

# Adaptação de conteúdos Web para o ambiente WAP

Arlindo Santos

Instituto Politécnico de Bragança, Bragança, Portugal

[acsantos@ipb.pt](mailto:acsantos@ipb.pt)

## Resumo

Um dos actuais desafios da Internet surge a partir do momento que a tecnologia WAP permite que os dispositivos móveis acedam à informação disponível na Internet. No entanto apenas podem aceder a conteúdos multimédia criados de acordo com uma especificação própria, que difere daquela que é usada para disponibilizar conteúdos para os computadores pessoais. E como um dos actuais entraves para o sucesso da tecnologia WAP é sem dúvida a escassez de conteúdos quando comparada com quantidade de informação Web, é necessário criar mecanismos de enriquecimento através da adaptação da informação existente na Web ou simplesmente através do desenvolvimento de conteúdos próprios. Deste modo foi desenvolvida uma aplicação que efectua a adaptação automática de conteúdos da Web para o ambiente WAP com resultados positivos..

**Palavras chave:** Web, WAP, XSL, WML, HTML, XML, XHTML, acesso móvel, conversão, segmentação.

## 1 Introdução

O acesso à Internet a baixo custo a partir das redes wireless e a explosão do número de dispositivos móveis com capacidade de comunicação estão a provocar uma mudança na forma como se desenvolvem as aplicações informáticas e nas expectativas dos utilizadores relativamente às funcionalidades que estes oferecem [Silva 1999].

A Internet móvel adoptou o protocolo Wireless Application Protocol (WAP) para estender a Internet para além dos computadores pessoais. Embora muito se tenha falado sobre as suas limitações e falhas, tal facto também ocorreu com a Internet na sua fase inicial [Nielsen 2000], e hoje ela tornou-se uma ferramenta fundamental para as mais diversas actividades.

O objectivo primordial desta tecnologia é fornecer informação aos utilizadores móveis, em qualquer lugar e a qualquer hora, mas a quantidade da informação ainda são pouco significativas. Em Abril de 2000 o número de páginas WAP estimava-se entre 50 000 e 1 500 000 [Arehart 2000] e, segundo o WAP Forum [WAP 2001], em Maio de 2001 existiam cerca de 12 000 sites WAP em mais de 100 países. Em contrapartida, estima-se em cerca de 8 milhões o número de web sites em 2001 [OLC 2001], e num futuro próximo, de acordo com a IDC [IDC 2001], o número de páginas web ascenderá a cerca de 8.000 milhões. A acrescentar a estes números, foi efectuada uma pesquisa pela palavra “HTML” e “WML” em [www.google.com](http://www.google.com), no dia 18 de Dezembro de 2001, e obteve-se um universo de 270 000 000 páginas que tinham associadas a palavra “HTML” e 549 000 páginas associadas à palavra “WML”, numa razão de aproximadamente 500:1. Estes dados reflectem o estado de imaturidade da Internet móvel [Arehart 2000] comparado com a actual Internet.

De acordo com estes dados pode-se concluir que a probabilidade de um utilizador móvel desejar obter informação (HTML) da Web num formato não suportado pelo telemóvel (WML) é significativa.

## 2 Adaptação de HTML para WML

Falar da adaptação dos conteúdos existentes na Web para os dispositivos WAP é falar da adaptação da linguagem de normalização de conteúdos da Web, o HTML, para a linguagem normalizada para o ambiente WAP, o WML. Mas a adaptação pode ser utilizada noutros domínios, nomeadamente a adaptação de qualquer documento para outro formato diferente do original. Este processo faz parte do quotidiano de qualquer pessoa que utiliza um computador pessoal, como por exemplo converter um documento da Microsoft Word 97 para Microsoft Word 2000, ou então para o formato PDF, como se pode ver na figura 1. Estas tarefas são realizadas ocasionalmente, uma a uma e não de forma automática. Mas se o utilizador pretendesse converter  $n$  documentos, com  $n$  suficientemente grande, tornava-se imperativo desenvolver um processo automático, com ou sem a ajuda do utilizador, para reduzir o tempo de conversão.

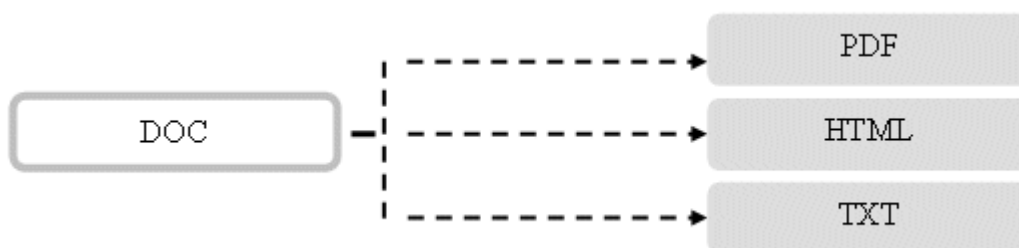


Figura 1 – Conversão de um documento em vários formatos

Como a quantidade dos conteúdos disponíveis no formato WML são pequenas quando comparada com o número de páginas HTML, tornar-se-ia vantajoso tanto para os utilizadores como para os fornecedores de conteúdos móveis, que estas páginas fossem acessíveis aos dispositivos WAP.

