

Faculdade Anchieta de Ensino Superior do Paraná – FAESP
Curso de Bacharelado em Sistemas de Informação

Desenvolvimento de um Sistema de Controle Acadêmico utilizando
tecnologias *Wireless*

Monografia submetida a Faculdade Anchieta de Ensino Superior do Paraná como requisito parcial para a obtenção do grau de Bacharel em Sistemas de Informação.

Professoras Responsáveis: Fernanda R. Ramos
Liliane

Alunos: Sandro Sovinski
Edgar Magno

Curitiba, 2005.

SUMÁRIO

SUMÁRIO.....	ii
LISTA DE FIGURAS	iii
LISTA DE TABELAS.....	iv
LISTA DE ABREVIATURAS.....	v
1. INTRODUÇÃO.....	07
1.1 Objetivo Geral	09
1.2 Objetivos Específicos	09
1.3 Justificativa.....	10
2. FUNDAMENTAÇÃO TEÓRICA.....	11
2.1 Tecnologia WAP	11
2.2 Protocolo WAP.....	12
2.3 Origem do Protocolo WAP.....	12
2.4 Benefício no uso do WAP	13
2.5 Modelo de Rede WAP.....	15
2.6 Segurança utilizada no WAP	19
2.6.1 Funcionamento do SSL – <i>Security Socktes Layer</i>	20
2.7 Arquitetura de Protocolos WAP	21
2.7.1 WAE – <i>Wireless Application Environment</i> (Camada de Aplicação)	22
2.7.2 WSP – <i>Wireless Session Protocol</i> (Camada de Sessão).....	23
2.7.3 WTP – <i>Wireless Transaction Protocol</i> (Camada de Transação).....	24
2.7.4 WTLS – <i>Wireless Transport Layer Security</i> (Camada de Segurança)	26
2.7.5 WDP – <i>Wireless Datagram Protocol</i> (Camada de Transporte)	27
2.8 Tecnologia Wap <i>Push</i>	28
2.9 A Estrutura do Wap <i>Push</i>	29
2.10 WML e WMLScript	30
2.10.1 <i>Cards e Decks</i>	32
2.10.2 WMLScript.....	34
2.10.3 Acesso a conteúdo WML e WMLScript	35

2.10.4 Segurança e Controle de Acesso.....	36
REFERÊNCIAS BIBLIOGRÁFICAS	38

LISTA DE FIGURAS

Figura 1 – Modelo de programação WAP	16
Figura 2 – Exemplo de modelo de rede WAP	18
Figura 3- Segurança WAP	20
Figura 4- Camadas de arquitetura WAP – Fonte: WAPmaster	22
Figura 5- Fonte Adaptado de (Cani, 2000).....	24
Figura 6- Fonte Adaptado de (Cani, 2000).....	25
Figura 7- Fonte Adaptado de (Cani, 2000).....	25
Figura 8 – Diferença de fluxo entre a tecnologia <i>pull</i> e <i>push</i>	28
Figura 9 – Arquitetura <i>push</i> básica.....	29
Figura 10 – Arquitetura <i>push</i> utilizando o <i>Gateway Proxy Push</i>	29
Figura 11 – Arquitetura <i>push</i> com os protocolos	30
Figura 12 – Arquitetura lógica do agente usuário WML.....	35
Figura 13- Arquitetura lógica do agente usuário WML (sem utilização do <i>gateway</i>)	36

LISTA DE TABELAS

Tabela 1 – Comparação entre <i>Web</i> e WAP	14
--	----

LISTA DE ABREVIATURAS

DTD – *Document Type Definition*

HDML – *Handheld Device Markup Language*

HTML – *Hipertext Markup Language*

HTTP – *Hipertext Transport Protocol*

IP – *Internet Protocol*

PDA – *Personal Digital Assitent*

PDA – *Hypertext Preprocessor*

PPG – *Gateway Proxy Push*

RAM – *Random Access Memory*

ROM – *Random Anly Assitent*

SSL – *Security Sockets Layer*

TCP – *Transmission Control Protocol*

UDP - *User Datagram Protocol*

UI – *User Interface*

XML – *Extensible Markup Language*

WAE – *Wireless Application Layer*

WAP – *Wireless Application Protocol*

WBMP – *Wireless Bitmapl*

WDP – *Wireless Transport Protocol*

WML – *Wireless Markup Language*

WMLS – *Wireless Markup Language Script*

WSP – *Wireless Session Protocol*

WTA – *Wireless Telephony Application*

WTLS – *Wireless Transport Layer Security*

WTP – *Wireless Trasaction Protocol*

WWW – *Word Wide Web*

1. INTRODUÇÃO

É fato que os usuários tornam-se cada vez mais dependentes dos serviços oferecidos via Internet, mas para acessá-los se faz necessário que eles estejam conectados à rede através de um fio. Entretanto, milhões de usuários passam muito tempo em trânsito e o fato de necessitarem de um cabo para a conexão torna-se um empecilho (Silva, 2000).

A internet vem crescendo cada vez mais, sendo uma área de atualização com constantes novidades, e de muito investimento por grandes empresas (Dias, 2000).

Num tempo não muito distante dos dias atuais, possuir um aparelho de telefone celular era um fato de *status* para as pessoas. Somente uma fatia muito pequena da população, a qual fosse caracterizada por seu poder aquisitivo elevado, teria condições de exibir esses aparelhos móveis. Aliado a essa situação, surgia o advento das redes de computadores livres dos fios que interconectavam. Era a chegada de uma inovação na maneira de se comunicar, deixando mais livre a localização desses terminais de acesso à rede (Henkel, 2001).

Conforme Silva (2000), nos últimos anos, as tentativas de acabar com estas barreiras e transformar a Internet em plataforma de serviços sem fio não foram bem sucedidas, pois o leque de padrões era muito extenso. O WAP (*Wireless Application Protocol*, ou Protocolo para Aplicações Sem Fio) surgiu como uma promessa de ser um protocolo largamente aceito, com capacidade de reconhecer os serviços WWW (*World Wide Web*), além de oferecer serviços diversos como telefônicos e mensagens eletrônicas.

Mas agora, além de se disseminar, a telefonia móvel quer comemorar o rito de passagem para a sua maioria por meio da oferta de serviços que até algum tempo poucos suspeitavam que um dia seria transportado para a tela de aparelhos compactos. De certa forma, um dos primeiros passos nesse sentido já vem sendo dado por meio de tecnologia como o WAP, que capacitam esses terminais a entrar no mundo da transmissão de dados (Henkel, 2001).

Em 1995 quando a Ericsson iniciou um esforço para desenvolver um protocolo geral que poderia oferecer uma variedade de serviços para redes sem fio.

Outras companhias estariam logo nos calcanhares da Ericsson, desenvolvendo outras tecnologias para competir com este mercado prestes a explodir. Entre elas dois grandes competidores incluindo Nokia e Phone.com, formalmente conhecido como Unwired Planet.

Como mais companhias entraram na dança, começando a desenvolver tecnologias similares, tornou-se óbvio que a cooperação entre as várias organizações era o único caminho para que qualquer companhia tivesse lucros deste prospecto. Após um tempo de intenso desenvolvimento e negociações, esta iniciativa culminou na formação do WAP Forum (www.wapforum.org). O WAP Forum juntou as melhores idéias, resultando na publicação do WAP 1.0 nos idos de 1997. O resultado foi um protocolo contendo as tecnologias HDML (*Handle Device Markup Language*) da Phone.com, o ITTP (*Intelligent Terminal Transfer Protocol*) da Ericsson e o TTML (*Tagged Text Markup Language*) da Nokia.

