

# TV Digital e mobilidade: Acesso a web services e interação por meio de dispositivos móveis

Eucássio G. Lima Jr., Antônio O. Alves, José M. L. Silva, Nairon S. Viana, Maria C. Machado  
Universidade Estadual do Piauí (UESPI),  
Núcleo de Pesquisa e Soluções em Tecnologia da Informação (PSTI)  
Rua João Cabral, 2231, Campus do Pirajá, Teresina-PI, Brasil  
{eucassiojr, aj.oliveira.alves, magnoleal89, nairon.viana}@gmail.com,  
mariacostam@hotmail.com

## RESUMO

Este trabalho aborda a utilização da TV Digital Interativa (TVDI) brasileira como ferramenta de acesso a conteúdo Web, por meio do consumo de web services, bem como a interação por meio de uma interface móvel, como um smartphone. O trabalho apresenta o início da construção de um protocolo de troca de mensagens entre os dispositivos móveis, hoje em ascensão, e receptores Ginga-NCL. O modelo proposto é implementado em dois aplicativos: *CineLua*, um ambiente social via TVDI com informações sobre a grade de programação de um cinema; e *LuaNews*, um serviço que permite ao usuário enviar informações de um programa de TVDI para serem visualizadas diretamente em seu dispositivo móvel, sem interferir na experiência coletiva de assistir TV. São apresentados componentes do modelo, suas principais funções, resultados alcançados e algumas dificuldades na construção da plataforma.

## PALAVRAS-CHAVE

TVDI, interatividade, Web, Usabilidade.

## 1. INTRODUÇÃO

Com a crescente ascensão do uso de dispositivos móveis (*smartphones* e *tablets*), a expansão da Internet e a introdução da TV Digital Interativa (TVDI) nos lares brasileiros, surge a necessidade de uma integração cada vez maior entre essas plataformas [1]. Tal convergência oferece serviços inovadores aos usuários, uma vez que podem ser explorados os potenciais de cada plataforma em conjunto.

Dentro dessa linha de convergência, o presente trabalho propõe e implementa uma arquitetura de convergência TVDI-Web-Mobile, através da configuração de componentes de software que representam cada domínio. De uma forma geral, o trabalho explora a utilização de outros elementos em conjunto com a TVDI que dão suporte a aplicações interativas mais ricas, como aquelas vinculadas a algum conteúdo web ou as que oferecem a possibilidade de conectar TV e dispositivo móvel (celulares, PDAS, smartphones, etc.).

A pesquisa desenvolvida apresenta-se como resultado parcial do projeto *Desenvolvimento de Competências para a TV Digital Interativa Brasileira*, no qual foram investigadas as soluções existentes para a construção de aplicações dentro do contexto de convergência TVDI-Web-Mobile. A seguir são apresentados componentes e esquemas de funcionamento dos dois cenários implementados (*CineLua*, na seção 2 e *LuaNews* na seção 3), resultados (seção 4) e conclusões acerca do trabalho (seção 5).

## 2. ARQUITETURA DO CINELUA

A arquitetura do projeto *CineLua* constitui-se de uma extensão do modelo Ginga-NCL incluindo dois componentes sobre ele, possibilitando o uso do protocolo SOAP (escolhido para realizar acesso a web services) e o do parser de arquivos JSON (formato de dados utilizado para a troca de objetos). Nos casos onde não há uma interface para comunicação, pode-se criar um WS (Web Service), utilizando diversos provedores de conteúdo (Sites, feeds RSS, redes sociais), que faz acesso direto as fontes da informação (site de um cinema, por exemplo), remove a formatação HTML, e agrega outras fontes criando uma estrutura específica, podendo ser SOAP ou REST. Do modelo apresentado, apenas a camada que faz consumo do WS seria alterada para o uso de REST, sendo substituído o uso do módulo NUSOAP pela classe TCP.lua, fornecendo métodos de acesso ao protocolo REST.

A facilidade de atualização do WS também favorece sua adoção, pois caso haja mudança da estrutura dos provedores de dados (como a atualização de alguma informação do web site ou a adoção de uma nova API para acesso às mídias digitais) apenas o WS seria atualizado em um servidor web, processo mais simplificado que um *deploy* de uma aplicação Ginga-NCL/Lua, por exemplo. Outra característica é a aplicação de técnicas de otimização como *cache* e escalabilidade de servidores [2].