### 2.1 O estado da arte

Os fornecedores de conteúdos Web estão a ser pressionados para converterem os conteúdos existentes do formato HTML para o formato WML de forma a serem visualizados nos dispositivos móveis. Esta tendência inicial de “injecção” de conteúdo no formato WML está errada segundo Nielsen [Nielsen 2000], e sacrifica a usabilidade para reduzir tempo de comercialização. A maioria dos web sites disponibilizados no formato WML são desenvolvidos à custa da simples conversão das páginas de HTML desenhadas para computadores pessoais [Notess 2000], sem ter em consideração as diferenças entre os vários tipos de dispositivos de saída.

Segundo estudos [Nielsen 2000] realizados nesta área, o uso dos telemóveis não é um entrave à utilização da tecnologia WAP, mas sim a falta de serviços. Uma razão para o fraco desempenho é que os programadores não têm em consideração o modelo WAP. De acordo com um documento do grupo W3C de 15 Março de 1999 [W3C 2001], há muitas características do telemóvel que diferem do computador pessoal, tanto em termos de restrições de hardware e largura de banda, como ao nível da interface.

Em contraste com a grande quantidade de páginas HTML, só algumas páginas WML estão actualmente disponíveis, um pouco à semelhança das dificuldades encontradas pela Web no início os anos 90 [Nielsen 2000]. Também existe uma falta de consonância entre os vários fabricantes de dispositivos, resultando numa incompatibilidade séria entre os vários modelos [WAP 2001]. Devido a este facto, manter páginas WML acarreta mais custos pois é necessário criar os *decks* de forma a serem visualizados pelos diferentes modelos.

## 2.2 Métodos de conversão

Existem milhões de páginas HTML na Internet, e algumas delas são úteis para os utilizadores móveis. Então, a solução seria aproveitar essa informação e fornece-la aos dispositivos móveis, utilizando para o efeito um filtro de informação.

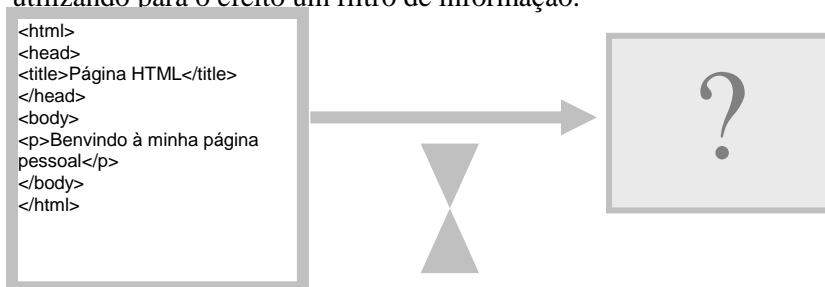


Figura 2 – Filtro de informação HTML/WML

Embora já existam no mercado ferramentas comerciais e não comerciais que permitem filtrar a informação ou, por outras palavras, realizar a conversão entre os vários formatos de linguagem de *markup*, essas não representam um processo mágico [Rysavy 2001], pois não será efectivamente fácil condensar a grande quantidade de informação visualizada em ecrãs com dimensões superiores a 640x480 *pixels* e com milhões de cores para os pequenos ecrãs monocromáticos dos dispositivos móveis.

Segundo Arehart [Arehart 2000], existem actualmente quatro perspectivas sobre a adaptação do conteúdo da Web para o WAP: automática, manual, personalizada e integrada.

### (1) Automática

O conversor automático envolve o desenvolvimento de software que possa converter um documento projectado para um dispositivo de saída diferente do dispositivo móvel e adaptá-lo de forma a ser visualizado neste dispositivo. As vantagens advêm do facto de acelerar o processo da comercialização, do baixo custo e de serem independentes do desenho original da página HTML. As desvantagens também são visíveis na medida em que adicionam tempo de processamento e a avaliação da usabilidade no resultado final é imprevisível.

Os conversores automatizados podem ser classificados de acordo com a inclusão ou não de configuração. Empresas como a Phone.com [Phone 2001] e a Argo ActiGate [Argo 2001] incluem conversores automáticos no Gateway WAP cujos algoritmos são os mesmos para todas as páginas. Este processo diminui o tempo de comercialização e o tempo de processamento. A configuração de um conversor pressupõe que cada página HTML é tratada como única, com o intuito de produzir melhores resultados. As ferramentas Oracle Portal-to Go [Oracle 2001], Spyglass Prism [OpenTV 2001] e Orchid WebShaper [IBM 2001] enquadram-se neste tipo.

### (2) Manual

O conversor automático será sempre uma solução temporária e, com o decorrer do tempo, as soluções que utilizam XML para descrever o conteúdo e os documentos XSL para descrever a apresentação farão parte do quotidiano dos fornecedores de conteúdos.

