

# **Desenvolvimento de Servidor RSS para TV Digital Interativa**

---

Aluno: Everton Schroeder

Orientador: Mauro Marcelo Mattos

# Tópicos abordados

---

- Introdução e objetivos
  - Conceitos
    - Apresentação do sistema de TVDI
      - Aplicações
      - Tecnologias envolvidas
      - Principais padrões tecnológicos
      - Sistema Brasileiro de TV Digital – SBTVD
    - RSS
      - Características estruturais
      - Principais formas de uso
  - Trabalhos utilizados como referência
  - Desenvolvimento da aplicação
  - Comentários finais
-

# Televisão

- ❑ Sistema eletrônico de transmissão e recepção imagens e som de forma instantânea



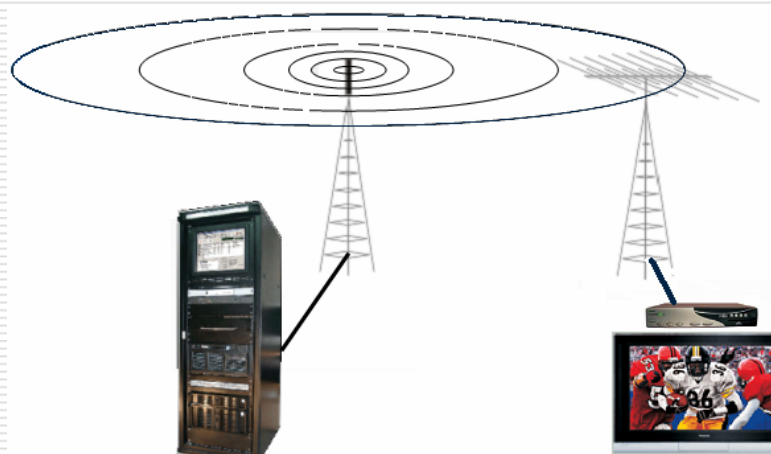
• Produção  
Vídeo  
Áudio

Transmissão



• Recepção  
Canal

- ❑ Funciona a partir da análise e conversão de imagem e som em ondas eletromagnéticas



# TV Digital Interativa

---

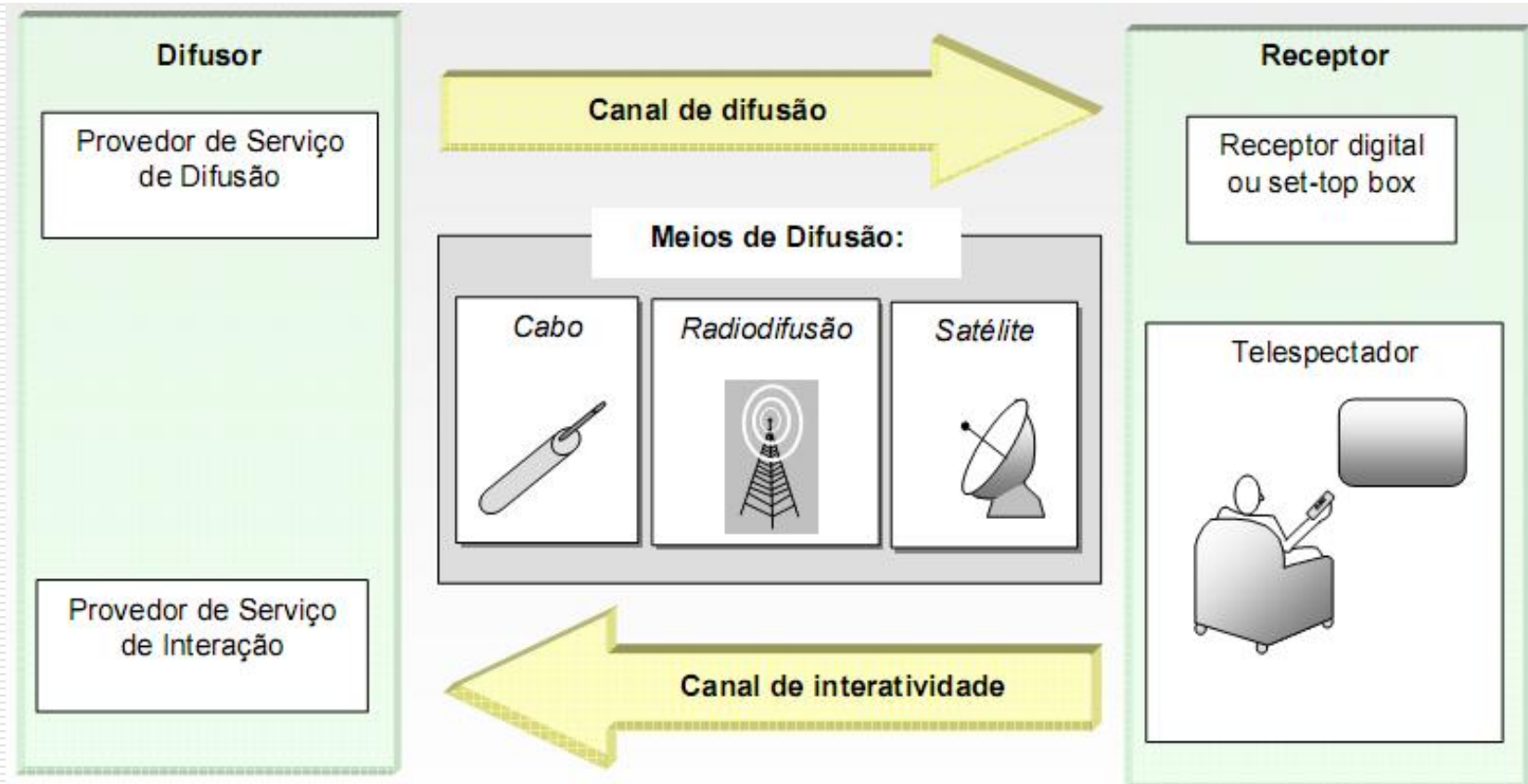
- Características da TVDI
    - Melhor qualidade de imagem (*widescreen*) e som (*surround*)
    - Transmissão do sinal sem degradação (interferências)
    - Capacidade de transporte de dados em conjunto com áudio e vídeo
    - Interatividade com o telespectador e disponibilidade de novos serviços
      - Guia de programação eletrônica
      - Serviços governamentais
      - Programação educacional
      - Comércio eletrônico
      - Vídeo sob demanda
    - Expansão da TV - nova forma de comunicação
      - **NÃO** é TV via internet
      - Agrega várias tecnologias existentes
        - TV, VoIP, SMS, RSS, etc.
-

