

ADAPTAÇÃO DE INTERFACES WEB PARA DISPOSITIVOS MÓVEIS

André A. dos Santos¹, Douglas T. S. Finkler¹, Giani Carla Ito², Maurício Ferreira², Nilson Sant'anna², Marilson M. dos Santos¹

¹Departamento de Sistemas de Informação
Universidade Paranaense (UNIPAR) – Cascavel, PR – Brasil

²Laboratório de Computação Aplicada
Instituto Nacional de Pesquisas Espaciais (INPE) – São José dos Campos, SP - Brasil

{andreaugusto_cvel, dtsfinkler}@hotmail.com,
{giito,nilson}@lac.inpe.br, mauricio@ccs.inpe.br,
marilson_ms@hotmail.com

Abstract. *This article will present the proposal of an architecture for creation of Web interfaces for mobile devices that it has as objective of create interfaces to adapt the characteristics of client. So that the process is possible, she will be used a proposal of World Wide Web Consortium (W3C), denominated Composite Capability Preference Profile (CC/PP).*

Resumo. *A proposta deste artigo é apresentar uma arquitetura para a Geração de Interfaces Adaptativas (GIA) que tem como objetivo criar interfaces que se adaptem às características de dispositivos móveis. Para que este processo seja possível utilizar-se-á uma forma de reconhecimento para dispositivos móveis proposta pela World Wide Web Consortium (W3C), denominada Composite Capability Preference Profile (CC/PP).*

1. Introdução

Com a revolução da computação móvel, o acesso à informação a qualquer instante e em qualquer lugar faz necessário que as aplicações desenvolvidas para a web sejam capazes de adaptar suas interfaces conforme as características dos diversos dispositivos móveis existentes. Segundo [Menkhaus & Pree, 2002] a nova variedade de dispositivos tem um profundo impacto na maneira como as interfaces de usuário de serviços baseados na web são construídas.

Neste contexto, diversos estudos estão sendo realizados em busca de interfaces mais dinâmicas envolvendo vários desafios, como ambientes heterogêneos, limitações físicas do aparelho, entre outras. As interfaces adaptativas se apresentam promissoras na tentativa de superar os problemas atuais de complexidade na interação homem-computador.

As páginas acessadas por dispositivos móveis são exibidas de forma desconfigurada e, dependendo da compatibilidade dos componentes da interface, algumas vezes podem não ser exibidas corretamente. [Kostakos & O'Neill, 2003] afirmam que os problemas relacionados à usabilidade são apresentados tomando uma interface desenvolvida para monitores de computadores de mesa, bem como uma

interface gráfica de um sistema operacional para estes tipos de computadores e colocando-o com um mínimo de alterações em um *Pocket PC* que possui uma tela de área pequena e dispositivos de entrada muito diferentes.

Este trabalho visa apresentar uma ferramenta para a Geração de Interfaces Adaptativas (GIA) que partindo de um modelo genérico tem como objetivo a criação de interfaces que se adaptem às características distintas de dispositivos móveis, retornando dinamicamente para cada cliente.

Utilizar-se-á uma forma de reconhecimento para dispositivos móveis proposta pela *World Wide Web Consortium* (W3C), denominada *Composite Capability Preference Profile* (CC/PP) que tem a finalidade de proporcionar um mecanismo estruturado e universal para descrever e transmitir informações sobre as características de um dispositivo móvel para um servidor.

Este artigo está dividido em seis seções descritas a seguir. A seção 2 abordará os trabalhos relacionados, a seção 3 o Repositório de Perfis CC/PP, a seção 4 sobre a Arquitetura para Geração de Interfaces Adaptativas (GIA), a seção 5 abordará o Ambiente Visual e Técnica de Desenvolvimento, a seção 6 apresentará os Testes Realizados e a seção 7 a Conclusão.

2. Trabalhos Relacionados

No campo das aplicações, muito se tem discutido a respeito de associar às unidades móveis o trabalho de adaptação e mecanismos que permitem identificar o ambiente, dividindo a responsabilidade entre cliente e servidor.

