

UNIVERSIDADE DO EXTREMO SUL CATARINENSE - UNESC

CURSO DE CIÊNCIA DA COMPUTAÇÃO

GÉLIO JOSÉ DA SILVA JÚNIOR

**PROTÓTIPO DE VISUALIZAÇÃO DE IMAGENS DE CÂMERAS DE
VIGILÂNCIA POR MEIO DO SUBSISTEMA GINGA-NCL DO
SISTEMA BRASILEIRO DE TV DIGITAL**

CRICIÚMA, JULHO DE 2011

GÉLIO JOSÉ DA SILVA JÚNIOR

**PROTÓTIPO DE VISUALIZAÇÃO DE IMAGENS DE CÂMERAS DE
VIGILÂNCIA POR MEIO DO SUBSISTEMA GINGA-NCL DO
SISTEMA BRASILEIRO DE TV DIGITAL**

Trabalho de Conclusão de Curso apresentado
para obtenção do Grau de Bacharel em Ciência
da Computação da Universidade do Extremo
Sul Catarinense.

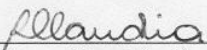
Orientador: Prof. MSc. Paulo João Martins

CRICIÚMA, JULHO DE 2011

GÉLIO JOSÉ DA SILVA JÚNIOR

**Protótipo de Visualização de Imagens de Câmeras de Vigilância por meio
do Subsistema Ginga-NCL do Sistema Brasileiro de TV Digital**

Submetido ao corpo docente do Curso de Ciência da Computação da
Universidade do Extremo Sul Catarinense como um dos requisitos para obtenção do grau
de Bacharel em Ciência da Computação.

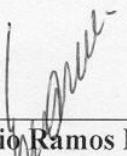


Profa. MSc. Ana Claudia Garcia Barbosa
Coordenadora do Curso de Ciência da Computação


Banca Examinadora:



Prof. MSc. Paulo João Martins (UNESC)
Orientador



Prof. MEng. Evânio Ramos Nicoleit (UNESC)



Prof. Esp. Fabrício Giordani (UNESC)

Dedico aos meus pais e amigos, que estão
sempre presentes em minha vida.

AGRADECIMENTOS

Primeiramente agradeço aos meus pais e minha irmã pelo acompanhamento de toda fase, tanto da elaboração deste trabalho como de toda a faculdade, na qual foram exigidas inúmeras ajudas e compreensões às dificuldades encontradas.

Ao Paulo pelas ajudas prestadas, tanto como orientador, professor e amigo.

A todos os professores do curso de Ciência da Computação pelas experiências compartilhadas durante estes anos, bem como ao departamento pelo apoio e estrutura.

Aos colegas de classe, mesmo que muitos tenham ficado pelo caminho, tornaram-se amigos, principalmente por compartilharmos um gosto em comum, a paixão por tecnologia.

Ao quarteto inseparável, Gélío, Guilherme, Lucas e Thaffareu, que desde a primeira fase da faculdade andam juntos pelos corredores da universidade, compartilhando conhecimentos, gerando sabedoria. Criando novas formas de ganhar dinheiro e filosofando sobre os mais diversos e inusitados assuntos. Ao Diego, a Priscila e ao Anderson, que nas últimas fases aumentaram o quarteto, tornando-se todos eternos amigos.

Aos companheiros de viagem de Praia Grande – Criciúma pelas mais de quatro voltas na Terra. Muitas teorias discutidas, problemas enfrentados e amizades construídas.

Aos amigos de Praia Grande e região pela força moral, que sempre me deixam com autoestima elevada, podendo assim enfrentar os desafios da vida.

A minha namorada Sara por entender minha vida corrida e estar ao meu lado, dando força para continuar.

Aos meus professores da Escola Bulcão Viana, que hoje são meus colegas de trabalho, mas que mesmo assim, me ensinam a cada dia, pela troca de experiências, como o mundo e a vida são realmente.

Nunca desista de algo que você pensa todo dia!

RESUMO

A segurança é um requisito primordial na sociedade atual. A vigilância de locais é altamente utilizada para que se diminua a ocorrência de atos ilícitos em diversos locais, tanto públicos quanto privados. No setor público, há diversos locais que são vigiados por órgãos competentes, porém essas imagens não são disponibilizadas para a população. Essas imagens poderiam ser divulgadas ao público, de forma que os civis teriam a possibilidade de visualizar imagens em determinado local, para obter informações de tempo ou movimento, por exemplo. O objetivo do trabalho é disponibilizar essas imagens a população, com inclusão em âmbito nacional. Para essa realização é necessário a utilização de um meio bem difundido, onde foi utilizado o Sistema Brasileiro de TV Digital (SBTVD). O estudo apresenta diversos conceitos sobre câmeras de vigilância e seus tipos de sistemas. Apresenta também conceitos e características do SBTVD, bem como inovações desse sistema em relação a outros existentes no mundo. Para o desenvolvimento da aplicação foram utilizadas as ferramentas Eclipse e alguns plugins, a máquina virtual do Gingga e um emulador de máquinas virtuais. Durante a pesquisa foi constatado que o padrão brasileiro de TV digital ainda está em fase de desenvolvimento e que muitas normas ainda não estão implementadas, atrapalhando a conclusão de alguns objetivos. O software desenvolvido disponibiliza as imagens de câmeras de vigilância aos telespectadores, que podem escolher entre várias opções de câmeras. A aplicação desenvolvida na linguagem NCL segue os padrões do *middleware* do padrão nipo-brasileiro, tem uma interface amigável e simples, com efeitos de transparência. Para a comunidade científica o trabalho identifica algumas limitações do Gingga e analisa o Exibidor NCL GHTV, documentando-as e indicando algumas possibilidades de trabalhos futuros na área de TV digital interativa.

Palavras-chave: TV Digital, Videovigilância, Interatividade, Gingga; SBTVD.

ABSTRACT

Safety is a primary requirement in society nowadays. The sites monitoring is highly used for the reduction of illegal acts which occur in several locations, both public and private. In the public sector, there are several sites that are supervised by competent bodies, but these images are not available to the public. These images could be showed to the public, so that civilians would be able to view images in a certain place, for getting information of time or movement, for example. The objective of this study is to provide these images to people, with inclusion in the national level. For this realization it is necessary to use an easily available way, which it is used the Brazilian Digital Television System (SBTVD). The study presents various concepts about surveillance cameras and their types of systems. It also presents concepts and SBTVD features and innovations of this system over others in the world. For the application development were used Eclipse tools and some plugins, the Ginga's virtual machine and a virtual machine emulator. During the research it was found that the Brazilian digital TV standard is still under development and many rules are not implemented yet, hindering the completion of some goals. The developed software provides the images of surveillance cameras to viewers, who can choose from several options of cameras. The developed application in NCL follows the standards of the standard middleware nipo-Brazilian, and has a friendly and simple interface with transparency effects. For the scientific community, the work identifies some limitations of Ginga and analyzes the GHTV NCL Exhibitor, documenting them and pointing out some possibilities for future work in the area of interactive digital TV.

Keywords: Digital Television, Surveillance, Interactivity, Ginga; SBTVD.

LISTA DE ILUSTRAÇÕES

Figura 1. Sistema Analógico	23
Figura 2. Sistema Digital	24
Figura 3. Tecnologias usadas em cada camada da arquitetura de TV Digital	31
Figura 4. <i>Middleware</i>	33
Figura 5. Arquitetura <i>Middleware</i> Ginga	35
Figura 6. Sistema Remoto	55
Figura 7. Terminal	55
Figura 8. Controle Remoto	57
Figura 9. Diagrama de casos de uso	58
Figura 10. Primeira versão de Interface	59
Figura 11. Versão 1.1 de Interface	60
Figura 12. Segunda versão de Interface	61
Figura 13. Código regras	62
Figura 14. Tela inicial	63
Figura 15. Código Ícone Interatividade	63
Figura 16. Versão final de Interface	64
Figura 17. Código controle de ações	65
Figura 18. Código regiões	66
Figura 19. Código Ocultar	66
Figura 20. Câmera de vigilância sendo exibida	67
Figura 21. Código botão azul	67
Figura 22. Sistema Vigilância TV Digital	68
Figura 23. Teste em ambiente Windows	70

Figura 24. Teste em ambiente Linux	71
Figura 25. Declaração de mídia remota.....	71
Figura 26. Máquina virtual acessando Motion	73
Figura 27. Gíngua não consegue carregar vídeo	74

LISTA DE ABREVIATURAS E SIGLAS

CFTV	Circuito Fechado de Televisão
BML	<i>Broadcast Markup Language</i>
FTP	<i>File Transfer Protocol</i>
HDTV	<i>High Definition Television</i>
HTTP	<i>HiperText Transfer Protocol</i>
IDE	<i>Integrated Development Environment</i>
LAVID	Laboratório de Aplicações de Vídeo Digital
MG	Minas Gerais
NCL	<i>Nested Context Language</i>
NCM	<i>Nested Context Model</i>
PUC-RIO	Pontifícia Universidade do Rio de Janeiro
PVR	<i>Personal Video Recorder</i>
RSE	<i>Remote System Explorer</i>
RTP	<i>Real Time Protocol</i>
SBTVD	Sistema Brasileiro de TV Digital
SMS	<i>Short Message Service</i>
SO	Sistema Operacional
SP	São Paulo
SSH	<i>Secure Shell</i>
STB	<i>Set-Top Box</i>
TI	Tecnologia da Informação
TV	Televisão
UFPB	Universidade Federal da Paraíba
UNESC	Universidade do Extremo Sul Catarinense
VCR	<i>Video Cassete Recorder</i>
VHS	<i>Video Home System</i>

SUMÁRIO

1 INTRODUÇÃO	14
1.1 OBJETIVO GERAL	16
1.2 OBJETIVOS ESPECÍFICOS.....	16
1.3 JUSTIFICATIVA	16
1.4 ESTRUTURA DO TRABALHO.....	19
2 SEGURANÇA	20
2.1 SEGURANÇA VIGIADA	21
2.2 CIRCUITO FECHADO DE TELEVISÃO	22
2.2.1 Sistemas Analógicos.....	22
2.2.2 Sistemas Digitais	23
2.3 PROJETOS PÚBLICOS DE VIGILÂNCIA	24
3 TV DIGITAL	27
3.1 SISTEMA BRASILEIRO DE TV DIGITAL.....	29
3.2 ARQUITETURA SBTVD	30
3.3 <i>MIDDLEWARE</i> GINGA.....	32
3.3.1 Ginga-CC:	34
3.3.2 Ginga-NCL.....	35
3.3.3 Ginga-J	35
3.4 INTERATIVIDADE.....	36
3.4.1 Interatividade Local	38
3.4.2 Interatividade Intermitente	38
3.4.3 Interatividade Permanente	39
3.5 RECEPTORES DIGITAIS	39

4 LINGUAGEM ESPECÍFICA SBTVD	41
4.1 APLICAÇÃO	41
4.2 NCL	42
4.3 NESTED CONTEXT MODEL (NCM)	43
4.3.1 Regiões	43
4.3.2 Descritores	44
4.3.3 Portas	45
4.3.4 Âncoras	45
4.3.5 Contextos	46
4.3.6 Nó de mídia	46
4.3.7 Elos	46
4.3.8 Conectores	47
5 TRABALHOS CORRELATOS	48
5.1 SISTEMA BRASILEIRO DE TV DIGITAL – SBTVD UMA ANÁLISE POLÍTICA E TECNOLÓGICA NA INCLUSÃO DIGITAL	48
5.2 TV DIGITAL PARA DISPOSITIVOS PORTÁTEIS - <i>MIDDLEWARES</i>	49
5.3 CANAL DE RETORNO EM TV DIGITAL	50
5.4 ESTUDO SOBRE A TELEVISÃO DIGITAL E DESENVOLVIMENTO DE UM PROTÓTIPO INTERATIVO BASEADO NO SUBSISTEMA PROCEDURAL DO MIDDLEWARE GINGA	51
6 SISTEMA VIGILÂNCIA TV DIGITAL	52
6.1 METODOLOGIA	52
6.2 FERRAMENTAS DE DESENVOLVIMENTO	53
6.2.1 Eclipse	54

6.2.2 Vmware	56
6.3 CONTROLE REMOTO	56
6.4 DESENVOLVIMENTO DA APLICAÇÃO.....	57
6.4.1 Primeira Versão	58
6.4.2 Segunda Versão	60
6.4.3 Versão Final	62
6.5 RESULTADOS	68
6.6 DIFICULDADES ENCONTRADAS.....	72
CONCLUSÃO.....	76
REFERÊNCIAS	78
APÊNDICE A: LOG ERRO EXECUÇÃO DO CANAL DE INTERATIVIDADE PELO GINGA.....	81
ANEXO A: COMO ESTRUTURAR SEU AMBIENTE DE DESENVOLVIMENTO PARA O GINGA-NCL.....	86

1 INTRODUÇÃO

A criminalidade e suas manifestações estão presentes em todas as regiões do Brasil. Devido à sua extensão territorial e pela diversidade cultural não há uma solução uniforme que atenda em todas as áreas o problema de segurança (SOARES, 2006). Porém, uma prática aplicada atualmente são as câmeras de vigilância, que estão espalhadas por vários locais devido ao baixo custo de instalação e manutenção.

A videovigilância é entendida como sinônimo de segurança, principalmente quando são denominadas “Câmeras de Segurança”. Essa associação de conceitos é constituída sócio-historicamente fazendo um recuo à modernidade descrita por Foucault (2000), onde dizia que “ser vigiado” significava “estar seguro”. A visão atual para a alta demanda desse tipo de sistema dá-se ao fato que a vigilância não objetiva garantir a segurança da população vigiada, mas garantir que os mesmos se adéquem as normas e valores, podendo ser punidos caso isso não ocorra (CORREA; CUNHA, 2009).

No Brasil, os projetos de instalação desses sistemas têm se multiplicado, tanto em ambientes públicos quanto em ambientes particulares. Um exemplo bem conhecido são as câmeras do Parque da Luz em São Paulo, que é vigiado dia e noite pela polícia (KANASHIRO, 2008).

As imagens geradas podem ser monitoradas em tempo real e/ou armazenadas para visualização posterior. A maioria dos sistemas é feito em Circuito Fechado de TV (CFTV) onde o acesso é feito somente local, porém há a possibilidade de disponibilizar essas imagens remotamente, onde as imagens poderiam ser vistas de qualquer local com acesso a *Internet*. Isso já está altamente difundido nos sistemas privados de vigilância, mas não no ambiente público.

A cada dia que passa surge novas tecnologias que podem ser exploradas na área de acesso remoto. Uma delas é o Sistema Brasileiro de Televisão Digital (SBTVD) que foi

instituído pelo governo federal brasileiro por meio do Decreto Presidencial 4.901 de 26 de novembro de 2003, com o objetivo principal de implantar uma plataforma tecnológica digital baseada em televisão que promova a inclusão social por meio do acesso à informação.

A estratégia do SBTVD foi adotada devido ao fato de 90% das residências brasileiras terem aparelho de televisão, sendo assim – após o rádio - a tecnologia mais difundida no país. Assim, ele tem como meta de ser porta de entrada para uma significativa parte da população que não tem acesso ao mundo da informação (LAMAS et al, 2005).

O SBTVD adota a arquitetura tradicional de radiodifusão para transmitir a informação, onde de um lado têm-se as emissoras transmitindo, e do outro os usuários recebendo.

Com essa nova tecnologia, tem-se a capacidade de interagir com a TV digital por meio do padrão de *middleware* estabelecido, o Ginga (NBR 15606, 2008). Para haver uma interatividade entre ambos, há a necessidade de um meio de comunicação fazendo o caminho inverso do sinal (SOARES, 2008). Uma funcionalidade marcante é a capacidade de utilizar as teclas play/pause na programação corrente. Dependendo do nível de interação, é possível ter as funções básicas de um computador conectado a Internet (LAMAS et al, 2005). Por meio desse nível pode-se criar uma conexão entre um sistema de segurança e uma TV digital.

Com sua alta difusão e com um cenário de interação permanente, este trabalho propõe ajudar na segurança, seja empresarial ou domiciliar, pública ou privada, obtendo acesso a imagens de câmeras de vigilância utilizando o canal de interatividade, propiciando que seja acessado a qualquer hora (24/7) de qualquer ambiente que tenha os recursos necessários.

1.1 OBJETIVO GERAL

Desenvolver um protótipo para exibição de imagens de câmeras de vigilância por meio do subsistema Ginga-NCL do Sistema Brasileiro de TV Digital (SBTVD).

1.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos dessa pesquisa consistem em:

- a) estudar e utilizar os conceitos de TV Digital;
- b) descrever e utilizar os padrões do SBTVD;
- c) verificar e descrever os recursos disponíveis no *Middleware* Ginga;
- d) utilizar os recursos no desenvolvimento de aplicações por meio do canal de interatividade;
- e) desenvolver um aplicativo para exibição de imagens de vídeo baseado no padrão do SBTVD

1.3 JUSTIFICATIVA

A necessidade de segurança é uma das características marcantes da sociedade do século XXI. A busca por soluções concretas para o problema da criminalidade é um dos maiores desafios, não só para os governantes, como também para toda a sociedade. As taxas de criminalidade nunca foram tão altas, tanto nas grandes quanto nas pequenas cidades, deixando a população preocupada com o aumento e convívio neste panorama (CARVALHO, E. A. 2008).

Precisa-se aumentar a sensação de segurança da população e uma das armas utilizadas atualmente é a vigilância. Com o alto custo na contratação de mais pessoas, os sistemas com câmeras estão aumentando a cada dia o seu espaço na realidade dos brasileiros.

Com a mentalidade que estar sendo vigiado é estar seguro, a sensação de segurança da população aumenta quando está sendo monitorado por câmeras de vigilância. Com elas ligadas, criminosos (ladrões ou até mesmo próprios funcionários) ficam apreensivos em fazer algum ato ilícito, diminuindo efetivamente a ocorrência desses atos no local vigiado.

As câmeras podem estar em locais físicos totalmente distintos podendo ser monitoradas por apenas uma pessoa, que pode acionar pessoas ou alarmes a distância, combatendo assim a criminalidade que está ocorrendo em tempo real.

Se o combate à criminalidade por meio da videovigilância realmente for eficaz, seu uso aumentará cada vez mais. Para realmente ser eficiente existe a necessidade de poder monitorar essas câmeras dos lugares mais variados possíveis, para isso, o uso de uma tecnologia altamente difundida se torna necessário.

O uso de TV digital aumenta em todo o mundo, em cada região com seus padrões, trazendo muitas vantagens tanto para os telespectadores quanto para as emissoras. Para os telespectadores têm-se uma melhor qualidade de imagem e a possibilidade de interatividade com a programação. Para as emissoras que podem transmitir o sinal de sua programação em formato digital, que mesmo já o produzindo, tem de transmitir em sinal analógico para seu público devido ao meio de transmissão atual. Outra grande vantagem é a possibilidade de novo tipo de propaganda que pode ser gerada, como, por exemplo, compras em tempo real no *T-commerce*¹.

¹ T-Commerce: Possibilidade de compras pela televisão

Com uma tecnologia altamente difundida estando em uma etapa de transição, vê-se a possibilidade de explorar esse ramo, obtendo vantagens em todas as áreas possíveis e necessárias para a sociedade.

A possibilidade de utilizar o canal de interatividade do Sistema Brasileiro de Televisão Digital aumenta o horizonte para aplicação da monitoração de câmeras de videovigilância. Pode ser utilizado devido ao fato de ter uma interação permanente com o usuário final, disponibilizando uma comunicação com o mundo externo, diferentemente da simples programação da televisão.

O desenvolvimento de um software que receba as imagens de câmeras de vigilância e a disponibilize na tela da televisão é possível porque a TV digital difere-se da TV analógica pela forma como os sinais são modulados. No formato digital as informações estão em formato binário onde existe a possibilidade de se aplicar técnicas computacionais, conseguindo enviar dados diferentes do conteúdo televisivo (ZIMERMANN, 2007). Esses dados podem ser um software, o qual será executado no Set-Top Box² (STB).

Com um software rodando em um servidor e um STB poderá haver comunicação, podendo acessar as imagens de câmeras instaladas em locais específicos. Esses locais poderão ser públicos, onde, por exemplo, antes de pegar a estrada, poderá checar como está o trânsito no momento e escolher a melhor rota. E também ter acesso a câmeras privadas, que, por exemplo, pode-se conferir como está o andamento da produção em sua fábrica, e tudo isso por meio do aparelho de televisão.

Com o estudo nesta área, irá se verificar as tecnologias aplicadas para a transmissão completa dos dados, compreendendo a parte de armazenamento e programação para o SBTVD para que possa ser aplicada de maneira geral, em todos os ramos necessários,

²Aparelho necessário para decodificação do sinal da TV digital.

havendo assim a possibilidade de criação de softwares de acesso remoto de imagens de câmeras de vigilância pela TV digital.

1.4 ESTRUTURA DO TRABALHO

O trabalho está dividido em quatro capítulos principais. O Capítulo 1 apresentou uma introdução geral sobre a pesquisa, apontando os objetivos gerais e específicos, bem como a introdução e a justificativa. O segundo capítulo descreve sobre segurança e câmeras de vigilância.

TV digital, seus conceitos, técnicas e principais características são abordados no terceiro capítulo. O Capítulo 4 descreve sobre ferramentas e tecnologias que serão utilizadas para o desenvolvimento do software capaz de realizar o objetivo desse trabalho. O quinto capítulo descreve sobre alguns trabalhos correlatos, em âmbito nacional, estadual e também sobre um trabalho desenvolvido na universidade.

O Capítulo 6 apresenta a metodologia e ferramentas utilizadas na elaboração do código da aplicação, bem como as versões da aplicação, os resultados e dificuldades encontradas. O trabalho é encerrado com a Conclusão, que indica algumas possibilidades de trabalhos futuros.

2 SEGURANÇA

A violência existente nas cidades está em constante crescimento. Para conter essa criminalidade há a necessidade de controle, com isso o crescimento e inovação dos sistemas para proteção são de suma importância para a sociedade. Esses sistemas têm a finalidade de proteger pessoas e patrimônios (MOREIRA, 2007).

Mesmo que a maior responsabilidade pela segurança pública seja do Estado, a população e empresas podem e devem ajudar, montando seus próprios sistemas de proteção.

Geralmente há algum tipo desses sistemas em todos os locais do cotidiano de pessoas. Existem para proteção de invasões externas ou de pessoas não autorizadas. Essa preocupação existe desde as primeiras cidades construídas pela humanidade, onde se construía muros altos e poços de água para defesa de quem se encontrava dentro de determinado ambiente.

