, o que dificultava o trabalho e ajustes.”

- **Você utilizaria esta ferramenta de autoria para desenvolver conteúdo interativo no seus trabalhos não acadêmicos? Por quê?**

João Munhoz: “Sim, pois a ferramenta cumpriu as minhas expectativas em relação a um software de gerenciamento de conteúdo interativo.”

Grazielle Pasqual Schneider: “Sim, porque não há muitas opções, e esta ferramenta, apesar de passível de melhoras, não é tão difícil de aprender.”

As questões aplicadas neste primeiro projeto foram importantes para o *preview*, que foi implementado na versão atual da ferramenta de autoria para visualização das interatividades inseridas no projeto. Esta visualização ainda é falha e requer melhorias. Quanto à conversão de vídeo apontada como limitação neste projeto, foram tomadas duas ações; a primeira foi possibilitar a abertura de vídeos no padrão H264 e a segunda foi permitir a escolha pelo usuário, no momento de gerar a aplicação, sobre o tamanho de vídeo que deverá ser gerado o projeto.

4.3.2 Mãos

O projeto “Mãos” é um programa televisivo pensado para ser interativo. Trata-se de um vídeo que fala sobre a utilização das mãos de pessoas em suas profissões. Apesar da utilização de recursos simples de interatividade — apenas imagens — este projeto apresenta grande qualidade ao demonstrar a preparação de um projeto televisivo pensado na integração em toda a cadeia produtiva. O programa consiste em um vídeo apresentando as mãos de um profissional, e a interatividade permite que o telespectador tenha a oportunidade de conhecer o profissional que está apresentando. O projeto mãos teve como resultado um vídeo interativo que pode ser observado na Figura 34.

- **Por que você precisou da ferramenta?**

Fernando Antônio Crocomo: “Em 2005 participei do Sistema Brasileiro de TV Digital com a produção de vídeos e a inserção

de aplicativos de interatividade. Um dos grandes desafios era a programação dos aplicativos com especialistas em computação. O Célula passou a ser uma ferramenta fundamental para facilidade e autonomia na aplicação de perguntas e outros recursos interativos em vídeos.”



Figura 34: Projeto “Mãos”

- **Quais foram as suas principais dificuldades em usar a ferramenta da autoria?**
Fernando Antônio Crocomo: “Não foram muitas, mas a princípio o entendimento da interface e a utilização dos planos de fundo.”
- **A similaridade com outras ferramentas de produção de vídeo ajuda na localização dos recursos?**
Fernando Antônio Crocomo: “É fundamental. É uma continuidade da lógica de edição não-linear de vídeos.”
- **Quais recursos você sentiu falta na ferramenta e sugere como uma possível implementação?**
Fernando Antônio Crocomo: “Mais opções de planos de fundo e fontes. É uma questão de consolidação da ferramenta com *updates*.”

- **Você utilizaria esta ferramenta de autoria para desenvolver conteúdo interativo no seus trabalhos não acadêmicos? Por quê?**

Fernando Antônio Crocomo: “Utilizaria sem dúvida alguma, porque a ferramenta tem potencial para uso profissional.”

Este projeto foi realizado com a intenção de testar a finalidade da ferramenta, uma vez que o profissional que realizou este projeto colaborou com as definições dos requisitos da ferramenta de autoria. Dos comentários deste projeto, foi possível vislumbrar o potencial econômico da ferramenta, o que nos levou a registrar o *software* junto ao INPI. Quanto ao recursos de fontes, apontado como limitação, foi aumentada de uma para dez a quantidade de fontes disponíveis para uso.

4.3.3 Design Gráfico

Esta avaliação é um parte do trabalho de mestrado da acadêmica Dafne F Arbex. Voltado para ensino a distância, onde é apresentada uma teleaula, a interatividade é utilizada para complementar o conteúdo com vídeos alternativos, *links* para outras aulas e questionários sobre a aula. O resultado do projeto desenvolvido pode ser observado na Figura 35.

- **Por que você precisou da ferramenta?**

Dafne F. Arbex: “Precisei da ferramenta para desenvolver um menu contendo informações extras sobre o vídeo principal.”