Para a construção da interface do usuário desenvolveu-se uma estrutura que une componentes NCL para a exibição de conteúdo estático, com objetos Lua que permitem a construção e a reutilização de módulos, criando funcionalidades básicas já existentes em outras plataformas, como exemplo painéis com rolagem de texto.

Dessa forma visa-se alcançar dois objetivos principais: desenvolver uma API de manipulação de web services; utilizar classes Lua para tratar eventos no sistema e oferecer uma resposta de forma automatizada à interface gráfica (GUI) da aplicação. Os componentes utilizados para tais fins são:

---

WebMedia'11: Proceedings of the 17<sup>th</sup> Brazilian Symposium on Multimedia and the Web. VIII Workshop on Ongoing Undergraduate Research.  
October 3 -6, 2011, Florianópolis, SC, Brazil.  
ISSN 2175-9650.  
SBC - Brazilian Computer Society

- **NCLua SOAP**: módulo responsável pelo consumo de Web Services nas versões 1.1 e 1.2 do protocolo SOAP. Adaptado de [3].
- **Json4Lua**: Cliente LUA que implementa a codificação e decodificação JSON de objetos lua.
- **Web Service PHP**: conjunto de classes interpretação do conteúdo do site de um cinema, que o inspeciona a partir da página inicial na busca de informações sobre os filmes em cartaz. Esse componente realiza também uma busca no *Twitter* sobre o que está sendo comentado a respeito de cada um dos filmes, respondendo com os dados junto com as informações sobre o cinema, de forma que o processamento maior seja do lado do servidor web, diminuindo ao máximo o número de requisições por parte do Set-Top Box.

O início da interação com o usuário se dá quando o *CineLua*, por meio do *NCLuaSOAP* realiza uma chamada ao web service localizado em um servidor web remoto. O método *getDadosCinema* é chamado no servidor remoto recebendo o nome do cinema do qual ele deseja obter informações, enviando como resposta para o cliente um XML com as informações desejadas no formato apresentado na Figura 1.

```
<SOAP-ENV:Body>
  <ns1:getDadosCinemaResponse xmlns:ns1="urn:cinelua">
    <return xsi:type="xsd:string">
      {
        "nome": "riverside", "filmes": [
          {
            "titulo": "O Discurso do Rei",
            "genero": "- Drama",
            "duraÃ§Ão": "1:58h",
          }
        ]
      }
    </return>
  </ns1:getDadosCinemaResponse>
</SOAP-ENV:Body>
```

Figura 1: XML de resposta do web service.

Do lado da TVDI o NCLua, após estabelecer a conexão de rede por meio de objetos Lua [4], recebe o XML e o decodifica. Ao receber o retorno em formato JSON este é transformado em objetos Lua, que podem ser facilmente manipulados. Nesse momento a classe *TwittController* realiza a formatação da informação a ser exibida com as informações sobre o filme em cartaz e os *tweets* relacionados.

### 3. ARQUITETURA DO LUANEWS

No que diz respeito a integração da TVDI com dispositivos móveis apresenta-se o *LuaNews*. Para esse estudo utilizou-se aparelhos com sistema operacional Android 1.6 ou superior. O Android é um conjunto completo de software para dispositivos móveis: um sistema operacional, middleware e chave de aplicações móveis, construído sobre o kernel do Linux, ele é open source, podendo ser livremente ampliado para incorporar novas tecnologias de ponta à medida que surgem [5].

Apesar dessa constante evolução o modelo pode ser aplicado em diversas versões do SO sem prejuízo, devido ao uso de componentes das camadas bases do sistema, tendo que adaptar apenas a camada de visualização, para fazer uso dos componentes específicos para smartphones (até a versão 2.3) ou para tablets (versão 3.x). O modelo proposto pode ser aplicado às demais

plataformas móveis, como IOS, Windows Phone, Symbian, Web OS dentre outras, desde que sigam o mesmo padrão.

Na Figura 2 nota-se que o modelo divide-se em duas partes principais: *TVDI app* e *Mobile app*, tendo como meio de comunicação entre uma conexão wireless. A primeira exhibe ao telespectador a interface principal, na qual se escolhe qual aplicativo irá conectar-se a *TVDI app* e iniciar a interação. Na segunda tem-se o controle da interface da TVDI, bem como a execução em paralelo da aplicação da TV.