Durante toda a história tiveram-se problemas com segurança. Moreira (2007) cita que a sociedade enfrentou graves problemas na revolução industrial que foi quando se criou a primeira corporação profissional pública para esse problema. A primeira guerra mundial ficou marcada pelo medo e a pressão a qual a população foi submetida. O impacto sofrido na segunda guerra mundial foi menor do que a primeira, pois a segurança pessoal, de patrimônio e de informações estava mais evoluída devido aos recursos investidos. No período da guerra fria, o investimento foi intensificado, principalmente da informação, devido esse ser um dos focos da guerra.

Porém, mesmo com o passar do tempo e com diversas formas para combater o crime, sempre há uma parte da população que é voltada para a criminalidade, sendo influenciados por diversos fatores, como guerras e pobreza. O comportamento dessas pessoas

é altamente influenciado ao quanto de violência esta pessoa possa estar sendo submetida (MOREIRA, 2007).

As técnicas de proteção vão evoluindo para prevenir e proteger pessoas e locais da criminalidade a qual possam estar expostas. As novas tecnologias também vêm ao encontro da segurança, sendo usadas como aliada no combate ao crime.

A cada dia surgem inovações na área tecnológica. Sensores dos mais variados tipos são comumente utilizados. A biometria é explorada principalmente para restrição de acesso a determinados locais, assim como o uso de cartões ou chaves.

2.1 SEGURANÇA VIGIADA

Dentre as várias tecnologias existentes em prol da segurança, as câmeras de vigilância são utilizadas em larga escala. Elas estão espalhadas em vários lugares, tanto públicos quanto particulares.

Elas têm como característica atribuir sensação de segurança aos locais em que estão aplicadas. Como descreveu Foucault (2000) estar sendo vigiado significa estar seguro. Por estarem sendo vigiados, indivíduos têm receio em praticar atos ilícitos, sendo forçados a praticar as normas e valores pré-definidos pela sociedade e pelas leis vigentes (CORREA; CUNHA, 2009).

As câmeras, denominadas popularmente como de segurança, possibilitam que informações sejam adquiridas por pessoas ou equipamentos a distância. Esses podem tomar atitudes ou decisões para prevenir ou providenciar algo na hora em que está ocorrendo o ato, entre outras serventias, que serão detalhadas no próximo subcapítulo.

2.2 CIRCUITO FECHADO DE TELEVISÃO

Circuito Fechado de Televisão (CFTV) é um sistema de monitoração por meio de câmeras supervisionadas. Tem como propósito a segurança e/ou vigilância dos locais/pessoas onde estão instaladas. É composto basicamente por câmeras de vídeo, software específico para gerenciamento das imagens e um meio para a transmissão do sinal das câmeras até o servidor (JANES, 2009).

O CFTV abrange a transmissão das imagens capturadas pelas câmeras até um local que seja monitorado, podendo ser armazenadas em algum gravador ou em um computador.

A monitoração dessas imagens pode ser feita localmente, onde geralmente se tem um centro de controle e vigilância, ou pode ser feita de maneira remota pela rede (MOREIRA, 2007). Outro detalhe a ser observado é a iluminação do local, se o local vigiado for aberto, há a possibilidade de instalar câmeras para uso diurno e noturno, porém, se o local for fechado, pode-se utilizar câmeras específicas para a iluminação existente.

2.2.1 Sistemas Analógicos

Janes (2009) diz que os sistemas de CFTV analógicos é um tipo de sistema que utilizam câmeras analógicas. Transmitem estas informações até o gerenciador destas imagens, geralmente com cabo coaxial.

Nesses sistemas as imagens são exibidas em monitores para a vigilância e são armazenadas em fitas VHS, sendo essa a principal característica do sistema analógico, conforme Figura 1.

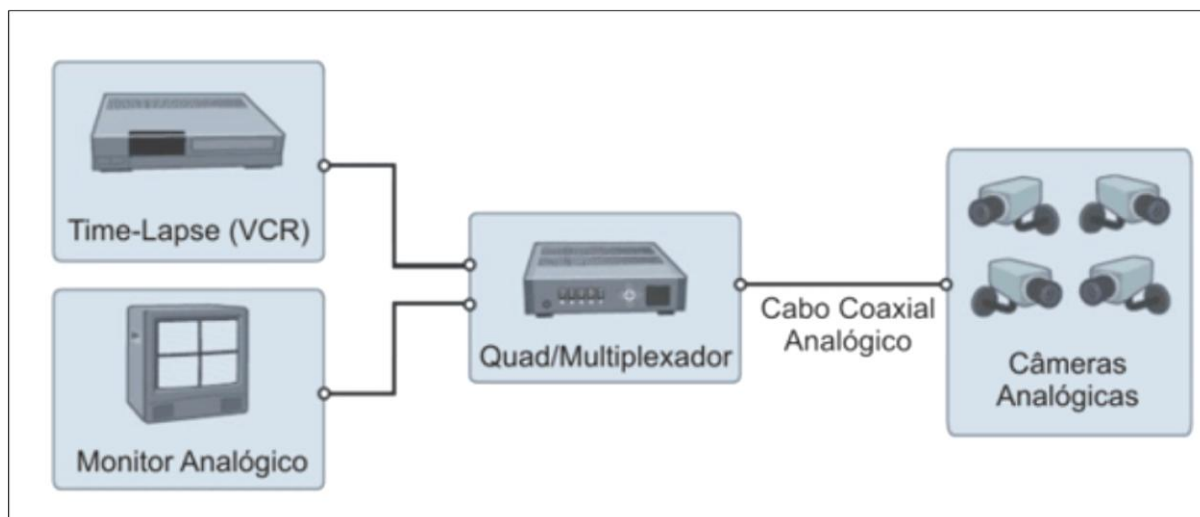


Figura 1. Sistema Analógico
 Fonte: NETCOM (2009)

O *Time-Lapse* (Video Cassete Recorder - VCR) é um vídeo cassete feito para gravar em fitas VHS de longa duração.

O multiplexador é responsável por possibilitar visualizar a imagem de várias câmeras em um único monitor. Ele tem também a responsabilidade de digitalizar a imagem para o *Time-Lapse* gravar.

Os monitores são projetados para funcionar em tempo integral, pois geralmente a vigilância é feita 24 horas por dia.

2.2.2 Sistemas Digitais

O sistema digital, assim como o analógico, tem a função de monitorar e gravar as imagens capturadas em um local específico, porém ele o faz de forma mais eficiente.

As câmeras desse sistema são digitais, gerando melhor qualidade e precisão de imagens. Para o gerenciamento são utilizados computadores e softwares. As câmeras são ligadas na rede, onde se deve ter um servidor capturando as imagens e armazenando-as. Elas

podem ser acessadas ao vivo tanto pela rede local quanto remotamente pela Internet (JANES, 2009). A Figura 2 demonstra o sistema digital.

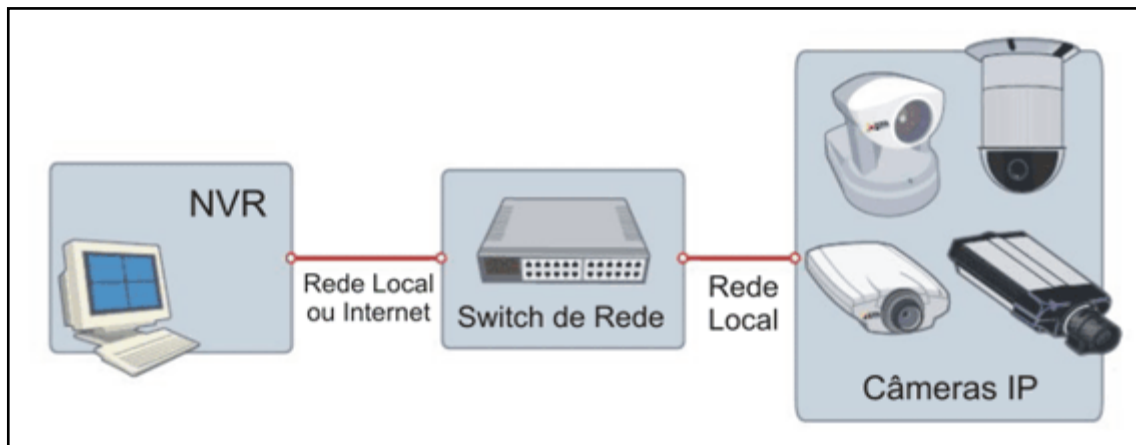


Figura 2. Sistema Digital
Fonte: NETCOM (2009)

Há uma grande facilidade em encontrar um vídeo específico se comparado ao sistema analógico, pois os arquivos salvos são ordenados ou nomeados de acordo com dia ou localização da câmera.

Outra grande vantagem são os sensores de movimento, que ao detectarem movimento iniciam a gravação, não acumulando horas de vídeo sem que haja nenhuma movimentação em frente à câmera.

2.3 PROJETOS PÚBLICOS DE VIGILÂNCIA

Kanashiro (2008) comenta que no mundo inteiro a instalação de câmeras de vigilância nas ruas das cidades é cada vez mais comum. E no Brasil não é diferente, várias cidades estão implantando esses sistemas de videovigilância.

A pioneira nesse tipo de projeto foi Praia Grande / SP, instalando uma interligação entre as repartições públicas da cidade. Esta rede, conhecida como Infovia³, possibilitou a instalação de diversas câmeras que vigiam os locais com maior movimento, prédios públicos e monumentos (CARVALHO, E. A. 2008).

A necessidade de monitoração desses locais existe principalmente devido à segurança de bens materiais e de pessoas. As imagens dessas câmeras também são utilizadas como objetos de investigações policiais e como provas de crimes.

Segundo Carvalho (2008) os índices de ocorrências policiais e de depredações em equipamentos públicos municipais diminuíram consideravelmente nas cidades onde foram implantados os sistemas de videovigilância.

Outro projeto conhecido nessa área é o Projeto Olho Vivo⁴ da cidade de Belo Horizonte / MG. Um sistema de vigilância digital que tem uma central de monitoramento em tempo real e com alta definição. Esse sistema conta com profissionais capacitados, onde cada um é responsável por quatro câmeras, podendo visualizar a região com mais detalhes, graças à capacidade de poder controlá-las. O projeto opera de forma plena desde 2004.

Nesse sistema é possível uma pessoa vigiar quatro lugares ao mesmo tempo, podendo acionar o despachante⁵ que, à distância, aciona a autoridade competente ao determinado problema. Com o decorrer da operação, o despachante também pode informar se está sendo abordada a pessoa certa ou por onde ela está fugindo.

Em relação à legalidade em instalação de câmeras de vigilância, a Constituição Federal prevê que não se pode violar a intimidade e a vida privada das pessoas, porém no estado de Minas Gerais onde esse projeto está em funcionamento existe uma lei estadual que legaliza sua utilização de para fins de segurança, desde que informado a presença das mesmas e que não sejam instaladas em locais íntimos, como vestiários, banheiros e provadores. Essas

³ Mais informações em: http://www.unicamp.br/unicamp/unicamp_hoje/jornalPDF/ju362pag04.pdf

⁴ Mais informações em: http://www.biblioteca.pucminas.br/teses/TratInfEspacial_CarvalhoEA_1.pdf

⁵ Pessoa responsável por abrir o chamado.

imagens também não podem ser exibidas a terceiros, exceto em casos de processos judiciais (CARVALHO, E. A. 2008).

O próximo capítulo será feita uma descrição sobre TV digital, padrões existentes no mundo e o padrão nipo-brasileiro com suas características e funcionalidades inovadoras que possibilitam o desenvolvimento da aplicação proposta.

3 TV DIGITAL

A TV, como qualquer outro meio de comunicação, está em constante evolução e adaptação às novas tecnologias e necessidades sociais. Muitas mudanças ocorreram desde o surgimento dos primeiros canais de televisão. A cor deixou a televisão muito mais elegante e atraente e aumentou o número de canais, podendo o telespectador ter escolhas entre o que assistir.

Em um sistema televisivo existem basicamente dois cenários: quem emite os conteúdos e quem os assiste, Soares e Barbosa (2009) fazem uma analogia ao sistema de cliente e servidor. O primeiro cenário é quem desenvolve os programas e conteúdos que serão transmitidos para os telespectadores/clientes (CRUZ; MORENO; SOARES, 2008). A forma como é transmitido esse conteúdo está em uma fase de transição, do analógico para o digital. Essa mudança se dá na digitalização da transmissão do sinal, trazendo a possibilidade de tratamento dos dados transmitidos. A primeira mudança notória entre esses dois sistemas é a qualidade de imagem e de som (SOARES; BARBOSA, 2009).

Na transmissão analógica o conteúdo transmitido é de natureza contínua, varia continuamente com o tempo. É necessário modular o sinal para enviar da emissora e demular ao chegar o sinal no receptor. O sinal enviado pode sofrer alterações durante o percurso, alterações essas dadas em decorrência de problemas de atenuação por perdas de energia do sinal transmitido, ruídos provocados por outros sinais e/ou distorções por atraso. No modelo analógico o resultado mais comum desses problemas são os “fantasmas” na imagem exibida pela televisão.

No sistema digital, sendo que o conteúdo digital se dá de forma discreta, é possível aplicar algoritmos para recuperação de informações, conservando a qualidade do sinal, obtendo uma imagem praticamente perfeita àquela enviada pela emissora. Ela é exibida

perfeitamente, sem “chuviscos”, ou, se o sinal estiver muito fraco e os algoritmos de recuperação não conseguirem recuperar o sinal, não irá ser exibido nada na tela.

A banda do canal de frequência utilizado para transmissão de conteúdos televisivos é de 6 MHz. No modelo analógico, os sinais enviados da emissora não podem ser compactados, utilizando assim toda a banda disponível para transmissão.

No modelo digital, há como comprimir o sinal, possibilitando que mais dados sejam enviados nos mesmos canais (MONTEZ; BECKER, 2005). Os dados possíveis de serem enviados podem ser escolhidos de diversas formas. Uma das alternativas é a transmissão de vários programas de televisão de menor qualidade simultaneamente, podendo ser tanto da mesma emissora, quanto de emissoras distintas, processo esse conhecido como multiprogramação. Outra, e principal alternativa, é a transmissão de um único programa de maior qualidade (HDTV) que ocupa toda a faixa de frequência, assim transmitindo também áudio no padrão 5.1 (multicanal) e imagem de 1.920 pixels por linha e 1080 linhas em 30 quadros por segundo, o que dá uma qualidade de áudio e vídeo muito superior ao disponível no sistema analógico (SOARES; BARBOSA, 2009).

No sistema analógico é necessário que haja um canal sem uso entre os canais de transmissão devido à interferência que ocorre entre ambos. Montez e Becker (2005) comentam que em Florianópolis existem os canais 2, 4, 6, 9, 12, 15, 18 e 20 em uso, necessitando de 120 MHz para transmitir os oito canais sendo que teoricamente precisariam de somente 48 MHz. No sistema digital, um não interfere em outro, não sendo necessária essa banda de frequência livre entre eles, melhorando e aproveitando o meio de comunicação disponível.

A necessidade de aperfeiçoamento da utilização do espectro de frequência é causada pelo avanço das tecnologias de comunicação, entre elas o uso do rádio, celular e da própria TV, que estão cada vez mais complexos e trafegando maior quantidade de dados.

3.1 SISTEMA BRASILEIRO DE TV DIGITAL

A evolução do sistema de televisão analógico para o digital está ocorrendo em diversos países do mundo. Para isso é necessário aderir a um padrão ou criar seu próprio sistema de transmissão (MONTEZ; BECKER, 2005).

As pesquisas para a implantação de TV digital no Brasil iniciaram em 1994, onde foram estudados e testados os padrões que já estavam em funcionamento. Segundo Tonieto (2006), em meados de 2000 encerrou-se a discussão técnica sobre o padrão de TV Digital a ser adotado pelo Brasil.

No Brasil foram feitos testes com os três padrões internacionais existentes. Dentre os padrões ASTC – americano, DVB – europeu e o ISDB – japonês, os que obtiveram melhores desempenhos foram os padrões Europeu e Japonês. Pesquisas de técnicos brasileiros resultaram na escolha do padrão japonês, incorporando tecnologias desenvolvidas por diversos grupos de pesquisas aqui no Brasil. O modelo ISDB-T também foi considerado superior em relação ao desempenho na recepção do sinal em ambientes fechados e por terminais fixos ou móveis.

Assim, surge o Sistema Brasileiro de TV Digital (SBTVD) que é um padrão nipo-brasileiro, com base japonesa e adaptado às condições e necessidades brasileiras, integrando uma nova arquitetura com canal de interatividade.

De acordo com o art. 1º do Decreto 4.901 e seus incisos, temos os seguintes modelos no SBTVD (BRASIL, 2003):

- a) **modelo social:** para promover a inclusão social, a diversidade cultural do País e a língua nacional através de tecnologia digital, criando uma rede de educação à distância e o acesso a informações;

- b) **modelo tecnológico:** estimular a pesquisa e o desenvolvimento e propiciar a expansão de tecnologias e indústrias nacionais relacionadas com tecnologia da informação (TI) e comunicação;
- c) **modelo político:** estimular a evolução das atuais exploradoras de serviço de televisão analógica, bem como a possibilidade de novas empresas, propiciando a expansão do setor e possibilitar o desenvolvimento de produtos e serviços decorrentes de tecnologia digital. Ainda visar o aperfeiçoamento do uso do espectro de frequência, como já citado no item 3.
- d) **modelo econômico:** contribuir para a convergência tecnológica e empresarial dos serviços de comunicação e incentivar a produção de conteúdo em âmbito local e regional.

De acordo com Tonieto (2006) para chegar ao êxito desse decreto presidencial, foram designados dois comitês: o de Desenvolvimento e o Consultivo e mais um Grupo Gestor, o qual tem a finalidade de definir as políticas da TV digital.

O SBTVD está muito bem estruturado, e como consequência, vários países já aderiram ao padrão brasileiro de TV digital e outros estão estudando a possibilidade de adoção. Segundo Montez e Becker (2005) esse tipo de evolução acontece de forma lenta e gradativa, não substituindo a tecnologia antiga de forma repentina, porém é uma mudança que está ocorrendo de forma global.

3.2 ARQUITETURA SBTVD

Por ser um sistema altamente complexo, o SBTVD tem uma arquitetura de referência dividida em camadas. Cada uma é composta por uma arquitetura de hardwares e softwares com a finalidade de oferecer serviços padronizados para a camada superior e

utilizar os serviços oferecidos pela camada inferior (FERNANDES; LEMOS; SILVEIRA, 2004). A divisão de tarefas pode ser observada na Figura 3.

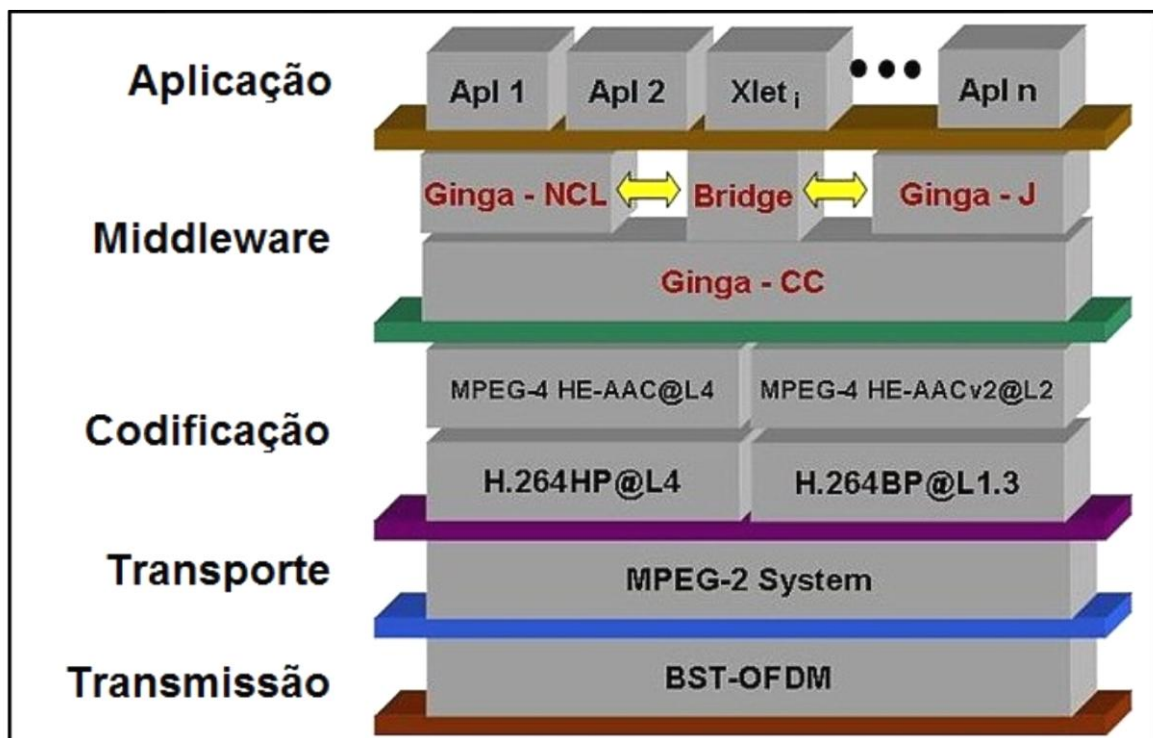


Figura 3. Tecnologias usadas em cada camada da arquitetura de TV Digital
Fonte: FERNANDES, J; LEMOS, G; SILVEIRA, G (2004)

Tonieto (2006) classifica os componentes de um sistema de TV digital em cinco conjuntos de padrões:

- a) transmissão de modulação, que é subdividido em três partes:
 - transmissão e recepção: no difusor é a parte responsável por colocar o sinal da TV digital no ar e no receptor é o responsável pela sintonia do sinal,
 - modulação e demodulação: no emissor é o responsável pela modulação do fluxo de transporte codificado e no receptor é responsável pela demodulação do sinal em um fluxo de transporte codificado,
 - codificação e decodificação: amplamente conhecido como *codec*⁶, codifica e decodifica o fluxo de transporte no emissor e no receptor, respectivamente;

⁶ Acrônimo de Codificador/Decodificador, responsável por codificar/decodificar sinais.

- b) transporte: na emissora, é a parte responsável pela multiplexação de vários programas para que seja enviado em um único fluxo de transporte. No receptor, faz a demultiplexação de acordo com a programação escolhida pelo usuário;
- c) compressão e codificação: faz a compressão e a descompressão dos sinais de áudio e vídeo na emissora e no receptor, respectivamente;
- d) *middleware*: é um software que, por meio de serviços, deixa transparente as funções das camadas de transporte e transmissão para a aplicação.
- e) aplicações: é a camada visível ao usuário, que fará a interação e executará os aplicativos, sendo suportada pelas camadas inferiores.

3.3 MIDDLEWARE GINGA

Já visto na seção 3.2, o *middleware* é uma camada de software existente entre o Hardware/SO e os aplicativos de TV digital. Tem como finalidade oferecer diversas facilidades para o desenvolvimento de conteúdo e aplicativos, tornando transparentes as camadas de comunicação e transporte, fazendo com que aplicativos rodem da mesma maneira, independente de hardware ou sistema operacional (SO) (DAMASCENO, 2008).