- **Quais foram as suas principais dificuldades em usar a ferramenta da autoria?**

Dafne F. Arbex: “A aplicação do *design*, como escolha de família tipográfica, paleta de cores e ferramentas de desenho poderiam estar contempladas no aplicativo para enriquecer os futuros trabalhos.”

- **A similaridade com outras ferramentas de produção de vídeo ajuda na localização dos recursos?**

Dafne F. Arbex: “Sim. Um exemplo é o Final CUT, porém o que falta é inserir interatividade para trabalhar com EAD também.”

- **Quais recursos você sentiu falta na ferramenta e sugere como uma possível implementação?**

Dafne F. Arbex: “Recursos de design.”

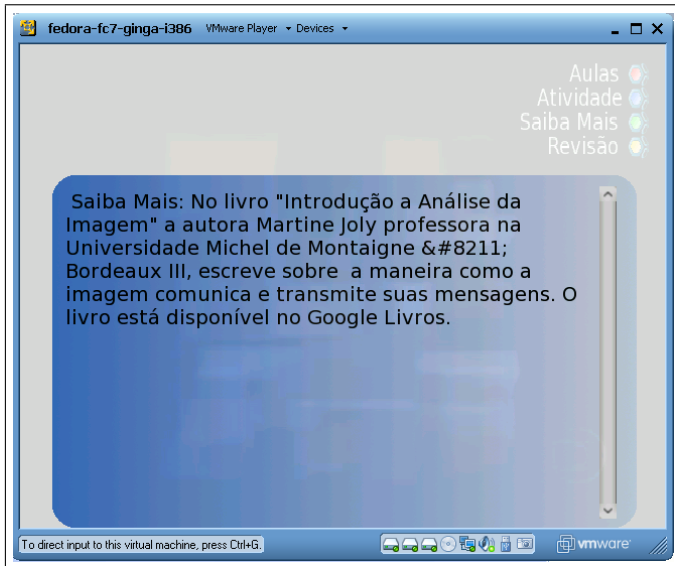


Figura 35: Saiba mais: parte do projeto de dissertação do curso de Design Gráfico

- **Você utilizaria esta ferramenta de autoria para desenvolver conteúdo interativo no seus trabalhos não acadêmicos? Por quê?**

Dafne F. Arbex: “Sim. Porque é uma boa ferramenta para estudantes e profissionais que não querem ou não precisam entender de programação. Porém, ainda existem limitações que precisam ser trabalhadas.”

O projeto apresentado foi o único fora da área na qual a ferramenta foi proposta (jornalismo). Os comentários apresentam outra realidade e as limitações apresentadas necessitam de pesquisa e auxílio de um profissional da área de *design*. Com isso as sugestões apresentadas nos comentários deste projeto são deixadas como sugestões de trabalhos futuros.

5. CONCLUSÃO E TRABALHOS FUTUROS

Quanto aos objetivos levantados para este trabalho, todos foram cumpridos. Identificaram-se e documentaram-se através desta dissertação conceitos relevantes sobre televisão digital interativa, voltando-se principalmente para o padrão Ginga, utilizado nas transmissões terrestres no Brasil. No segundo objetivo específico foi elaborado um conjunto de ferramentas para o desenvolvimento de aplicações interativas. Estas ferramentas foram voltadas principalmente para jornalistas, tendo como principal ponto a construção de uma ferramenta de autoria para TVDi. Esta ferramenta de autoria foi integrada a *templates* desenvolvidos, que servem como base para a produção de interatividade. Já para os jornalistas poderem armazenar e testar projetos, foi elaborado e testado um servidor que permitiu a fácil execução das aplicações interativas desenvolvidas na ferramenta de autoria. Além disso, para comprovar que o conjunto de ferramentas desenvolvido é útil e de fácil utilização, foram realizados testes com o desenvolvimento de três projetos, um por alunos de jornalismo, um por um professor e profissional de jornalismo e um por uma estudante de *design*. O cumprimento destes objetivos permitiu que o objetivo geral deste trabalho, de propor e testar um conjunto de ferramentas para a criação e gerenciamento de aplicações com interatividade em vídeos para TVDi, com foco na produção por jornalistas, fosse atingido.