Para disponibilizar conteúdos para os dispositivos WAP, o programador deve escrever outras páginas, estáticas ou dinâmicas, de forma a fornecer uma estrutura e conteúdo de acordo com o meio de divulgação. Embora esta solução seja fácil para web sites pequenos, para web sites complexos, a solução acarreta um aumento substancial dos custos.

### (3) Personalizada

Esta solução permite aceder a uma página HTML e ao mesmo tempo o utilizador tem acesso ao *deck* WML correspondente embutido no código HTML. Assim que a página HTML é obtida, uma aplicação procura uma anotação que indica se a página possui *tags* personalizadas para a tradução. No caso de resposta afirmativa, é carregado o código personalizado que identifica o documento WML.

### (4) Integrada

Esta solução combina o método automático e manual (XML e XSL) que oferece garantias para uma razoável adaptação das páginas HTML para páginas WML. O software da Sun Microsystems iPlanet Wireless Server™ [iPlanet 2001] segue esta linha de orientação, permitindo ainda a configuração personalizada.

Esta solução também pode ser adoptada no lado do programador quando existe necessidade de desenvolver páginas para outro formato. O desenvolvimento de um produto que dá ao programador a possibilidade de conhecer apenas as características básicas da linguagem e é responsável pela conversão automática seguindo determinadas regras, configuradas ou não pelo programador, permite que o processo de desenvolvimento de conteúdos em formato WML seja facilitado com a diminuição do tempo de execução e, como consequência, o número de páginas disponíveis aumentaria.

## 2.3 Problema da segmentação

Além da conversão descrita, há um item problemático: a segmentação do conteúdo resultante. O tamanho de cada *deck* WML, devido às limitações do dispositivo (memória, processamento, hardware, etc.), deve ser um elemento a ter em consideração. Por exemplo o Nokia 7110 apenas pode movimentar cerca de 1,4 KB de informação compactada, o Nokia 9210 cerca de 8 KB. Ao invés, os browsers dos computadores pessoais não limitam o tamanho da página. Isto levanta uma pergunta pertinente: como proceder à tarefa da segmentação? Seguindo a lógica, só é possível segmentar o conteúdo de forma adequada para cada dispositivo cliente se à priori existir informação sobre a sua capacidade. Porém, se o dispositivo actualizar as suas características haveria necessidade de actualizar a aplicação. Na prática é adoptado o limite do dispositivo menos capaz [Móro 2001].

Segmentar o conteúdo pressupõe dividi-lo em vários *decks* para apresentar o resultado nas condições desejadas. Como cada *deck* WML pode ser dividido em vários *cards*, a limitação também se lhes aplica.

Browser	Tamanho máximo
UP.Browser 3.2	1.492 Bytes
UP.Browser 4.x	2.048 Bytes

Ericson R320	Aproximadamente 3.000 Bytes
Ericson R380	Aproximadamente 3.500 Bytes
Ericson MC218	Mais de 8.000 Bytes
Nokia 7110	Aproximadamente 1.500 Bytes

Tabela 1 – Limitação do tamanho do *deck* nos dispositivos WAP

## 2.4 Técnicas aplicadas à conversão

A conversão passa pela aplicação de um conjunto de técnicas que têm como objectivo entregar ao utilizador final apenas informação útil. Sob este ponto de vista é imperativa a remoção de informação não relevante e, ao mesmo tempo, a informação a visualizar deve ser adequada às características do dispositivo de saída. Assim a conversão deve considerar algumas restrições [W3C 2001] [Openwave 2001] [Hjelm 2000]:

### (1) Níveis de abstracção

Os dados devem ser fáceis de visualizar a um nível superficial, assim como é necessário criar mecanismos de acesso rápido a um nível com menor abstracção. Se tomarmos como exemplo o acesso à informação noticiosa, os títulos das mesmas devem estar disponíveis numa primeira fase, e apenas é visualizado o resumo da notícia quando o utilizador pretende mais informação menos abstracta como o resumo. E finalmente poderá aceder ao corpo da mesma.

### (2) Conversão da estrutura

Devido às diferentes características de visualização, é necessário ignorar a estrutura de cada página HTML. Considerando que uma página HTML está dividida em várias secções, a conversão é relativamente fácil: a cada secção corresponde um *card* e usam-se *hyperlinks* para navegar entre os vários *cards*. Porém, devido à limitação de tamanho de cada *deck*, este método pode não ser apropriado.