Tonieto (2006) diz que é muito importante a portabilidade em sistemas de TV digital, pois haverá diferentes configurações de receptores para executar as aplicações. O *middleware* abstrai essas diferenças, como exemplificado na Figura 4.

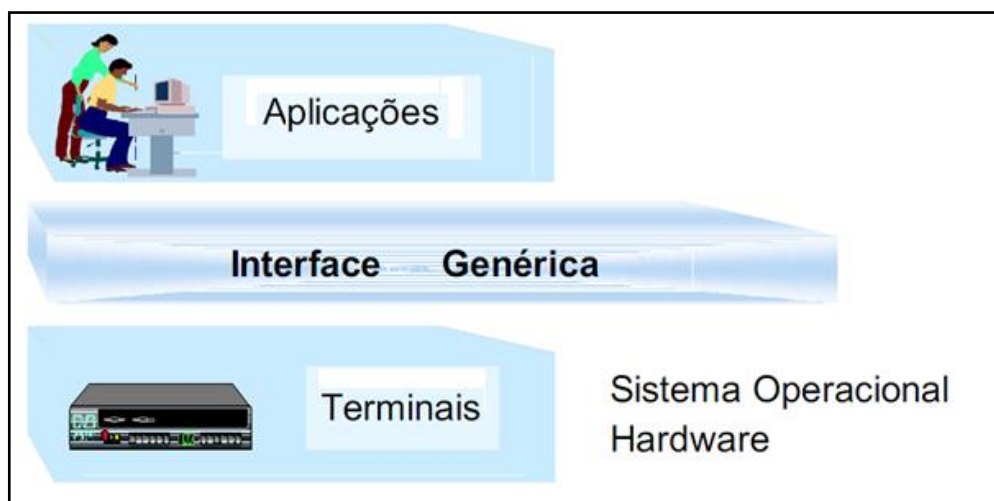


Figura 4. *Middleware*
Fonte: TONIETO, M. (2006)

A Figura 4 mostra seu escopo, provendo os mesmos serviços, independente de qual receptor ou SO estará executando a aplicação.

Para que as funções possam operar de modo eficaz, ela segue diversos requisitos, Tonieto (2006) cita quatro principais:

- a) confiabilidade: o sistema tem de ser confiável, pois telespectadores não estão habituados a esperar, assim, dependendo da aplicação que esteja carregando, pode levar muito tempo ou ocorrer falhas, o que pode levar a ter uma má reputação de toda infraestrutura envolvida;
- b) segurança: o *middleware* deve conter técnicas de segurança da informação durante a troca de dados para que possam, por exemplo, realizar transações com senhas ou compras com cartões de crédito;
- c) extensibilidade: o *middleware* tem de ser capaz de suportar futuros softwares e hardwares, pois há um contínuo processo de inovação tecnológica;
- d) reflexibilidade: considerando a variedade de STBs, é importante que se tenha a capacidade de reflexão, que é oferecer, além da interface padrão, uma metainterface, cuja finalidade é disponibilizar a possibilidade para a aplicação de melhorias em características com o usuário, e mudanças para melhoria de

desempenho, mudando o ambiente, adequando às necessidades de um dispositivo ou protocolo específico.

Soares e Barbosa (2009, p. 22) fazem uma importante afirmação:

O middleware é um dos componentes mais importantes de um sistema de TV digital porque, na prática, é ele que regula as relações entre duas indústrias de fundamental importância: a produção de conteúdos e a de fabricação de aparelhos receptores. Do ponto de vista do software, podemos dizer, sem exagero, que ao definir o middleware estamos, de fato, definindo um sistema de televisão. Dominar o conhecimento dessa tecnologia é estratégico para um país, pois o não-domínio certamente acarretaria também o não-domínio de seu uso.

O SBTVD tem como principal destaque o seu *middleware*, o Ginga. Ele é resultado de anos de pesquisas feitas pelo laboratório TeleMídia⁷ da Pontifícia Universidade Católica do Rio de Janeiro (PUC-Rio) e pelo Laboratório de Aplicações de Vídeo Digital (LAVID)⁸ da Universidade Federal da Paraíba (UFPB). Ginga agrupa diversas tecnologias e inovações que o tornam muito avançado e adequado à realidade do país.

Segundo Soares (2008) a arquitetura do Ginga pode ser dividida em três módulos principais que serão descritos a seguir.

3.3.1 Ginga-CC:

O Ginga é dividido em duas classes de aplicações, as declarativas e as procedurais. O subsistema lógico que disponibiliza funcionalidades comuns entre elas é o Ginga Common-Core (Ginga-CC) (CRUZ; MORENO; SOARES, 2008).

⁷ Mais informações em: <http://www.telemidia.puc-rio.br/>

⁸ Mais informações em: <http://www.lavid.ufpb.br/>

3.3.2 Ginga-NCL

É o subsistema lógico responsável pelo processamento das aplicações declarativas do *middleware* do SBTVD, desenvolvidas na linguagem de programação Nested Context Language (NCL) que será melhor explicada no item 4.2. Esse subsistema é obrigatório nos receptores portáteis e fixos.

3.3.3 Ginga-J

É o subsistema lógico responsável pelo processamento das aplicações imperativas desenvolvidas na linguagem de programação Java. Ele é obrigatório apenas nos receptores fixos.

O desenvolvimento da parte Java do Ginga é liderado pelo grupo de pesquisa do LAVID. A Figura 5 exemplifica a forma como está estruturado o *middleware* Ginga.

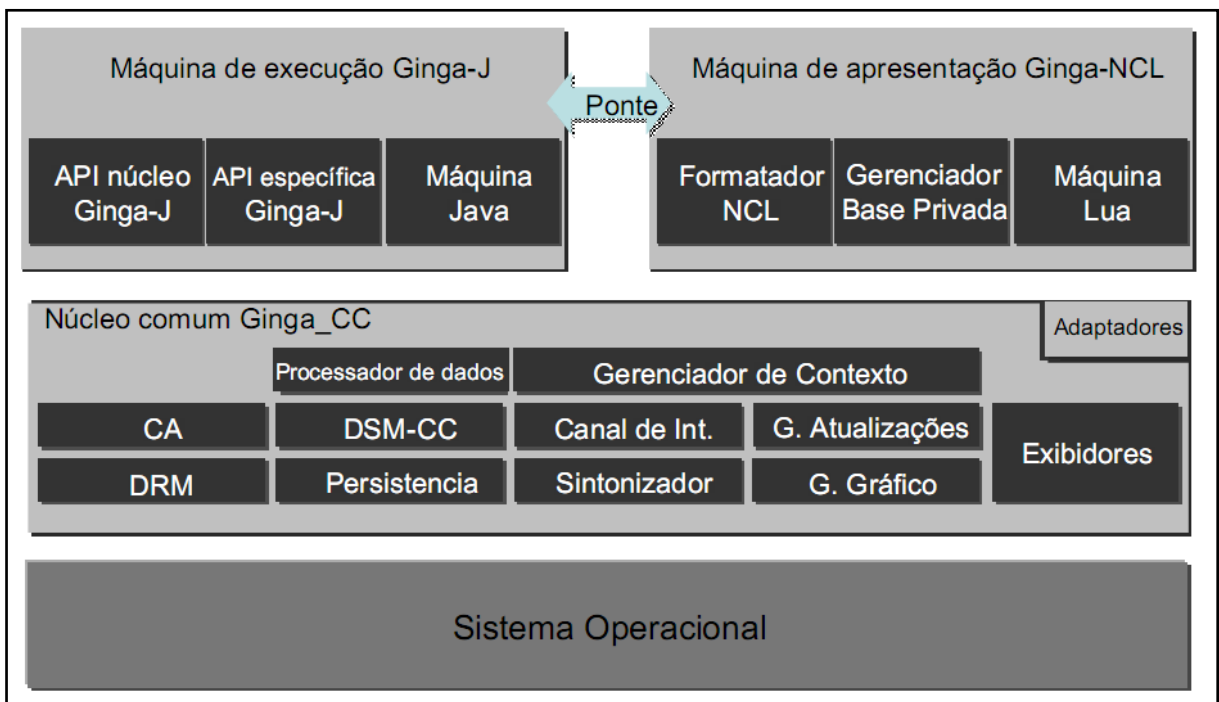


Figura 5. Arquitetura *Middleware* Ginga
Fonte: SOARES, L. (2008)

A Figura 5 ilustra os conceitos e a estrutura apresentada nos subcapítulos 3.3.1, 3.3.2 e 3.3.3 que falam sobre os subsistemas existentes no *middleware* Ginga.

3.4 INTERATIVIDADE

Para Montez e Becker (2005) interatividade acontece quando um usuário pode participar ou influenciar na modificação imediata do que está ocorrendo. Eles exemplificam que jornais e livros seriam pouco interativos e que videogames e teleconferências seriam de alta interatividade. Tonieto (2006) coloca que interatividade acontece quando há a participação direta do interessado àquilo que está ocorrendo/utilizando.

Devido essa abrangência no conceito de interatividade, tanto Tonieto (2006) quanto Montez e Becker (2005) comentam que os termos relacionados com interatividade, estão sendo utilizados mais como propaganda do que com seu real significado, para chamar atenção de seus produtos/serviços.

Na televisão, a interatividade já existiu se apresentou de diversas formas. Uma das mais conhecidas era no programa *Você Decide*, da TV Globo, o qual era um programa que apresentava parte de uma história e no intervalo da programação, o público decidia seu final.

Para essa interatividade era necessário um meio externo de comunicação, o telefone. Para a decisão eram informados dois ou mais números de telefone, e o que tivesse maior número de ligações ao final da votação seria o ganhador. Em programas até hoje ainda são largamente utilizados esse tipo de interatividade, tanto por serviços de voz citado anteriormente quanto o serviço amplamente utilizado por SMS⁹.

Com o advento de TV digital e do *middleware* Ginga, a comunicação que antes era apenas unidirecional, o qual somente a emissora enviava conteúdos aos telespectadores,

⁹ Short Message Service: Serviço de mensagem de texto que habilita mensagens curtas a serem enviadas ou transmitidas por aparelhos específicos.

transforma-se em uma comunicação bidirecional, o qual possibilita o antes mero espectador, fazer escolhas, solicitar e receber respostas personalizadas, tornando-se assim um usuário e tendo diversas possibilidades e experiências diante da televisão (TONIETO, 2006).

Com o canal de interatividade não é mais necessário a utilização de um meio externo para haver comunicação entre o usuário e a emissora.

O usuário tem a possibilidade de escolher aquilo que quer assistir, como por exemplo, qual câmera escolher dentre as disponíveis em um jogo de futebol. Tem também a possibilidade de ampliar imagens, utilizar as funções do tipo play/pause ou até comprar um produto (*T-Commerce*) que possa estar sendo oferecido em uma aplicação interativa.

Recursos avançados como utilização de e-mail por meio da TV, nomeado como *T-mail*¹⁰ e outros serviços como *T-Government*¹¹, *T-Educação*¹², *T-Saúde*¹³, *T-Bank*¹⁴ ou qualquer outro serviço que possa estar disponível. Assim, quem está assistindo deixa de ser apenas espectador, tornando-se parte da programação, interagindo com o que lhe interessa.

Damasceno (2008) comparando os outros padrões de TV digital existentes no mundo, diz que algumas das funcionalidades foram desenvolvidas especificamente para a realidade brasileira, tornando-se inovadoras, principalmente o canal de interatividade.

Segundo Eduardo Carvalho (2008) a interatividade no Sistema Brasileiro de TV digital pode ser dividida em três níveis principais.

¹⁰T-Mail: Acesso de e-mail pela televisão.

¹¹T-Government: Acesso de aplicativos do governo pela televisão.

¹²T-Educação: Possibilidade de acesso a conteúdos educacionais pela televisão.

¹³T-Saúde: Acesso a aplicativos voltados a saúde, como, por exemplo, agendar uma consulta.

¹⁴T-Bank: Acesso a serviços bancários.

3.4.1 Interatividade Local

Segundo o próprio Eduardo Carvalho (2008) e Soares e Barbosa (2009), o sistema de TV digital terrestre pode ser interativo mesmo sem o canal de retorno, o conteúdo é recebido da emissora e executado no receptor¹⁵.

As funcionalidades pertinentes a esse nível são, entre outras, a possibilidade de habilitação de opções que foram recebidas da emissora, como áudio adicional ou legendas, por exemplo.

3.4.2 Interatividade Intermitente

Nesse nível de interação a aplicação é enviada pela emissora, executada no receptor e retornada informações do telespectador para a emissora por meio de um canal de retorno unidirecional.

A conexão pode ser feita na hora em que ir enviar os dados de volta para a emissora e interrompido logo após ter enviado as informações.

Soares e Barbosa (2009) exemplificam esse tipo de interatividade com a possibilidade de *T-Commerce*, onde pode haver toda a propaganda via *broadcast*¹⁶ e somente os interessados irão fazer a conexão para efetuar a compra do produto.

¹⁵ Mais detalhes sobre receptores no subcapítulo 3.5.

¹⁶ Uma transmissão ser enviada a mais de um receptor simultaneamente.

3.4.3 Interatividade Permanente

Existe um canal de retorno bidirecional, conhecido também como canal de interatividade, que deve estar ativo em tempo integral, onde há a possibilidade não só de retornar dados à emissora, mas também fazer downloads de dados utilizados pelas aplicações.

Numa breve apresentação da aplicação, é nesse cenário que esse trabalho irá atuar. Uma aplicação que deve ser enviada por difusão pela emissora e ao ser carregada no receptor, fazer download das imagens das câmeras de vigilância por meio do canal de interatividade e exibi-las na tela da televisão.

Soares e Barbosa (2009) ainda citam um quarto nível, o qual seria a Interatividade Plena, que possibilitaria a chamada “TV social (*social TV*)” ou “TV comunitária (*community TV*)”, que é a possibilidade um grupo de usuários telespectadores de um mesmo programa poderem trocar dados entre si.

3.5 RECEPTORES DIGITAIS

Eduardo Carvalho (2008) apresenta o receptor digital, que também conhecido como Set-Top Box (STB), é o aparelho responsável por captar o sinal digital da emissora e tratá-lo, fornecendo como saída o conteúdo a ser assistido e as aplicações interativas. Para que se possa desfrutar de toda essa interação disponível é preciso que o receptor digital proporcione suporte ao canal de interatividade. Tonieto (2006) apresenta cinco tipos de receptores digitais, que são:

- a) tipo 1: terminal zapper – o terminal mais simples e conseqüentemente mais barato. Disponibiliza apenas funções básicas, não disponibilizando nenhum tipo de interatividade;

- b) tipo 2: terminal com aplicações residentes – Terminal também simples, que disponibiliza apenas a possibilidade de algumas aplicações residentes, não possuindo suporte para o canal de interatividade;
- c) tipo 3: com suporte a carga de aplicações transmitidas por *broadcast* – este tipo de receptor dá suporte às aplicações de interatividade local, onde são enviadas pela emissora junto com o áudio e o vídeo do canal selecionado. Como não tem o canal de interatividade, não é possível enviar dados para a emissora;
- d) tipo 4: receptor com canal de interatividade – com suporte a todas as funcionalidades do receptor de tipo 3, este receptor ainda pode enviar dados para a emissora, comunicando-se com aplicações remotas. Essa interação é feita pelo *middleware* Ginga usando o canal de interatividade;
- e) tipo 5: receptor com funcionalidades avançadas – além de todas as funcionalidades do receptor com canal de interatividade, pode-se também comunicar-se com outros dispositivos, como celulares, computadores, dispositivos para auxílio de deficientes físicos. Tem também a funcionalidade de gravação do conteúdo transmitido (PVR – Personal Video Recorder). Com hardware mais robusto, esse tipo de receptor é o de preço mais elevado e o que contém funcionalidades avançadas do *middleware*.

Para a captura do sinal digital é necessário uma antena específica. O receptor pode ser um aparelho separado ou estar embutido na televisão. A maioria dos televisores vendidos atualmente já vem com o conversor integrado.

No entanto, para que todas essas funcionalidades sejam possíveis, existe uma poderosa linguagem de programação, capaz de integrar e exibir toda essa gama de possibilidades na tela de uma TV.

4 LINGUAGEM ESPECÍFICA SBTVD

Nesse trabalho foi citada a falta de segurança enfrentada pela sociedade atual e a nova tecnologia de TV digital que está sendo adotada que aumentará a gama de possibilidades e recursos. Nesse cenário ingressa a convergência entre diferentes dispositivos e tecnologias para auxiliar na solução de problemas existentes.

4.1 APLICAÇÃO

Câmeras de vigilância, tanto públicas quanto privadas, podem ser vigiadas localmente ou remotamente como já citado no Item 2.

O objetivo do governo brasileiro é disponibilizar TV digital em todo o território nacional até o limite de 2013, e até em 2016 fazer o desligamento da transmissão analógica.

Partindo dessas duas premissas, este trabalho tem como objetivo disponibilizar imagens de câmeras de vigilância em qualquer local que haja transmissão do sinal televisivo e que tenha a disponibilidade do canal de interatividade. Será exibido o conteúdo normal da programação da emissora e estará disponível um ícone para acessar a aplicação interativa.

Ao acionar esse ícone, abrirá a aplicação que terá ícones para acesso rápido às câmeras mais utilizadas ou algum lugar para entrar com o endereço da câmera desejada. Ao escolher, a aplicação deverá carregar a imagem da câmera de vigilância por meio do canal de interatividade e mostrar a imagem na tela da televisão para que o telespectador possa visualizar.

A aplicação tem intuito de apenas exibir as imagens capturadas por terceiros e disponibilizadas à população. Caso sejam câmeras particulares, deve-se ter tratamento de

controle de acesso e/ou criptografia nos dados, funções essas não tratadas nesse trabalho. Não é abordado a fundo questões como legalidade de vigilância ou quebra de privacidade.

Para o desenvolvimento da aplicação interativa nos padrões do SBTVD deve-se utilizar a linguagem específica definida. Como já apresentado no subcapítulo 3.3.2, a linguagem padrão do Sistema Brasileiro de TV Digital é a Nested Context Language (NCL).

4.2 NCL

A NCL é uma linguagem multimídia declarativa de autoria brasileira, desenvolvida pelo Laboratório Telemídia da PUC-Rio. Ela separa claramente os conteúdos de mídia e a estrutura da aplicação, definindo como os objetos de mídia são estruturados e relacionados no tempo e no espaço.

Soares e Barbosa (2009) dizem que, como uma linguagem de cola, a NCL não restringe e nem prescreve os tipos de conteúdo dos objetos de mídia de uma aplicação. As aplicações interativas podem ou não estarem diretamente relacionadas com o conteúdo exibido pela programação da televisão. Porém quanto mais próximo do que está passando na TV, maior a chance do usuário se interessar em executar o aplicativo.

Alguns aspectos são específicos por serem destinados a televisão, pois devido à impossibilidade de previsão de o telespectador iniciar uma aplicação interativa é necessária a possibilidade de produção de programas não-lineares¹⁷ ao vivo. Outra característica, ainda não muito explorada, é o suporte a múltiplos dispositivos. Outra característica é a adaptabilidade que torna possível que o Set-Top Box adapte o conteúdo de acordo com sua localização ou capacidade de processamento (SOARES NETO et al, 2006).

¹⁷ Termo se dá em contraposição à forma sequencial – linear – na TV analógica onde só existe um caminho de exibição.

4.3 NESTED CONTEXT MODEL (NCM)

Segundo Soares e Barbosa (2009, p. 35): “Toda linguagem declarativa é baseada em um modelo conceitual de dados”. O modelo conceitual de dados tem a finalidade de representar os conceitos estruturais dos dados, os eventos e os relacionamentos entre si. Também é função do modelo definir as regras estruturais e operações sobre os dados para manipulação e atualização das estruturas.

O NCM utiliza conceitos que são comumente utilizados em documentos hipermídia: nós e elos. Segundo Soares Neto et al (2006) um nó pode ser de dois tipos: nó de conteúdo ou de mídia, que é associado a um elemento de mídia ou nó de composição ou contexto.

Uma visão básica sobre vários elementos do modelo NCM será abordado nos próximos subcapítulos com base em Soares Neto et al (2006) e Soares e Barbosa (2009).

4.3.1 Regiões

Uma região é uma área definida no dispositivo de saída e é o local onde um nó será exibido. Em um programa NCL é definido pelo menos a região base e nesta podem ser definidas outras diversas. Ela tem como finalidade posicionar os nós em locais específicos da tela.

Cada região tem diversos atributos:

- a) id: identificador da região;
- b) title: título da região;
- c) left: coordenada X do lado esquerdo;

- d) top: coordenada Y do lado superior;
- e) right: coordenada X do lado direito;
- f) bottom: coordenada Y do lado inferior;
- g) width: largura da região;
- h) height: altura da região;
- i) zindex: posição do eixo Z, geralmente utilizado para definir regiões sobrepostas umas as outras.

As coordenadas são relacionadas a região pai. Caso seja a primeira, a tela se torna pai. Dependendo da aplicação, o programador escolhe entre configurar os atributos right, bottom, width e height ou os atributos top, left, width e height.

4.3.2 Descritores

Um descritor é que define como e onde um nó de mídia será apresentado e exibido. Os atributos pertinentes ao descritor são:

- a) id: identificador do descritor;
- b) player: atributo opcional que identifica a ferramenta que será utilizada para exibir o objeto de mídia;
- c) explicitDur: define a duração do objeto de mídia;
- d) region: define em que região será exibido o objeto de mídia;
- e) descriptorParam: define parâmetros do descritor.

4.3.3 Portas

Porta oferece acesso externo ao conteúdo de um contexto. Possui os seguintes atributos:

- a) id: identificador da porta;
- b) component: objeto ao qual a porta se refere;
- c) interface: nome do ponto de interface de destino no contexto.

4.3.4 Âncoras

Âncoras são marcos de entrada para os nós de mídia ou contextos. Soares e Barbosa (2009, p. 37) fazem uma metáfora utilizando o conjunto de animais racionais:

[...] Podemos dizer que o conjunto de animais racionais é composto por seres humanos(objetos de mídia). Todo ser humano tem um identificador, por exemplo, seu CPF. Partes do ser humano são suas âncoras: coração, cérebro etc. O corpo, como um todo, é a âncora que delimitaria todo o conteúdo. Da mesma forma que nós de mídia têm tipos (áudio, vídeo etc.), os seres humanos também podem ser classificados quanto ao sexo, raça etc.

Assim, o objetivo de utilizá-las é com o intuito de manipular partes de um nó de mídia ou contexto, tanto como origem quanto como destino de elos. As âncoras podem ainda ser de dois tipos:

- a) âncora de conteúdo: determina o intervalo de tempo de execução e/ou a região no espaço.
- b) âncora de propriedade: referem-se a atributos de um nó, tanto de origem quanto de destino, que poderão ser manipulados pelos elos.