A principal contribuição deste projeto é a disponibilização, para jornalistas, de uma ferramenta que não requer um profissional da área de computação para o desenvolvimento de programas interativos. Contudo, outras contribuições podem ser relatadas neste projeto, como: a interdisciplinaridade entre cursos para especificação, elaboração e teste do projeto; esclarecimentos dos conceitos de TV Digital; arquitetura para desenvolvimento e teste de aplicações interativas; a continuidade na produção de *templates*, justamente pela forma que a ferramenta de autoria se integra com eles; e trabalhos, de certa forma frutos deste, bem como a possibilidade de novos projetos.

5.1 TRABALHOS FUTUROS

Como primeiro trabalho futuro podemos citar um projeto para aumento da abrangência na cadeia de produção de programas televisivos, que consiste na criação de um exibidor capaz de acessar os aplicativos gerados pela ferramenta da autoria, possibilitando o gerenciamento e exibição destes programas em um parque de transmissão de televisão terrestre.

Outro projeto apresentado foi a produção de novos modelos de interatividade para a ferramenta de autoria, possibilitando a inserção de conteúdo interativo para as mais diversas áreas, iniciando-se por EAD, pelo seu potencial, mas passando também por entretenimento.

Mais um projeto que pode ser citado é a melhoria da usabilidade e recursos disponíveis na ferramenta de autoria para facilitar e abranger seu uso para diferentes áreas, aumentando assim o público alvo de uma ferramenta como esta. Estes três projetos citados já estão em andamento na forma de continuação desta dissertação, e outros projetos estão em planejamento ou em elaboração; dentre eles podemos citar:

- Aumento da cadeia de produção: podemos falar sobre a criação de roteiros de televisão dentro da própria ferramenta, possibilitando que produtores, editores e roteiristas trabalhem de forma integrada sem a necessidade de retrabalho em cada etapa da produção de programas televisivos interativos;
- Outra funcionalidade muito importante que deveria ser disponibilizada na ferramenta de autoria é a visualização, em tempo de desenvolvimento, da interatividade que esta sendo produzida;
- Para auxiliar na criação de novos modelos, podemos imaginar a elaboração de tutoriais e documentações técnicas que permitam o aprendizado de pessoas leigas acerca do processo necessário para produção de modelos interativos integrados com a ferramenta de autoria;
- Voltando-se para o tema de convergência digital, podemos citar a criação de novos geradores para a ferramenta, possibilitando expandir os programas produzidos para fora da TV Digital, de forma simples, atingindo áreas como *Internet Protocol Television* (IPTV) e EAD na internet através da tecnologia Flash e em dispositivos móveis, tanto no formato de TV Digital Móvel quando em aplicações nativas para dispositivos móveis.

REFERÊNCIAS

- Alticast Home Page. *Alticast Home Page*. 2009. Disponível em: <"http://www.alticast.com/main.html">. Acesso em: 18 de Jan. 2009.
- ARIB STANDARD. *ARIB STD-B23*: Application Execution Engine Platform for Digital Broadcasting. [S.l.], 2004.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS E TÉCNICAS. *NBR 15601*: Televisão digital terrestre - Sistema de transmissão. [S.l.], abr. 2008.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR 15606-1*: Televisão digital terrestre - Codificação de dados e especificações de transmissão para radiodifusão digital Parte 1: Codificação de dados. Rio de Janeiro, abr. 2008.
- ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. *NBR 15606-2*: Televisão digital terrestre: Codificação e especificação de transmissão para radiodifusão digital: Parte 2: Gíngua-NCL para receptores fixos e móveis: Linguagem de aplicação XML para codificação de aplicações. Rio de Janeiro, abr. 2008.
- ATSC. *ATSC, Advanced Television Systems Committee (ATSC) website*. 2009. Disponível em: <"http://www.atsc.org">. Acesso em: 18 de Jan. 2009.
- ATSC. *ATSC SALUTES THE 'PASSING' OF NTSCr*. 2009. Disponível em: <"http://www.atsc.org/communications/press/2009-06-12-NTSCnderline_End_Final.php">. Acesso em: 18 de Jan. 2009.
- ATSC A/53 Part 1. *A/53: ATSC Digital Television Standard, Parts 1 - 6, 2007*. 2009. Disponível em: <"http://www.atsc.org/standards/a\53\Part\1\6\2007.pdf">. Acesso em: 13 de Ago. 2009.
- AZEVEDO, R. G. de A.; TEIXEIRA, M. M.; NETO, C. de S. S. *NCL Eclipse: Ambiente Integrado para o Desenvolvimento de Aplicações para TV Digital Interativa em Nested Context Language*. SRBC - XXVII Simposio Brasileiro de Redes de Computador e Sistemas Distribuidos, 2009.
- BELAÑO, C.; VIEIRA, V. R. *TV digital no Brasil e no mundo: estado da arte*. *Revista de Economía Política de Las Tecnologías de La Información Y Comunicación*, v. 33, n. 2, p. 102-134, maio 2004.