# Impacto na sociedade

---

- Ampliação da gama de serviços disponibilizados ao telespectador
  - Alternativas para desenvolvimento de conteúdo (equipamentos, programas) específicos para esse tipo de mídia
  - Inclusão digital
    - O número de pessoas que possuem TV (90% das residências no Brasil) é bem maior do que as que possuem PC com acesso a Internet – 6% (Becker, 2004).
-

# Cadeia de transmissão e recepção de sinal televisivo



# Objetivos do trabalho

---

- Desenvolvimento de uma aplicação interativa para o ambiente de TVDI
    - Utilizar o canal de retorno para requisitar informações
    - Receber e traduzir mensagens enviadas pelo serviço difusor no formato RSS
-

# Infra-estrutura

## Etapas envolvidas

---

### □ Transmissão/multiplexação de conteúdo MPEG-2

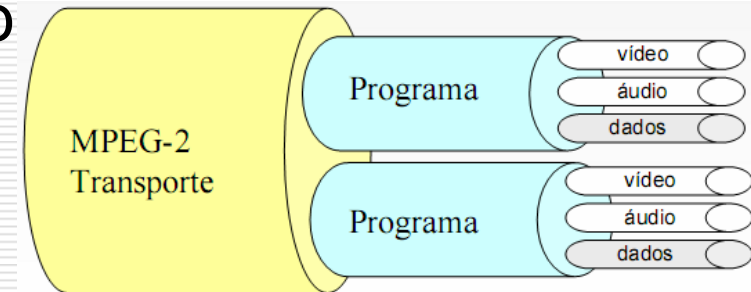
- Edição e otimização da mídia a ser transmitida
- Correção de erros na amostragem

### □ Modulação (QAM, QPSK, COFDM ou 8-VSB)

- Empacotamento da mídia para difusão
- Envio do fluxo para a frequência em que será distribuído