4.3.5 Contextos

Os contextos são utilizados para estruturação do documento. Os atributos mais triviais são:

- a) id: identificador do contexto.
- b) refer: a quem o contexto está referenciando.

Soares Neto et al (2006) frisa que o elemento *body* é considerado um caso especial de contexto, pois representa o documento como um todo.

4.3.6 Nó de mídia

O nó de mídia ou de conteúdo é a mídia propriamente dita, como vídeo, áudio, imagem ou texto. Para cada nó de mídia deve haver um descritor que ajustará sua apresentação. Tem os seguintes atributos:

- a) id: identificador do nó de mídia;
- b) type: tipo de mídia que terá no nó;
- c) src: endereço relativo do arquivo a ser carregado;
- d) descriptor: identificação do descritor que regula o objeto de mídia;
- e) refer: referência a outro nó de mídia.

4.3.7 Elos

Elos, conhecidos como *links*, ligam os nós por meio de conectores. Os atributos dos elos são:

- a) id: identificador do elo;

- b) xconnector: identifica o conector associado ao *link*;
- c) linkParam: define parâmetros para o *link*;
- d) bind: define os componentes envolvidos no elo;
- e) bindParam: define parâmetros para o bind do elo.

4.3.8 Conectores

O NCM define que o sincronismo é feito por mecanismos de casualidade e restrição, que são definidos pelos conectores. Assim, existem dois tipos de conectores, os causais e os de restrição.

Soares Neto et al (2006) comenta que para TV digital, a linguagem NCL usa somente conectores causais, que definem dois ou mais papéis, indicando condições de ativação do elo e as ações que deverão ser realizadas quando ele for ativado.

5 TRABALHOS CORRELATOS

Alguns trabalhos foram essenciais para a realização dessa pesquisa. Ajudaram a entender conceitos, mostrando e aplicando-os na prática. A seguir estão descritos sucintamente os principais e mais relevantes.

5.1 SISTEMA BRASILEIRO DE TV DIGITAL – SBTVD UMA ANÁLISE POLÍTICA E TECNOLÓGICA NA INCLUSÃO DIGITAL

Este trabalho é uma dissertação de mestrado na área de computação aplicada elaborado na Universidade Estadual do Ceará. Elaborado por Marcia Terezinha Tonieto em 2006.

Apresenta um relato histórico sobre a construção do SBTVD e conta com uma pesquisa de campo sobre inclusão social, característica pela qual o governo quer implantar o sistema de TV digital em todo o território nacional.

A pesquisa foi realizada tanto com pessoas especialistas como pessoas leigas no assunto, para que haja valor às análises realizadas (TONIETO, 2006).

Essa pesquisa foi elaborada quando estava iniciando a implantação do SBTVD, assim, o resultado dessa pesquisa aponta recomendações para auxiliar na implantação do Sistema Brasileiro de TV Digital terrestre, levando em consideração a Estrutura de Análise do Modelo de Referência, utilizando com o base fatos, análises e pesquisas de opinião, bem como decisões formais e informais do governo.

5.2 TV DIGITAL PARA DISPOSITIVOS PORTÁTEIS - *MIDDLEWARES*

Este documento é um estudo documentado por grandes nomes da TV digital brasileira: Vitor Medina Cruz, Marcio Ferreira e Luiz Fernando Gomes Soares. Todos integrantes do laboratório Telemidia da PUC-Rio, onde as linguagens NCL e Lua foram desenvolvidas.

O trabalho oferece um embasamento teórico muito importante para iniciantes, pois apresenta um amplo estudo sobre as tecnologias de TV digital.

Inicia conceituando V digital e falando sobre características dos dispositivos portáteis. Descreve sobre diversos Sistemas Operacionais para esses dispositivos.

Com relação direta com TV digital, o trabalho descreve sobre sua organização e seus *middlewares*, citando o modelo europeu e em seguida falando sobre o Sistema Brasileiro de TV Digital, inserindo no contexto o Ginga e seus subsistemas Ginga-NCL e Ginga-J.

Interessante para iniciantes no estudo do SBTVD, pois trata de assuntos básicos, como, por exemplo, o desenvolvimento do *middleware* Ginga foi feito por duas universidades. A PUC-Rio responsável pelo ambiente declarativo (Ginga-NCL), enquanto a Universidade Federal da Paraíba pelo ambiente procedural (Ginga-J) (CRUZ; MORENO; SOARES, 2008).

Esse estudo concluiu que o Ginga-NCL e a Broadcast Markup Language (BML) seriam, na época, os mais apropriados *middlewares* para dispositivos portáteis.

5.3 CANAL DE RETORNO EM TV DIGITAL

Um artigo elaborado por Filipi Zimmermann na Universidade Federal de Santa Catarina que apresenta uma revisão bibliográfica sobre a comunicação por meio do canal de retorno entre dispositivos televisivos.

Explica como funciona todo o processo de transmissão de TV digital, incluindo a transmissão da emissora ao receptor e como é feito o retorno desses dados pelo canal específico, com a finalidade de dar suporte aos profissionais de TI que queiram utilizar os recursos de interatividade em suas aplicações.

São abordados assuntos como os tipos de interatividade televisiva, como é feito a difusão e o acesso, de que forma funciona o receptor e o canal de retorno.

Zimmermann (2007) conceitua canal de interatividade como sendo formado por dois canais de comunicação: o sinal que vem da emissora, com a programação corrente e com aplicativos interativos, e o canal de retorno, o qual envia dados à emissora, formando assim uma comunicação entre ambos.

Ele coloca também que apesar das diversas possibilidades para soluções de problemas, existem diversos empecilhos para utilização do canal de interatividade, como a rede de comunicação entre telespectador e emissora, as aplicações e sua usabilidade e que os provedores de conteúdo tem de estar bem estruturados para atender a demanda.

5.4 ESTUDO SOBRE A TELEVISÃO DIGITAL E DESENVOLVIMENTO DE UM PROTÓTIPO INTERATIVO BASEADO NO SUBSISTEMA PROCEDURAL DO MIDDLEWARE GINGA

Trabalho desenvolvido recentemente na Universidade do Extremo Sul Catarinense – UNESC, no curso de Ciência da Computação e defendido no último semestre de 2010. Bortolotto (2010) aborda diversos conceitos do Sistema Brasileiro de TV Digital e de seus subsistemas Ginga-NCL e Ginga-J, onde o ambiente procedural foi amplamente abordado, devido ser o foco do desenvolvimento.

A finalidade do trabalho foi desenvolver um protótipo no ambiente procedural, utilizando a linguagem de programação Java, com a implementação de referência OpenGinga, servindo como demonstrativo do potencial de aplicações que podem ser feitas, desde a parte de interface, processamento e utilização dos recursos providos pelo *middleware* Ginga.

Foi desenvolvido um aplicativo de *T-Commerce*, onde é possível escolher entre diversos produtos para comprar por meio do controle da televisão. A ideia é de estar passando um comercial na programação corrente da emissora, e aparecer o aplicativo, que disponibiliza a compra pela própria televisão.

6 SISTEMA VIGILÂNCIA TV DIGITAL

Foi proposta a elaboração de um software que disponibilize as imagens de determinadas câmeras de vigilância pelo Sistema Brasileiro de TV Digital Terrestre.

O protótipo será desenvolvido no ambiente declarativo do *middleware* Ginga, utilizando a linguagem NCL, a qual já foi apresentada e estudada no corpo do presente documento.

A aplicação para o SBTVD trará para o telespectador a possibilidade de saber, por exemplo, como está o movimento ou o tempo em diversos locais da região, onde poderá selecionar a câmera que deseja assistir.

6.1 METODOLOGIA

No início do trabalho, foi realizado todo o embasamento teórico e definição de conceitos sobre o universo de televisão digital. Estudou-se desde a história da TV, sua evolução até o sistema digital. No contexto de TV digital, foi estudado seu funcionamento, características de transporte, envio e recebimento, como também o suporte a interatividade proporcionada pelo *middleware* Ginga.

Durante todo o trabalho foi feito o levantamento bibliográfico com a finalidade de encontrar materiais científicos para pesquisa do assunto. A maioria dos materiais utilizados está disponível em sites de universidades ou nos sites dos fabricantes. Os materiais de universidades são em sua grande maioria teses de mestrados. Para o entendimento da linguagem NCL, foi efetuada a compra de um livro, que é referência no assunto.

Foi estudada a aplicabilidade de videovigilância, exemplificando como e onde é utilizada essa técnica, quais as tecnologias mais utilizadas para segurança e qual o envolvimento e importância de diversificar as formas de monitoramento da vigilância.

O Sistema Brasileiro de TV Digital, mesmo sendo uma tecnologia nova, têm diversos materiais publicados, tanto em congressos e eventos, quanto em pesquisas científicas, tratando sobre a tecnologia, leis e conceitos científicos. A participação em eventos e fóruns na Internet trouxe contatos com diversas pessoas que ajudaram na busca de informações necessárias.

Tanto o canal de interatividade quanto o *middleware* Ginga estão muito bem documentados e já foram estudados por diversas universidades, assim facilitando a compreensão e escrita desse trabalho.

O desenvolvimento e os testes serão feitos baseados no conhecimento já adquirido e com suporte de toda a comunidade NCL¹⁸, TVD¹⁹ e devdtv²⁰.

6.2 FERRAMENTAS DE DESENVOLVIMENTO

Para o desenvolvimento dessa aplicação foi necessário a utilização de diversas ferramentas, montando assim um ambiente de desenvolvimento capaz de auxiliar na codificação, execução e visualização de aplicações NCL. Todas as ferramentas são livres, disponíveis para *download* na Internet.

Para iniciar precisa-se de uma máquina virtual do Ginga para poder emular uma TV com interatividade. Esta máquina²¹ é disponibilizada pelo Laboratório Telemídia da PUC-

¹⁸ Clube NCL <http://clube.ncl.org.br/>

¹⁹ Grupo de Discussão da TV Digital Interativa www.tvdi.inf.br

²⁰ Desenvolvimento para TV digital <http://groups.google.com/group/devdtv?hl=pt-BR>.

²¹ <http://www.telemidia.puc-rio.br/pt/index.html>

Rio. Segue a explicação sucinta do necessário para configurar o ambiente de desenvolvimento, para mais informações ver Anexo A.

6.2.1 Eclipse

A IDE utilizada para implementação foi o Eclipse, pois dentro dela é possível a instalação de diversos plugins²², necessários para desenvolvimento de toda a aplicação. O Eclipse está disponível para download²³ e não é preciso instalar, é só executar o arquivo binário. Como ele é desenvolvido em Java, é necessário ter a máquina virtual instalada.

Para o código NCL, é necessário a instalação do plugin NCL Eclipse, que auxilia e agiliza o desenvolvimento. Ele pode ser instalado diretamente pelo sistema automático do Eclipse. Adicionando o endereço²⁴ e clicando em OK ele instalará automaticamente.

Como o código deve estar dentro da máquina virtual do Ginga, vai ser preciso acessá-lo ou enviar arquivos para ele. Com o putty pode-se criar uma conexão SSH. Porém para editar os arquivos NCL optou-se por utilizar outro plugin do Eclipse, que irá editá-los remotamente. Denominado Remote System Explorer (RSE) ele disponibiliza serviços de SSH e FTP. Sua instalação²⁵ também está disponível pelo sistema automático.

No RSE o acesso aos arquivos remotos é feito de maneira facilitada, ajudando na produtividade. Sem ele, seria necessário editar os arquivos na máquina local e sempre que fosse executar, teria que copiar os arquivos para a máquina virtual e então executá-los. Fazer todo esse processo diversas vezes pode se tornar cansativo e desestimulante, perdendo muito em produtividade. O plugin é mostrado na Figura 6.

²² Programa de computador usado para adicionar funções a outros programas maiores, provendo alguma funcionalidade especial ou muito específica..

²³ <http://www.eclipse.org/downloads/>

²⁴ <http://www.laws.deinf.ufma.br/ncleclipse/update>

²⁵ <http://download.eclipse.org/dsdp/tm/updates/3.0>

Estando no diretório da aplicação, é necessário dar o comando “/misc/launcher.sh main.ncl” para iniciar a aplicação, isso por que o código principal está dentro do arquivo “main.ncl”.

6.2.2 VMware

Para emular o ambiente de uma televisão com recurso de interatividade é necessário um *player* de máquinas virtuais. Para rodar o Ginga foi utilizado o VMware Player, disponível para *download*²⁶. Após é só abrir a máquina virtual, que já vem com as configurações padrões necessária para rodar.

6.3 CONTROLE REMOTO

Para compreender algumas funções das aplicações de TV digital é necessário saber o básico sobre o controle remoto das TV's. A Figura 8 mostra um controle remoto de um STB da Proview²⁷.

²⁶ http://downloads.vmware.com/d/info/desktop_downloads/vmware_player/3_0

²⁷ Fabricante de Set-Top Box



Figura 8. Controle Remoto

O controle de Set-Top Box ou de televisão digital que tenha conversor para captação do sinal digital, contem os botões para interatividade. Os botões são como os mostrados acima na Figura 8. São quatro botões nas cores: vermelho, verde, amarelo e azul. São sempre dispostos nessa ordem de cores.

6.4 DESENVOLVIMENTO DA APLICAÇÃO

Com o conhecimento técnico-científico, foi dado início ao desenvolvimento da aplicação declarativa na linguagem NCL. A aplicação foi desenvolvida no modelo de prototipação, onde foram criadas telas e a partir desse modelo foi desenvolvido o código. O telespectador vai controlar a aplicação por meio do controle remoto da televisão. O diagrama de casos de uso da Figura 9 demonstra as ações possíveis.

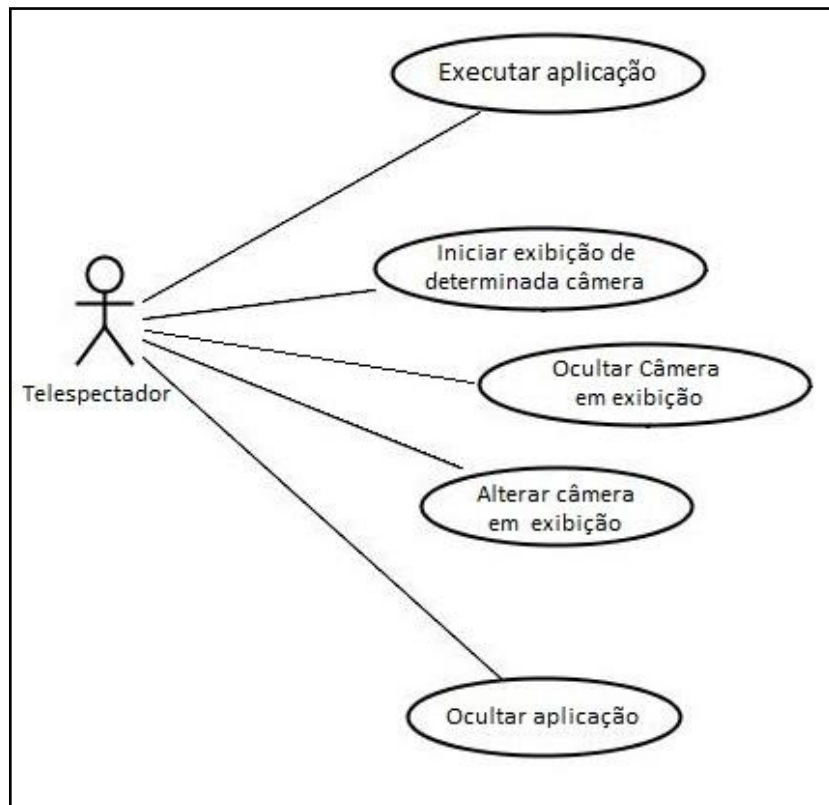


Figura 9. Diagrama de casos de uso

Com testes foi constatado que poderiam ser feitas modificações na interface da aplicação para melhorar a usabilidade, definindo assim um modelo para o projeto final.

6.4.1 Primeira Versão

A Figura 10 mostra uma tela com quatro botões, que fariam a seleção entre as opções.

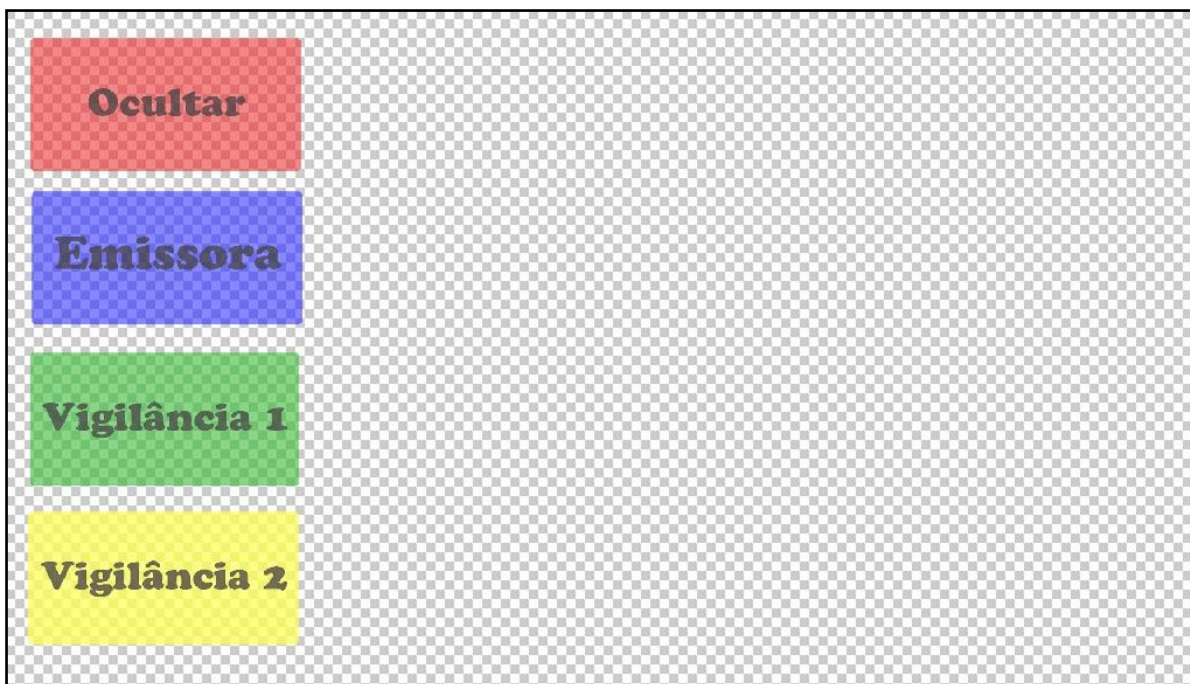


Figura 10. Primeira versão de Interface

A Figura 10 ilustra a primeira versão da aplicação, onde a parte xadrez cinza representa transparência. A aplicação iniciaria com o vídeo enviado pela emissora por meio do carrossel de dados. Ainda sem a preocupação em iniciar a aplicação somente quando o telespectador desejasse, ela já iniciava automaticamente ao ser recebida pelo STB.

Como haveria quatro opções, foi colocado cada botão de uma cor, que representaria cada botão do controle remoto. Então a seleção seria feita pelos botões de interatividade do controle remoto.

O primeiro botão (vermelho) servia para ocultar a aplicação inteira, ficando na tela somente a imagem da emissora. Todos os componentes seriam abortados pela aplicação. O segundo botão (azul) selecionaria novamente a imagem da emissora, caso alguma câmera de vigilância tivesse sido selecionada previamente.

A exibição de uma câmera de vigilância seria iniciada, caso o terceiro botão (verde) fosse selecionado. A imagem de outra câmera de vigilância seria acessada caso o quarto botão (amarelo) fosse acionado. Tanto a programação corrente da emissora, quanto as imagens das câmeras de vigilância seriam executadas em tela cheia.

Lembrando que os botões também têm transparência, a imagem de fundo é visível, porém os botões atrapalham um pouco a visibilidade.

Como a ordem dos botões não estava correta, foi decidido alterar a ordem dos botões para ter uma melhor usabilidade. A Figura 11 mostra como ficaria.



Figura 11. Versão 1.1 de Interface

Com a ordem dos botões alterados, a aplicação ficou muito mais usual para telespectadores em geral, mas os botões ainda atrapalham a visibilidade total das imagens das câmeras de vigilância.

6.4.2 Segunda Versão

A segunda versão de interface e de programação foi totalmente refeita. A Figura 12 mostra uma tela onde os vídeos são redimensionados para que nada fique na frente.



Figura 12. Segunda versão de Interface

A interface mostra um fundo com leve degradê²⁸, uma barra para o título da aplicação e com 4 botões, agora sem transparência. Mostra também a imagem de uma televisão, que os vídeos seriam redimensionados para dentro de sua área de tela.

A aplicação conta agora com um esquema de regras que, quando iniciada a aplicação, pode-se selecionar algum componente pelos botões UP e DOWN do controle remoto. Para cada elemento que ganhar o foco, a aplicação seta um valor em uma variável interna, caso ela seja selecionada, teclando a tecla ENTER do controle, a aplicação sabe qual elemento iniciar dentro da área estipulada. O código fonte das regras está definido na Figura 13.

²⁸ Área onde duas cores são sobrepostas, cada uma com suas intensidades, formando uma transição suave entre as cores.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<ncl id="regras" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
  <head>
    <ruleBase>
      <rule id="r1" var="opcao" comparator="eq" value="1"/>
      <rule id="r2" var="opcao" comparator="eq" value="2"/>
      <rule id="r3" var="opcao" comparator="eq" value="3"/>
      <rule id="r4" var="opcao" comparator="eq" value="4"/>
    </ruleBase>
  </head>
</ncl>
```

Figura 13. Código regras

Nesse esquema, ao selecionar o primeiro componente, o vídeo da emissora é selecionado para ficar aparecendo. Se o segundo componente for selecionado, a aplicação é totalmente ocultada. Dependendo do botão da câmera de vigilância que for selecionado, a respectiva imagem da câmera de vigilância é iniciada na tela da televisão.

6.4.3 Versão Final

Ao receber uma aplicação interativa, a televisão inicia a aplicação, porém o telespectador escolhe a hora que deseja que toda a aplicação tome conta de sua atenção. Para isso, a tela inicial conta com um ícone de interatividade, como mostra a Figura 14.



Figura 14. Tela inicial

O ícone dessa aplicação é uma câmera de vigilância. O fundo da tela da televisão está passando um *videocast* da UNESC.

Após vários testes foi verificado que os vídeos de câmeras de vigilância são melhores observados se não tiver nada em sua frente. Quanto ao vídeo da emissora, não há tanta importância em ter seu foco perdido, principalmente nas laterais, pois o conteúdo principal sempre se encontra mais ao centro da tela.

Ao ser acionado o botão vermelho (RED) do controle a aplicação inicia o contexto “App” da aplicação, como pode ser observado na Figura 15.