[Coninx, et. al. 2003], apresentou a arquitetura *Dynamically Generating Interfaces for Mobile and Embedded Systems* (DIGYMES) composta por um *framework* que tem como finalidade a criação e *design*, de um sistema interativo para diversos tipos de dispositivos móveis e embutidos, utilizando o conceito de separação entre as camadas lógica e de apresentação. Tem como proposta principal, a criação de uma interface que preserve a consistência e usabilidade. Estas interfaces podem se adaptar a diversos tipos de aparelhos, mantendo a mesma funcionalidade, sem precisar ser redesenhada. Uma das limitações do DIGYMES é que o mesmo não garante uma apresentação visual agradável da interface perante a migração da mesma para dispositivos móveis, alegando que desta forma o sistema apresenta maior flexibilidade.

[Menkhaus 2002] desenvolveu o *Multi User Interfaces Single Application* (MUSA), que tem como objetivo a redução no tempo de desenvolvimento, custos, melhora da manutenção e flexibilidade de utilização, através da separação em camadas da interface, aplicação lógica e comunicação. O sistema MUSA suporta a adaptação envolvendo grande diversidade de dispositivos e usando várias estratégias de adaptação.

Os trabalhos estudados, MUSA descrito por [Menkhaus 2002] e DIGYMES por [Coninx, et. al. 2003], utilizaram o conceito de separação por camadas, fazendo uma clara distinção entre o modelo de implementação de cada nível: interface do usuário e modelo lógico. Ambos têm como objetivo a adaptação da interface do usuário no ambiente móvel, mas utilizam técnicas diferentes. Os autores argumentam uma forma de desenvolvimento que seja menos dependente de propriedades de um dispositivo único, justificando desta forma a necessidade do conceito de adaptação da interface do usuário.

As arquiteturas propostas apresentam-se limitadas no sentido de considerar aspectos como modelo lógico, usuários e modelos de plataformas, deixando sem

modelagem os aspectos contextuais como, por exemplo, a identificação em tempo de execução do tipo de dispositivo móvel.

3. Repositório de Perfis CC/PP

O *Composite Capabilities/Preferences Profile* (CC/PP) é uma especificação do W3C para expressar características dos dispositivos e das preferências de usuários. O perfil CC/PP é uma descrição das potencialidades do dispositivo e das preferências do usuário de acordo com [Hanumansetty 2004]. Tem como objetivo proporcionar um mecanismo estruturado e universal para descrever e transmitir informações sobre as capacidades de um cliente Web para um servidor, de forma que o conteúdo seja direcionado a essas características. Os perfis podem estar armazenados em um servidor Web local ou remoto, segundo [Butler 2002].

De acordo com [Sila e Swick 1999], o CC/PP é baseado no *Resource Description Framework* (RDF), que é uma linguagem utilizada pela W3C para modelagem de metadados e descrição de documentos *eXtensible Markup Language* (XML), permitindo uma maior flexibilidade na criação de novos vocabulários. O perfil CC/PP é construído em uma hierarquia de dois níveis, o primeiro chamado de componentes de um perfil e o segundo de atributos de um perfil.

Para que sejam tratadas as requisições de dispositivos que possuam perfis CC/PP, existe uma biblioteca denominada *Delivery Context Library* (DELI) que foi desenvolvida pela HP (*Hewlett Packard*). [Silva, Ribeiro e Ito 2005] descrevem um exemplo de uso e aplicação do DELI integrado ao CC/PP.

4. Arquitetura para Geração de Interfaces Adaptativas (GIA)

A geração de interfaces proposta pela arquitetura GIA utiliza a descrição de uma interface genérica que tem como finalidade principal atender as solicitações de diversos usuários a uma mesma aplicação considerando o tipo de dispositivo móvel para responder devidamente a solicitação em tempo de execução.