### □ Recepção/demultiplexação

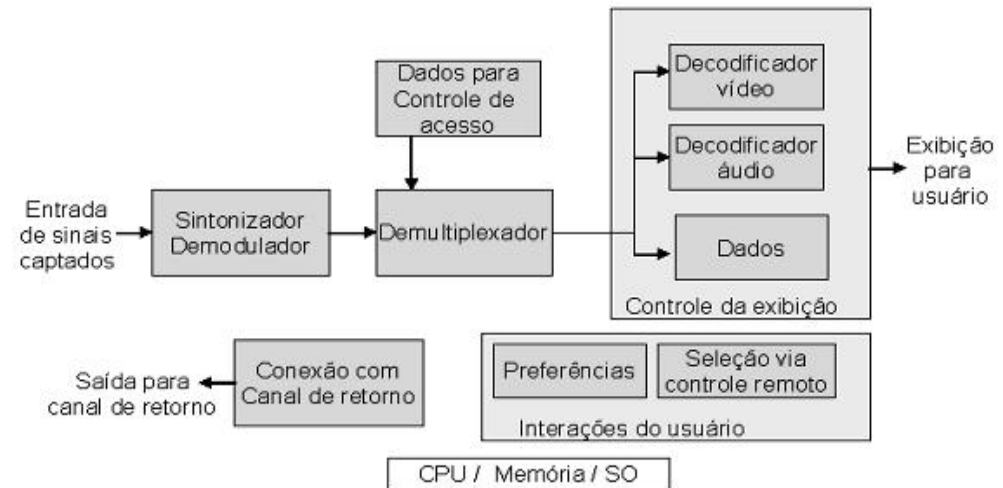
- Receptor digital/Set-top box





# Set-top boxes

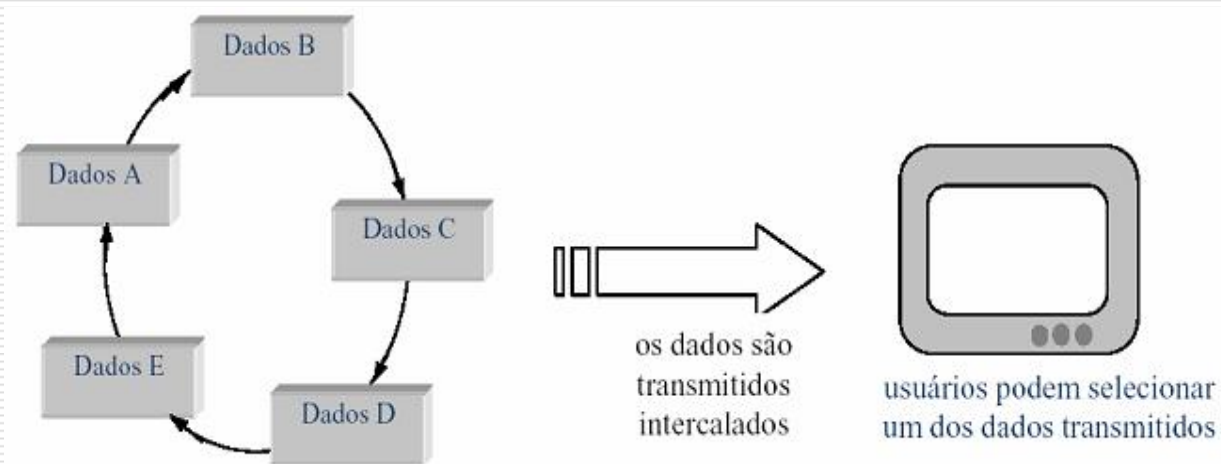
- ❑ Decodificação do fluxo televisivo
  - áudio, vídeo e dados
- ❑ Processamento de atividades paralelas
  - verificar direitos de acesso ao conteúdo
  - executar aplicações
  - interação com o telespectador
- ❑ Suporte a periféricos
  - dispositivos de entrada e saída



# Digital Storage Media - Command and Control (DSM-CC)

---

- Conjunto de protocolos responsáveis pelo controle sobre fluxos de transmissão MPEG
  - Carrossel de dados
  - Canal de retorno
  - Sistema de acesso condicional



# Canal de retorno

---

- Níveis de interatividade
    - Local
    - Parcial (unidirecional)
    - Total (bidirecional)
  - Tecnologias para implementação
    - conexão discada
    - ADSL
    - GSM
  - Utilização dos mesmos protocolos usados para Internet
    - TCP/IP
      - HTTP
      - SMTP
-