O WAP Push funciona como uma mensagem curta (SMS) que é enviada para o celular do usuário. A partir desta mensagem de texto, com uma informação objetiva e introdutória, o usuário pode utilizá-la como link para saber mais sobre o assunto, sem precisar de navegar em busca dos links desejados.

Um exemplo prático desta aplicação foi a parceria entre a Openwave, Telesp Celular e Folha Online durante a Copa do Mundo de 2002. Os clientes da Telesp Celular recebiam mensagens de texto sobre tabela, gols, resultados, com um link para mais informações sobre os jogos, que eram acessados de acordo com a necessidade e o interesse imediato do usuário. Em suma: o usuário pagava apenas pela informação que ele realmente desejava acessar.

Esta experiência foi bem sucedida e surpreendeu os envolvidos no negócio. Segundo Ana Busch, diretora executiva da Folha Online, em declaração ao site da Openwave, o sistema "Push superou nossas expectativas. Nosso tráfego aumento em 134%, uma vez que nosso conteúdo sobre a Copa estava disponível pela aplicação". A aplicação também aumentou em 40% o uso de dados *wireless* da Telesp Celular na Copa do mundo.

Segundo Fábio Póvoa, vice-presidente de negócios da Compera, "o conceito da informação necessária ao seu processo de negócio ir até o usuário traz um diferencial importante e pode tornar o WAP um protocolo muito utilizado, pela utilização de tecnologias já existentes nos padrões atuais de redes celulares (SMS e WAP) bem como com um início da cultura *wireless*, tanto para o setor corporativo quanto para os usuários finais". Segundo o executivo, o WAP Push tem particular potencial de uso em aplicações de gerenciamento de equipes em campo, com uma mensagem inicial solicitando determinada ordem de serviço e com link WAP para as demais informações necessárias para o retorno do status da operação.

Por esta e outras possibilidades é que a empresa sueca Ericsson prevê que em 2004 haverá 600 milhões de usuários conectados à internet via WAP. E não fica apenas nesta previsão: a empresa acha que já em 2003 os usuários que acessam a internet por celular superarão em número os usuários que se conectam por meio de computadores pessoais.

O presente trabalho visa o desenvolvimento de uma aplicação WEB utilizando da tecnologia WAP Push, que permite o envio de informações da instituição de ensino para seus alunos e professores de um servidor web para um celular com acesso WAP. Para tanto, será utilizada no desenvolvimento do trabalho a linguagem WML (*Wireless Markup Language*) para o desenvolvimento do WapSite, a linguagem PHP (*Hypertext Preprocessor*) para fazer o intercâmbio com a base de dados, o MySQL como ferramenta de banco de dados, a linguagem HTML (*Hipertext Markup Language*) para o desenvolvimento da *homepage*, um simulador de aparelho celular, para que a instituição consiga enviar estas informações para os alunos e professores.

1.1 Objetivo Geral

O trabalho tem por objetivo realizar um estudo sobre a tecnologia WAP e WAP Push, resultando na especificação e implementação de um sistema de apoio de envio de mensagens, que disponibilizará à instituição de ensino uma forma mais eficiente de enviar as notas e faltas dos alunos.

1.2 Objetivos específicos

O trabalho tem como objetivos específicos:

- Demonstrar o funcionamento da Tecnologia WAP e sua arquitetura de comunicação entre seus protocolos;
- Demonstrar o funcionamento da Tecnologia WAP *Push* e suas aplicações;
- Desenvolver uma aplicação através da utilização de um protótipo de telefonia móvel para enviar links informativos para os clientes.

1.3 Justificativas

Através desta tecnologia poderemos enviar link's para clientes, proporcionando a eles comodidade no recebimento de informações, neste caso informações com notas e faltas das disciplinas da instituição de ensino.

2. FUNDAMENTAÇÃO TEÓRICA

2.1 Tecnologia Wap

A possibilidade de se acessar a Internet através de um telefone celular ou de um PDA (*Personal Digital Assitent*) já está presente. Vários são os recursos disponíveis, ou seja, aparelhos celulares capazes de exibir páginas Web. Essa “Internet de bolso” vem revolucionando o mercado ocasionando a união das empresas da área comercial, tecnológica e de telecomunicações para criação de uma rede que alcance tanto usuários já acostumados com a Internet tradicional como aqueles que ainda nem possuem computador ou nunca acessaram a rede (Henkel, 2001).

Segundo Demétrio (2000), WAP é uma especificação para um conjunto de protocolos de comunicação com o intuito de normalizar a forma como os dispositivos sem fio (tais como: telefones celulares, palm tops, emissores/receptores de radio, etc.) acessam a Internet.

Sob o ambiente evolutivo que esta acontecendo, tem-se a tecnologia WAP que, segundo Henkel (2001) é definida como: “WAP provê um método para comunicação através de redes sem fio rapidamente, com segurança e eficiência. A comunicação pode tomar lugar utilizando os dispositivos de telefones celulares, *paggers*, rádios e PDA’s. Estas comunicações não estão limitadas a páginas estáticas como Internet fora antes, mas sim, oferece a oportunidade de integrar bancos de dados, contudo dinâmico e comercio eletrônico trafegando via dispositivo WAP”.

Atualmente a Internet é amplamente utilizada por usuários de computadores que possuem alto poder de processamento, grande quantidade de memória, alta largura de banda e transmissão de dados em redes geralmente confiáveis. Até então todas as tecnologias criadas para acesso à Internet eram destinadas a esse tipo de plataforma, mas no caso de aparelhos menos “robustos” como *handhelds*, telefones celulares e outros era necessário criar uma forma de acesso que considerasse algumas características particulares desses equipamentos, como limitado poder de processamento, pouca memória (RAM e ROM), consumo restrito de energia, pequenas telas, diferentes dispositivos de entrada de dados (como o teclado de um telefone celular ou de um *handheld*), estreita largura de banda, maior latência na transmissão de dados e pouca estabilidade durante a conexão, foi pensando em um ambiente desse tipo que surgiu o WAP (Cani, 2000).

2.2 Protocolo WAP

O *Wireless Application Protocol* (WAP) é um ambiente de aplicação e um grupo de protocolo de comunicação para dispositivos sem fio, criado para permitir – de uma forma independente de fabricante, vendedor e tecnologia – acesso à Internet e a um conjunto de serviços de telefonia avançados [IEC 00]. O protocolo WAP pode ser visto como sendo a ponte que interliga o mundo dos dispositivos móveis e a Internet – assim como as Intranets das corporações – e oferece a habilidade para integrar uma gama de serviços móveis de valor agregado para os usuários, independente da rede e do terminal móvel. Assim, o usuário pode acessar a mesma informação disponível em um microcomputador de mesa através de um dispositivo móvel, com a vantagem de que o conteúdo está disponível ao mesmo tempo em que o usuário se movimenta.

2.3 Origem do Protocolo WAP

A especificação WAP teve origem em 1997, quando um grupo de empresas formado pela Ericsson, Nokia, Motorola e Unwired Planet deram origem ao chado *WAP Forum*, que ainda hoje é o órgão responsável pela publicação das especificações assim como pela homologação de produtos WAP. A proposta inicial do *WAP Fórum* foi a de definir um ampla especificação pra a indústria a fim de permitir o desenvolvimento de aplicações sobre redes de comunicação sem fio. O *WAP Fórum* é um fórum aberto a qualquer participante da indústria relacionada à tecnologia WAO – como fabricantes de terminais móveis, infra-estrutura, provedores de serviços e de conteúdo, operadoras e desenvolvedores de *software* – e atualmente possui 550 membros, incluindo a Microsoft, América Online, Intel e IBM.