```

<!-- Quando selecionar o icone de interatividade, deve iniciar o contexto App -->
<link id="Icône" xconnector="con#onSelection1StartN">
  <bind role="onSelection" component="icone">
    <bindParam name="keyCode" value="RED"/>
  </bind>
  <bind role="start" component="App"/>
</link>
  
```

Figura 15. Código Ícone Interatividade

Nesse contexto estão os botões, o fundo, a barra de título e a ajuda do programa. A imagem recebida via *broadcast* é exibida em tela cheia, sendo que os menus, barras e imagens têm transparência.



Figura 16. Versão final de Interface

Ao iniciar essa tela, é possível escolher imagem de qual câmera deseja visualizar. Nessa versão não é mais utilizado as regras e nem a variável interna para seleção das câmeras de vigilância. É feito um controle manual de qual componente está com o foco, e a partir disso é feito o controle de qual tecla é acionada.

Se estiver na primeira opção, somente é permitido selecionar, por meio da tecla ENTER, ou então ir para baixo pela tecla DOWN. A Figura 17 mostra de quando a opção Câmera 1 estiver com o foco, como a aplicação deve se comportar.

```

<!-- Quando o component CAMERA1 estiver selecionado -->
<link id="Camera1DOWN" xconnector="con#onSelection1StartNStopN">
  <bind component="opcaoCamera1Sel" role="onSelection">
    <bindParam name="tecla" value="CURSOR_DOWN"/>
  </bind>
  <bind component="opcaoCamera2Sel" role="start"/>
  <bind component="opcaoCamera1" role="start"/>
  <bind component="opcaoCamera1Sel" role="stop"/>
  <bind component="opcaoCamera2" role="stop"/>
</link>

<link id="Camera1UP" xconnector="con#onSelection1StartNStopN">
  <bind component="opcaoCamera1Sel" role="onSelection">
    <bindParam name="tecla" value="CURSOR_UP"/>
  </bind>
  <bind component="opcaoOcultarSel" role="start"/>
  <bind component="opcaoCamera1" role="start"/>
  <bind component="opcaoCamera1Sel" role="stop"/>
  <bind component="opcaoOcultar" role="stop"/>
</link>

```

Figura 17. Código controle de ações

Se o componente que está com o foco for um dos dois intermediários, é permitido ir para cima (tecla UP), para baixo (tecla DOWN) ou então selecionar o componente (ENTER).

Quando uma câmera é selecionada, a aplicação irá pegar um vídeo de uma câmera de vigilância e irá mostrar na tela da televisão, em tamanho proporcionalmente grande em relação à tela da televisão. A região desse vídeo, declarado na aplicação como “rgResultado”, ocupará 68% (sessenta e oito por cento) da largura e 80% (oitenta por cento) da altura da tela.

A Figura 18 mostra o código das regiões da aplicação.

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<ncl id="regioes" xmlns="http://www.ncl.org.br/NCL3.0/EDTVProfile">
  <head>
    <!-- Regiões -->
    <regionBase>
      <region id="rgTV" width="100%" height="100%">
        <region id="rgFundo" left="0%" top="0%" width="100%" height="100%" zIndex="2"/>
        <region id="rgEmissora" left="0%" top="0%" width="100%" height="100%" zIndex="0"/>
        <region id="rgOpcaoOcultar" left="1%" top="16%" width="25.5%" height="12%" zIndex="3"/>
        <region id="rgOpcaoCamera1" left="1%" top="29%" width="25.5%" height="12%" zIndex="3"/>
        <region id="rgOpcaoCamera2" left="1%" top="42%" width="25.5%" height="12%" zIndex="3"/>
        <region id="rgOpcaoCamera3" left="1%" top="55%" width="25.5%" height="12%" zIndex="3"/>
        <region id="rgOpcaoOcultarSel" left="0.5%" top="13.8%" width="26%" height="15.5%" zIndex="4"/>
        <region id="rgOpcaoCamera1Sel" left="0.5%" top="26.8%" width="26%" height="15.5%" zIndex="4"/>
        <region id="rgOpcaoCamera2Sel" left="0.5%" top="39.8%" width="26%" height="15.5%" zIndex="4"/>
        <region id="rgOpcaoCamera3Sel" left="0.5%" top="52.8%" width="26%" height="15.5%" zIndex="4"/>
        <region id="rgResultado" left="30%" top="18%" width="68%" height="80%" zIndex="3"/>
        <region id="rgIcone" left="93%" top="2%" width="6%" height="10%" zIndex="3"/>
      </region>
    </regionBase>
  </head>
</ncl>

```

Figura 18. Código regiões

Os valores são feito de forma proporcional, pois não se sabe o tamanho do televisor que executará a aplicação. Assim, não há a necessidade de verificar o tamanho da tela antes de executar o aplicativo.

A aplicação ilustrada na Figura 16 vem por *default* selecionado a opção “Ocultar Câmeras”. Se alguma câmera estivesse sendo exibida ela seria ocultada quando este componente fosse selecionado pela tecla ENTER, como mostrado na Figura 19.

```

<!-- Quando a opção ocultar for selecionada, a aplicação deve
ocultar qualquer câmera que esteja sendo visualizada na tela da TV -->
<link id="Selecao_Ocultar" xconnector="con#onSelection1AbortN">
  <bind component="opcaoOcultarSel" role="onSelection">
    <bindParam name="tecla" value="ENTER"/>
  </bind>
  <bind component="camera1" role="abort"/>
  <bind component="camera2" role="abort"/>
  <bind component="camera3" role="abort"/>
</link>

```

Figura 19. Código Ocultar

Na Figura 20 tem uma câmera em execução, a qual está sobreposta ao conteúdo exibido pela emissora.



Figura 20. Câmera de vigilância sendo exibida

O menu informa que para sair da aplicação tem de se apertar a tecla azul do controle remoto. Ao fazer isso, a aplicação encerra o contexto “App”, o qual contém os componentes de seleção de câmeras, o componente para ocultar as câmeras, o fundo e a barra de título, ficando visível apenas o ícone de interatividade, o qual pode ser acessado novamente pelo botão vermelho do controle remoto.

A tecla BLUE tem a função de ocultar a aplicação, como pode ser observado na Figura 21.

```

<!-- Quando a tecla BLUE for selecionada, a aplicação deve parar o contexto App,
ficando visível apenas o icone de interatividade -->
<link id="Selecao_Sair" xconnector="con#onSelectionNStopN">
  <bind role="onSelection" component="fundo">
    <bindParam name="tecla" value="BLUE"/>
  </bind>
  <bind component="App" role="stop"/>
</link>

```

Figura 21. Código botão azul

A tecla azul só disparará esta ação se o contexto “App” já estiver sido iniciado, pois ela seleciona o componente fundo, o qual só é iniciado no contexto “App”.

6.5 RESULTADOS

Como resultado desta pesquisa obteve-se um aplicativo, apresentado e explicado anteriormente, que exibi imagens de câmeras de vigilância, proporcionando ao telespectador a possibilidade de interação com a aplicação, podendo escolher a imagem de qual câmera deseja visualizar.

É uma aplicação fácil de usar, característica necessária no ambiente de televisão, pois atende a praticamente todo tipo de pessoa, não tendo um público alvo determinado.

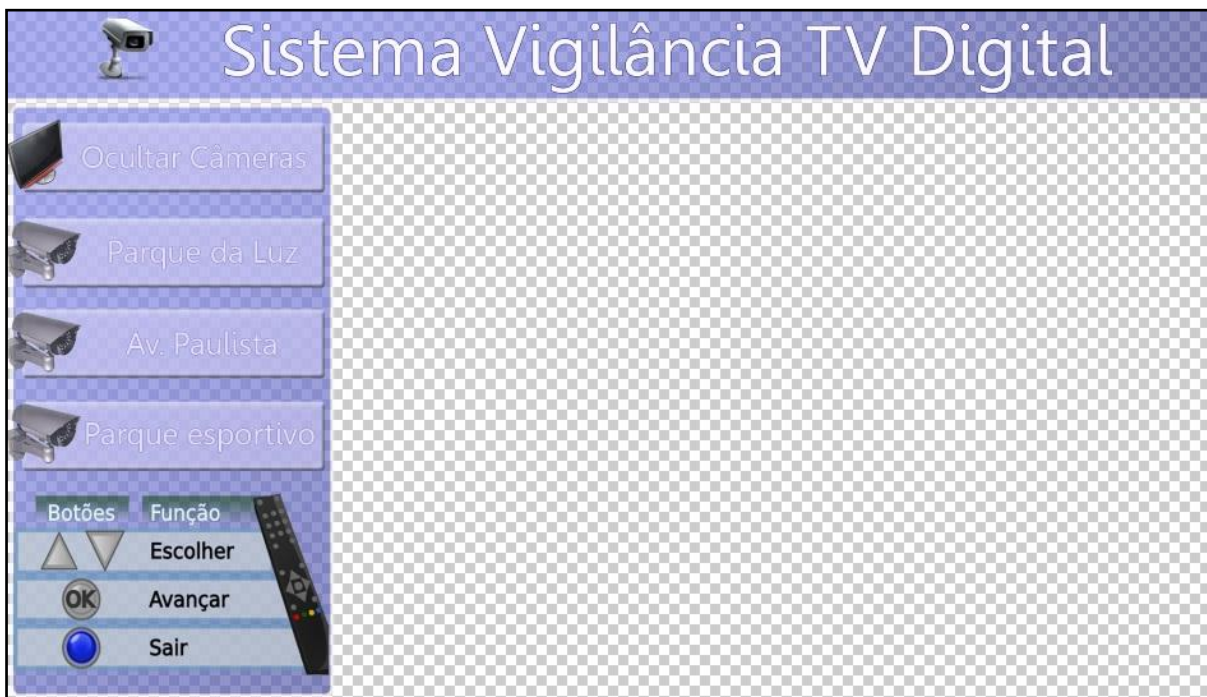


Figura 22. Sistema Vigilância TV Digital

Verificou-se que o middleware Ginga ainda não está completamente desenvolvido, atrapalhando assim na obtenção de êxito em todos os objetivos propostos. Porém a aplicação contém a ideia central de todo o trabalho, que é exibir as imagens de câmeras de vigilância em uma aplicação interativa.

Com o estudo de conceitos e aplicando-os em uma aplicação pode-se compreender como funciona o SBTVD abrindo assim um leque para novos estudos e aplicações.

Quando o *middleware* Ginga estiver totalmente implementado, as aplicações poderão ser muito mais ricas, cativando cada vez mais adeptos e difundindo cada vez mais a inclusão digital no Brasil. Utilizar o canal de interatividade para transmissão de arquivos de mídias, que poderão estar sendo atualizados ou modificados dependendo do perfil do telespectador, será uma das diversas possibilidades para maior interação e experiência em frente à televisão.

O Exibidor de NCL da GHTV²⁹ é uma implementação do padrão do SBTVD que está em desenvolvimento. É uma ferramenta que está sob a licença Creative Commons Attribution-NonCommercial-NoDerivs 3.0 Unported (CC BY-NC-ND 3.0)³⁰, onde o seu código fonte não pode ser usado comercialmente, porém pode ser utilizada em projetos livres. A versão 0.4.2, lançada no dia 14/06/2011 (dia da entrega desse trabalho) suporta a transmissão de arquivos pelo canal de interatividade.

O GHTV é um pequeno exibidor de documentos NCL, que conta com diversas funções, como suporte a HTML, NCL, mídias de texto, URL's HTTP contendo imagens, áudio, vídeo, HTML e texto, notificação de quando uma aplicação contém erros e registro no sistema operacional de mídias NCL.

Mesmo com diversas funcionalidades, ele ainda está em desenvolvimento e não trata várias características necessárias para apenas executar a aplicação desse projeto. O exibidor apresentou problemas com as transparências das imagens e com os conectores do código fonte. Como não há tempo hábil para migrar a aplicação para este exibidor, foram feitos apenas pequenos testes, verificando que realmente é possível o acesso de arquivos de

²⁹ <http://ghtv.com.br/>

³⁰ <http://ghtv.com.br/codigofonte/>

áudio, vídeo e imagem pelo canal de interatividade. A Figura 23 mostra o teste feito no sistema operacional Windows.

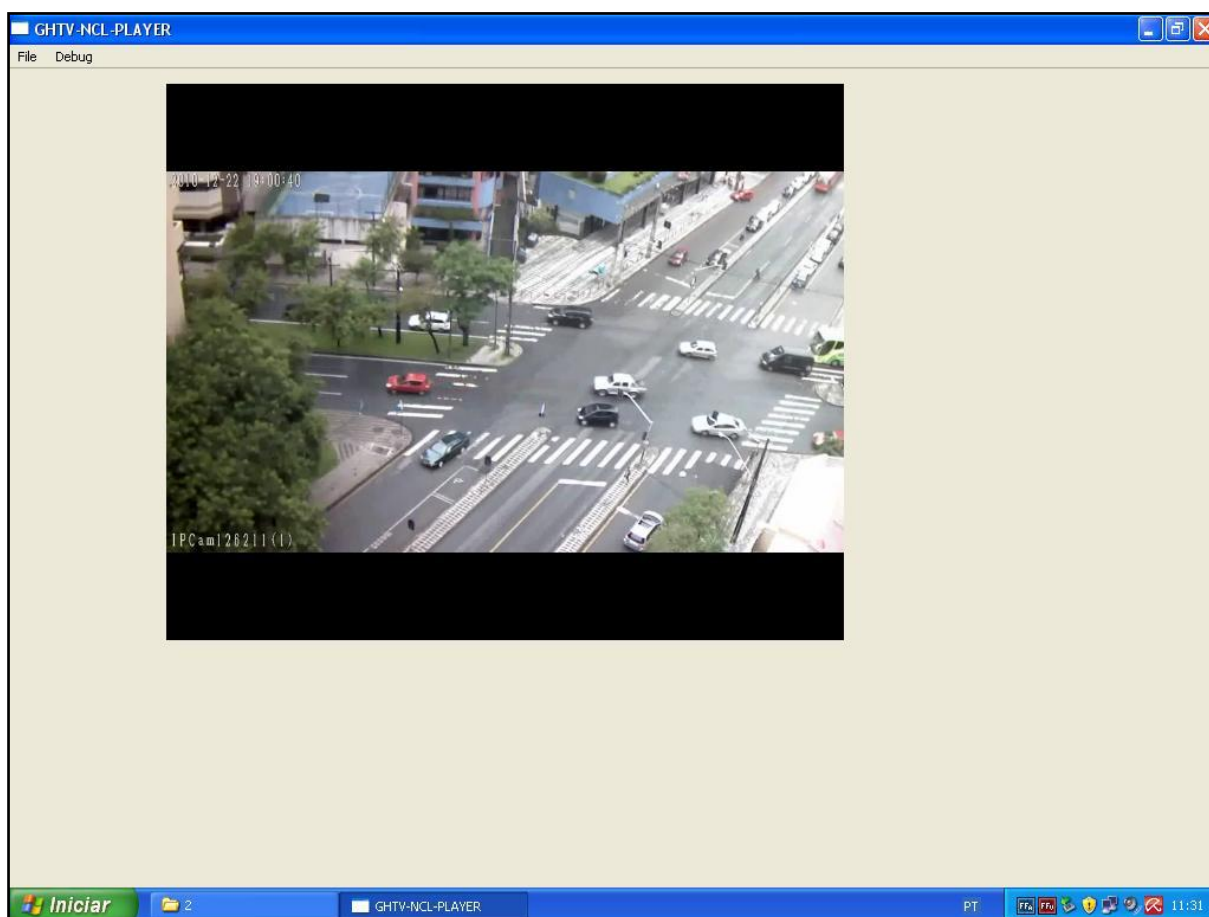


Figura 23. Teste em ambiente Windows

No ambiente Windows foi feito o teste para acessar um vídeo remoto, onde obteve-se êxito. Foi feito apenas o acesso a um arquivo de vídeo, pois o código foi elaborado no notepad³¹, porque o ambiente de desenvolvimento não estava configurado.

No ambiente Linux³² foi feita a execução do mesmo vídeo, iniciando com o mesmo vídeo da aplicação.

³¹ Editor de texto simples do Sistema Operacional Windows

³² Ubuntu 11.04 natty narwhal Kernel 2.6.38.2-8



Figura 24. Teste em ambiente Linux

No ambiente Linux o vídeo apresentou-se com sucesso. Para executar um vídeo pela rede é necessário que o arquivo que se deseja acessar esteja disponível no servidor. Para acessá-lo basta informar seu endereço quando declarar a mídia.

```

<!-- Mídias que serão iniciadas automaticamente -->

<port id="iniciarEmissora"          component="emissora"/>
<port id="iniciarInteratividade"    component="icone"/>

<!-- Declaração das mídias globais -->
<media id="emissora"  src="../1/media/videos/FeiraProfissoes2010.avi"      descriptor="dEmissora">
  <property name="bounds" value="0%,0%,100%,100%" />
</media>

<media id="icone"    src="http://192.168.254.6/videos/videoAvenida.avi"    descriptor="dIcone"/>

```

Figura 25. Declaração de mídia remota

No código da Figura 25 pode-se observar que quando a aplicação iniciar, ela vai carregar o componente emissora, que é um vídeo local e iniciará o componente ícone, o qual está no endereço de uma máquina da rede. O resultado é apresentado na Figura 24.

O mesmo código foi testado novamente no Ginga, o qual não executa a aplicação gerando um *log* de erro, conforme Apêndice A.

Na aplicação desse trabalho, por exemplo, poderiam ter sido exibidas as imagens de câmeras de vigilância em tempo real se o HiperText Transfer Protocol (protocolo HTTP) suportasse a transmissão de streaming. Mais detalhes serão abordados na Seção 6.6 onde serão discutidas as dificuldades encontradas.

6.6 DIFICULDADES ENCONTRADAS

Durante todo o processo de aprendizagem de uma nova linguagem e do desenvolvimento de uma solução para o problema encontrado encontrou-se diversas dificuldades.

Para o conhecimento dos conceitos e técnicas utilizadas para programação em NCL foi amplamente utilizado o livro Programando em NCL 3.0: desenvolvimento de aplicações para o *middleware* Ginga de Luiz Fernando Gomes Soares e Simone Diniz Junqueira Barbosa.

NCL, como já apresentado, é uma linguagem de programação brasileira. Por ser muito nova e ter sido regulamentada há pouco tempo, ainda não está toda implementada.

A ideia inicial do trabalho seria exibir imagens em tempo real de câmeras de vigilância por meio do canal de interatividade do SBTVD. Para isso seria necessário o Real Time Protocol (RTP) que ainda não foi implementado, tornando-se assim impossível o desenvolvimento por esse protocolo.

Foi instalado um programa, denominado Motion, que captura imagens de uma câmera, salva em um local e também as disponibiliza pela *web* pelo HiperText Transfer Propocol (HTTP). Sem sucesso novamente, a máquina virtual do Ginga não consegue

interpretar o que está recebendo. Para testar se o problema não estava na ferramenta VMware, foi emulado uma máquina do Windows XP e checado se era possível acessar as imagens do Motion rodando, tanto na máquina local quanto em outra máquina da rede. Esse teste confirmou que era possível o acesso das imagens via HTTP pela máquina virtual e pode ser visualizado na Figura 26.

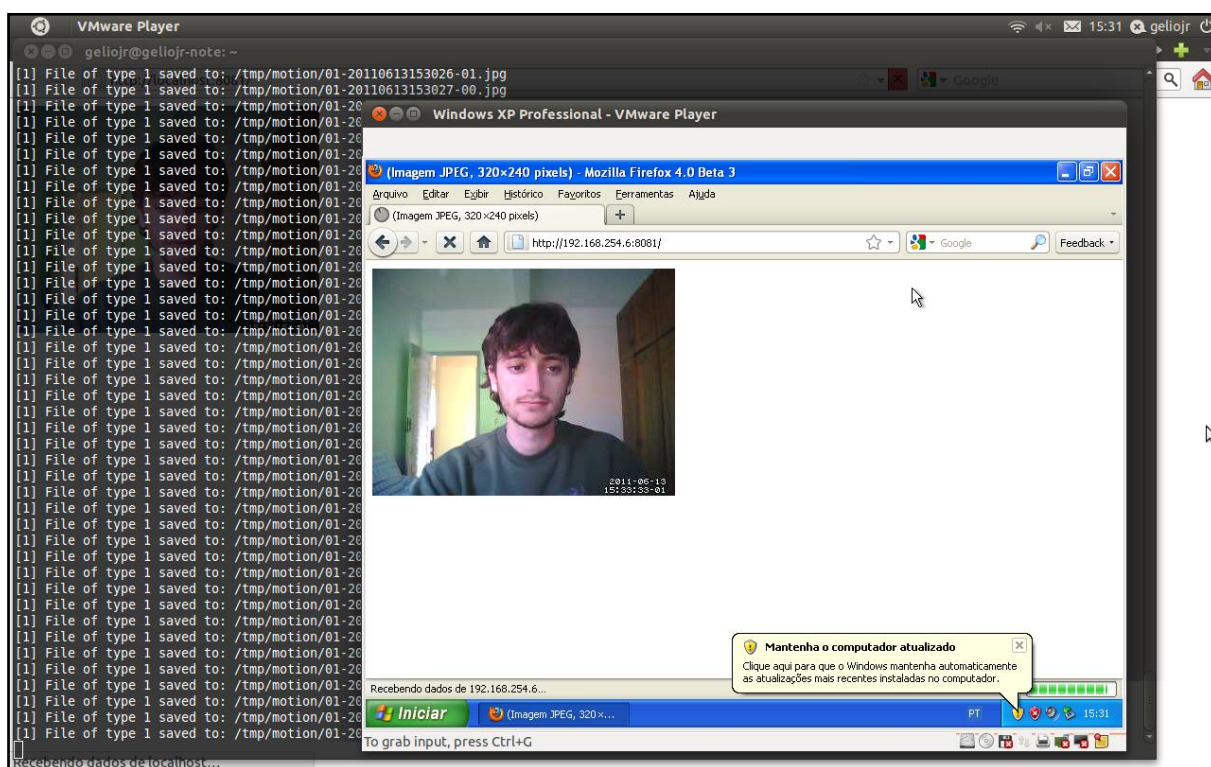


Figura 26. Máquina virtual acessando Motion

Em consulta aos grupos de discussões foi constatado que a VM do Ginga ainda suporta somente HTML pelo protocolo HTTP, não podendo assim ser acessado um vídeo por meio desse protocolo de comunicação. Com novas versões da implementação do *middleware* Ginga haverá a possibilidade de acessar outros tipos de mídias.