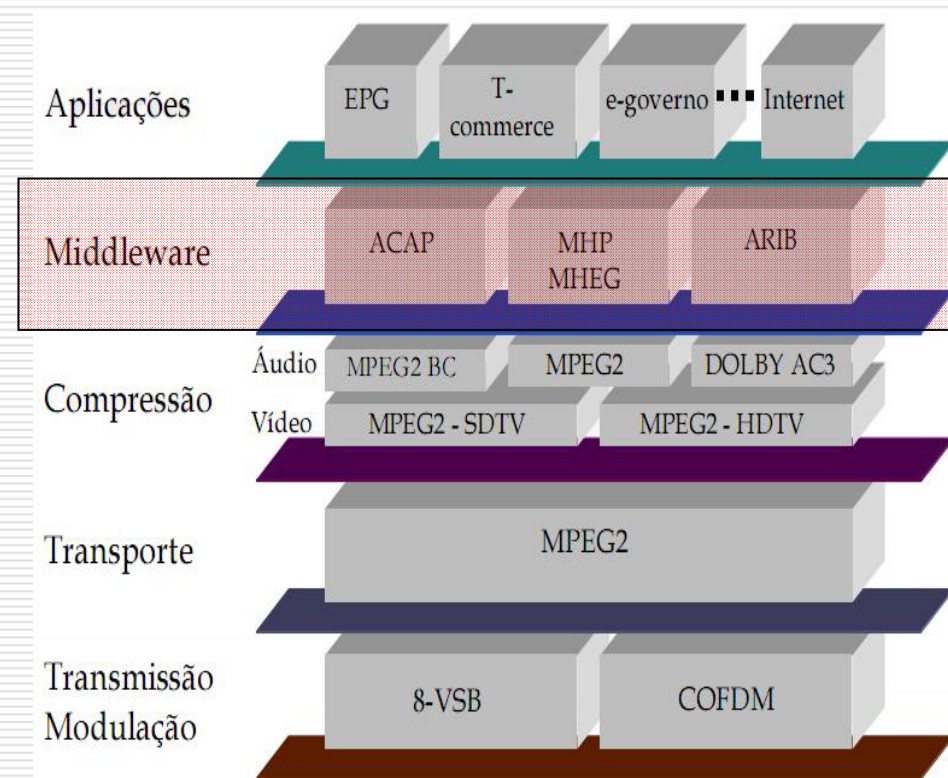
# Padrões de TVDI e *middlewares*

## □ Middleware

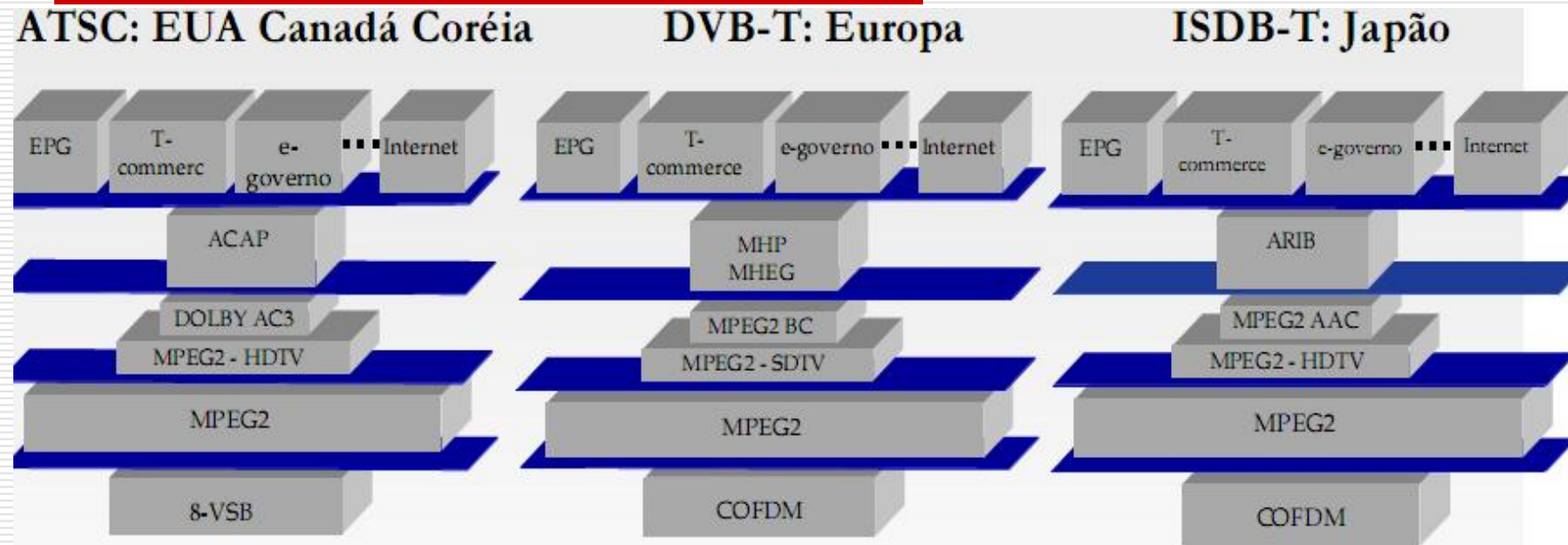
- Camada para padronização de aplicações para TVDI
- Intermediação entre aplicação e sistema operacional do STB