As diretrizes definidas pelo *WAP Fórum* têm a finalidade de alcançar os seguintes objetivos [RIB 00]:

- Tornar disponível conteúdo da Internet e serviços de dados avançados aos telefones celulares digitais ou outros terminais sem fio;
- Criar uma especificação global de protocolos que trabalhes através de diferentes tecnologias de redes sem fio;

- Incentivar a criação de conteúdo e aplicações que são providos através de uma variedade de redes sem fio e terminais móveis;
- Adotar, sempre que possível, e estender padrões e tecnologias existentes.

O WAP *Fórum* possui um programa de certificação de produtos onde fabricantes se credenciam com o objetivo de obter um certificado de conformação de seus produtos com especificação WAP corrente (atualmente WAP versão 1.2). Da mesma forma possui um programa de registro de desenvolvedores e também um programa de verificação de conteúdo [WAP 00a].

O telefone celular digital, o *Personal Digital Assistant* (PDA), o *pager*, além de outros dispositivos sem fio homologados pelo WAP *Fórum*, formam um grande conjunto de aparelhos que se beneficiam de serviços oferecidos pela especificação WAP, como *home banking*, comércio eletrônico, correio eletrônico, notícias e entretenimento.

2.4 Benefícios do Uso de WAP

Para as operadoras de rede sem fio, WAP promete reduzir custos e incrementar a base de usuários tanto pela melhoria de serviços existentes quanto pela criação de uma nova gama de serviços e aplicações agregadas. Novas aplicações podem ser introduzidas de forma rápida e fácil sem a necessidade adicional de infra-estrutura ou modificações no terminal móvel. Isto possibilita às operadoras diferenciarem-se pela forma como personalizam seus serviços e informações, fazendo disto uma interessante vantagem competitiva [IEC 00].

Os usuários de terminais móveis se beneficiam pela facilidade e segurança no acesso a informações relevantes disponíveis na Internet, sendo possível, por exemplo, o acesso a informações contidas em bancos de dados de corporações para uso em aplicações de *m-commerce*. Outros benefícios se refere à grande variedade de fabricantes que já suportam a iniciativa do WAP *Fórum*, permitindo que usuários tenham significativa liberdade na escolha de terminais móveis e aplicações.

Além disso, as inúmeras aplicações propiciadas pelo WAP-*Push*, como recebimento de informações meteorológicas e de condições de rodovias, trazem uma vantagem adicional às

tradicionais aplicações WWW, permitindo que o usuário receba informações relevantes em tempo real sem necessidade de requisita-las [WAP 99].

O WAP utiliza padrões de Internet como XML (*Extensible Markup Language*), UDP (*User Datagram Protocol*) e IP (*Internet Protocol*). Muitos destes protocolos são baseados em padrões da Internet como HTTP (*Hypertext Transfer Protocol*) e TLS (*Transport Layer Security*), mas têm sido otimizados para as restrições e baixa estabilidade de conexão [BRA 98a, BRA 98, POS 80, FIE 99].

Padrões utilizados na Internet, como HTML, HTTP, TLS e TCP (*Transmission Control Protocol*) são ineficientes sobre redes de terminais móveis, pois requerem um grande quantidade de dados em formato texto a serem enviados [IEC 00]. Dessa forma, o conteúdo de um arquivo HTML, padrão não pode ser mostrado de forma eficiente em telas de tamanho reduzido, como as de telefones móveis e *paggers*. Como decorrência, uma série de novos padrões e protocolos foram desenvolvidos, tendo por base os já consagrados na Internet e *Web*. A tabela abaixo relaciona os principais protocolos e padrões WAP, associando-os por funcionalidade, aos correspondentes na *Web*.

<i>Web</i>	<i>WAP</i>
HTML	WAE – WML
JavaScript	WMLScript
HTTP – Sessão Transação	WAP – Sessão WTP – Transação
TLS – Segurança	WTLS – Segurança
UDP – Datagrama	WSP Datagrama
IP – Endereço	<i>Bearer</i> - Endereço

Tabela 1 – Comparação entre *Web* e WAP

No padrão WAP, além da utilização de transmissão binária, permitindo que os dados sejam comprimidos, as sessões têm capacidade de adequação a problemas de cobertura de intermitente e podem operar sobre uma ampla variedade de formas de transporte.

Também, fazendo-se uma comparação com relação aos recursos disponíveis em um microcomputador pessoal, percebe-se que além das restrições citadas anteriormente, existem outras dificuldades a serem superadas, como a menor capacidade de processamento das CPUs, quantidade reduzida de memória RAM e/ou ROM, consumo de potencia restrito, visores de menor tamanho e dispositivos de entrada diferentes, o que justifica a necessidade de otimização e adaptação de padrões amplamente utilizado na Internet, como os citados acima [RIB 00].

As linguagens WML (*Wireless Markup Language*) e WMLScript (*Wireless Markup Language Script*) atualmente são utilizadas para a produção de conteúdo WAP, podendo este ser visualizado tanto em telas com apenas duas linhas, disponíveis em um dispositivo básico, como também em telas totalmente gráficas e maiores que acompanham dispositivos mais recentemente lançados.

O conjunto de protocolos WAP é desenvolvido para minimizar o tamanho de largura de banda requerido e maximizar o número de tipos de redes sem fio que podem distribuir conteúdo WAO [IEC 00]. Devido a ser projetado para otimizar o trabalho através de uma variedade de interfaces moveis, o WAP permite que um número maior de provedores de serviço e fabricantes de dispositivos móveis trabalhe em uma única especificação, e, além disso, tem-se a certeza de que cada aplicação, desenvolvida utilizando-se um padrão, trabalhará em uma diversidade de tipos de redes. Os fabricantes podem utilizar o mesmo *software* em todas as linhas de produto, o que reduz o tempo de desenvolvimento e simplifica o suporte. Também, como uma interface móvel independente, é possível, de forma rápida e fácil, estender a especificação.

Além dessas características, a especificação do protocolo WAP é projetada com o objetivo de encorajar a fácil e aberta interoperabilidade entre seus componentes fundamentais [RIB 00]. Assim, qualquer componente que possui compatibilidade com a especificação WAP pode interoperar com qualquer outro componente também compatível com o WAP. Provedores de conteúdo e serviços podem escolher equipamentos e *software* de múltiplos fabricantes, selecionando cada parte da solução de acordo com as suas necessidades.

2.5 Modelo de Rede WAP

O modelo de programação WAP é similar ao da *World-Wide Web*[WAP 98].

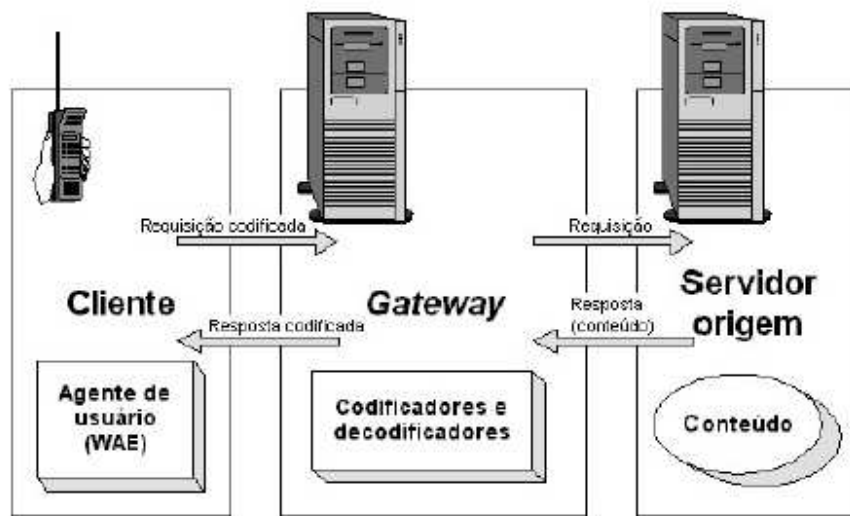


Figura 1 – Modelo de programação WAP

O conteúdo e as aplicações WAP são especificadas em um conjunto de formatos padronizados e baseados nos formatos já familiares da WWW, sendo transportados utilizando um conjunto de protocolos de comunicação também baseados nos utilizados na *Web*. Um *microbrowser* disponível no terminal sem fio, semelhante a um *Web browser* coordena a interface com o usuário.