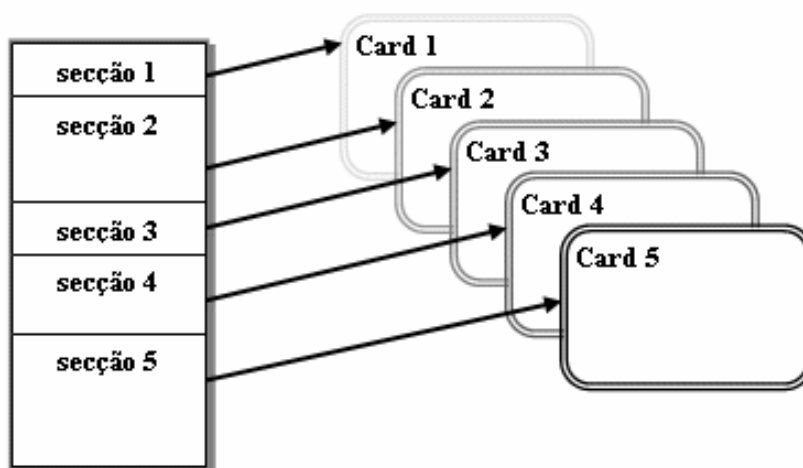


Figura 3 – Estrutura de conversão de uma página HTML bem estruturada para um deck WAP

No entanto, a maioria das páginas HTML presentes na Internet ou não estão correctamente estruturadas ou é impossível definir secções. A violação sintáctica do HTML pode ser corrigido por algum software, como por exemplo o HTML Tidy [Raggett 2001]. Porém existem ambiguidades semânticas que necessitam do envolvimento do programador WAP, caso

contrário o resultado da conversão poderá oferecer páginas que não obedecem às exigências dos dispositivos e poderá comprometer a apresentação da informação.

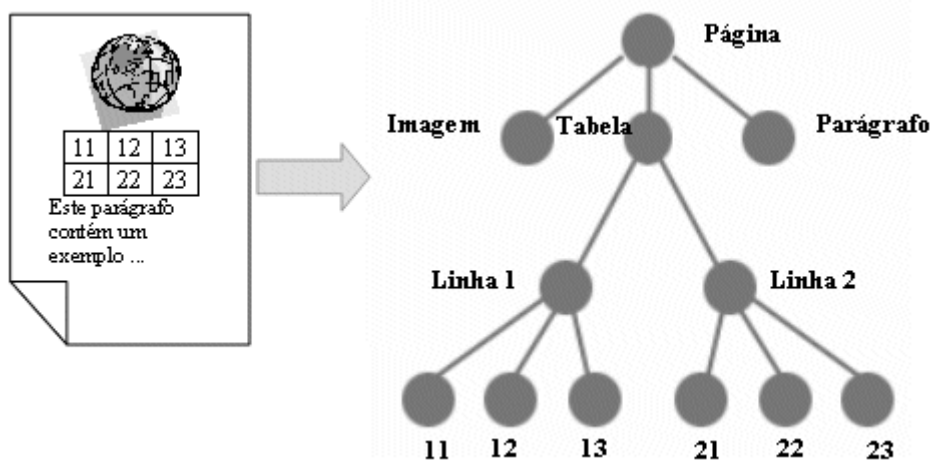


Figura 4 – Estrutura de conversão de uma página HTML mal estruturada para um *deck* WAP

O conteúdo das páginas HTML pode ser estruturado segundo a tipologia de uma árvore, em que cada *nodo* representa um elemento HTML como ilustra a figura 4. Desta forma é possível criar mecanismos que mantêm a estrutura da informação e alteram a estrutura de apresentação.

### (3) Formulários

Os dispositivos WAP também suportam formulários, mas estes têm a limitação do mecanismo de entrada de dados, sendo necessário minimizar a acção do utilizador.

### (4) Informação associada à apresentação

Os dispositivos móveis, devido ao seu ecrã pequeno, limitam a capacidade de visualização e por conseguinte o uso de CSS torna-se irrelevante. Assim, o conversor deve ter em atenção que o conteúdo deve ser legível sem CSS.

### (5) Tabelas

O uso de tabelas gera na maioria dos dispositivos móveis perda de informação pois estes não têm *scroll* horizontal. Se para tabelas simples o processo de reestruturação é relativamente fácil, o mesmo não se pode dizer para tabelas com algum grau de complexidade. Para tabelas com uma estrutura simples pode-se aplicar uma técnica que consiste em incorporar num único *card* os *hyperlinks* para cada coluna de informação a apresentar num novo *card*.

O Digester [Bickmore 99] é um sistema que converte automaticamente conteúdo de documento Web para dispositivos móveis (PDAs, dispositivos WAP, etc.), e aplica uma técnica que consiste em considerar cada célula da tabela como um sub-documento.

1	2	3
	4a	
	4.1	
	4.2	
	4b	

	5	6
--	---	---

Tabela 2 – Sub-documentos HTML com tabelas

### (6) Frames

As *frames* podem ser úteis, mas nos dispositivos com ecrãs pequenos só é possível visualizar uma *frame* de cada vez.

Caso as *frames* sejam usadas nas páginas HTML, é necessário criar mecanismos de visualização dos conteúdos, como por exemplo atribuir a cada *frame* um nome para que o utilizador a possa seleccionar e aceder à informação.