BENOIT, H. *Digital Television: Satellite, Cable, Terrestrial, IPTV, Mobile TV in the DVB Framework 3rd ed.* [S.l.]: Elseviere, 2008.

BORGES, K. S. et al. **Integração de diferentes áreas tecnológicas para o desenvolvimento de aplicações interativas na TV Digital.** *TVDI2008 - VI Fórum de Oportunidades em TV Digital Interativa*, Poços de Caldas, Minas Gerais, Mai 2008.

BULTERMAN1, D. et al. **GRiNS: A GRaphical INTERface for creating and playing SMIL documents.** *WWW7 Conference, Computer Networks and ISDN Systems*, v. 30, p. 519–529, 1998.

CAMPAGNOLO, J. et al. **SBTVD.** 2004. Disponível em: <"http://www.ntdi.ufsc.br/sbtvd_especial.html">. Acesso em: 21 de Set. 2009.

Cardinal Home Page. *Cardinal Systems Home Page.* 2009. Disponível em: <"http://www.cardinalsystems.com">. Acesso em: 18 de Jan. 2009.

CATTELAN, R. G. et al. **Watch-and-comment as a paradigm toward ubiquitous interactive video editing.** *ACM Trans. Multimedia Comput. Commun. Appl.*, ACM, New York, NY, USA, v. 4, n. 4, p. 1–24, 2008. ISSN 1551-6857.

CHESTERFIELD, J.; RODRIGUEZ, P. **DeltaCast: efficient file reconciliation in wireless broadcast systems.** ACM, New York, NY, USA, p. 93–106, 2005.

CONTEJEAN, E. et al. **A3PAT, an approach for certified automated termination proofs.** ACM, New York, NY, USA, p. 63–72, 2010.

CROCOMO, F. et al. *Sistema Brasileiro de TV Digital Interativa - RFP 6 - Serviço de Saúde.* 2009. Disponível em: <"http://www.ntdi.ufsc.br/sbtvd_especial.html">. Acesso em: 21 de Set. 2009.

CROCOMO, F. A. *TV Digital e produção interativa: a comunidade manda notícias.* 1ª ediçãO. ed. [S.l.]: UFSC, 2007.

DVB Project Office. *Introduction to MHP and GEM.* 2009. Disponível em: <"http://www.mhp.org/introduction.htm">. Acesso em: 18 de Jan. 2009.

ECMA International. **ECMAScript Language Specification**. 2009. Disponível em: <"http://www.ecma-international.org/publications/standards/Ecma-262.htm">. Acesso em: 18 de Jan. 2009.

FERNANDES, J.; LEMON, G.; ELIAS, G. **Introdução à Televisão Digital Interativa: Arquitetura, Protocolos, Padrões e Práticas**. *Jornada de Atualização em Informática do Congresso da Sociedade Brasileira de Computação, JAI-SBC*, 2004.

FILHO, G. L. S.; LEITE, L. E. C.; BATISTA, C. E. C. F. **Ginga-J: The Procedural Middleware for the Brazilian Digital TV System**. *Journal of the Brazilian Computer Society*, v. 13, p. 47–56, 2007.