O protocolo WAP define um conjunto de componentes que possibilitam a comunicação entre os terminais móveis e os servidores de rede, o que inclui:

- Um modelo de nomes padrão [WAP 98a] – URI (*Universal Resource Identifier*) e URL (*Uniform Resource Locator*), que consistem em padrões utilizados na WWW, são utilizados. O recurso URL é usado para identificar o conteúdo WAP nos servidores de origem e URI identifica recursos locais em um dispositivo, como por exemplo, as funções de controle são chamadas;
- Topologia de conteúdo – Todo conteúdo WAP é caracterizado por um tipo específico consistente com a topologia WWW. Isto permite que agentes de usuários WAP processem de forma correta o conteúdo, pois o mesmo é baseado em um tipo pré-definido;

- Padrão de formatos de conteúdos – Os formatos de conteúdo WAP são baseados em tecnologia WWW, incluindo informações de calendário, cartão de negócios eletrônicos, imagens e linguagem de *script*;
- Padrão de protocolo de comunicação – Os protocolos de comunicação WAP permitem a comunicação entre as requisições feitas pelo *browser* do terminal móvel e o servidor de rede *Web*.

O protocolo WAP utiliza a tecnologia *proxy* para realizar a conexão entre o domínio sem fio e a WWW.

Na especificação do WAP, o acesso a uma determinada aplicação é conseguido através do envio de uma requisição do dispositivo móvel (na forma de URL) ao *proxy* WAP. Isto permite que o conteúdo e aplicação sejam armazenados em servidores WWW comuns e que este conteúdo e aplicações sejam desenvolvidos utilizando tecnologias também usadas na WWW, como *CGI Scripting*.

O *proxy* WAP, que é a interface entre a rede da operadora e a Internet, deve incluir as seguintes funcionalidades [RIB 00]:

- Um mecanismo, normalmente denominado *gateway* de protocolos, cuja função é traduzir as requisições provenientes da pilha de protocolos WAP para o conjunto de protocolos WWW (HTTP e TCP/IP);
- Codificadores e decodificadores de conteúdo, que têm o objetivo, respectivamente, de codificar e decodificar a informação desenvolvida em WML, para que esta seja transportada através da rede sem fio de forma compacta, o que diminui o tráfego de dados necessários.

Um exemplo de modelo de rede WAP [WAP 98] poder ser visualizado através da figura abaixo.

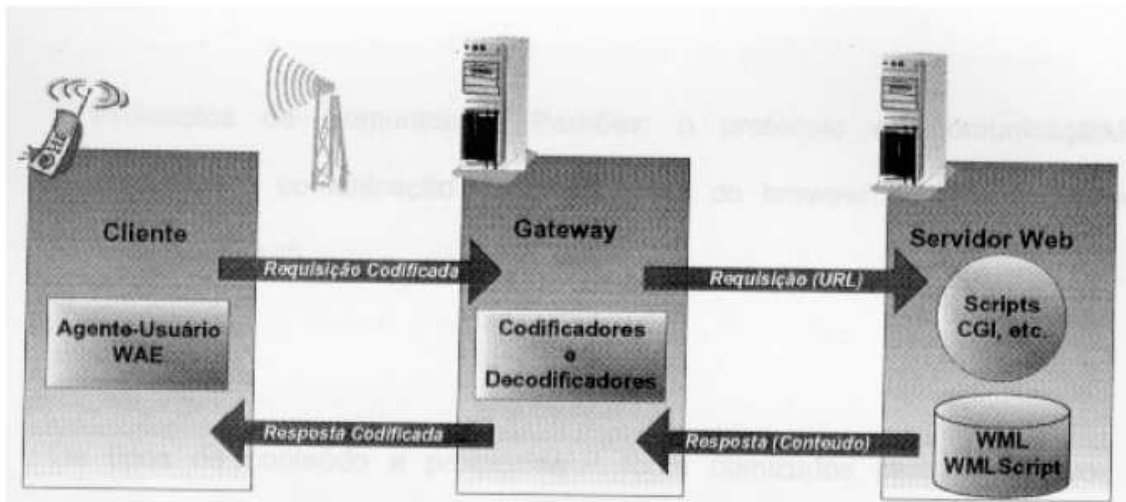


Figura 2 – Exemplo de modelo de rede WAP

Neste exemplo, o cliente WAP se comunica com dois servidores na rede sem fio. O *proxy* WAP produz as requisições WAP para requisições WWW e, desse modo, permite que o cliente WAP submeta requisições ao servidor *Web*. O *proxy* também codifica as respostas do servidor *Web* em um formato binário compactado entendido pelo cliente. Podemos também observar que há necessidade de comunicação entre o cliente e o *proxy* WAP sempre que o usuário deseja acessar serviços disponibilizados na Internet. Já o acesso ao servidor WTA (*Wireless Telephony Application*), que provê serviços de telefonia, é feito diretamente. Além disso, percebe-se que existem duas formas de acesso ao servidor WWW. Se este servidor apresentar o conteúdo no formato WML, as informações são enviadas diretamente ao *proxy* WAP, caso contrário, quando as informações estão disponíveis em HTML, é necessário que este conteúdo seja filtrado antes. A função deste filtro conforme pode ser observado na figura, é a de converter conteúdo no formato HTML em WML e posteriormente envia-lo ao *proxy* WAP [RIB 00].

O uso de *gateway* de protocolos (ou *gateway* WAP) permite que o conteúdo e as aplicações, criadas para serem acessadas através de um dispositivo móvel, possam ser hospedadas em um servidor WWW comum, beneficiando-se de toda a estrutura e tecnologia da Internet existente.

A comunicação entre o servidor WWW e o *gateway* WAP é realizada através do protocolo HTTP v1.1. Dessa forma, os usuários podem utilizar URLs diferentes para acessar

conteúdo HTML ou WML; ou até mesmo usar um único URL para disponibilizar ambos formatos, deixando a cargo do servidor WWW a escolha de qual conteúdo enviar (decisão tomada de acordo com o navegador utilizado pelo cliente) [SEC 00].

2.6 Segurança utilizada no WAP

A segurança tem sido, e permanece sendo, uma das maiores preocupações dos usuários da Internet e dos vendedores e provedores de serviços que desejam usa-la como canla para o mercado.

Com o aumento significativo de dispositivos, aplicações e serviços *wireless* surge há preocupação com a questão da segurança, ou seja, até que ponto o acesso é seguro e livre de intrusos. Nas aplicações de comercio eletrônico via telefones celulares, PDA á ou *Pagers* com acesso à Internet, essa questão é muito mias importante do que no modelo convencional via computador *desktop*. Isso acontece porque se deve garantir a privacidade, a confiabilidade e a qualidade nas operações realizadas (Henkel, 2001).