## □ Padrões existentes

- ATSC - DASE (EUA)
- DVB - MHP (Europa)
- DIBEG - ISDB (Japão)



# Padrões de TVDI e *middlewares*



- ❑ ATSC: alta definição de imagens (HDTV)
- ❑ DVB: interatividade e novos serviços
- ❑ ISDB: recepção portátil e HDTV
- ❑ Todos os padrões de *middleware* adotam JAVA, têm uma base comum (DAVIC, HAVI e JavaTV)

# DVB-MHP

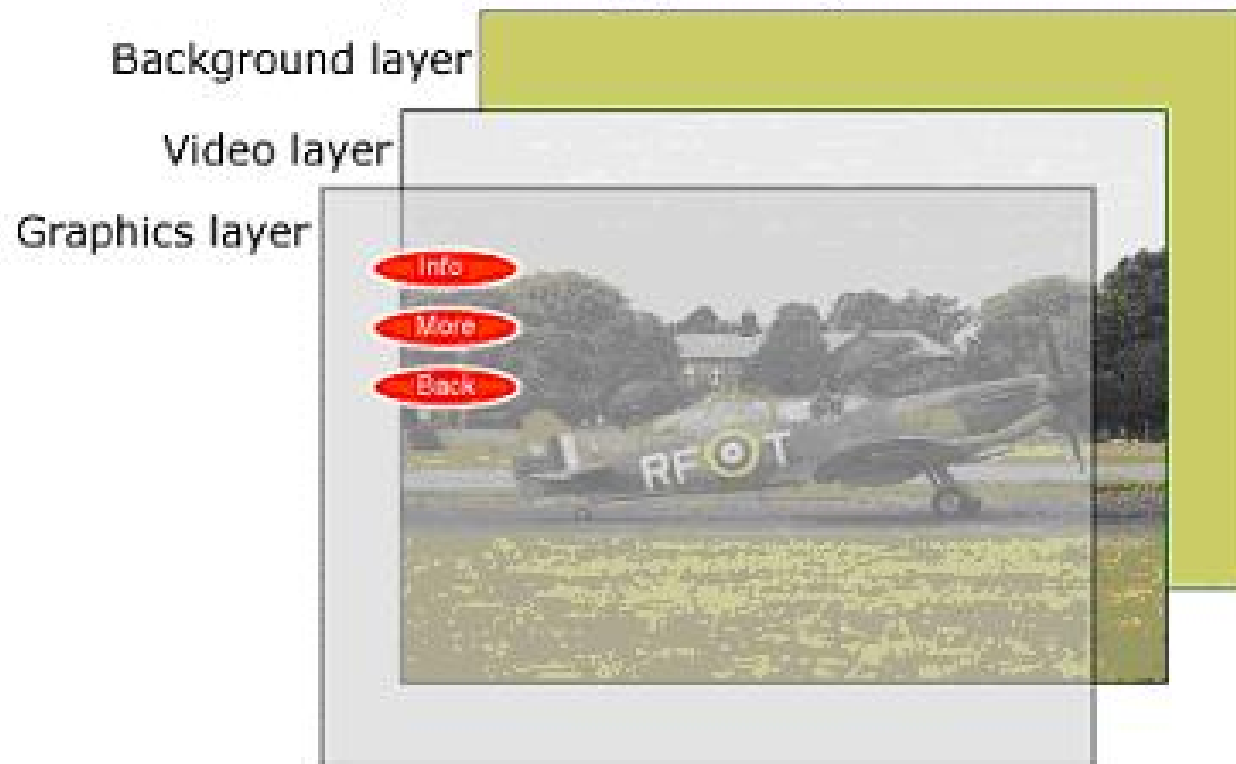
---

- Middleware baseado em JavaTV
  - APIs auxiliares
    - gerenciamento de recursos de hardware
      - dispositivos de entrada/saída
    - gerenciador de aplicações
    - interface com o usuário
      - recursos de mídia (JMF)
      - componentes visuais (AWT)
-