### (7) Imagens

Nem todos os dispositivos móveis suportam imagens e, mesmo naqueles que as suportam, são pequenos ícones monocromáticos com informação visual deteriorada. Existe software disponível com capacidade de conversão das imagens suportadas pela Web para o formato WBMP. Porém a conversão e apresentação devem ter em consideração que estas são limitadas em tamanho, devido às limitações do ecrã e do *deck*.

Normalmente a técnica a aplicar passa pela substituição da *tag* <IMG> pelo atributo *alt*, ou então redimensionar a imagem de acordo com o dispositivo de saída.

### (8) Texto

Devem-se usar frases curtas e simples de modo a revelar o nível seguinte de abstracção e evitar que o utilizador aumente de nível. Este é um princípio básico da acessibilidade na Internet [Hsu 1994] e também deve ser um princípio básico nos dispositivos móveis.

A maioria dos ecrãs dos dispositivos WAP só é capaz de exibir aproximadamente 20 caracteres por linha. No entanto, o resultado da conversão pode gerar parágrafos ou frases de texto excessivamente grandes, sendo necessárias múltiplas acções do utilizador para visualizar todo o conteúdo.

Alguns browsers WAP são capazes de exibir texto com fluxo automático. Mas outras técnicas podem se aplicadas de forma a minimizar a intervenção do utilizador, como por exemplo dividir o texto em vários *cards* ou para cada parágrafo, substituir o texto inicial por um *hyperlink* que aponta para o texto integral.

### (9) Hyperlinks

Uma grande vantagem das páginas HTML são os *hypelinks*, que permitem manter uma ligação aberta para outros web sites. Num PC com o apontador do *mouse* pode-se facilmente activar um *hyperlink*, no entanto, nos dispositivos WAP esta operação torna-se mais incómoda.

Alguns conversores automáticos produzem *cards* de WML com texto e *hyperlinks* misturados. Uma das técnicas a aplicar será extrair os *hyperlinks* para um *card* único e criar mecanismos de acesso ao *card*, permitindo que o utilizador visualize o conteúdo textual separado dos *hyperlinks*.

### (10) Scripts e eventos

Os dispositivos têm limite de memória e processamento e muitos *scripts* e eventos não são suportados, como o JavaScript. Existe uma solução que passa pela conversão das linguagens de *script* para permitir a sua adaptação ao novo ambiente. Esta tarefa é crucial quando o programador Web faz uso do *script* para manipular os dados do formulário. No entanto, este processo é irrealista se pensarmos que cada programador tem o seu estilo de programação e se

pensarmos que a maior parte dos *scripts* de cliente apenas têm acção directa sobre a apresentação.

### 3 Requisitos de um Sistema de Conversão

Existe um conjunto de exigências naturais que devem servir como orientadoras no desenvolvimento de um sistema de conversão:

- Apresentação dos *decks* WML em vários *cards* tendo em atenção as normas da acessibilidade e usabilidade;
- Os *decks* de WML devem ser limitados em tamanho;
- Converter os elementos multimédia presentes no documento HTML para o documento WML;
- Qualquer que seja o documento de entrada, o documento de saída deve estar formatado correctamente.

Fundamentalmente estamos perante a existência de duas linguagens diferentes. Uma (HTML) que é altamente irregular e outra (WML) que foi projectada de acordo com regras bem delineadas. Então um dos requisitos do sistema é tornar o formato dos documentos Web “inteligentes”, isto é, reestruturar a informação de forma a ser bem interpretada.

Um sistema de adaptação a desenvolver poderá abranger diferentes áreas de utilização: aplicações *offline*, módulo embutido no Gateway WAP, ou um serviço fornecido pelo servidor WAP.

- O desenvolvimento de uma aplicação *offline* é útil para o programador se o objectivo é efectuar a conversão automática ou semi-automática do conteúdo dos web sites criados. Um sistema desta natureza facilita o trabalho de duplicação da informação para os dois tipos de dispositivos: PC e telemóveis.
- O módulo para o Gateway WAP converte o conteúdo dos documentos HTML e outros elementos multimédia embutidos, sempre que o utilizador os pretenda visualizar no seu dispositivo uma página HTML.
- Por fim, pode ser disponibilizado um serviço aos utilizadores WAP que tenham interesse em visualizar as páginas HTML de forma automática. Desta forma, sempre que um utilizador pretenda aceder a informação Web deverá, na fase inicial, aceder ao servidor que fornece este serviço e efectuar a escolha da página HTML.

### 4 Conversor HTML2WML

Com base nos três exemplos práticos apresentados na secção anterior, a aplicação desenvolvida tem como objectivo fornecer o serviço WAP. Assim, foram desenvolvidos vários módulos para atingir esse objectivo.

1. Formulário para introdução da URL;
2. Acesso à página HTML;
3. Conversão para XHTML;
4. Conversão de XHTML para WML;
5. Segmentação em vários *decks* e *cards*;

## 6. Visualização do resultado.

### (1) Formulário para introdução da URL

Com o apoio de uma ferramenta que permite a edição de documentos WML, implementou-se um documento que serve para introdução da URL da página HTML. O desenho seguiu as normas de acessibilidade e usabilidade para os dispositivos WAP

O servidor recebe o pedido do formulário e, depois de preencher o campo da URL, o utilizador dá ordem para iniciar o processo de conversão.