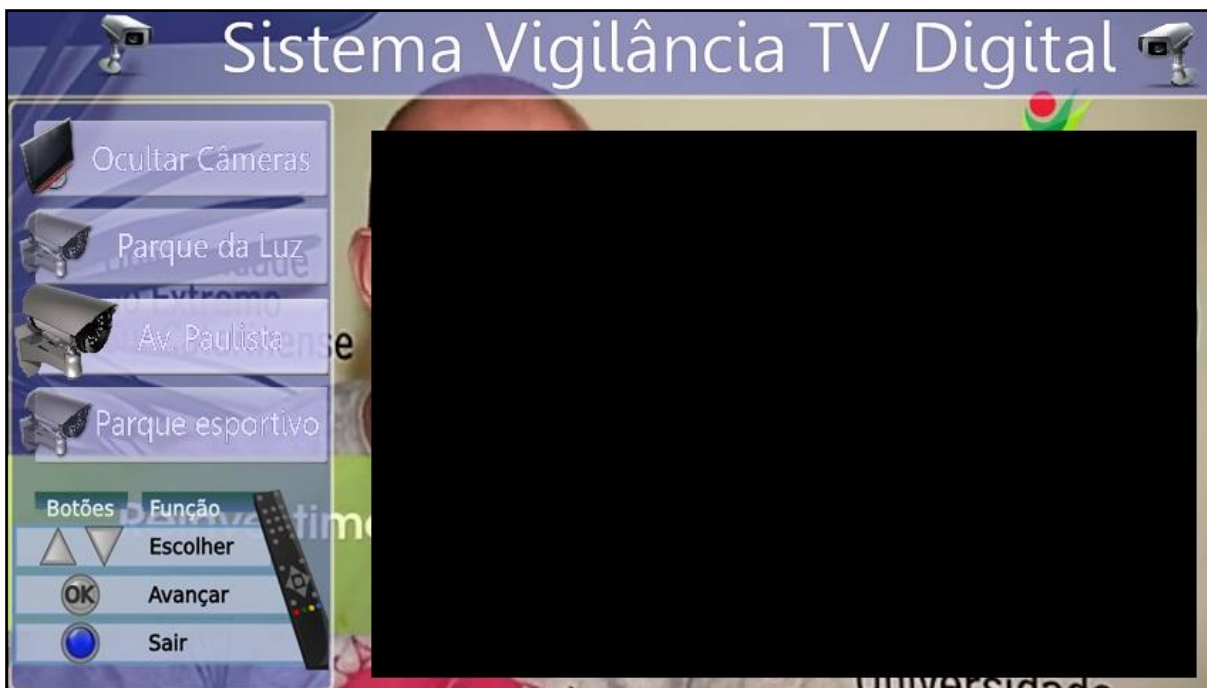


Figura 27. Ginga não consegue carregar vídeo

O *middleware* Ginga não consegue carregar nenhum tipo de mídia que não esteja na máquina local. Na Figura 27 têm-se a aplicação rodando, e foi selecionado a câmera 2, que estava com o *link* para acessar o vídeo de uma máquina externa. Fica só a tela preta e tentando carregar.

O Exibidor NCL GHTV³³ é um exibidor de aplicativos escritos em NCL. Ele foi testado durante todo o trabalho nas versões 0.4 e 0.4.1. Até a véspera da entrega desse trabalho ele estava na versão 0.4.1 e, sabia-se que iria ser lançada a versão 0.4.2 que daria suporte a transmissão de arquivos de mídias por meio do canal de interatividade pelo protocolo HTTP. Este iria suportar vídeos, imagens, áudio e HTML por meio do HTTP. Com esse exibidor será possível acessar os vídeos das câmeras de vigilância remotos e exibi-los na tela da televisão.

No dia da entrega desse trabalho foi lançada a versão 0.4.2 do GHTV, a qual realmente incorpora o recurso para acesso a mídias remotas. A mesma foi testada durante o dia todo para viabilizar a entrega do sistema plenamente operacional. Porém esse exibidor não

³³ Mais informações em <http://ghtv.com.br>

trata efeitos de transparência nas imagens, o que impossibilita de apenas executar o código NCL nele e funcionar perfeitamente.

Para disponibilizar os vídeos em uma máquina remota, foi necessário a instalação de um servidor web. Foi instalado o apache³⁴, onde os arquivos de vídeo ficam disponíveis pela porta 80.

³⁴Servidor web livre <http://www.apache.org/>

CONCLUSÃO

A pesquisa resultou no estudo da disponibilidade de imagens de câmeras de vigilância pelo Sistema Brasileiro de TV Digital. Verificou-se conceitos e características de câmeras de vigilância, Circuitos Fechados de TV, conceitos sobre o SBTVD, capacidades e inovações do padrão nipo-brasileiro de TV Digital.

Com isso foi possível a elaboração de um protótipo de software que disponibiliza a visualização de imagens de câmeras de vigilância na televisão de todo telespectador que tenha acesso ao sinal digital. Foram utilizadas ferramentas como o Eclipse e alguns plugins, a máquina virtual Ginga e o emulador de máquinas virtuais VMware Player.

Com o estudo, constatou-se que o *middleware* Ginga ainda apresenta limitações, faltando diversas implementações. Não se pode, ainda, devido a limitações de desenvolvimento do Ginga, acessar os vídeos de câmeras de vigilância por meio do canal de interatividade. Porém a aplicação os exibe, recebendo-os via *broadcast*.

O Exibidor NCL GHTV consegue acessar arquivos remotos por meio do canal de interatividade, mas pelo recente lançamento da versão com esse suporte não foi possível migrar a aplicação para o mesmo. Foram feitos apenas pequenos testes para confirmar sua veracidade de funcionamento, encontrando assim problemas com as transparências das imagens.

A aplicação resultante é bem intuitiva, característica fundamental, pois atende a todos os tipos de público. Tem uma interface amigável e simples, com transparência, visando ser ao máximo usual e discreta, para não atrapalhar a programação corrente.

Para a comunidade científica, o trabalho colabora identificando as limitações do protocolo HTTP do *middleware* brasileiro, documentando diversas características e conceitos do padrão do SBTVD e alguns detalhes do Exibidor NCL GHTV.

Deixa-se como possibilidade para trabalhos futuros, a evolução deste trabalho das seguintes formas:

- a) utilização do canal de interatividade: quando a máquina virtual do Ginga suportar o transporte de vídeos pelo canal de interatividade pode-se acessar as imagens por este canal, não sendo necessária a utilização da banda do carrossel de dados. Para isso é preciso apenas alterar o nome do arquivo que está relativo, colocando o endereço do servidor web que contém o vídeo especificado.
- b) uso de imagens de câmeras em tempo real: quando a máquina virtual do Ginga suportar o transporte de streaming, pode-se ter um servidor de streaming pegando as imagens de câmeras em tempo real e disponibilizando-as aos telespectadores. Para isso também é necessário colocar apenas o endereço do servidor quando se declara a mídia no código NCL.
- c) restrição de acesso: pode ser implementado para que apenas certas pessoas possam ter acessos a câmeras específicas. Para isso pode ser feito a implementação de senhas de acesso.
- d) prioridade de acesso: pode haver a necessidade de certas conexões terem prioridades sobre outras. Por exemplo, a polícia deve ter mais prioridade no acesso a imagens de câmeras que telespectadores comuns, então para isso é necessário haver controle no servidor. Isso ocorrerá somente usando o canal de interatividade.

REFERÊNCIAS

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. **NBR 6023**: informação e documentação: referências: elaboração. Rio de Janeiro, 2002.

_____. **NBR 10520**: informação e documentação: citações em documentos: apresentação. Rio de Janeiro, 2002.

_____. **NBR 15606:Middleware**. Rio de Janeiro, 2008. Disponível em: <http://www.dtv.org.br/download/pt-br/ABNTNBR15606_2D3_2007Vc2_2008.pdf>. Acesso em: abr. 2010.

BRASIL. **Decreto nº 4901**, de 26 de novembro de 2003. Disponível em: <<http://www.mc.gov.br/tv-digital/decreto-no-4901-de-26-de-novembro-de-2003/>>. Acesso em abr. 2010.

BORTOLOTTO, Franco Garcia. **Estudo sobre a televisão digital e desenvolvimento de um protótipo interativo baseado no subsistema procedural do Middleware ginga**. Criciúma, 2010.

CARVALHO, Ederson de Assis. **Projeto olho vivo: “A Iris dos olhos da segurança pública” Uma análise geográfica**. 100f. Dissertação (Mestrado) PUC-MG, Belo Horizonte, 2008. Disponível em: <http://www.biblioteca.pucminas.br/teses/TratInfEspacial_CarvalhoEA_1.pdf>. Acesso em: set. de 2010.

CARVALHO, Eduardo Rodrigues. **Uma plataforma modular para testes com interatividade na TV digital brasileira**. 105f. Dissertação (Mestrado) – USP, São Paulo, 2008. Disponível em: <http://www.teses.usp.br/teses/disponiveis/3/3142/tde-30052008-110812/publico/Dissertacao_Eduardo.pdf>. Acesso em: out. de 2010.

CORRÊA, Luisa; CUNHA, Maicon. **Câmeras no jornal: cartografando o discurso jornalístico sobre a vídeo-vigilância no Brasil**. Rio de Janeiro, 2009. Disponível em: <http://www2.pucpr.br/ssscla/papers/SessaoB_A11_pp92-111.pdf>. Acesso em: abr. de 2010.

CRUZ, Vitor Medina; MORENO, Marcio Ferreira; SOARES, Luiz Fernando Gomes. **TV Digital Para Dispositivos Portáteis- Middlewares**. Rio de Janeiro, 2008. Disponível em: <http://www.dbd.puc-rio.br/depto_informatica/08_03_cruz.pdf>. Acesso em: abr. de 2010.

DAMASCENO, Jean Ribeiro. **Middleware Ginga**. Niterói, 2008. Disponível em: <<http://www.midiacom.uff.br/~debora/fsmm/trab-2008-2/middleware.pdf>>. Acesso em: out de 2010.

FERNANDES, Jorge; LEMOS, Guido; SILVEIRA, Gledson. **Introdução à televisão digital interativa: arquitetura, protocolos, padrões e práticas**. Salvador, 2004. Disponível em: <<http://www.cic.unb.br/~jhcf/MyBooks/itvdi/texto/itvdi.pdf>>. Acesso em out. de 2010

FOUCAULT, Michel. **Vigiar e Punir**. Tradução Raquel Ramalhete. Petrópolis: Vozes, 2000. 262 p.

JANES, Ricardo. **Estudo sobre sistemas de segurança em instalações elétricas automatizadas**. 121f. Dissertação (Mestrado) – USP, São Paulo, 2009. Disponível em: <http://www.teses.usp.br/teses/disponiveis/3/3143/tde-29062009-181507/publico/Dissertacao_Ricardo_Janes.pdf>. Acesso em: set. de 2010.

KANASHIRO, Marta Mourão. **Surveillance Cameras in Brazil: exclusion, mobility regulation, and the new meanings of security**. Surveillance Studies Network, 2008. Disponível em: <[http://www.surveillance-and-society.org/articles5\(3\)/brazil.pdf](http://www.surveillance-and-society.org/articles5(3)/brazil.pdf)> Acesso em: abr. 2010.

LAMAS, Amilton C. et al. **Canal de Interatividade em TV Digital**. Cad. CPqD Tecnologia. Campinas, v.1, n.1, p. 29-36, jan./dez. 2005. Disponível em: <http://www.cpqd.com.br/cadernosdetecnologia/Vol1_N1_jan_dez_2005/pdf/artigo2_Lamas.pdf>. Acesso em: abr. 2010

MONTEZ, Carlos; BECKER, Valdecir. **TV Digital Interativa: conceitos, desafios e perspectivas para o Brasil**. Florianópolis: Ed. da UFSC, 2005. 2ª edição.

MOREIRA, Katia B. R. **Diretrizes para projeto de segurança patrimonial em edificações**. 186f. Dissertação (Mestrado) USP, São Paulo, 2007. Disponível em: <http://www.teses.usp.br/teses/disponiveis/16/16132/tde-17052010-090837/publico/Dissertacao_Katia_B_R_Moreira_FAUUSP_2007.pdf>. Acesso em: set. de 2010.

NETCOM. **CFTV digital**. Disponível em: <http://www.netcomam.com.br/serv_cftv_digital.html>. Acesso em set. de 2010.

SOARES NETO, C. S. et al. **Desenvolvimento de aplicações declarativas para TV digital interativa**. Rio de Janeiro, 2006. Disponível em: < <http://www.telemidia.puc-rio.br/sites/telemidia.puc-rio.br/files/MCAplDeclarativa.pdf> >. Acesso em: nov de 2010.

SOARES, Luiz Eduardo. **Segurança Pública**: presente e futuro. Estudos Avançados. Rio de Janeiro, 2006. Disponível em: <<http://www.scielo.br/pdf/ea/v20n56/28629.pdf>>. Acesso em: abr. 2010.

SOARES, Luiz Fernando Gomes. **As múltiplas possibilidades do *middleware* Ginga**. São Paulo, 2008. Disponível em: <<http://www.producaoprofissional.com.br/files/editions/ed1216040609.pdf>>. Acesso em: abr. 2010.

SOARES, L. F. G; BARBOSA, S. D. J. **Programando em NCL 3.0**: Desenvolvimento de aplicações para o *middleware* Ginga. Rio de Janeiro: Ed. Campus, 2009.

TONIETO, Marcia Terezinha. Sistema Brasileiro de TV digital – SBTVD Uma análise política e tecnológica na inclusão digital. 267f. Dissertação (Mestrado) - UECE/CEFET, Fortaleza, 2006. Disponível em: <<http://mpcomp.pgcomp.uece.br/admin/arquivos/MarciaTonieto2006.PDF>>. Acesso em: out. de 2010.

ZIMERMANN, Filipi. **Canal de retorno em TV Digital**: Tecnologias e abordagens para efetivação da interatividade televisiva. Florianópolis, 2007. Disponível em: <http://projetos.inf.ufsc.br/arquivos_projetos/projeto_617/canal_retorno__artigo__filipi_zimmermann.pdf>. Acesso em abr. 2010.

APÊNDICE A: Log erro execução do Canal de Interatividade pelo Ginga

```
root@gingavm:~# /misc/launcher.sh tcc/2/main.ncl
```

```
argv = 'tcc/2/main.ncl' nclFile = 'tcc/2/main.ncl'
```

```
ginga main() NCLFILE = /root/tcc/2/main.ncl
```

```
commandline read: ginga
```

```
~~~~~| DirectFB 1.4.11 |~~~~~
```

```
(c) 2001-2010 The world wide DirectFB Open Source Community
```

```
(c) 2000-2004 Convergence (integrated media) GmbH
```

```
-----
```

```
(* DirectFB/Core: Single Application Core. (2010-12-01 00:03)
```

```
(* Direct/Thread: Started 'VT Switcher' (-1) [CRITICAL OTHER/OTHER 0/0] <838860
```

```
8>...
```

```
(* Direct/Thread: Started 'VT Flusher' (-1) [DEFAULT OTHER/OTHER 0/0] <8388608>
```

```
...
```

```
(* DirectFB/FBDev: Found 'VESA VGA' (ID 0) with frame buffer at 0xd0000000, 166
```

```
4k (MMIO 0x00000000, 0k)
```

```
(* Direct/Thread: Started 'Linux Input' (-1) [INPUT OTHER/OTHER 0/0] <8388608>.
```

```
..
```

```
(* DirectFB/Input: Power Button (1) 0.1 (directfb.org)
```

```
(* Direct/Thread: Started 'Linux Input' (-1) [INPUT OTHER/OTHER 0/0] <8388608>.
```

```
..
```

```
(* DirectFB/Input: AT Translated Set 2 keyboard (2) 0.1 (directfb.org)
```

(*) Direct/Thread: Started 'Linux Input' (-1) [INPUT OTHER/OTHER 0/0] <8388608>.

..

(*) DirectFB/Input: ImPS/2 Generic Wheel Mouse (3) 0.1 (directfb.org)

(*) Direct/Thread: Started 'Hotplug with Linux Input' (-1) [INPUT OTHER/OTHER 0/0] <8388608>...

(*) DirectFB/Input: Hot-plug detection enabled with Linux Input Driver

(*) Direct/Thread: Started 'PS/2 Input' (-1) [INPUT OTHER/OTHER 0/0] <8388608>..

.

(*) DirectFB/Input: IMPS/2 Mouse 1.0 (directfb.org)

(*) Direct/Thread: Started 'Keyboard Input' (-1) [INPUT OTHER/OTHER 0/0] <8388608>...

(*) DirectFB/Input: Keyboard 0.9 (directfb.org)

(*) DirectFB/Genefx: MMX detected and enabled

(*) DirectFB/Graphics: MMX Software Rasterizer 0.6 (directfb.org)

(*) DirectFB/Core/WM: Default 0.3 (directfb.org)

(*) FBDev/Mode: Setting 854x480 RGB32

(*) FBDev/Mode: Switched to 854x480 (virtual 854x480) at 32 bit (RGB32), pitch 3424

(#) DirectFBError [gingacc-systemio DFBDeviceScreen TestConfig error:]: Not supported!

InputManager::initializeInputIntervalTime imperative = '50' declarative = '200'

ginga main() NCLFILE = /root/tcc/2/main.ncl

PresentationEngineManager::openNclFile document process time: 0

(*) Direct/Interface: Loaded 'PNG' implementation of 'IDirectFBImageProvider'.

ComponentManager::releaseComponentFromObject 'ImageProvider'

ContextManager::addContextListener

SystemInfo::getSystemLanguage can't find 'system.language, return por

BroadcastSocketService::discoverBroadcastAddress interfaceName: 'eth0

RemoteEventService::new RemoteEventService()

PrivateBaseContext::createPrivateBase '-1'

FormatterMediator::printData(0x8e549e0) baseId: -1 playerId: /root/tcc/2/main.nc

l devClass: 0 x: 0 y: 0 w: 854 h: 480 enableGfx: 1 parentDocId: docId:

FormatterMediator::addDocument(0x8e549e0) call addDocument '/root/tcc/2/main.ncl

,

TimeStamp: 3.956

ComponentManager::releaseComponentFromObject 'NclDocumentConverter'

PrivateBaseContext::addDocument id='ProjetoTVDi' url='/root/tcc/2/main.ncl' all

done

/misc/launcher.sh: line 14: 1018 Segmentation fault /usr/local/sbin/ginga

--ncl \$1

~~~~~| DirectFB 1.4.11 |~~~~~

(c) 2001-2010 The world wide DirectFB Open Source Community

(c) 2000-2004 Convergence (integrated media) GmbH

-----

(\* DirectFB/Core: Single Application Core. (2010-12-01 00:03)

(\* Direct/Thread: Started 'VT Switcher' (-1) [CRITICAL OTHER/OTHER 0/0] <838860

8>...

(\* Direct/Thread: Started 'VT Flusher' (-1) [DEFAULT OTHER/OTHER 0/0] <8388608>

...

(\* DirectFB/FBDev: Found 'VESA VGA' (ID 0) with frame buffer at 0xd0000000, 166  
4k (MMIO 0x00000000, 0k)

(\* Direct/Thread: Started 'Linux Input' (-1) [INPUT OTHER/OTHER 0/0] <8388608>.

..

(\* DirectFB/Input: Power Button (1) 0.1 (directfb.org)

(\* Direct/Thread: Started 'Linux Input' (-1) [INPUT OTHER/OTHER 0/0] <8388608>.

..

(\* DirectFB/Input: AT Translated Set 2 keyboard (2) 0.1 (directfb.org)

(\* Direct/Thread: Started 'Linux Input' (-1) [INPUT OTHER/OTHER 0/0] <8388608>.

..

(\* DirectFB/Input: ImPS/2 Generic Wheel Mouse (3) 0.1 (directfb.org)

(\* Direct/Thread: Started 'Hotplug with Linux Input' (-1) [INPUT OTHER/OTHER 0/  
0] <8388608>...

(\* DirectFB/Input: Hot-plug detection enabled with Linux Input Driver

(\* Direct/Thread: Started 'PS/2 Input' (-1) [INPUT OTHER/OTHER 0/0] <8388608>..

.

(\* DirectFB/Input: IMPS/2 Mouse 1.0 (directfb.org)

(\* Direct/Thread: Started 'Keyboard Input' (-1) [INPUT OTHER/OTHER 0/0] <838860  
8>...

(\* DirectFB/Input: Keyboard 0.9 (directfb.org)

(\* DirectFB/Genefx: MMX detected and enabled

(\* DirectFB/Graphics: MMX Software Rasterizer 0.6 (directfb.org)

(\* DirectFB/Core/WM: Default 0.3 (directfb.org)

(\* Direct/Interface: Loaded 'PNG' implementation of 'IDirectFBImageProvider'.

(\*) Direct/Interface: Loaded 'FT2' implementation of 'IDirectFBFont'.

(\*) FBDev/Mode: Setting 854x480 RGB32

(\*) FBDev/Mode: Switched to 854x480 (virtual 854x480) at 32 bit (RGB32), pitch 3

424

root@gingavm:~#

## **ANEXO A: Como Estruturar Seu Ambiente de Desenvolvimento para o Ginga-NCL**

É muito recompensador desenvolver uma aplicação para TV digital e observar sua criação sendo executada num set-top box e exibida na TV, porém quem deseja começar a desenvolver seus primeiros aplicativos pode encontrar algumas dificuldades. No primeiro contato com o desenvolvimento para TV digital geralmente o iniciante acaba deparando-se com muitas informações sobre normas, plataformas, ambiente de testes, APIs, usabilidade etc. Tantas informações podem causar confusão deixando o desenvolvedor perdido, e pior, podem até causar a desistência e conseqüente decepção com o desenvolvimento de aplicações para TV digital.

Neste tutorial abordaremos passo a passo a estruturação de um ambiente de desenvolvimento e testes para as aplicações interativas desenvolvidas para o middleware Ginga-NCL.

### **Ambiente de desenvolvimento**

Quando falamos em ambiente de desenvolvimento estamos nos referindo ao conjunto de ferramentas que tem por finalidade auxiliar tanto na codificação como na execução e visualização de nossas aplicações interativas.

Para codificarmos nossas aplicações iremos utilizar uma IDE muito popular conhecida como Eclipse. O uso de uma IDE é bastante recomendado, pois, como veremos no decorrer deste tutorial, ela oferece uma série de funcionalidades, como um sistema avançado de plugins, para facilitar a codificação de nossas aplicações.

Para simularmos uma TV com interatividade iremos utilizar uma imagem VMware de uma máquina virtual com o middleware Ginga-NCL instalado. Esta imagem é

chamada de Ginga-NCL Virtual STB e é disponibilizada pelo laboratório Telemídia da PUC-Rio.

## Download e instalação do Ginga-NCL Virtual STB

Para executarmos o Ginga-NCL Virtual STB é necessário ter instalado um player para a máquina virtual VMware. Existem duas opções gratuitas: VMware Server<sup>35</sup> e VMware Player.