Segurança é permitir que pessoas estranhas não tenham acesso aos dados pessoais e senhas. Segurança é um conceito utilizado freqüentemente com pouco rigor e conceitos equivocados. Como ponto de partida, pode-se tomar como parâmetro a arquitetura de segurança do modelo de referencia OSI (TANEMBAUM, 1997), a qual adota serviços como os listados a seguir:

- Autenticação: é a certificação de que as partes envolvidas na comunicação são quem dizem ser. Este processo tem a finalidade de detectar o ataque *spoofing*, onde um indivíduo tenta ocultar sua identidade e assumir a de outro para obter vantagens ilícitas;
- Integridade: objetiva garantir que mensagens não foram adulteradas durante a transmissão. Se houver alteração, o sistema de segurança deve ter um método de descobrir e informar o ocorrido;
- Não repúdio: implementa mecanismo para que impossibilite a negação da participação ou da não ocorrência de uma transação, por parte dos participantes. Ou seja, houve a transação, mas o participante nega sua existência ou participação na mesma. Isto requer mecanismos de autenticação;

- Privacidade: pressupondo-se que a comunicação seja interceptada, deve-se garantir que a informação não seja entendida ou usada por terceiros. Normalmente, utiliza-se criptografia;
- Autorização: determina se um indivíduo tem o direito de executar determinada ação, em determinado instante.

Como se pode observar na figura abaixo, existem duas partes diferenciadas no modelo de segurança WAP. Na parte direita da figura, o *Gateway* WAP simplesmente usa SSL (*Secure Socket Layer*) para estabelecer uma comunicação segura com o servidor *Web*, assegurando a privacidade, integridade e autenticação do servidor.

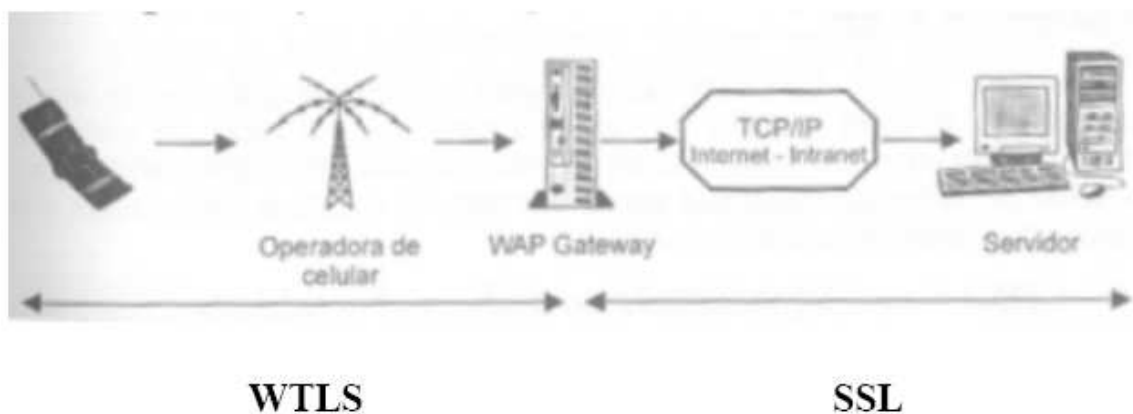


Figura 3 – Segurança WAP

Na parte esquerda, o *Gateway* apanha os dados codificados com SSL do servidor *Web* e os transforma para poder transmitir, usando WAP e o protocolo de segurança WTLS. As solicitações desde o telefone até o servidor *Web* refazem o caminho inverso. Em resumo, o *Gateway* faz o papel de ponte entre os protocolos WTLS e SSL (Demétrio, 2000).

2.6.1 Funcionamento do SSL – *Secure Socket Layer*

O elemento-chave do protocolo SSL é um código de encriptação público. Para isto usa-se um par de códigos e algoritmos matemáticos que convertem o texto em texto codificado e vice-

versa. Esse par de códigos consta de um código registrado público e outro privado, que é mantido em segredo pelo proprietário. Uma mensagem codificada com o código público só pode ser decodificada com o código privado. O inverso é verdadeiro. Assim como uma porta que só pode ser aberta com duas chaves (Demetrio, 2000).

A encriptação de código público é muito útil para pequenas quantidades de dados, mas torna-se lenta quando se utilizam grandes quantidades. Para o último caso usa-se a encriptação privada ou simétrica.

O protocolo SSL combina ambas as técnicas para realizar as transações. Em um primeiro contato, conhecido como negociação, o remetente e o destinatário vão se comunicar utilizando o código público, o que assegurará a privacidade durante o resto da transação.

Para proporcionar a integridade, o protocolo SSL usa algoritmos que criam uma assinatura digital matemática de cada mensagem. Se a mensagem modifica-se pelo caminho, o destinatário comprovará que a assinatura digital não coincide com a mensagem e a rejeitará.

A autenticidade de ambas as partes é confirmada por intermédio de certificados digitais, que evitam que terceiro interceptem as transmissões. Os certificados asseguram que o remetente e o destinatário são quem dizem ser.

Quando o navegador solicita uma conexão com o servidor, ele mostra o seu certificado. O navegador comprova se o certificado é válido, e caso positivo, prossegue com a transação, seguindo então para codificação/decodificação dos dados (Demetrio, 2000).

2.7 Arquitetura de Protocolos WAP

A arquitetura é formada por um ambiente escalável e extensivo para o desenvolvimento de aplicações para dispositivos de comunicação móveis. Isto se realiza por meio do protocolo WAP, que é composto por várias camadas de subprotocolos (figura 4), em que cada camada pode ser acessada pela camada superior e por outros serviços e aplicações (Demétrio 2000).

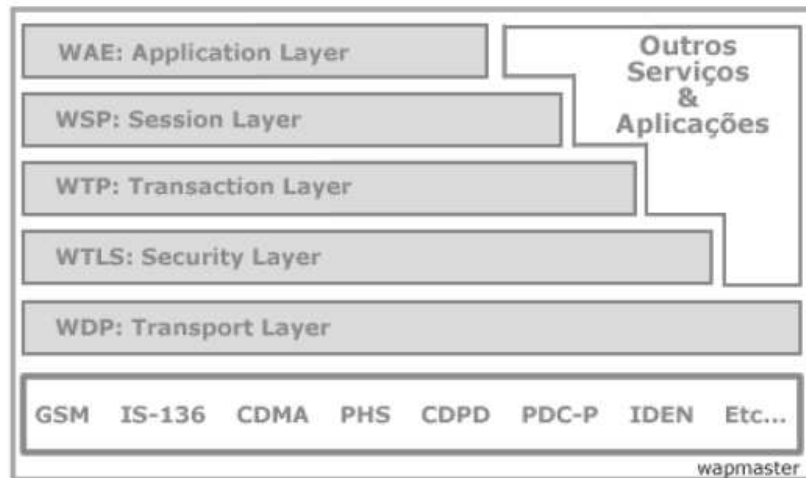


Figura 4 – Camadas de arquitetura WAP – Fonte: (WAPmaster 2001)

A arquitetura da camada WAP permite a outros serviços e aplicações utilizar as funções da pilha de camadas WAP por meio de interfaces bem definidas. Ainda, as aplicações externas podem acessar diretamente as camadas de sessão, transação, segurança e transporte (Demétrio, 2000).

2.7.1 WAE – *Wireless Application Environment* (Camada de Aplicação)

O WAE é a camada de aplicação de propósito geral baseada na combinação das tecnologias WWW e de telefonia móvel. Seu principal objetivo é estabelecer uma interação entre operadores e os provedores de serviços, para permitir a construção de aplicações de uma forma eficiente, independente do dispositivo em que vão ser executadas (celulares, palms, etc.). O WAE inclui um micro-navegador com as seguintes funções (Demétrio, 2000):

- a) WML (*Wireless Markup Language*), uma linguagem leve, otimizada, semelhante a HTML, porém específica para uso em dispositivos móveis;
- b) WMLScript, uma linguagem similar ao JavaScript que otimiza a WML;
- c) Serviços de telefonia e interfaces de programação WTA (*Wireless Telephony Application*);
- d) Suporte a imagens, calendários, agenda, etc.

O WAE não especifica formalmente nenhum agente-usuário, ou seja, suas características e capacidades são deixadas para o implementador. O WAE somente define serviços e formatos fundamentais que são necessários para assegurar a interoperabilidade entre diferentes implementações.