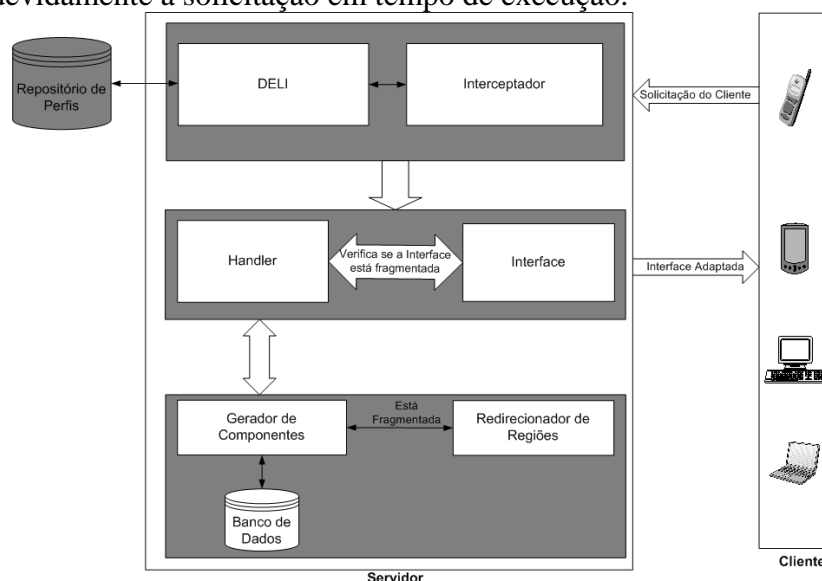


Figura 1 - Arquitetura para Geração de Interfaces Adaptativas

Outro fator de destaque é permitir uma implementação que esteja voltada a um ambiente multi-plataforma, ampliando assim, a relação um para um, para uma relação

um para n, ou seja, uma aplicação pode gerar várias interfaces. O processo de adaptação ocorre durante a solicitação facilitando o processo de desenvolvimento de interfaces, pois um único projeto de interface poderá ser exibido em múltiplos tipos de dispositivos.

A figura 1 ilustra a arquitetura GIA composta por dois níveis: cliente e servidor; onde os clientes podem fazer solicitações de serviços munidos de diversos tipos de PDA's dentre eles *Palms*, *Pocket PC's* ou *Smartphones*. No servidor estão localizados o interceptador, o *framework* DELI, o redirecionador de regiões, o gerador de componentes, o *handler* e o banco de dados.

Para exemplificar um cenário de utilização da arquitetura GIA, supõe-se que o usuário faz uma solicitação à aplicação descrita em JSP através de um dispositivo móvel ou computador pessoal. O interceptador captura-a e faz uma chamada ao DELI, para identificar o dispositivo e localizar no repositório de perfis externo as informações relevantes, tais como, o tamanho de tela, *browser*, sistema operacional, etc. Após este processo, o interceptador envia os parâmetros para o *handler* que por sua vez ativará o gerador, o qual usará as informações dos componentes armazenados em banco de dados para converter através de *tag-libs*, o código para a linguagem apropriada ao dispositivo.

Caso a interface esteja fragmentada o redirecionador inserirá na página o algoritmo que possibilitará a navegação entre as regiões e a interface será exibida no dispositivo solicitante.

5. Ambiente Visual e Técnica de Desenvolvimento

Conforme a arquitetura proposta na seção 4, foi implementada na linguagem *Object Pascal* através do *Borland Delphi 7* um ambiente visual que agrega a funcionalidade da arquitetura GIA e requisitos funcionais para a criação de interfaces adaptativas. A linguagem de programação utilizada para o desenvolvimento nesta ferramenta será a *Java Server Pages* (JSP).

Ao iniciar a construção da página o desenvolvedor deverá utilizar uma técnica de programação que propõe a divisão do código em regiões mapeando a interface de forma hierárquica, mantendo uma relação lógica entre elas de acordo com a funcionalidade atribuída, que tem por objetivo a intercalação do código.

Cada região poderá ser composta por sub-regiões e também por componentes visuais como textos, botões, *links*, *combos*, imagens, etc. Foram determinados 9 tamanhos de região, numa escala aritmética de 100 a 500 *pixels* com coeficiente de variação de 50. Para determinar o tamanho mínimo e máximo de uma região analisou-se que o menor tamanho da tela dos dispositivos móveis, incluindo celulares e PDA's, aproximam-se de 100 *pixels* e o maior de 500 *pixels*. Caso o tamanho da tela do dispositivo ultrapasse 500 *pixels* a interface não será fragmentada, sendo mostrada na íntegra. Para viabilizar o processo de adaptação, foi estabelecido que seja mostrada a menor região que mais se aproxima da resolução do aparelho.

A arquitetura GIA permite utilizar durante o processo de desenvolvimento uma forma de reduzir o conteúdo que será visualizado no dispositivo móvel, pois as restrições relacionadas ao tamanho da tela do dispositivo tornam necessárias em alguns casos, a classificação das informações. A seleção do conteúdo que será exibido é determinada de acordo com as delimitações das regiões.

Durante o processo de desenvolvimento, a interface poderá ser visualizada de forma completa ou fragmentada, como ilustra a figura 2.