# MHP – Camadas da interface gráfica

---

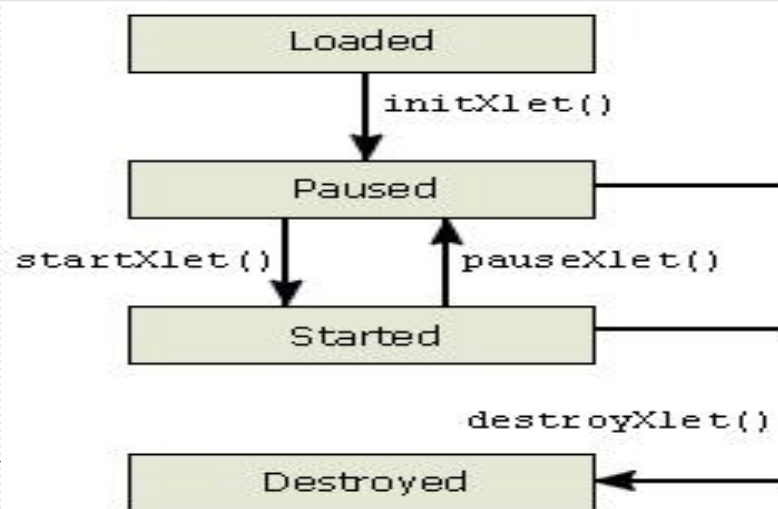
Layers in the MHP graphics subsystem.



# API JavaTV

---

- API desenvolvida com recursos direcionados para os *Set-top boxes*
  - plataforma mais enxuta
- Aplicações em JavaTV - Xlets
  - ciclo de vida controlado pelo gerenciador de aplicações do STB





# Sistema Brasileiro de TV Digital

## Objetivos

---

- Papel da TV no Brasil
    - Principal instrumento de informação e entretenimento da população
  
  - Incentivar desenvolvimento tecnológico no Brasil para a área
  
  - Transição entre sistema analógico e digital respeitando condições sócio-econômicas
  
  - Inclusão digital
    - Maior acesso a informações para a população
    - Serviços educacionais
    - Modernização de serviços administrativos
-

# Sistema Brasileiro de TV Digital SBTVD

---

- Histórico (MINISTÉRIO DAS COMUNICAÇÕES, 2003)
    - 1991 – Comissão Assessora para Assuntos de Televisão (COM-TV)
    - 1994 – Abert/SET
    - 1998/2000 – Anatel importa equipamentos e realiza primeiros testes
      - Estudo dos três padrões
    - 2001 – Relatório com resultados dos testes
    - 2003 – Opção por criar um sistema brasileiro – SBTVD
      - Formação dos comitês (Desenvolvimento, Consultivo e Gestor)
    - 2004 – Chamadas públicas para desenvolvimento
    - 2006 – Definição do modelo de referência
    - 2007 – Primeiras transmissões digitais no país (01/12)
-

## ***Rich Site Summary - RSS***

---

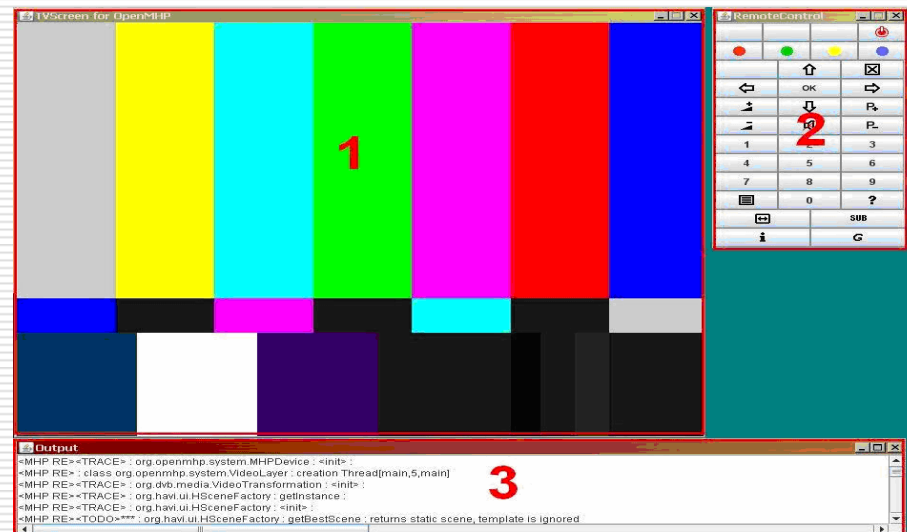
- ❑ Criado em 1997 pela UserLand e comprado pela Netscape
  - ❑ Principal propósito: prover notícias e atualizações por meio de *feeds*
  - ❑ Utiliza-se do formato XML para organização do documento
-