2.7.2 WSP – *Wireless Session Protocol* (Camada de Sessão)

O WSP permite a aplicação de uma interface para o estabelecimento de sessões. Para isso foram criados dois tipos de protocolos diferentes (Demétrio, 2000):

- a) o primeiro para conexão mediante o serviço de transações WTP (*Wireless Transaction Protocol*);
- b) e o outro permite acessar diretamente a camada WDP (*Wireless Datagram Protocol*), sem a necessidade de estabelecer uma conexão direta com as outras camadas, o que melhora o rendimento de aplicações que não necessitam de confirmação de envio de dados. Permite a troca eficiente de dados entre as aplicações.

A diferença básica é que nos serviços orientados à conexão são disponibilizados facilidades de gerenciamento e segurança na transmissão entre o cliente (*browser WAP*) e o servidor (*Gateway WAP*). Caso por algum motivo tome impossível a transmissão, a mesma pode ser interrompida e retomada mais tarde partindo do ponto de parada anterior. Existe um rígido controle entre pacotes enviados e pacotes recebidos, esse controle obviamente gera um tráfego um pouco maior do que nos serviços sem conexão. Já no caso dos serviços sem conexão não existe nenhum controle de recebimento dos pacotes enviados pelo dispositivo, ou seja, os serviços não são confirmados. Nesse caso a sessão de transmissão não é considerada segura, sua principal vantagem é que gera menos tráfego na rede (Cani, 2000).

O Núcleo do WSP é um formato binário utilizado pelo HTTP, o que permite a inclusão de cabeçalho e informação. Dessa forma, os métodos utilizados pelo http/1.1 são suportados garantindo compatibilidade com o mesmo. Convém salientar que o WSP, por si só, não interpreta a informação do cabeçalho nas requisições/respostas e, também, o seu ciclo de vida não esta

ligado ao transporte, uma vez que a sessão pode ser suspensa quando estiver ociosa, para fins de liberação de recursos na rede ou economia de bateria do dispositivo (Henkel, 2001).

2.7.3 WTP – *Wireless Transaction Protocol* (Camada de Transação)

É um protocolo simplificado, idealizado para situações de baixa largura de banda, caso específico das comunicações sem fio (Demétrio, 2000).

A camada WTP da pilha de protocolos WAP é a responsável por controlar o envio e a recepção das mensagens, oferecendo suporte de transações, o que acrescenta confiança ao serviço de datagrama provido pela camada WDP, uma vez que libera a camada superior a retransmissões reconhecimentos necessários quando esse serviço é utilizado (Henkel, 2001)

Quando uma transação WTP se inicia, ela é definida com sendo o Inicializador, enquanto que a resposta à mesma é indicada como sendo Responder. Uma classe de transação é então criada pelo Inicializador e indicada na mensagem de execução a ser enviada ao Responder. Existem três classes de transações que são (Henkel, 2001):

- a) classe 0: o Inicializador envia uma mensagem para o Responder. A mensagem de execução não é confiável, portanto não possui mensagem de resultado (figura 5);

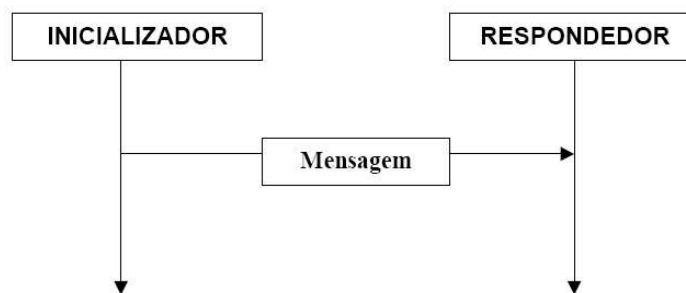


Figura 5 - Fonte: Adaptado de (Cani, 2000).

- b) classe 1: a mensagem de execução é confiável, porém, não possui mensagem de resultado. O Inicializador envia uma mensagem para o Responder, após o

recebimento da mensagem pelo Respondedor o mesmo retorna para o Inicializador um sinal de reconhecimento da mensagem conforme segue abaixo (figura 6);

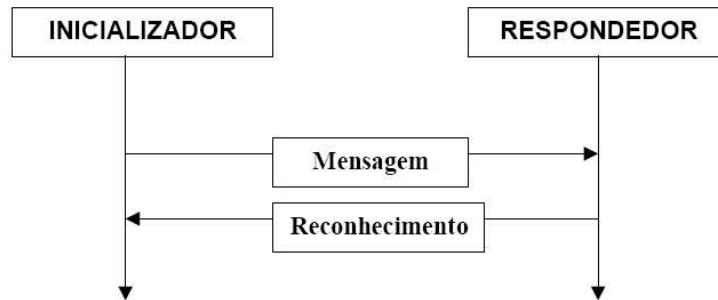


Figura 6 - Fonte: Adaptado de (Cani, 2000).

- c) classe 2: a mensagem de execução é confiável, mas com somente uma mensagem de resultado confiável. O Inicializador manda uma mensagem para o Respondedor e após o recebimento da mensagem pelo respondedor, o mesmo faz o processamento da mensagem e envia o resultado para o Inicializador. Ao receber o resultado o Inicializador envia ao Respondedor um sinal de reconhecimento (Figura 7).

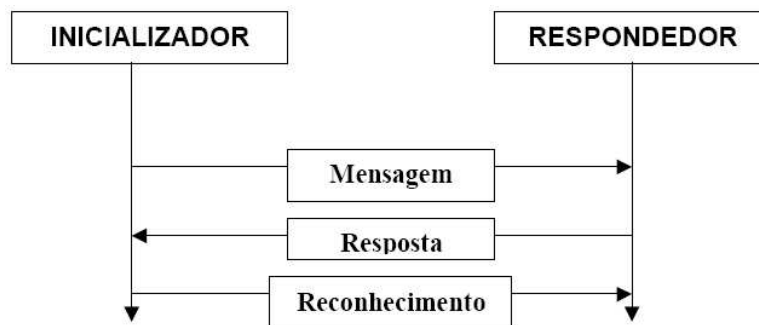


Figura 7 - Fonte: Adaptado de (Cani, 2000).

O protocolo possui varias características, tais como (Silva, 2000):

- a) o alcance da confiabilidade é feito através do uso de identificadores únicos de transação, reconhecimento, remoção de duplicação e retransmissões.

- b) não possui fase de criação e destruição de conexão explícita, uma vez que este procedimento de abertura e/ou fechamento de conexão impõe uma excessiva sobrecarga na rede de comunicação.
- c) possui segurança de usuário-a-usuário opcional, isto é, o usuário do WTP pode confirmar cada mensagem recebida.
- d) faculta ao último reconhecimento da transação a contenção de informações fora da banda relacionada à transação, como medidas de desempenho.
- e) usa a concatenação para transportar múltiplas unidades de dados do protocolo em uma única unidade de dados de serviços, que será transportada pelo WDP.
- f) baseia-se na mensagem, isto é, a unidade básica de intercâmbio é uma mensagem inteira e não uma cadeia de bytes.
- g) usa mecanismos para minimizar o número de transações sendo retornadas como resultado de pacotes duplicados.
- h) possibilita o cancelamento de transações atual, incluindo o descarte dos dados não enviados tanto do cliente quanto no servidor. O cancelamento pode ser disparado pelo usuário para cancelar um serviço requisitado.
- i) dá segurança através de mensagem de execução confiável. Tanto o sucesso quanto a falha são reportados, isto é, se uma execução não pode ser manipulada pelo Responder, então uma mensagem de cancelamento será retornada para o Inicializador ao invés do resultado.
- j) permite transações assíncronas, isto é, o responder envia o resultado assim que os dados se tornarem disponíveis.