Figura 2 – Formas de visualização da interface

Para que seja possível o processo de fragmentação de uma página pelo *handler* da ferramenta, foram definidas *tag-libs*, que delimitam e identificam as regiões determinadas pelo desenvolvedor durante a implementação da interface, conforme o exemplo a seguir:

```
<prefix:regiao id="topo" seq="2">
  <prefix:componente tipo="fonte" propriedades="size=2; color=#000000" >
    Exemplo de Identificação de Região
  </prefix:componente>
</prefix:regiao>
```

As *tag-libs* têm como função também representar de forma genérica o respectivo componente de cada dispositivo. Quando utilizadas, o gerador de componentes os substituem por componentes válidos convertendo os modelos utilizados na estrutura da interface por um modelo cadastrado em banco que satisfaça necessidades do cliente.

6. Testes Realizados

Para a realização da análise proposta foi desenvolvida uma interface escrita em JSP. Para tal foi configurado o servidor *Web Apache TomCat 5.5*, o *Internet Explorer 6.0* da máquina local e os simuladores *Openwave Phone Simulator*, de celular e *Palm Os Simulator*, da Palm.

Observa-se na figura 3 que a exibição no navegador *Internet Explorer 6.0* na resolução de *1024x768 pixels* mostra que a página está adaptada conforme as características deste *browser*.

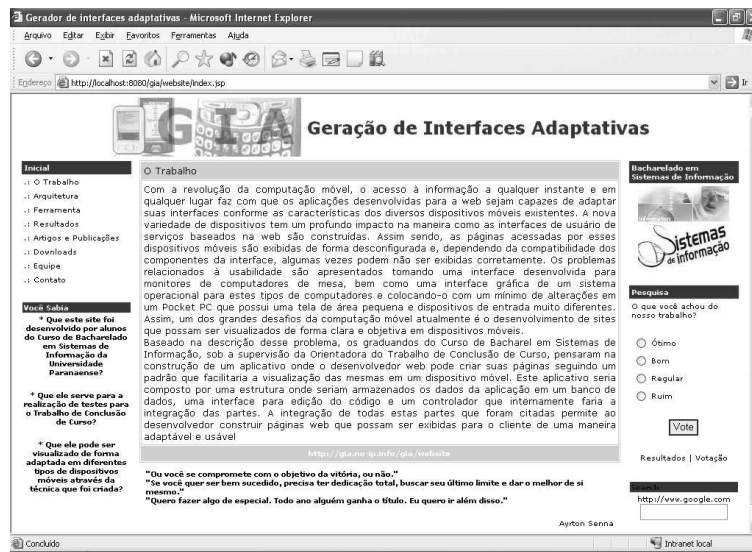


Figura 3 – Interface visualizada no *browser Internet Explorer 6.0*

A partir da captura de telas do *Palm OS Simulator*, observa-se na figura 4 que a interface também se adaptou as características do dispositivo. Devido à tela do *Palm* ser maior em relação a do *Openwave*, o conteúdo visualizado na tela é maior, o que resulta na menor rolagem da barra horizontal. O *handler* inseriu no topo e no final de cada região um menu que possibilita a navegação entre as telas.

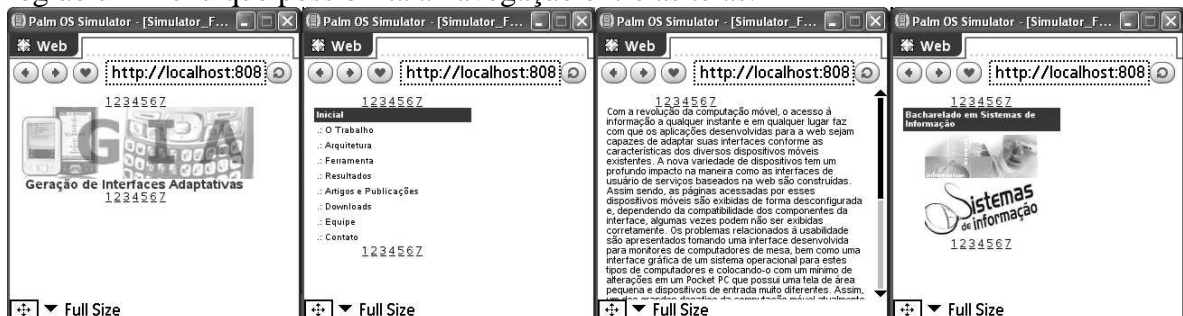


Figura 4 – Interface visualizada no *Palm OS Simulator*

Na mesma interface através do simulador *Openwave Phone Simulator*, observa-se através da figura 5 que o *site* foi adaptado às características do dispositivo pois não gerou barra horizontal e o conteúdo pôde ser visualizado de uma maneira clara e objetiva.