# RSS – estrutura de um *feed*

```
<?xml version="1.0"?>
<rss version="2.0">
<channel>
  <title>Canal de notícias</title>
  <link>http://www.nomedapagina.com/</link>
  <description>Exemplo de um feed RSS</description>
  <copyright>2007</copyright>
  <image>
    <title>Título do canal</title>
    <width>88</width>
    <height>31</height>
    <url>http://www.nomedapagina.com/logo.gif</url>
  </image>
  <item>
    <title>Item contendo uma notícia</title>
    <link>http://www.nomedapagina.com/item1.htm</link>
    <description>Exemplo de um item</description>
    <pubDate>Sun, 22 Dec 2007 22:16:32 GMT</pubDate>
  </item>
</channel>
</rss>
```

# Emuladores

- ❑ Objetivo
  - simular execução em ambiente de TVDI
- ❑ Disponíveis em código aberto
  - XletView
  - OpenMHP



***Interface do OpenMHP***

# Trabalhos correlatos

---

- Darós (2004)
    - Aplicação de T-comércio
  - Casas (2006)
    - Interpretador de documentos RSS
  - Andreatta (2006)
    - Portal para aplicações colaborativas
-

# Ferramentas utilizadas para o desenvolvimento

---

- Modelagem
    - Rational Rose
    - JUDE
  - Implementação
    - Eclipse
  - Execução e testes
    - OpenMHP
-