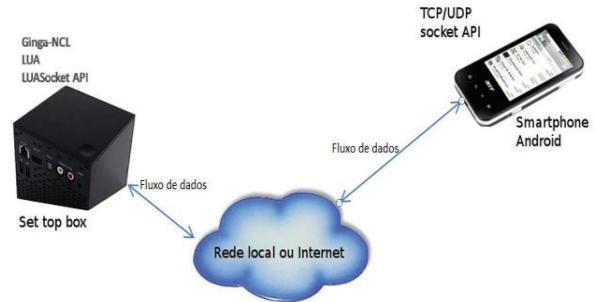


Figura 2. Visão Geral do Modelo do Sistema.

#### 1.1 Localização de dispositivos

Embora o *Ginga-NCL* já possua um modelo de suporte a múltiplos dispositivos de interação [6] optou-se por desenvolver um mecanismo específico para o cenário do *LuaNews*. No momento em que a aplicação Lua inicia, ela passa a escutar a porta 2007 aguardando uma conexão. Utiliza-se o módulo oficial *Socket* de Lua, que provê suporte aos protocolos TCP e UDP na camada de transporte e vários outros na camada de aplicação. Ao conectar-se, o cliente envia por meio do socket estabelecido, uma string contendo o seu IP na rede local e seu nome de exibição, que passam a ser guardados em uma tabela de identificação de dispositivos, junto com o próprio socket (img dispositivo). Como resposta, o *LuaNews* envia o caracter '0' indicando que a operação foi bem sucedida.

#### 1.2 Conexão de dispositivo

O *LuaNews* exhibe uma lista de todos os clientes online na rede. Por meio do componente *panelScroll*, desenvolvido no *CineLua* cada usuário é listado e o telespectador pode escolher um para se comunicar. Quando a conexão se estabelece o cliente recebe uma string com os seguintes códigos: 3#1.

Na primeira parte é informada a ação a ser executada e na segunda o componente a ser criado. Nesse caso, 3 indica que deve ser criado um componente, e o 1 significa que deve ser um componente para entrada de texto. Como o cliente é uma plataforma Android é criado um *android.widget.EditText*.

Com a interface móvel criada, o usuário entra com o texto a ser localizado dentre as notícias relacionadas com a programação que está sendo exibida e submete seu conteúdo. A mensagem é enviada ao *LuaNews*, que ao recebê-la a processa e exhibe o resultado da busca pela palavra chave recebida. O foco da aplicação volta para a TVDI, onde o telespectador então escolhe a notícia dentre os resultados e pressiona a tecla vermelha, para enviá-la ao dispositivo móvel.

A notícia é codificada, da seguinte forma: "comando\_a\_executar#título\_da\_notícia#conteúdo\_da\_notícia" gerando o formato descrito no exemplo da Figura 3:

1#Acesso à internet móvel cresce 82%#O número de acessos à tecnologia 3G cresceu 82% de julho de 2010 a junho deste ano, segundo dados da Agência Nacional de Telecomunicações (Anatel) compilados pela consultoria em telecomunicações Telemo. As conexões ativas de internet móvel tanto de celular quanto por uso de modem passaram de 14,6 milhões para 26,6 milhões nesses 12 meses...

Figura 3. Mensagem codificada para envio ao smartphone.

A aplicação mobile, ao receber a notícia, persiste no banco de dados o conteúdo e o título da notícia. Sua tela principal é atualizada exibindo junto as notícias armazenadas off-line, a nova recebida. Nesse ponto, como não há mais interação entre as plataformas, envia-se a última mensagem: comando\_fim\_app#. Ao recebê-la a aplicação de TVDI encerra-se fechando o socket de comunicação com o dispositivo Android.

#### 4. RESULTADOS

Os testes do *CineLua* e *LuaNews* foram realizados em um ambiente emulado (PC) com as seguintes configurações: Processador Intel Pentium E2180 (Dual Core), Mem. 2Gb, HD 320Gb, com adaptador Ethernet (conexão com taxa de 56Mbps) e Sistema Operacional Microsoft Windows 7 Professional.

Entre os principais resultados a serem destacados pode-se citar o tempo para a recuperação dos dados do web service: cerca de 3,0seg. Considerando as limitações do ambiente real do STB os resultados são aceitáveis. Vale ressaltar também o sucesso no controle da aplicação TVDI por meio do dispositivo Android, por meio da troca de mensagens entre os mesmos. As Figuras 4 e 5 ilustram, respectivamente, as telas para as aplicações *LuaNews* e *CineLua*.