2.7.4 WTLS – *Wireless Transport Layer Security* (Camada de Segurança)

O protocolo da camada de segurança na arquitetura WAP provê para as camadas superiores de interface de transporte seguro, preservando o serviço de interface de transporte abaixo dele. A camada WTLS é modular, isto é, o seu uso não depende do nível de segurança requerido em uma dada aplicação, ou seja, é uma camada opcional da pilha WAP. Além da segurança, o WTLS fornece uma interface para gerenciamento de conexões seguras (Silva, 2000).

O WTLS disponibiliza os seguintes serviços:

- a) integridade de dados: garante que não haverá alteração do conteúdo nos dados enviados entre o cliente e o servidor, e vice-versa;
- b) privacidade: assegura que não haverá como um interceptador descriptar a mensagem, a não ser o transmissor ou receptor;
- c) autenticação do cliente: limita o acesso através do servidor de conteúdo, ou seja, apenas quem tem permissão pode acessar determinado conteúdo;
- d) autenticação do servidor: certifica que o servidor acessado, é realmente o desejado.

Aplicações são capazes de habilitar ou desabilitar características do WTLS dependendo das suas necessidades de segurança ou das características da rede. Por exemplo, privacidade pode ser desabilitada em redes que já fornecem esse serviço em uma camada de mais baixo nível (Jesus, 2000).

2.7.5 WDP – *Wireless Datagram Protocol* (Camada de Transporte)

É a camada que transporta os dados, responsável pelo envio e recebimento de mensagens pelos vários tipos de redes. Como o protocolo WDP proporciona uma interface entre protocolos superiores WAE, WSP e WTLS, é capaz de funcionar de forma independente da rede telefônica em que esteja trabalhando, ou de se adaptar às especificações desta rede (Demetrio, 2000 e Dias, 2000).

Os serviços oferecidos pelo WDP incluem endereçamentos de aplicações por número de porta e segmentação opcional, além de reunião e detecção de erros opcionais. Estes serviços permitem às aplicações operarem transparentemente sobre diferentes transportadores disponíveis. O número da porta identifica a entidade mais acima do WDP que pode ser WTP ou WSP ou mesmo uma aplicação como o correio eletrônico (Silva, 2000).

2.8 Tecnologia WAP *Push*

A estrutura do WAP *Push* introduz uma maneira, dentro do esforço WAP, para transmitir informações ao dispositivo sem uma anterior ação do usuário. No conhecido modelo cliente/servidor, um cliente requisita um serviço ou informação a um servidor, que então responde transmitindo a informação ao cliente. Isto é conhecido como tecnologia “*pull*”: o cliente puxa (*pull*) a informação do servidor (figura 8). A WWW é um exemplo típico de tecnologia *pull*, onde um usuário entra com uma URL (a requisição) que é enviada ao servidor, e o servidor responde pelo envio de uma página *Web* (a resposta) ao usuário.

Em constraste disto, há a tecnologia *push*, que também é baseada no modelo cliente/servidor, mas onde não há uma requisição explícita do cliente antes do servidor transmitir o conteúdo.

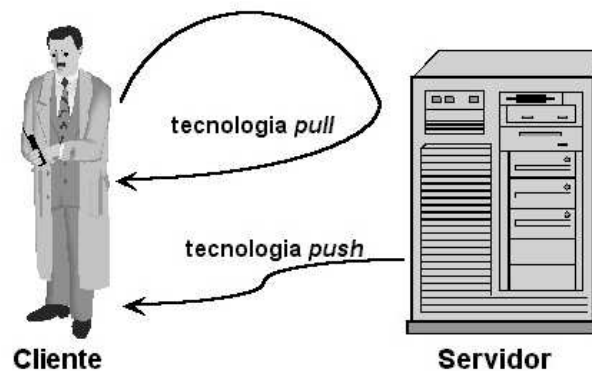


Figura 8 – Diferença de fluxo entre as tecnologias *pull* e *push*.

Outra forma de explicar isso é que enquanto as transações “*pull*” da informação são sempre originadas pelo cliente, as transações “*push*” são originadas pelo servidor.

2.9 A Estrutura do WAP Push

Uma operação *push* em WAP ocorre quando um originador *push* envia o conteúdo ao cliente utilizando um dos seus protocolos. Nesta forma simples a arquitetura pareceria como na Figura 9.

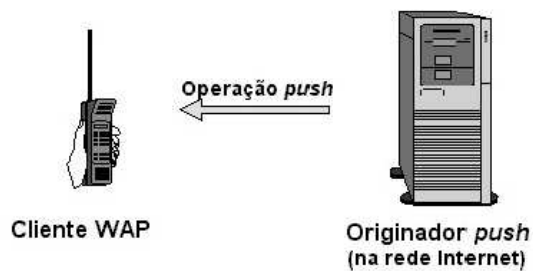


Figura 9 – Arquitetura *push* básica.

Contudo, o originador *push* não compartilha nenhum protocolo com o cliente WAP: o originador *push* está na Internet e o cliente WAP no domínio WAP (figura 10). Por isso, o originador *push* não pode se comunicar com o cliente WAP sem um mecanismo intermediário, então nós vamos processar inserir um *gateway* para fazer essa tradução.

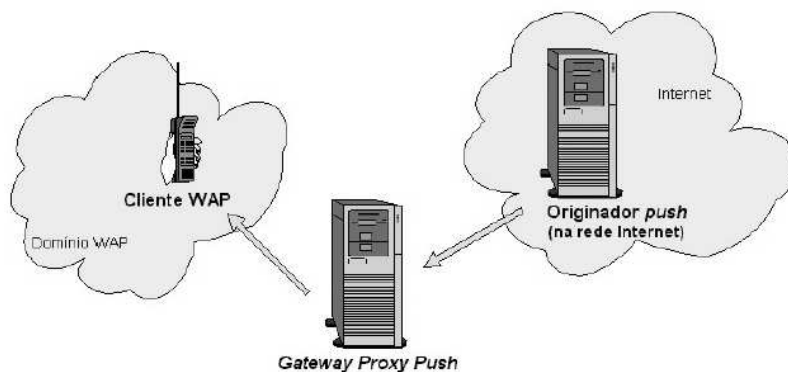


Figura 10 – Arquitetura *push* utilizando o *Gateway Proxy Push*

Assim, o originador *push* contata o *Gateway Proxy Push* (PPG) do lado da Internet, despachando conteúdo para o cliente destino utilizando protocolos da Internet. O PPG faz o que é

necessário para repassar o conteúdo “empurrando” para o domínio WAP, e o conteúdo é transmitido pela rede de terminais móveis até o cliente destino.

Adicionalmente, para prover simples serviços de *proxy gateway*, o PPG pode estar encarregado de notificar o originador *push* sobre o final do resultado da operação de *push*, e ele pode esperar pelo cliente para aceitar ou rejeitar o conteúdo em redes móveis de dois sentidos. Isto também pode proporcionar ao originador *push*, juntamente com o cliente, serviços de procura, deixando o originador *push* selecionar as preferências de conteúdo especial para este cliente particular.

No lado da Internet, o protocolo de acesso ao PPG é chamado de protocolo de acesso *push*. No lado WAP, o protocolo é chamado de “*Push Over-the-Air*”. Dessa maneira, um esquema revisado parece alguma coisa como mostrado na figura 11.