# Especificação

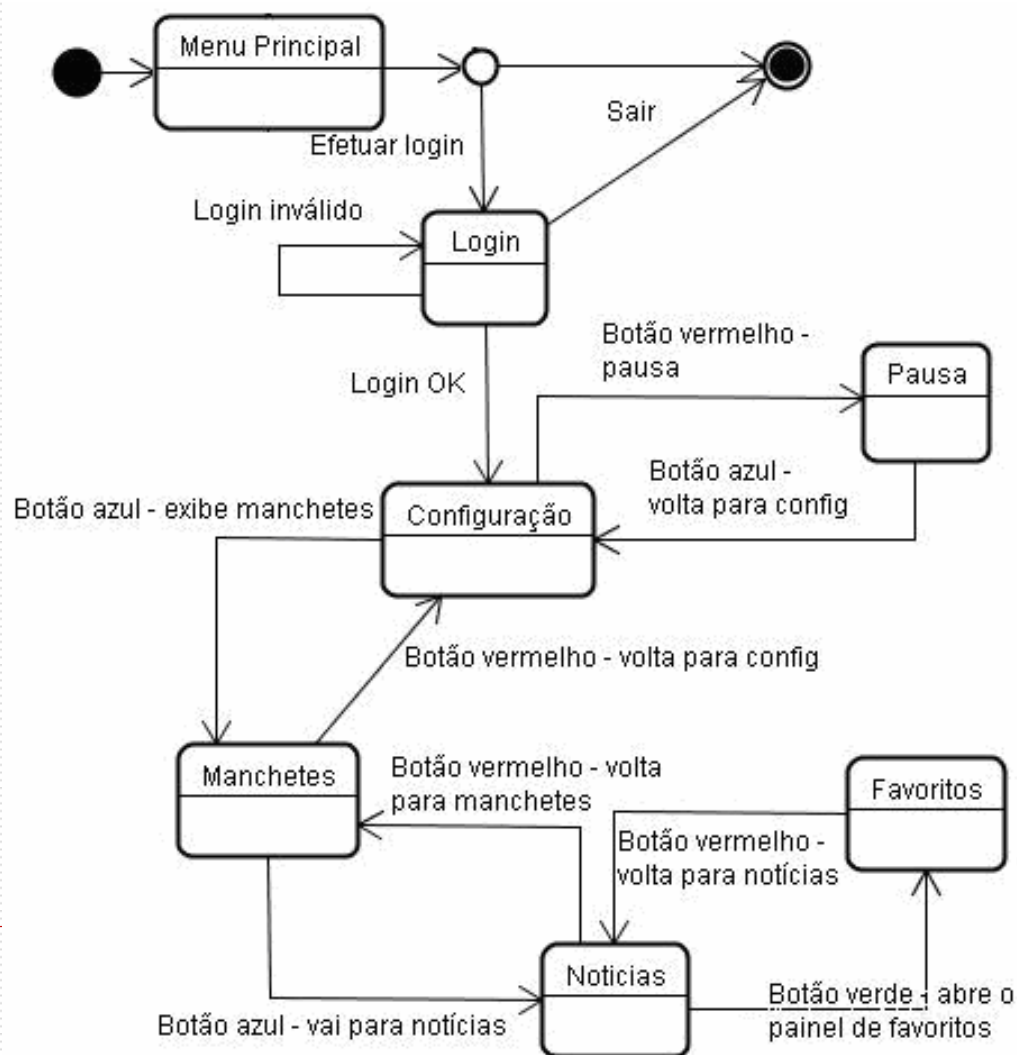
## Classes implementadas

---

- Aplicação
    - Xlet
  - Telas
    - Menu Principal
    - Login
    - Configurações
    - Manchetes
    - Notícias
    - Favoritos
  - Componentes visuais / entrada de informações
    - TextBox
    - TextBox editável
    - Teclado Numérico
  - Negócio
    - ItemRSS
    - Depósito feeds
  - Comunicação
    - Interface Canal Retorno
-

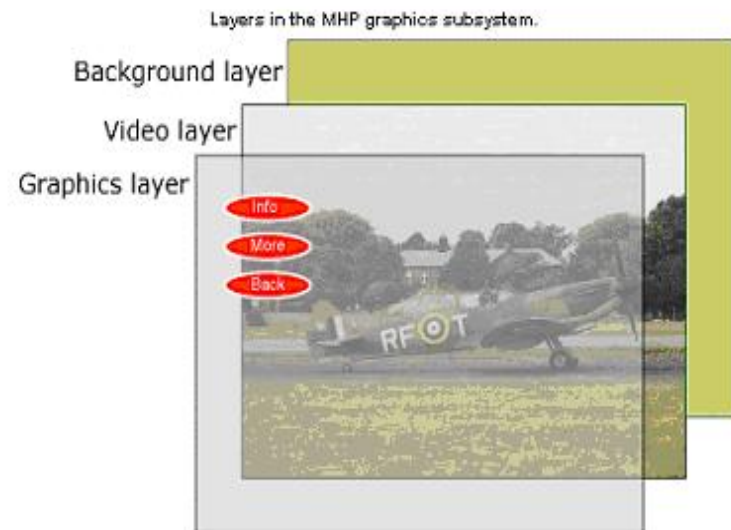


# Especificação Estados da aplicação



# Interface gráfica

- API: HAVi
  - Classe org.havi.ui
- Renderizada na camada gráfica
  - HGraphicsDevice
- Cada tela da aplicação é adicionada a uma instância de HScene
  - Container
- Tratamento de eventos do controle remoto
  - KeyEvent



# Conexão com canal de retorno

---

- ❑ org.dvb.net.rc
  - RCInterfaceManager – armazena as interfaces disponíveis para conexão
  - RCInterface – interface para conexão com o canal de retorno
  - ConnectionRCInterface – efetua conexão
- ❑ Tipo de conexão para buscar informações: java.net.HttpURLConnection

```
RCInterfaceManager rcm = RCInterfaceManager.getInstance();
RCInterface[] interfaces = rcm.getInterfaces();
if (interfaces[0] instanceof ConnectionRCInterface) {
    ConnectionRCInterface myInterface;
    myInterface = (ConnectionRCInterface)interfaces[0];
    try {
        myInterface.reserve();
        ConnectionParameters myConnectionParameters;
        myConnectionParameters = new ConnectionParameters
            ("0199604235697", "username", "password");
        myInterface.setTarget(myConnectionParameters);
        myInterface.connect();
        ...
        myInterface.disconnect();
        myInterface.release();
    }
}
```

# Interpretação do *feed*

---

## □ *Parser* XML: DOM

- Mapeamento direto dos elementos do arquivo para memória
  - Cada item de notícia é armazenado em uma lista para exibição na tela
-

# Conclusões

---

## Objetivo alcançado

- Aplicação utilizando interatividade total

## Conceitos abordados

- APIs utilizadas no desenvolvimento
- Estudos sobre tecnologias para canal de retorno

## Pesquisas futuras

- SBTVD
-

# Limitações e dificuldades encontradas

---

- ❑ Ambiente de emulação com implementação parcial dos recursos
  - ❑ Tecnologias de canal de retorno não estão definidas em termos de mercado
-

## **Sugestões para extensão**

---

- Segurança para comunicação com canal de retorno
  - Soluções para desenvolvimento de interface gráfica
  - Interpretação de conteúdo HTML para o ambiente de TVDI
-

# Apresentação do protótipo

---



# Dúvidas?

---

---

# Agradecimentos

---

Obrigado pela atenção

**Contato:**

**evertons@inf.furb.br**

**evertonschroeder@gmail.com**

---