### (2) Acesso à página HTML

A aplicação acede ao servidor Web para fazer o pedido da página HTML, a qual é enviada para o servidor WAP. De salientar que é efectuado apenas um pedido, ou seja, os elementos multimédia incorporados na página, tal como as imagens, são ignorados.

### (3) Conversão para XHTML

Este módulo consiste em converter o documento mal formatado num documento bem formatado: em XHTML. O XHTML usa os mesmos elementos e atributos que o HTML, sendo apenas necessário actualizar a sintaxe.

Este módulo desenvolvido em ASP utiliza o ActiveX designado por TidyCOM [Blavier 2001], que é recomendado pelo W3C como uma possível solução para efectuar esta tarefa. Este componente realiza um conjunto de tarefas que têm como objectivo tornar o documento “inteligente”, isto é, formatado de acordo com as regras de um documento XML.

### (4) Conversão de XHTML para WML

O quarto módulo efectua a conversão entre duas linguagens com regras bem definidas e comuns. Tendo em atenção que o documento XHTML contém a informação do conteúdo e da apresentação para os PCs, o objectivo neste passo é apresentar o conteúdo ao utilizador no formato WML para os dispositivos WAP.

O desenvolvimento deste módulo abordou duas perspectivas. Na primeira abordagem foi utilizada uma tabela de correspondências directa entre o XHTML e o WML e, fazendo uso da linguagem Perl implementou-se uma série de algoritmos tendo como objectivo a conversão. Na segunda abordagem o processo de conversão passa pela utilização do documento XSL. Este documento é somado ao documento XHTML e o resultado final é um documento WML. Ambas as abordagens utilizam um conjunto de algoritmos idênticos que têm como objectivo apresentar um documento WML constituído por apenas um *deck*.

- **Acessibilidade**

A conversão aplicada adiciona acessibilidade ao resultado final na medida em que evita a introdução de URL desnecessárias. Por exemplo, a página “index.htm”, cuja a URL é <http://www.estgm.ipb.pt/index.htm>, possui um *hyperlink* para a página “cursos.htm”. Então não será necessário introduzir novamente a URL completa para converter a página, e o resultado será dado pela seguinte URL: <http://www.estgm.ipb.pt/cursos.htm>.

- **Informação irrelevante**

O método utilizado pode permitir a conversão da informação irrelevante, o que aumentaria o tamanho dos dados transmitidos. Se um determinado elemento do *nodo* não tem correspondência, deve ser eliminado assim como todos os *sub-nodos*. Mas por vezes a eliminação dos *sub-nodos* não é a melhor opção porque pode representar informação importante. Desta forma, o módulo ignora os elementos dos *sub-nodos* quando conhece *a priori*

o resultado final, com base numa tabela de configuração que especifica os elementos cujo o conteúdo associado deve ser eliminado.

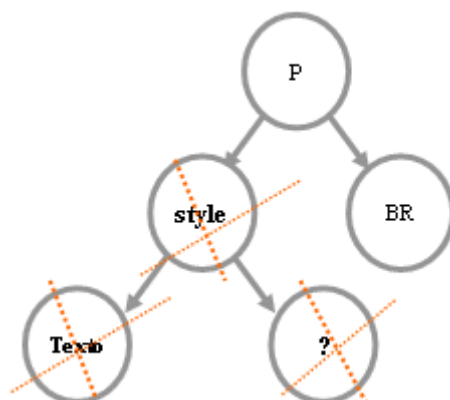


Figura 5 – Eliminação de conteúdo irrelevante

- **Erros de formatação**

Como o documento WML deve ser formatado de acordo com regras definidas na especificação desenvolvida pelo WAP Fórum, o módulo verifica os erros de formatação no documento final e corrige-os. O algoritmo implementado permite a configuração destas situações na medida em que limita a inclusão dos elementos que provocam um resultado incorrecto.

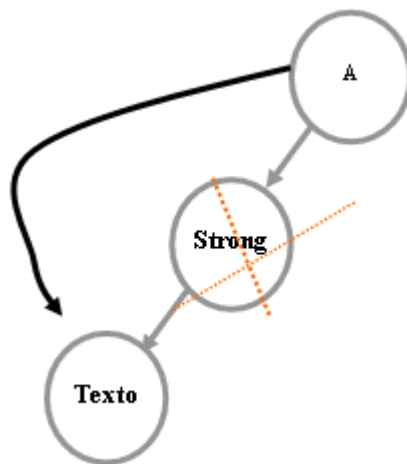


Figura 6 – Esquema de eliminação de parte da informação

- **Informação invisível**

O resultado final pode apresentar informação nula ou desnecessária. Por exemplo, se uma página apresenta 5 <br/> são visualizadas 4 linhas sem informação. Neste caso específico, apenas são necessárias 2 duplicações para separar a informação. Outra situação verifica-se quando não existe nos *sub-nodos* informação visível para o utilizador. Neste caso o nodo e todos os *sub-nodos* são eliminados. A aplicação permite configurar os elementos que representam informação visível com o objectivo de apresentar o mínimo de espaços nulos.