#### 5. Conclusões e trabalhos futuros

Nesse projeto foram apresentados dois modelos que possibilitam tanto o acesso do Ginga-NCL/Lua a web services, bem como a interação de dispositivos móveis com o mesmo. No primeiro, o *CineLua* fez uso do módulo NCLua SOAP, consumindo os web services, propiciando o acesso a conteúdo web de forma independente de linguagem de programação, pois os dados são recebidos no padrão JSON, podendo ser formatados da melhor forma a se apresentar na tela da TV. O modelo flexibiliza-se para trabalhar com REST, de forma a atender o cenário da exibição de informações obtidas de diversas fontes em uma aplicação Ginga-NCL.

No *LuaNews* demonstrou-se como podem ser utilizados dispositivos móveis para interagir com a aplicação de TVDI. No experimento realizado com dispositivos com sistema operacional Android, foram verificados os princípios necessários para essa comunicação: localização dos dispositivos, conexão, e troca de informações de forma que um aparelho reflita seu estado no outro.

Dessa forma, surge como trabalho futuro o aprimoramento do *LuaNews*, com a construção de um protocolo de comunicação entre a aplicação da TVDI e a móvel, de forma que possa independente da plataforma móvel (Android, IOS, Windows Phone, Symbian, etc) para haver a comunicação. Para isso basta que o cliente específico da plataforma implemente o protocolo. Com isso, qualquer aplicação Ginga-NCL poderia comunicar-se com o cliente móvel, desde que ambos implementem o modelo.

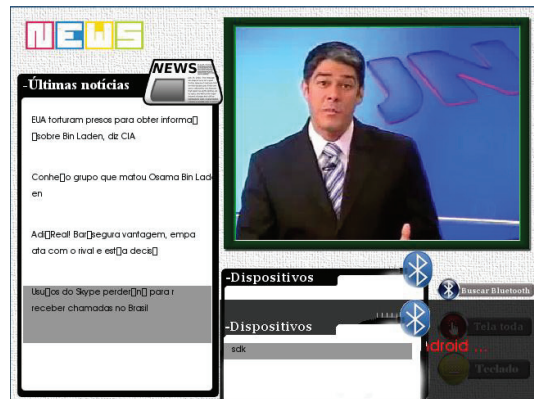


Figura 4. LuaNews: Notícia a ser enviada para o dispositivo.

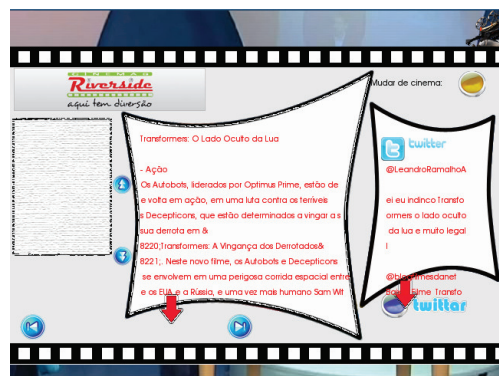


Figura 5. CineLua: Exibição dos filmes em cartaz.

#### 6. REFERÊNCIAS

- [1] VIANA, N.S.; LUCENA Jr., V. F. A Software Model Supporting the Management of Home Network Services through the Brazilian iDTV. In: 7th EUROPEAN CONFERENCE ON INTERACTIVE DIGITAL TELEVISION, EUROITV, 2009..
- [2] *Web Services Activity*. Disponível em: <<http://www.w3.org/2002/ws/>> Acesso em: 17/05/2011.
- [3] *NCLua SOAP: Acesso a Web Services em Aplicações de TVDI*. Disponível em: <<http://manoelcampos.com/tvd/nclua-soap/>> Acesso em: 29/03/2011.
- [4] Network support for the Lua language Disponível em: <<http://w3.impa.br/~diego/software/luasocket/>> Acesso em: 23/04/2011.
- [5] Guilherme Daher Ferreira; Guilherme Nogueira; Giovanni Comarella. *Ginga-NCL em Dispositivos Portáteis: Uma Implementação para a Plataforma Android - XVI Simpósio Brasileiro de Sistemas Multimídia e Web - WebMedia2010* Belo Horizonte, CE, Brasil 2010.
- [6] Romualdo M. de Resende Costa e Marcio Ferreira Moreno e Luiz; Fernando Gomes Soares. *Ginga-ncl: Suporte a múltiplos dispositivos*. In XV Simpósio Brasileiro de Sistemas Multimídia e Web - WebMedia2009, Fortaleza, CE, Brasil, 2009.