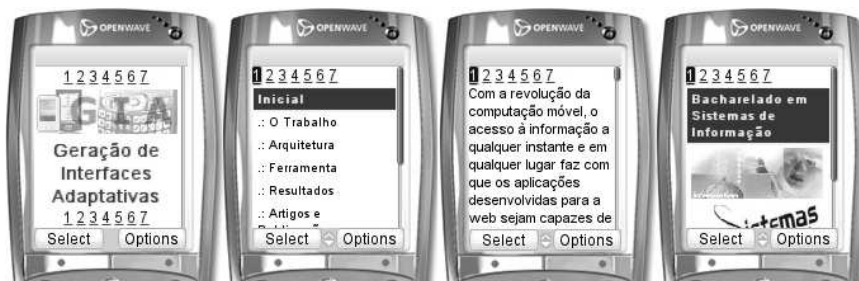


Figura 5 – Interface visualizada no *Openwave Phone Simulator*

7. Conclusão

Com o intuito de contribuir para uma maior facilidade de desenvolvimento no ambiente móvel este trabalho apresentou uma Arquitetura para Geração de Interfaces Adaptativas

tendo como principais vantagens o processo de adaptação da interface ao tipo de dispositivo móvel em tempo de execução, bem como a facilidade de desenvolvimento, pois um único código poderá ser visualizado em múltiplos dispositivos.

Os resultados obtidos através da implementação de estudos de caso apresentaram-se satisfatórios no contexto da adaptação de interfaces Web em um ambiente de desenvolvimento. Para tal, utilizou-se uma técnica de fragmentação do código em regiões que por meio de uma aplicação são adaptadas conforme o tipo de dispositivo que realiza uma requisição HTTP (*Hiper Text Transfer Protocol*).

8. Referências

- Butler, M. H. (2002) DELI: A Delivery Context Library for CC/PP and UAProf. Disponível em: <http://www.hpl.hp.com/personal/marbut/DeliUserGuideWEB.htm>. Acessado em 19/03/2006.
- Coninx, K., Luyten, K., Vandervelpen, C., Van Den Bergh, J., and Creemers, B. (2003) Dygimes: Dynamically Generating Interfaces for Mobile Computing Devices and Embedded Systems. In Chittaro, L., editor, *Mobile HCI*, volume 2795 of *Lecture Notes in Computer Science*, Springer.
- Hanumansetty, G. Reena. (2004) *Model Based Approach for Context Aware and Adaptive User Interface Generation*. Master's Thesis, Virginia Polytechnic Institute and State University, Falls Church, Virginia, USA.
- Hewlett Packard Labs. Disponível em <http://www.hpl.hp.com/personal/marbut>. Acessado em 10/06/2005.
- Kostakos V., O'Neill, E. (2003) *A Directional Stroke Recognition Technique for Mobile Interaction in a Pervasive Computing World*. Department of Computer Science, University of Bath.
- Menkhaus, G. (2002) *Adaptive User Interface Generation in a Mobile Computing Environment*. PhD Thesis, University of Salzburg, Austria.
- Menkhaus, G., Pree, W. (2002) *User Interface Tailoring for MultiPlatform Service Access*, Disponível em <http://portal.acm.org/citation.cfm?id=502760>. Acessado em 20/04/2006.
- Openwave Developer Network. Disponível em <http://developer.openwave.com/dvl/member/downloadManager.htm?softwareId=2>. Acessado em 10/02/2006.
- Palm OS Simulator. Disponível em <http://www.palmos.com/dev/tools/simulator/>. Acessado em 13/04/2006.
- Sila, O., Swick, R. (1999) *Resource Description Framework (RDF) Model and Syntax Specification*, Disponível em <http://www.w3.org/1999/status/REC-rdf-syntax-19990222/status>. Acessado em 13/06/2005.
- Silva, A. C. B., Ribeiro, N. N. T. J.; Ito, G. C. (2005) *Análise de um Repositório de Perfis para Reconhecimento de Dispositivos Móveis*. Anais, XII Simpep.
- World Wide Web Consortium (W3C). Disponível em <http://www.w3.org/>. Acessado em 09/03/2006.