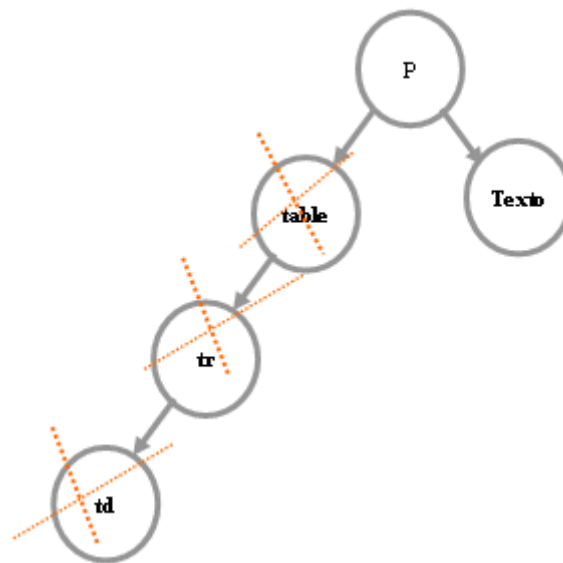


Figura 7 – Processo de eliminação de informação nula

- **Caracteres especiais**

Existe a necessidade de converter os caracteres especiais para a notação ASCII pelo facto de não poderem ser visualizados correctamente. Por exemplo o “á” deve ser substituído por “&#225;”.

- **Duplicação de elementos**

Segundo a especificação de um documento XML, não é possível ter nos *sub-nodos* elementos iguais aqueles que estão inseridos nos *nodos* superiores. No entanto o processo de conversão pode apresentar documentos WML incorrectos e por isso é necessário criar mecanismos para evitar o problema da duplicação.

- A solução mais fácil será ignorar, mas esta operação implica a perda de informação, seja ela relevante ou não.
- A segunda solução pressupõe que existe um erro de formatação do documento, e por conseguinte a tarefa a adoptar deve passar pela substituição de todos os elementos duplicados por outro elemento que não possua um par de *tags* tal como o elemento BR.
- A solução ideal passa por não perder informação e ao mesmo tempo manter as regras da conversão. Essa solução consiste em agrupar os *sub-nodos* num novo *card*. O sistema verifica se existe duplicação e, em caso afirmativo, o actual *nodo* e *sub-nodos* pertencem a um novo *card*. De referir que deve existir um *hyperlink* do actual *card* para o novo e vice-versa com o propósito de manter uma correcta navegação entre toda a informação.

**(5) Segmentação em vários decks e cards**

É necessário proceder à segmentação do conteúdo em vários *decks* e *cards*, com o propósito de contornar as limitações impostas pelos dispositivos WAP e apresentar o resultado final em condições de ser visualizado de forma fácil e clara.

Como a segmentação implica a divisão da informação, então torna-se imperativa a criação de mecanismos que permitam a navegação entre os vários elementos.

O módulo implementado aproveita o facto de a informação estar agrupada em *cards*, analisa o tamanho de cada *card* criado e este é adicionado ao *deck* actual se a operação não provocar um

excesso de tamanho do *deck*. No entanto a adopção deste algoritmo pode também não ser suficiente quando o tamanho do *card* é superior ao tamanho permitido, neste caso existe necessidade de o dividir.

Verifica-se outro problema quando a página HTML é convertida em  $n$  *decks* com tamanhos distintos. Por exemplo, o *deck 1* constituído pelo *card 1* e o *card 2* tem um tamanho de 400 bytes e o *deck 2* constituído pelo *card 3* com o tamanho de 1900 bytes. De forma a minimizar essa diferença, deve-se colocar parte do *card 3* no *deck 1* e outra parte no *deck 2*. Este processo é vantajoso pois permite harmonizar o tempo de *download* de todos os *decks* WML e minimiza o número de *decks* criados

## **(6) Visualização do resultado**

Finalmente, foi necessário criar um módulo responsável pelo envio do conteúdo WML para o utilizador. Pelo facto da conversão resultar em vários *decks* WML, o processo de envio do conteúdo passa pelo envio de vários documentos.

## **5 Conclusões**

### **5.1 Discussão dos resultados**

Foram implementadas duas fases para atingir o objectivo. Na primeira fase implementou-se a conversão, enquanto que na fase posterior foi necessário efectuar a segmentação do documento WML em vários *cards* e *decks* de forma a adaptá-lo às características dos dispositivos WAP. Cada fase foi projectada com o propósito de manter a sua independência. Ou seja, independentemente da abordagem utilizada na primeira fase, a segunda fase apresenta o resultado final devidamente segmentado. De referir que durante a fase de testes realizada, os resultados são positivos se a tabela de correspondência e outras tabelas de configurações criadas ou o documento XSL estão correctos.