Fórum SBTVD. [*Agência Estado*] - **Brasil exporta sistema de TV digital**. 2009. Disponível em: <"http://www.forumsbtvd.org.br/materias.asp?id=198">. Acesso em: 21 de Set. 2009.

GAWLINSKI, M. **Interactive Television Production**. [S.l.]: Focal Press, 2003.

GELEIJNSE, G.; ALIAKSEYEU, D.; SARROUKH, E. **Comparing text entry methods for interactive television applications**. ACM, New York, NY, USA, p. 145–148, 2009.

ginga.org.br. **Site Ginga.org.br**. 2009. Disponível em: <"http://www.ginga.org.br">. Acesso em: 18 de Jan. 2009.

GOMES, F. de J. L.; LIMA, J. V. de; NEVADO, R. A. de. **Uma interface multimodal para objetos de aprendizagem visualizados na TV digital**. Sociedade Brasileira de Computação, Porto Alegre, Brazil, Brazil, p. 284–287, 2008.

GUIMARÃES, R. L. **Composer: um ambiente de autoria de documentos NCL para TV digital interativa**. *Dissertação de Mestrado, Programa de Pós-Graduação em Informática, Pontifícia Universidade Católica do Rio de Janeiro*, 2007.

IERUSALIMSCHY, R.; FIGUEIREDO, L. H. de; CELES, W. **Lua 5.1 Reference Manual**. [S.l.]: Lua.org, 2006.

IERUSALIMSCHY, R.; FIGUEIREDO, L. H. de; CELES, W. **The Evolution of LUA**. *ACM SIGPLAN History of Programming Languages Conference*, p. 2–1 – 2–26, 2007.

LEE, J. **In-band mobile digital TV transmission technology for advanced television systems committee**. IEEE Computer Society Press, Los Alamitos, CA, USA, p. 382–382, 2008.

LUGMAYR, A.; NIIRANEN, S.; KALLI, S. *Digital interactive TV and metadata: Future broadcast multimedia*. [S.l.]: Springer, 2004.

MATSUMURA, K. et al. **Location-aware data broadcasting: an application for digital mobile broadcasting in Japan**. ACM, New York, NY, USA, p. 271–274, 2003.

MELONI, L. G. P. **Return Channel for the Brazilian Digital Television System-Terrestrial**. [*JBCS*] - *Journal of the Brazilian Computer Society*, v. 13, n. 1, Mar 2007.

MIRANDA, L. C. de; PICCOLO, L. S. G.; BARANAUSKAS, M. C. C. **Artefatos físicos de interaç ao com a TVDI: desafios e diretrizes para o cenário brasileiro**. Sociedade Brasileira de Computaç ao, Porto Alegre, Brazil, Brazil, p. 60–69, 2008.

MONTEZ, C.; BECKER, V. *TV Digital Interativa: conceitos, desafios e perspectivas para o Brasil 2ª ed.* [S.l.]: UFSC, 2005.

MORENO, M. F. *Um middleware declarativo para sistemas de TV digital interativa*. Dissertação (Mestrado) — PUC-Rio, 2006.

NETO, C. d. S. S. et al. *Construindo Programas Audiovisuais Interativos Utilizando a NCL3.0 e a Ferramenta Composer*. 2009. Disponível em: <"http://www.ncl.org.br/documentos-/TutorialNCL3.0-2ed.pdf">. Acesso em: 18 de Jan. 2009.

OBRIST, M. et al. **Connecting TV & PC: an in-situ field evaluation of an unified electronic program guide concept**. ACM, New York, NY, USA, p. 91–100, 2009.

OLIVEIRA, M. R.; FILHO, C. B.; FER, A. F. **iTV Project: An Authoring Tool for MHP and Ginga-J based on a Web Environment**. *ACM International Conference Proceeding Series*, v. 291, p. 179–182, 2008.

PAES, A. *Padrões de Middleware para TV Digital*. Dissertação (Mestrado) — Universidade Federal Fluminense, 2006.

PELLAN, B.; CONCOLATO, C. **Scalable multimedia documents for digital radio**. ACM, New York, NY, USA, p. 221–222, 2008.