Pela facilidade de instalação e entendimento, prosseguiremos com a instalação do VMware Player. Acesse a página de download<sup>36</sup> e escolha a melhor opção de acordo com o seu sistema operacional. Será necessário efetuar um pequeno cadastro para prosseguir.

Após a conclusão do download (tanto no Windows como no GNU/Linux) basta executar o arquivo baixado para iniciar uma instalação em modo guiado. As opções apresentadas são fáceis de escolher e a para a maioria delas é necessário apenas clicar em/ou escolher Next. Ao término da instalação pode ser solicitado a reinicialização do sistema. A figura abaixo mostra a tela inicial do VMware Player onde é possível ver o botão Open an existing Virtual Machine.



<sup>35</sup> [http://downloads.vmware.com/d/info/datacenter\\_downloads/vmware\\_server/2\\_0](http://downloads.vmware.com/d/info/datacenter_downloads/vmware_server/2_0)

<sup>36</sup> [http://downloads.vmware.com/d/info/desktop\\_downloads/vmware\\_player/3\\_0](http://downloads.vmware.com/d/info/desktop_downloads/vmware_player/3_0)

Após a instalação do VMware Player seguiremos com o download da imagem do Ginga-NCL Virtual STB. Para isso acesse a página de ferramentas<sup>37</sup> do site oficial. A imagem é disponibilizada num arquivo compactado (.zip) e para extraí-lo será necessário ter um descompactador como unzip<sup>38</sup> ou winzip<sup>39</sup> instalado em seu computador. Quando o download terminar, descompacte o arquivo onde desejar.

Para executar o Ginga-NCL Virtual STB basta abrir o VMware Player, clicar no botão Open an existing Virtual Machine, navegar até a pasta onde descompactou a imagem e clicar no arquivo .vmx. A figura abaixo ilustra essa operação.



Como é exibido na figura abaixo, após o carregamento da imagem teremos nosso Ginga-NCL Virtual STB pronto para os testes.



---

<sup>37</sup> <http://www.ncl.org.br/ferramentas/ubuntu-server10.10-ginga-i386.zip>

<sup>38</sup> <http://linux.die.net/man/1/unzip>

<sup>39</sup> <http://www.winzip.com/>

Agora já é possível acessar o Ginga-NCL Virtual STB através de uma conexão SSH. No Windows para abrirmos tal conexão com a máquina virtual podemos fazer uso dos programas Putty ou SSH Secure Shell Client; a maioria das distribuições GNU/Linux já oferece o cliente SSH<sup>40</sup> instalado. Contudo não iremos trabalhar com o Ginga-NCL Virtual STB desta forma, apresentaremos uma maneira mais fácil para trabalharmos com a máquina virtual.

### **Instalação do Eclipse**

Neste tutorial utilizaremos o Eclipse 3.5.2 (Galileo) Classic que pode ser obtido, gratuitamente, em sua página de download<sup>41</sup>. O Eclipse é desenvolvido em Java e não é necessário efetuar instalação, basta descompactar e executar o arquivo binário para iniciar sua utilização. Contudo é necessário ter a máquina virtual Java instalada. Na figura abaixo pode ser visto uma tela do Eclipse informando a versão que utilizamos.



---

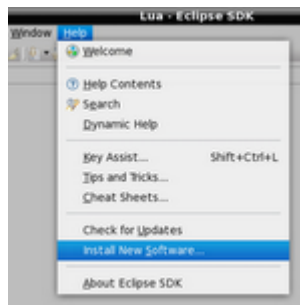
<sup>40</sup> <http://unixhelp.ed.ac.uk/CGI/man-cgi?ssh+1>

<sup>41</sup> <http://www.eclipse.org/downloads/>

## Instalação do plugin NCL Eclipse

O NCL Eclipse<sup>42</sup> é um plugin que auxilia e agiliza bastante o desenvolvimento de aplicações em NCL. A partir da versão 1.4 (a versão atual é a 1.5) foi disponibilizada a instalação através do sistema de instalação automática do Eclipse.

Para efetuar a instalação do NCL Eclipse inicie o Eclipse e acesse Help -> Install New Software como pode ser visualizado na figura abaixo.



Na tela de instalação (figura abaixo) clique em Add.



Após clicar em Add será exibida uma caixa de diálogo para a informação do nome e localização do site onde o Eclipse irá buscar a atualização. Entre com as informações:

Name: NCL Eclipse

Location: <http://www.laws.deinf.ufma.br/ncleclipse/update>

<sup>42</sup> <http://laws.deinf.ufma.br/~ncleclipse>

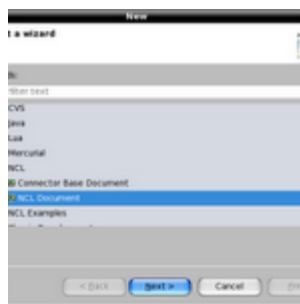


Clique em OK e o Eclipse irá procurar por atualizações no endereço informado. Para visualizar o NCL Eclipse desmarque a opção “Group items by category”. Escolha o NCL Eclipse, clique em Next e depois em Finish.



Após a instalação será solicitado a reinicialização do Eclipse, basta aceitar e aguardar a inicialização automática. Quando iniciar novamente o Eclipse já estará com o plugin NCL Eclipse instalado.

Para criar um novo documento NCL clique em File -> New -> Other ou utilize o atalho Ctrl+N. Na janela New escolha NCL -> NCL Document e clique em Next.



Como último passo iremos configurar o nome de nosso documento e clicar em Finish para criarmos o documento.



## Instalação do Remote System Explorer (RSE)

O RSE é um plugin para o Eclipse que oferece um conjunto de ferramentas para a conexão e trabalho com diferentes sistemas remotos, incluindo SSH e FTP. Para maiores detalhes sobre o plugin visite o site oficial<sup>43</sup>.

Mas o que um plugin para trabalho com sistemas remotos está fazendo aqui? Bem, podemos considerar o nosso Ginga-NCL Virtual STB um sistema remoto, apesar de muitas vezes ele residir no mesmo computador, e usufruir todas as facilidades oferecidas pelo RSE evitando o enfadonho processo de edição, cópia para a máquina virtual e acesso à máquina virtual para executar a aplicação. Quem passou por isso sabe o quanto isso pode ser desestimulante.

A instalação do RSE segue o mesmo procedimento dos outros plugins, Apenas substitua as informações do site por:

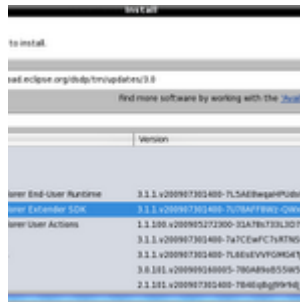
Name: RSE

Location: <http://download.eclipse.org/dsdp/tm/updates/3.0>

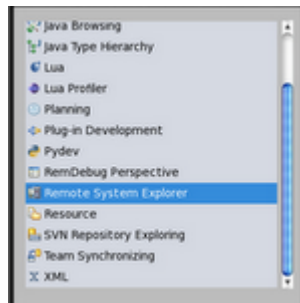
Neste site encontraremos muitas versões deste plugin, escolha a opção Remote System Explorer SDK 3.1.1 como na figura abaixo.

---

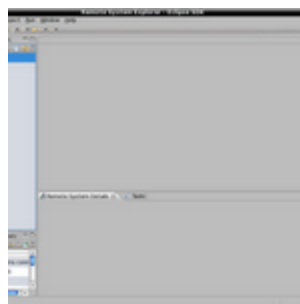
<sup>43</sup> <http://www.eclipse.org/dsdp/tm>



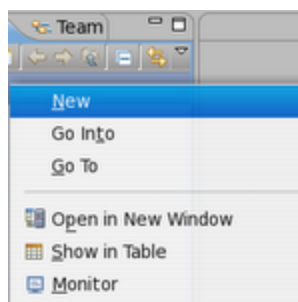
Após a instalação, podemos modificar o nosso workspace para a perspectiva oferecida pelo RSE, para isso acesse Window -> Open Perspective -> Other e escolha Remote System Explorer como na figura.



Com a nova perspectiva o workspace ficará parecido com este:



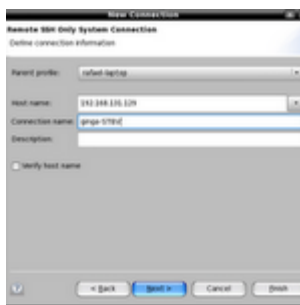
Agora iremos criar uma conexão com o Ginga-NCL Virtual STB (assegure-se que ele está rodando). Clique com o botão direito na aba Remote Systems e escolha New -> Connection. A figura abaixo ilustra essa operação.



Em seguida será necessário escolher o tipo de conexão com o sistema remoto, escolha SSH Only.



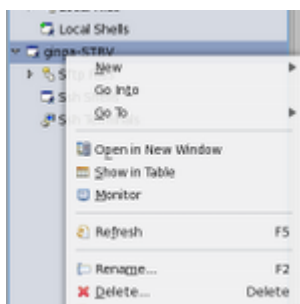
Após escolher o tipo de conexão é necessário configurar o Host name e atribuir um nome para a conexão, como pode ser visto na figura abaixo.



Na configuração do Host name coloque o endereço IP do seu Ginga-NCL Virtual STB.

Logo em seguida clique em Finish para criar sua conexão.

Para nos conectarmos ao Ginga-NCL Virtual STB clique com o botão direito sobre a conexão que acabou de criar e escolha a opção Connect.

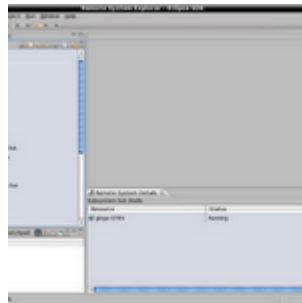


Em seguida será exibida a tela para o preenchimento do login e senha para a conexão. Seguindo as instruções presentes na tela de abertura do Ginga-NCL Virtual STB preencha com o login: root e senha: telemidia.



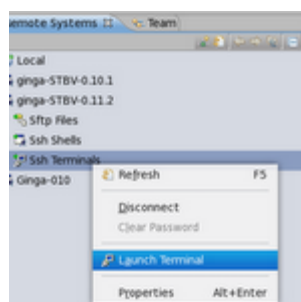
Para otimizar novas conexões marque a caixa Save password

Com a conexão estabelecida é possível ter acesso aos arquivos do Ginga-NCL Virtual STB como exibido na figura abaixo.



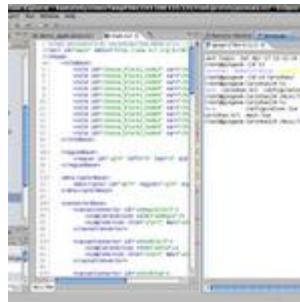
É possível executar ações, inclusive edição, sobre os arquivos remotos como se fossem arquivos locais; o RSE abstrai isso para o desenvolvedor. Experimente dar um duplo clique em um dos arquivos e você verá que ele abrirá para edição como qualquer outro arquivo e todas as modificações que você realizar são efetuadas diretamente no arquivo remoto. Também é possível criar diretórios e transferir arquivos, tudo da forma que você já faz no Eclipse.

Agora abriremos um terminal para execução de comandos diretamente no Ginga-NCL Virtual STB. Clique com o botão direito sobre Ssh Terminals e escolha Launch Terminal.



Com a utilização do terminal é possível executarmos nossa aplicação sem sair do Eclipse.

Na figura abaixo é possível observar uma tela do Eclipse com o ambiente integrado. O arquivo aberto, no centro da tela, é de uma aplicação de demonstração e está sendo editada remotamente. Do lado direito temos um terminal com o comando para a execução da aplicação que será exibida na tela do Ginga-NCL Virtual STB.



O RSE é um plugin muito abrangente e apresenta mais funcionalidades do que as apresentadas aqui. Se você utiliza alguma funcionalidade que não foi apresentada, contribua com comentários.

# Protótipo de Visualização de Imagens de Câmeras de Vigilância por meio do Subsistema Ginga-NCL do Sistema Brasileiro de TV Digital

Gélio José da Silva Júnior<sup>1</sup>, Paulo João Martins<sup>2</sup>

<sup>1</sup>Acadêmico do Curso de Ciência da Computação – Departamento de Ciência da Computação – Universidade do Extremo Sul Catarinense (UNESC) – Criciúma, SC - Brazil

<sup>2</sup>Professor do Curso de Ciência da Computação – Departamento de Ciência da Computação – Universidade do Extremo Sul Catarinense (UNESC) – Criciúma, SC - Brazil

geliojr@gmail.com, pjm@unesc.net

**Abstract.** *This paper describes the conclusion work submitted to obtain the degree of bachelor in Computer Science at UNESC, whose goal was to develop a prototype for viewing images from surveillance cameras for digital TV using the subsystem Ginga-NCL Brazilian Digital Television System. To carry out the same, we performed a literature search on surveillance and digital TV as well as application development for the SBTVD at programming language NCL.*

**Resumo.** *O presente artigo descreve o trabalho de conclusão de curso apresentado para obtenção do grau de Bacharel em Ciência da Computação da Universidade do Extremo Sul Catarinense, cujo objetivo foi desenvolver um protótipo para visualização de imagens de câmeras de vigilância pela TV digital, utilizando o subsistema Ginga-NCL do Sistema Brasileiro de TV Digital. Para a realização do mesmo, foi realizada uma pesquisa bibliográfica sobre vigilância e TV digital, bem como o desenvolvimento do aplicativo para o SBTVD na linguagem de programação NCL.*

## 1. Introdução

A criminalidade e suas manifestações estão presentes em todas as regiões do Brasil. Devido à sua extensão territorial e pela diversidade cultural não há uma solução uniforme que atenda em todas as áreas o problema de segurança (SOARES, 2006). Porém, uma prática aplicada atualmente são as câmeras de vigilância, que estão espalhadas por vários locais devido ao baixo custo de instalação e manutenção.

A videovigilância é entendida como sinônimo de segurança, principalmente quando são denominadas “Câmeras de Segurança”. Essa associação de conceitos é constituída sócio-historicamente fazendo um recuo à modernidade descrita por Foucault (2000), onde dizia que “ser vigiado” significava “estar seguro”. A visão atual para a alta demanda desse tipo de sistema dá-se ao fato que a vigilância não objetiva garantir a segurança da população vigiada, mas garantir que os mesmos se adéquem as normas e valores, podendo ser punidos caso isso não ocorra (CORREA; CUNHA, 2009).

No Brasil, os projetos de instalação desses sistemas têm se multiplicado, tanto em ambientes públicos quanto em ambientes particulares. Um exemplo bem conhecido são as câmeras do Parque da Luz em São Paulo, que é vigiado dia e noite pela polícia (KANASHIRO, 2008).

As imagens geradas podem ser monitoradas em tempo real e/ou armazenadas para visualização posterior. A maioria dos sistemas é feito em Circuito Fechado de TV (CFTV)

onde o acesso é feito somente local, porém há a possibilidade de disponibilizar essas imagens remotamente, onde as imagens poderiam ser vistas de qualquer local com acesso a *Internet*. Isso já está altamente difundido nos sistemas privados de vigilância, mas não no ambiente público.

A cada dia que passa surge novas tecnologias que podem ser exploradas na área de acesso remoto. Uma delas é o Sistema Brasileiro de Televisão Digital (SBTVD) que foi instituído pelo governo federal brasileiro por meio do Decreto Presidencial 4.901 de 26 de novembro de 2003, com o objetivo principal de implantar uma plataforma tecnológica digital baseada em televisão que promova a inclusão social por meio do acesso à informação.

Com essa nova tecnologia, tem-se a capacidade de interagir com a TV digital por meio do padrão de *middleware* estabelecido, o Ginga (NBR 15606, 2008). Para haver uma interatividade entre ambos, há a necessidade de um meio de comunicação fazendo o caminho inverso do sinal (SOARES, 2008). Uma funcionalidade marcante é a capacidade de utilizar as teclas play/pause na programação corrente. Dependendo do nível de interação, é possível ter as funções básicas de um computador conectado a Internet (LAMAS et al, 2005). Por meio desse nível pode-se criar uma conexão entre um sistema de segurança a uma TV digital.

Com sua alta difusão e com um cenário de interação permanente, este trabalho propõe ajudar na segurança, seja empresarial ou domiciliar, pública ou privada, obtendo acesso a imagens de câmeras de vigilância utilizando o canal de interatividade, propiciando que seja acessado a qualquer hora (24/7) de qualquer ambiente que tenha os recursos necessários.

## **2. Segurança**

A violência existente nas cidades está em constante crescimento. Para conter essa criminalidade há a necessidade de controle, com isso o crescimento e inovação dos sistemas para proteção são de suma importância para a sociedade. Esses sistemas têm a finalidade de proteger pessoas e patrimônios (MOREIRA, 2007).

Mesmo que a maior responsabilidade pela segurança pública seja do Estado, a população e empresas podem e devem ajudar, montando seus próprios sistemas de proteção.

Geralmente há algum tipo desses sistemas em todos os locais do cotidiano de pessoas. Existem para proteção de invasões externas ou de pessoas não autorizadas. Essa preocupação existe desde as primeiras cidades construídas pela humanidade, onde se construía muros altos e poços de água para defesa de quem se encontrava dentro de determinado ambiente.

Porém, mesmo com o passar do tempo e com diversas formas para combater o crime, sempre há uma parte da população que é voltada para a criminalidade, sendo influenciados por diversos fatores, como guerras e pobreza. O comportamento dessas pessoas é altamente influenciado ao quanto de violência esta pessoa possa estar sendo submetida (MOREIRA, 2007).

As técnicas de proteção vão evoluindo para prevenir e proteger pessoas e locais da criminalidade a qual possam estar expostas. As novas tecnologias também vêm ao encontro da segurança, sendo usadas como aliada no combate ao crime.

## 2.1. Segurança Viglada

Dentre as várias tecnologias existentes em prol da segurança, as câmeras de vigilância são utilizadas em larga escala. Elas estão espalhadas em vários lugares, tanto públicos quanto particulares.

Elas têm como característica atribuir sensação de segurança aos locais em que estão aplicadas. Como descreveu Foucault (2000) estar sendo vigiado significa estar seguro. Por estarem sendo vigiados, indivíduos têm receio em praticar atos ilícitos, sendo forçados a praticar as normas e valores pré-definidos pela sociedade e pelas leis vigentes (CORREA; CUNHA, 2009).

As câmeras, denominadas popularmente como de segurança, possibilitam que informações sejam adquiridas por pessoas ou equipamentos a distância. Esses podem tomar atitudes ou decisões para prevenir ou providenciar algo na hora em que está ocorrendo o ato, entre outras serventias, que serão detalhadas no próximo subcapítulo.

## 2.2. Projetos Públicos

Kanashiro (2008) comenta que no mundo inteiro a instalação de câmeras de vigilância nas ruas das cidades é cada vez mais comum. E no Brasil não é diferente, várias cidades estão implantando esses sistemas de videovigilância.

A pioneira nesse tipo de projeto foi Praia Grande / SP, instalando uma interligação entre as repartições públicas da cidade. Esta rede, conhecida como Infovia<sup>44</sup>, possibilitou a instalação de diversas câmeras que vigiam os locais com maior movimento, prédios públicos e monumentos (CARVALHO, E. A. 2008).

A necessidade de monitoração desses locais existe principalmente devido à segurança de bens materiais e de pessoas. As imagens dessas câmeras também são utilizadas como objetos de investigações policiais e como provas de crimes.

Segundo Carvalho (2008) os índices de ocorrências policiais e de depredações em equipamentos públicos municipais diminuíram consideravelmente nas cidades onde foram implantados os sistemas de videovigilância.

Outro projeto conhecido nessa área é o Projeto Olho Vivo<sup>45</sup> da cidade de Belo Horizonte / MG. Um sistema de vigilância digital que tem uma central de monitoramento em tempo real e com alta definição. Esse sistema conta com profissionais capacitados, onde cada um é responsável por quatro câmeras, podendo visualizar a região com mais detalhes, graças à capacidade de poder controlá-las. O projeto opera de forma plena desde 2004.

## 3. TV Digital

A TV, como qualquer outro meio de comunicação, está em constante evolução e adaptação às novas tecnologias e necessidades sociais. Muitas mudanças ocorreram desde o surgimento dos primeiros canais de televisão. A cor deixou a televisão muito mais elegante e atraente e aumentou o número de canais, podendo o telespectador ter escolhas entre o que assistir.

Em um sistema televisivo existem basicamente dois cenários: quem emite os conteúdos e quem os assiste, Soares e Barbosa (2009) fazem uma analogia ao sistema de cliente e servidor. O primeiro cenário é quem desenvolve os programas e conteúdos que serão

---

<sup>44</sup> Mais informações em: [http://www.unicamp.br/unicamp/unicamp\\_hoje/jornalPDF/ju362pag04.pdf](http://www.unicamp.br/unicamp/unicamp_hoje/jornalPDF/ju362pag04.pdf)

<sup>45</sup> Mais informações em: [http://www.biblioteca.pucminas.br/teses/TratInfEspacial\\_CarvalhoEA\\_1.pdf](http://www.biblioteca.pucminas.br/teses/TratInfEspacial_CarvalhoEA_1.pdf)

transmitidos para os telespectadores/clientes (CRUZ; MORENO; SOARES, 2008). A forma como é transmitido esse conteúdo está em uma fase de transição, do analógico para o digital. Essa mudança se dá na digitalização da transmissão do sinal, trazendo a possibilidade de tratamento dos dados transmitidos. A primeira mudança notória entre esses dois sistemas é a qualidade de imagem e de som (SOARES; BARBOSA, 2009).

### **3.1. Sistema Brasileiro de TV Digital**

A evolução do sistema de televisão analógico para o digital está ocorrendo em diversos países do mundo. Para isso é necessário aderir a um padrão ou criar seu próprio sistema de transmissão (MONTEZ; BECKER, 2005).

As pesquisas para a implantação de TV digital no Brasil iniciaram em 1994, onde foram estudados e testados os padrões que já estavam em funcionamento. Segundo Tonieto (2006), em meados de 2000 encerrou-se a discussão técnica sobre o padrão de TV Digital a ser adotado pelo Brasil.

No Brasil foram feitos testes com os três padrões internacionais existentes. Dentre os padrões ASTC – americano, DVB – europeu e o ISDB – japonês, os que obtiveram melhores desempenhos foram os padrões Europeu e Japonês. Pesquisas de técnicos brasileiros resultaram na escolha do padrão japonês, incorporando tecnologias desenvolvidas por diversos grupos de pesquisas aqui no Brasil. O modelo ISDB-T também foi considerado superior em relação ao desempenho na recepção do sinal em ambientes fechados e por terminais fixos ou móveis.

Assim, surge o Sistema Brasileiro de TV Digital (SBTVD) que é um padrão nipo-brasileiro, com base japonesa e adaptado às condições e necessidades brasileiras, integrando uma nova arquitetura com canal de interatividade.