Quanto a uma possível evolução da aplicação, os métodos desenvolvidos podem ser adaptados para implementar um componente ActiveX. Ou seja, dada a URL da página HTML, o componente é responsável pela correcta adaptação, utilizando um documento XSL externo. Este componente poderia ser então disponibilizado para o desenvolvimento de qualquer aplicação que tenha como objectivo fornecer a informação Web aos dispositivos WAP, online ou offline.

### **5.2 Considerações finais**

Actualmente o problema da conversão pode ser facilitado na medida em que a linguagem WML desapareceu. No WAP 2.0, o XHTML vem substituir o WML, e deste modo não será necessário conhecimentos extras acerca de outra linguagem de *markup*, cingindo-se o trabalho do programador apenas à elaboração de soluções para responder aos problemas associados aos diferentes dispositivo de visualização, desde os PCs aos dispositivos WAP.

Como o interesse é abranger o maior número de utilizadores, e visto que a tecnologia WAP está numa fase de afirmação, verifica-se que o processo de adaptação automática não corresponde eficientemente às pretensões estratégicas, embora, que em termos técnicos esta solução apresente resultados positivos.

Actualmente existem dois pontos importantes que devem ser tomados em consideração:

- Os utilizadores que procuram informação através de dispositivos WAP estão à espera de informação diferente daquela que lhes é oferecida a partir de um PC conectado à Internet;
- A ergonomia dos dispositivos móveis é completamente diferente dos PCs.

Com base nestas considerações, a estratégia a seguir será desenvolver conteúdos próprios para o novo ambiente. No entanto este processo acarreta um enorme esforço, mas o desenvolvimento de uma aplicação semi-automática permite contornar em parte os problemas. A aplicação é responsável pela conversão e pela segmentação e o programador tem a palavra final na medida em que tem liberdade para extrair/modificar/inserir para depois de testados os documentos WML podem ser disponibilizados.

## 6 Referências

- Arehart, C. et al, *Professional WAP*, 1ª ed. Wrox Press Ltd, 2000, p. 9-41, 2000.
- Argo Group. <http://www.argogroup.com>, 2001.
- Bickmore, T., Girgensohn, A. e Sullivan, J. W, *Web Page Filtering and Re-Authoring for Mobile Users*, The Computer Journal, Vol 42, Nº 6, 1999, p. 534-546.
- Blavier, A., *TidyCOM: A COM Wrapper for HTML Tidy's*, <http://perso.wanadoo.fr/ablavier/TidyCOM>, 2001.
- Hjelm, J., *Designing Wireless Information Services*, 1ª ed. John Wiley & Sons, Inc., ISBN 0-471-38015-6, 2000.
- Hsu, J., Johnston, W. and McCarthy, J., *Active outlining for HTML documents: An X-mosaic Implementation*, In 2ª Conferência Internacional de WWW, Chicago, 1994, <http://www.ncsa.uiuc.edu/SDG/IT94/Proceedings/HCI/hsu>, 2001.
- IBM, <http://www.ibm.com>, 2001.
- IDC, <http://www.idc.com>, 2001.
- IPanet, Sun Microsystems iPlanet Wireless ServerTM. <http://www.iplanet.com/>, 2001
- Móró, L., *XML-based content transformation in converged service development*, Tese de mestrado de 25 Abril de 2001 na Universidade de Lappeenranta, Finlândia, <http://ww.lut.fi/~leve/thesis>, 2001.
- Nielsen, J., *WAP Usability Report: Field Study Fall 2000*, <http://nngroup.com/reports/wap/>, 2001.
- Notess, G., *Reference Librarian*, Montana State University, Agosto/Setembro de 2000, Econtent p. 69-72.
- Online Computer Library Center, <http://www.oclc.org>, 2001.
- OpenTV, <http://www.opentv.com>, 2001.
- Openwave, *Top 10 Usability Guidelines for WAP Applications*, de 1 de Maio de 2001, <http://ww.openwave.com/resources/uiguide.html>, 2001.
- Oracle, <http://www.oracle.com>, 2001.
- Phone.com, <http://developer.openwave.com>, 2001.
- Raggett, D., *HTML Tidy*, <http://www.w3.org/People/Raggett/tidy>, 2001.
- Rysavy, P., *WAP: Untangling the Wireless Standard*, Network Computing, p. 109-113, 2001.

Silva M. J., “Mobilidade na Internet. In Alves et al.”, *O futuro da Internet*, 1ª ed. Edições Centro Atlântico, 1999. ISBN 972842608-9. p. 269.

W3C, *HTML 4.0 Guidelines for Mobile Access*, 15 de Março de 1999, <http://www.w3.org/TR/NOTE-html40-mobile>, 2001.

WAP Forum, <http://www.wapforum.org>, 2001.