Portal de Software Publico. **Repositório de arquivos, grupo Gíngã no portal do software publico**. 2009. Disponível em: <"http://www.softwarepublico.gov.br">. Acesso em: 18 de Jan. 2009.

RBSTV. **Portal RBS TV**. 2009. Disponível em: <"http://www.clicrbs.com.br/especial/rs/rbstvdigital/conteudo/detalhe/Padroes-SBTVD.html">. Acesso em: 18 de Jan. 2009.

REISMAN, R. R. **Rethinking Interactive TV – I want my Coactive TV**. 2009. Disponível em: <"http://www.teleshuttle.com/cotv/CoTVIntroWtPaper.htm">. Acesso em: 18 de Jan. 2009.

RODRIGUES, R. F.; MORENO, M. F.; SOARES, L. F. G. **Controle de Apresentação de Aplicações Declarativas em Receptores de Sistemas de TV Digital Interativa**. *XII Simpósio Brasileiro de Sistemas Multimídia e Web - WebMedia2006 - WebMedia2006*, v. 12, p. 184–192, 2006.

RODRIGUES, R. F.; SOARES, L. F. G. **Produção de Conteúdo Declarativo para TV Digital**. *XXXIII Seminário Integrado de Software e Hardware*, v. 33, p. 286–300, 2006.

RUDOLPH, J.; THIEMANN, P. **Mnemonics: type-safe bytecode generation at run time**. ACM, New York, NY, USA, p. 15–24, 2010.

SALEEMI, M. M.; BJÖRKQVIST, J.; LILIUS, J. System architecture and interactivity model for mobile tv applications. ACM, New York, NY, USA, p. 407–414, 2008.

Satori. **The Engine Survey: General results**. 2009. Disponível em: <"http://www.satori.org/2009/03/the-engine-survey-general-results-/">. Acesso em: 18 de Jan. 2009.

SCHIBELSKY, L. et al. **Understanding iDTV in a developing country and designing a T-gov application prototype**. ACM, New York, NY, USA, p. 379–385, 2008.

SILVEIRA, S. C. da. **Jornalismo e interatividade na Web 2.0: a produção de conteúdo pelo público em redes digitais**. *CELACOM - XI Colóquio Internacional sobre escola Latino Americana de Comunicação*, 2007.

SOARES, L. F. G. **Portal Laboratório TeleMídia**. 2009. Disponível em: <"http://www.telemidia.puc-rio.br/pt/index.html">. Acesso em: 18 de Jan. 2009.

SOARES, L. F. G.; RODRIGUES, R. F. *Nested Context Language 3.0 Part 8 - NCL Digital TV Profiles*. [S.l.]: Monorafica, PUC-RIO, 2009.

SVEDEN, M. *XletView Emulator*. 2009. Disponível em: <"http://www.xletview.org">. Acesso em: 18 de Jan. 2009.

TEIXEIRA, L. *Televisão digital*. [S.l.]: UCG, 2009.

VIANA, N. S.; LUCENA JR., V. F. de. **A software model supporting the management of home network services through the brazilian iDTV**. ACM, New York, NY, USA, p. 101–110, 2009.

VILJOEN, D. W.; CALITZ, A. P.; COWLEY, N. L. O. **A 2-D MPEG-4 multimedia authoring tool**. ACM, New York, NY, USA, p. 151–160, 2003.

W3C. *Portal W3C*. 2009. Disponível em: <"http://www.w3c.org">. Acesso em: 21 de Set. 2009.

YE, Y.; LI, Y.; SU, K. Architecture and software implementation of hdtv video decoder on a singlechip, mpeg decoder. p. 226 –230, 26-28 2006.

YIM, H.-J.; CHOY, Y.-C.; LIM, S.-B. **A study of GUI representation based on BIFs for enhanced mobile TV**. ACM, New York, NY, USA, p. 297–302, 2009.

ZALETELJ, J.; SAVIC, M.; MEZA, M. **Real-time viewer feedback in the iTV production**. ACM, New York, NY, USA, p. 149–152, 2009.