O SBTVD está muito bem estruturado, e como consequência, vários países já aderiram ao padrão brasileiro de TV digital e outros estão estudando a possibilidade de adoção. Segundo Montez e Becker (2005) esse tipo de evolução acontece de forma lenta e gradativa, não substituindo a tecnologia antiga de forma repentina, porém é uma mudança que está ocorrendo de forma global.

Por ser um sistema altamente complexo, o SBTVD tem uma arquitetura de referência dividida em camadas. Cada uma é composta por uma arquitetura de hardwares e softwares com a finalidade de oferecer serviços padronizados para a camada superior e utilizar os serviços oferecidos pela camada inferior (FERNADES; LEMOS; SILVEIRA, 2004).

### **3.2. Ginga**

O middleware é uma camada de software existente entre o Hardware/SO e os aplicativos de TV digital. Tem como finalidade oferecer diversas facilidades para o desenvolvimento de conteúdo e aplicativos, tornando transparentes as camadas de comunicação e transporte, fazendo com que aplicativos rodem da mesma maneira, independente de hardware ou sistema operacional (SO) (DAMASCENO, 2008).

Tonieto (2006) diz que é muito importante a portabilidade em sistemas de TV digital, pois haverá diferentes configurações de receptores para executar as aplicações e o responsável por abstrair essas diferenças é o middleware Ginga. Ele é resultado de anos de pesquisas feitas pelo laboratório TeleMídia<sup>46</sup> da Pontifícia Universidade Católica do Rio de

---

<sup>46</sup> Mais informações em: <http://www.telemidia.puc-rio.br/>

Janeiro (PUC-Rio) e pelo Laboratório de Aplicações de Vídeo Digital (LAVID)<sup>47</sup> da Universidade Federal da Paraíba (UFPB). Ginga agrupa diversas tecnologias e inovações que o tornam muito avançado e adequado à realidade do país.

Segundo Soares (2008) a arquitetura do Ginga pode ser dividida em três módulos principais: Ginga-CC, Ginga-NCL e Ginga-J.

O Ginga é dividido em duas classes de aplicações, as declarativas e as procedurais. O subsistema lógico que disponibiliza funcionalidades comuns entre elas é o Ginga Common-Core (Ginga-CC) (CRUZ; MORENO; SOARES, 2008).

O Ginga-NCL é o subsistema lógico responsável pelo processamento das aplicações declarativas do middleware do SBTVD, desenvolvidas na linguagem de programação Nested Context Language (NCL) que será melhor explicada no item 4.2. Esse subsistema é obrigatório nos receptores portáteis e fixos.

O Ginga-J é o subsistema lógico responsável pelo processamento das aplicações imperativas desenvolvidas na linguagem de programação Java. Ele é obrigatório apenas nos receptores fixos. O desenvolvimento da parte Java do Ginga é liderado pelo grupo de pesquisa do LAVID.

## **4. Sistema Vigilância TV Digital**

Foi proposta a elaboração de um software que disponibilize as imagens de determinadas câmeras de vigilância pelo Sistema Brasileiro de TV Digital Terrestre. O protótipo será desenvolvido no ambiente declarativo do middleware Ginga, utilizando a linguagem NCL.

A aplicação para o SBTVD trará para o telespectador a possibilidade de saber, por exemplo, como está o movimento ou o tempo em diversos locais da região, onde poderá selecionar a câmera que deseja assistir.

### **4.1. Metodologia**

No início do trabalho, foi realizado todo o embasamento teórico e definição de conceitos sobre o universo de televisão digital. Estudou-se desde a história da TV, sua evolução até o sistema digital. No contexto de TV digital, foi estudado seu funcionamento, características de transporte, envio e recebimento, como também o suporte a interatividade proporcionada pelo *middleware* Ginga.

Durante todo o trabalho foi feito o levantamento bibliográfico com a finalidade de encontrar materiais científicos para pesquisa do assunto. A maioria dos materiais utilizados está disponível em sites de universidades ou nos sites dos fabricantes. Os materiais de universidades são em sua grande maioria teses de mestrados. Para o entendimento da linguagem NCL, foi efetuada a compra de um livro, que é referência no assunto.

Foi estudada a aplicabilidade de videovigilância, exemplificando como e onde é utilizada essa técnica, quais as tecnologias mais utilizadas para segurança e qual o envolvimento e importância de diversificar as formas de monitoramento da vigilância.

O Sistema Brasileiro de TV Digital, mesmo sendo uma tecnologia nova, têm diversos materiais publicados, tanto em congressos e eventos, quanto em pesquisas científicas, tratando sobre a tecnologia, leis e conceitos científicos. A participação em eventos

---

<sup>47</sup> Mais informações em: <http://www.lavid.ufpb.br/>

e fóruns na Internet trouxe contatos com diversas pessoas que ajudaram na busca de informações necessárias.

Tanto o canal de interatividade quanto o *middleware* Ginga estão muito bem documentados e já foram estudados por diversas universidades, assim facilitando a compreensão e escrita desse trabalho.

#### 4.1.2. Ferramentas para desenvolvimento

Para o desenvolvimento dessa aplicação foi necessário a utilização de diversas ferramentas, montando assim um ambiente de desenvolvimento capaz de auxiliar na codificação, execução e visualização de aplicações NCL. Todas as ferramentas são livres, disponíveis para download na Internet.

Para iniciar precisa-se de uma máquina virtual do Ginga para poder emular uma TV com interatividade. Esta máquina<sup>48</sup> é disponibilizada pelo Laboratório Telemidia da PUC-Rio. O Eclipse foi a IDE utilizada para o desenvolvimento do código NCL, também foi utilizada a VMware para emular a máquina virtual do Ginga.

#### 4.2. Desenvolvimento da aplicação

Com o conhecimento técnico-científico, foi dado início ao desenvolvimento da aplicação declarativa na linguagem NCL. A aplicação foi desenvolvida no modelo de prototipação, onde foram criadas telas e a partir desse modelo foi desenvolvido o código. O telespectador vai controlar a aplicação por meio do controle remoto da televisão. O diagrama de casos de uso da Figura 1 demonstra as ações possíveis.

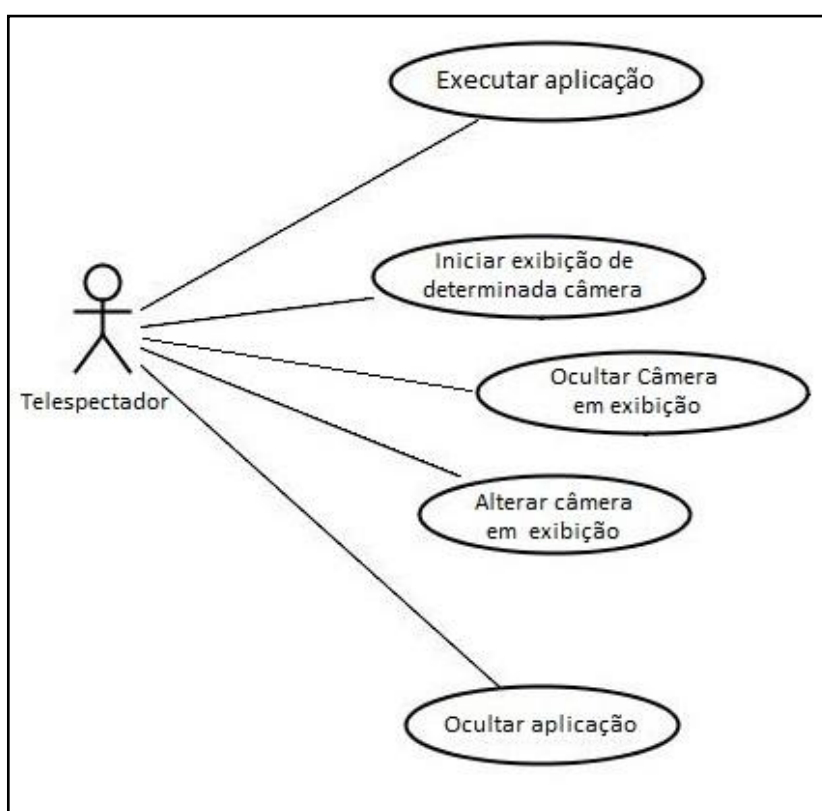


Figura 1. Diagrama de casos de uso

<sup>48</sup> <http://www.telemidia.puc-rio.br/pt/index.html>

Com testes foi constatado que poderiam ser feitas modificações na interface da aplicação para melhorar a usabilidade, definindo assim um modelo para o projeto final.

No decorrer de todo o trabalho foram desenvolvidas diversas versões da aplicação. A versão final conta com um ícone de interatividade, e ao receber uma aplicação interativa, a televisão inicia a aplicação, porém o telespectador escolhe a hora que deseja que toda a aplicação tome conta de sua atenção.

Após vários testes foi verificado que os vídeos de câmeras de vigilância são melhores observados se não tiver nada em sua frente. Quanto ao vídeo da emissora, não há tanta importância em ter seu foco perdido, principalmente nas laterais, pois o conteúdo principal sempre se encontra mais ao centro da tela.

Ao ser acionado o botão vermelho (RED) do controle a aplicação inicia o contexto "App" da aplicação, como pode ser observado na Figura 2.

```
<!-- Quando selecionar o ícone de interatividade, deve iniciar o contexto App -->
<link id="Ícone" xconnector="con#onSelection!StartN">
  <bind role="onSelection" component="ícone">
    <bindParam name="keyCode" value="RED"/>
  </bind>
  <bind role="start" component="App"/>
</link>
```

Figura 2. Código Ícone Interatividade

Nesse contexto estão os botões, o fundo, a barra de título e a ajuda do programa. A imagem recebida via *broadcast* é exibida em tela cheia, sendo que os menus, barras e imagens têm transparência.



Figura 28. Versão final de Interface

Ao iniciar essa tela, é possível escolher imagem de qual câmera deseja visualizar. Nessa versão não é mais utilizado as regras e nem a variável interna para seleção das câmeras de vigilância. É feito um controle manual de qual componente está com o foco, e a partir disso é feito o controle de qual tecla é acionada.

Se estiver na primeira opção, somente é permitido selecionar, por meio da tecla ENTER, ou então ir para baixo pela tecla DOWN.

Na Figura 4 tem uma câmera em execução, a qual está sobreposta ao conteúdo exibido pela emissora.



Figura 4. Câmera de vigilância sendo exibida

O menu informa que para sair da aplicação tem de se apertar a tecla azul do controle remoto. Ao fazer isso, a aplicação encerra o contexto “App”, o qual contém os componentes de seleção de câmeras, o componente para ocultar as câmeras, o fundo e a barra de título, ficando visível apenas o ícone de interatividade, o qual pode ser acessado novamente pelo botão vermelho do controle remoto.

## 5. Resultados

Como resultado desta pesquisa obteve-se um aplicativo, apresentado e explicado anteriormente, que exibi imagens de câmeras de vigilância, proporcionando ao telespectador a possibilidade de interação com a aplicação, podendo escolher a imagem de qual câmera deseja visualizar.

É uma aplicação de fácil de usar, característica necessária no ambiente de televisão, pois atende a praticamente todo tipo de pessoa, não tendo um público alvo determinado. Verificou-se que o middleware Ginga ainda não está completamente desenvolvido, atrapalhando assim na obtenção de êxito em alguns dos objetivos propostos. Porém a aplicação contém a ideia central de todo o trabalho, que é exibir as imagens de câmeras de vigilância em uma aplicação interativa.

Com o estudo de conceitos e aplicando-os em uma aplicação pode-se compreender como funciona o SBTVD abrindo assim um leque para novos estudos e aplicações.

Quando o *middleware* Ginga estiver totalmente implementado, as aplicações poderão ser muito mais ricas, cativando cada vez mais adeptos e difundindo cada vez mais a inclusão digital no Brasil. Utilizar o canal de interatividade para transmissão de arquivos de

mídias, que poderão estar sendo atualizados ou modificados dependendo do perfil do telespectador, será uma das diversas possibilidades para maior interação e experiência em frente à televisão.

O Exibidor de NCL da GHTV<sup>49</sup> é uma implementação do padrão do SBTVD que está em desenvolvimento. Ele é um pequeno exibidor de documentos NCL, que conta com diversas funções, como suporte a HTML, NCL, mídias de texto, URL's HTTP contendo imagens, áudio, vídeo, HTML e texto, notificação de quando uma aplicação contém erros e registro no sistema operacional de mídias NCL.

Mesmo com diversas funcionalidades, ele ainda está em desenvolvimento e não trata várias características necessárias para apenas executar a aplicação desse projeto. O exibidor apresentou problemas com as transparências das imagens e com os conectores do código fonte.

## 6. Dificuldades

Durante todo o processo de aprendizagem de uma nova linguagem e do desenvolvimento de uma solução para o problema encontrado encontrou-se diversas dificuldades.

A ideia inicial do trabalho seria exibir imagens em tempo real de câmeras de vigilância por meio do canal de interatividade do SBTVD. Para isso seria necessário o Real Time Protocol (RTP) que ainda não foi implementado, tornando-se assim impossível o desenvolvimento por esse protocolo.

Foi instalado um programa, denominado Motion, que captura imagens de uma câmera, salva em um local e também as disponibiliza pela *web* pelo HiperText Transfer Propocol (HTTP). Sem sucesso novamente, a máquina virtual do Ginga não consegue interpretar o que está recebendo.

Em consulta aos grupos de discussões foi constatado que a VM do Ginga ainda suporta somente HTML pelo protocolo HTTP, não podendo assim ser acessado um vídeo por meio desse protocolo de comunicação. Com novas versões da implementação do *middleware* Ginga haverá a possibilidade de acessar outros tipos de mídias.

O Exibidor NCL GHTV é um exibidor de aplicativos escritos em NCL. Na versão 0.4.2 o exibidor incorpora o recurso para acesso a mídias remotas. Porém esse exibidor não trata efeitos de transparência nas imagens, o que impossibilita de apenas executar o código NCL nele e funcionar perfeitamente.

## 7. Conclusão

A pesquisa resultou no estudo da disponibilidade de imagens de câmeras de vigilância pelo Sistema Brasileiro de TV Digital. Verificou-se conceitos e características de câmeras de vigilância, Circuitos Fechados de TV, conceitos sobre o SBTVD, capacidades e inovações do padrão nipo-brasileiro de TV Digital.

Com isso foi possível a elaboração de um protótipo de software que disponibiliza a visualização de imagens de câmeras de vigilância na televisão de todo telespectador que tenha acesso ao sinal digital. Foram utilizadas ferramentas como o Eclipse e alguns plugins, a máquina virtual Ginga e o emulador de máquinas virtuais VMware Player.

---

<sup>49</sup> <http://ghtv.com.br/>

Com o estudo, constatou-se que o middleware Ginga ainda apresenta limitações, faltando diversas implementações. Não se pode, ainda, devido a limitações de desenvolvimento do Ginga, acessar os vídeos de câmeras de vigilância por meio do canal de interatividade. Porém a aplicação os exibe, recebendo-os via broadcast.

O Exibidor NCL GHTV consegue acessar arquivos remotos por meio do canal de interatividade, mas pelo recente lançamento da versão com esse suporte não foi possível migrar a aplicação para o mesmo. Foram feitos apenas pequenos testes para confirmar sua veracidade de funcionamento, encontrando assim problemas com as transparências das imagens.

A aplicação resultante é bem intuitiva, característica fundamental, pois atende a todos os tipos de público. Tem uma interface amigável e simples, com transparência, visando ser ao máximo usual e discreta, para não atrapalhar a programação corrente.

Para a comunidade científica, o trabalho colabora identificando as limitações do protocolo HTTP do middleware brasileiro, documentando diversas características e conceitos do padrão do SBTVD e alguns detalhes do Exibidor NCL GHTV.

Deixa-se como possibilidade para trabalhos futuros, a evolução deste trabalho das seguintes formas:

- f) utilização do canal de interatividade: quando a máquina virtual do Ginga suportar o transporte de vídeos pelo canal de interatividade pode-se acessar as imagens por este canal, não sendo necessária a utilização da banda do carrossel de dados. Para isso é preciso apenas alterar o nome do arquivo que está relativo, colocando o endereço do servidor web que contém o vídeo especificado.
- g) uso de imagens de câmeras em tempo real: quando a máquina virtual do Ginga suportar o transporte de streaming, pode-se ter um servidor de streaming pegando as imagens de câmeras em tempo real e disponibilizando-as aos telespectadores. Para isso também é necessário colocar apenas o endereço do servidor quando se declara a mídia no código NCL.
- h) restrição de acesso: pode ser implementado para que apenas certas pessoas possam ter acessos a câmeras específicas. Para isso pode ser feito a implementação de senhas de acesso.
- i) prioridade de acesso: pode haver a necessidade de certas conexões terem prioridades sobre outras. Por exemplo, a polícia deve ter mais prioridade no acesso a imagens de câmeras que telespectadores comuns, então para isso é necessário haver controle no servidor. Isso ocorrerá somente usando o canal de interatividade.

## Referências

ASSOCIAÇÃO BRASILEIRA DE NORMAS TÉCNICAS. NBR 15606:Middleware. Rio de Janeiro, 2008. Disponível em: <[http://www.dtv.org.br/download/pt-br/ABNTNBR15606\\_2D3\\_2007Vc2\\_2008.pdf](http://www.dtv.org.br/download/pt-br/ABNTNBR15606_2D3_2007Vc2_2008.pdf)>. Acesso em: abr. 2010.

CARVALHO, Ederson de Assis. Projeto olho vivo: “A Iris dos olhos da segurança pública” Uma análise geográfica. 100f. Dissertação (Mestrado) PUC-MG, Belo Horizonte, 2008. Disponível em:

- <[http://www.biblioteca.pucminas.br/teses/TraInfEspacial\\_CarvalhoEA\\_1.pdf](http://www.biblioteca.pucminas.br/teses/TraInfEspacial_CarvalhoEA_1.pdf)>. Acesso em: set. de 2010.
- CARVALHO, Eduardo Rodrigues. Uma plataforma modular para testes com interatividade na TV digital brasileira. 105f. Dissertação (Mestrado) – USP, São Paulo, 2008. Disponível em: <[http://www.teses.usp.br/teses/disponiveis/3/3142/tde-30052008-110812/publico/Dissertacao\\_Eduardo.pdf](http://www.teses.usp.br/teses/disponiveis/3/3142/tde-30052008-110812/publico/Dissertacao_Eduardo.pdf)>. Acesso em: out. de 2010.
- CORRÊA, Luisa; CUNHA, Maicon. Câmeras no jornal: cartografando o discurso jornalístico sobre a vídeo-vigilância no Brasil. Rio de Janeiro, 2009. Disponível em: <[http://www2.pucpr.br/ssscla/papers/SessaoB\\_A11\\_pp92-111.pdf](http://www2.pucpr.br/ssscla/papers/SessaoB_A11_pp92-111.pdf)>. Acesso em: abr. de 2010.
- CRUZ, Vitor Medina; MORENO, Marcio Ferreira; SOARES, Luiz Fernando Gomes. TV Digital Para Dispositivos Portáteis- Middlewares. Rio de Janeiro, 2008. Disponível em: <[http://www.dbd.puc-rio.br/depto\\_informatica/08\\_03\\_cruz.pdf](http://www.dbd.puc-rio.br/depto_informatica/08_03_cruz.pdf)>. Acesso em: abr. de 2010.
- DAMASCENO, Jean Ribeiro. Middleware Ginga. Niterói, 2008. Disponível em: <<http://www.midiacom.uff.br/~deborafsmm/trab-2008-2/middleware.pdf>>. Acesso em: out de 2010.
- FERNANDES, Jorge; LEMOS, Guido; SILVEIRA, Gledson. Introdução à televisão digital interativa: arquitetura, protocolos, padrões e práticas. Salvador, 2004. Disponível em: <<http://www.cic.unb.br/~jhcf/MyBooks/itvdi/texto/itvdi.pdf>>. Acesso em out. de 2010
- FOUCAULT, Michel. Vigiar e Punir. Tradução Raquel Ramallete. Petrópolis: Vozes, 2000. 262 p.
- KANASHIRO, Marta Mourão. Surveillance Cameras in Brazil: exclusion, mobility regulation, and the new meanings of security. Surveillance Studies Network, 2008. Disponível em: <[http://www.surveillance-and-society.org/articles5\(3\)/brazil.pdf](http://www.surveillance-and-society.org/articles5(3)/brazil.pdf)> Acesso em: abr. 2010.
- LAMAS, Amilton C. et al. Canal de Interatividade em TV Digital. Cad. CPqD Tecnologia. Campinas, v.1, n.1, p. 29-36, jan./dez. 2005. Disponível em: <[http://www.cpqd.com.br/cadernosdetecnologia/Vol1\\_N1\\_jan\\_dez\\_2005/pdf/artigo2\\_Lamas.pdf](http://www.cpqd.com.br/cadernosdetecnologia/Vol1_N1_jan_dez_2005/pdf/artigo2_Lamas.pdf)>. Acesso em: abr. 2010
- MONTEZ, Carlos; BECKER, Valdecir. TV Digital Interativa: conceitos, desafios e perspectivas para o Brasil. Florianópolis: Ed. da UFSC, 2005. 2ª edição.
- MOREIRA, Katia B. R. Diretrizes para projeto de segurança patrimonial em edificações. 186f. Dissertação (Mestrado) USP, São Paulo, 2007. Disponível em: <[http://www.teses.usp.br/teses/disponiveis/16/16132/tde-17052010-090837/publico/Dissertacao\\_Katia\\_B\\_R\\_Moreira\\_FAUUSP\\_2007.pdf](http://www.teses.usp.br/teses/disponiveis/16/16132/tde-17052010-090837/publico/Dissertacao_Katia_B_R_Moreira_FAUUSP_2007.pdf)>. Acesso em: set. de 2010.
- SOARES, Luiz Eduardo. Segurança Pública: presente e futuro. Estudos Avançados. Rio de Janeiro, 2006. Disponível em: <<http://www.scielo.br/pdf/ea/v20n56/28629.pdf>>. Acesso em: abr. 2010.
- SOARES, Luiz Fernando Gomes. As múltiplas possibilidades do middleware Ginga. São Paulo, 2008. Disponível em: <<http://www.producaoprofissional.com.br/files/editions/ed1216040609.pdf>>. Acesso em: abr. 2010.
- SOARES, L. F. G; BARBOSA, S. D. J. Programando em NCL 3.0: Desenvolvimento de aplicações para o middleware Ginga. Rio de Janeiro: Ed. Campus, 2009.

TONIETO, Marcia Terezinha. Sistema Brasileiro de TV digital – SBTVD Uma análise política e tecnológica na inclusão digital. 267f. Dissertação (Mestrado) - UECE/CEFET, Fortaleza, 2006. Disponível em: <<http://mpcomp.pgcomp.uece.br/admin/arquivos/MarciaTonieto2006.PDF>>. Acesso em: out. de 2010.