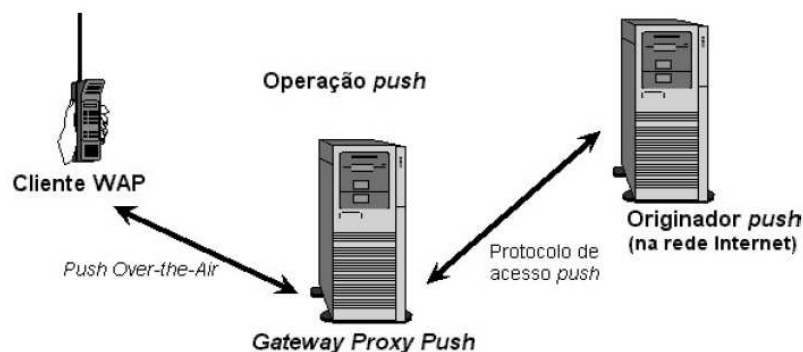


Figura 11 – Arquitetura *push* com os protocolos

O protocolo de acesso *push*, ou PAP utilizam mensagem XML que podem estar embutidos em vários protocolos utilizados na Internet como o HTTP.

2.10 WML e WMLScript

A WML [WAP 00] é uma linguagem descritora de documentos baseada em *tags*, assim como HTML. Embora tenha herdado aspectos de HTML [RAG 97] e HDML[KIN 97], é na realidade bem mais flexível, por tratar-se de um tipo de XML [BRA 98a], que é um conjunto simplificado da metalinguagem SGML. Ela é utilizada para especificar conteúdo e uma interface

de usuário para dispositivos WAP considerando a limitada capacidade dos dispositivos como telefones celulares e outros terminais móveis sem fio. A linguagem WML diferencia maiúsculas de minúsculas, assim como a linguagem C. Conforme já mencionado, WML também foi derivada de HDML (*Handheld Device Markup Language*), uma linguagem proprietária da empresa Phone.com e comercializada pela mesma antes da criação do padrão WAP.

A XML consiste em uma metalinguagem, a qual é utilizada para definir outras linguagens, assim como o que foi feito para criar a linguagem WML. Ela permite ao desenvolvedor definir qualquer conjunto de *tags*, que são comandos inseridos em um documento, que servem para especificar como este documento (ou parte dele) deve ser apresentado. Um conjunto de *tags* é agrupado em um conjunto de regras de gramática e descreve uma definição de tipo de documento (DTD – *Document Type Definition*). O WAP Fórum estabelece a DTD WML v1.1 que define sintaxe e elementos usados em um documento WML válido. Esta DTD encontra-se disponível em http://www.wapforum.org/DTD/wml_1.1.xml.

Se um telefone móvel é dito como “WAP *Compatible*” significa que ele possui uma porção de *software* embutido, conhecido como *microbrowser*, que é capaz de interpretar todas as entidades dispostas em um documento WML.

A linguagem WML apresenta uma variedade de características, incluído:

- Suporte para textos e imagens – WML provê aos autores (desenvolvedores) meios de especificar texto e imagens para serem apresentados ao usuário. Isto pode incluir *layout* e dicas de apresentação. Como outras linguagens de marcação, WML requer do autor especificar a apresentação em termos muito gerais e dá ao agente de usuário uma grande liberdade para determinar exatamente como a informação é apresentada ao usuário final;
- Suporte para dados de entrada do usuário – WML suporta vários elementos para solicitar entradas de dados do usuário. Todas as requisições para entrada de dados são feitas em termos abstratos, permitindo que o agente de usuário tenha liberdade para otimizar características para cada dispositivo particular. Os campos de entrada de textos podem ter mascaras para impedir que o usuário insira tipos de caracteres inválidos;

- Navegação e pilha de histórico – WML permite vários mecanismos de navegação utilizando URLs. Ele também expõe um mecanismo de histórico. Navegação inclui *hyperlinks* no estilo HTML, elementos de navegação entre *cards*, bem como histórico de elementos de navegação;
- Suporte de internacionalização – O conjunto de caracteres de documentos para WML é o *Universal Character Set* da ISSO/IEC-10646. Atualmente, este conjunto de caracteres é igual ao Unicode 2.0. Não há necessidade de que os *decks* WML sejam codificados usando a codificação completa do Unicode (exemplo, UCS-4). Qualquer conjunto de caracteres que sejam um subconjunto correto de caracteres do Unicode pode ser usado (exemplo, US-ASCII, ISSO-8859-1, UTF-8, Shift_JIS, etc.);
- Independência de interface homem-máquina – A especificação abstrata do *layout* e da apresentação do WML permite que fabricantes de dispositivos controlem a forma da interface homem-máquina para seus produtos, com cada um apresentando determinadas particularidades.
- Otimização para banda estreita – WML inclui uma variedade de tecnologias para otimizar a comunicação em um dispositivo de banda estreita. Isto inclui a possibilidade de especificar várias interações de usuários (*cards*) em uma transferência na rede (um *deck*);
- Gerenciamento de estado e contexto – Cada controle de entrada em WML pode introduzir variáveis. O estado das variáveis pode ser usado para modificar o conteúdo de um *card* parametrizado sem ter que haver comunicação com o servidor. Além disso, o tempo de vida do estado de uma variável pode ser maior que um único *deck* e pode ser compartilhado por vários sem ter que utilizar um servidor para salvar um estado intermediário entre as invocações dos *decks*.

2.10.1 *Cards e Decks*

A linguagem WML foi estruturada com o objetivo de utilizar a baixa taxa de transferência das redes sem fio e as pequenas telas dos dispositivos móveis de forma otimizada. Com esse objetivo, conceitos como *deck* e *card* foram utilizados. Um documento WML simples, que

apresenta elementos contidos dentro de um par da *tag* “<wml>”, é chamado de *deck*, e representa uma aplicação ou serviço. O *deck* é o primeiro elemento a ser lido pelo *microbrowser* e é nele que estão contidas as informações sobre a página, que são utilizadas para, por exemplo, identificar se a mesma pode ser visualizada pela versão do *microbrowser* disponível no dispositivo móvel.

Cada *deck* contém um ou vários *cards* (ou para melhor entendimento pode-se chamar de ‘cartões’), os quais podem ser definidos como sendo as telas de informação a serem visualizadas pelo usuário. O *card* pode conter texto, referências para outros sítios ou campos de entrada.

O *deck* é similar a um documento HTML, uma vez que é identificada por um URL [BER 98]. Dessa forma, um *deck* é uma unidade de conteúdo de transmissão que é enviada integralmente do servidor WAP ao dispositivo móvel do usuário. Isto significa que várias telas (ou seja, *cards*) podem ser enviadas a um cliente de uma única vez. Um *deck* não deve exceder 1400 *bytes*, o que não corresponde ao tamanho do arquivo WML e sim ao tamanho do arquivo binário codificado para ser transmitido, resultante do arquivo fonte WML.

Quando se desenvolve uma página para estar disponível na Internet, normalmente o arquivo que contém as instruções é salvo com a extensão HTML. Já quando se deseja que um determinado conjunto de informações seja acessado via WAP, a extensão do arquivo geralmente usada é WML.

As imagens possuem um formato especial chamado WBMP (*Wireless Bitmap*), que corresponde a imagens em preto e branco com tamanho de até 150 x 150 *pixels* – apesar de imagens coloridas também serem aceitas. Arquivos de imagens nos formatos JPEG, GIF ou outros utilizados na *Web* não podem ser diretamente referenciados em arquivos WML, necessitando serem convertidos para o formato WBMP, para que possam ser visualizados pelo *microbrowser* disponível no dispositivo móvel. Essa tarefa pode ser realizada de forma simples e rápida através da utilização de diversos programas gratuitos disponíveis, na Internet além de serviços de conversão *on-line* como o oferecido pela empresa Teraflops LTD. (<http://www.teraflops.com/wbmp/>). Além disso, como nem todos os terminais WAP têm capacidade de exibir imagens, pode-se utilizar a opção ‘alt’ (da mesma forma como é utilizado em um código HTML) para que esses terminais mostrem uma legenda no lugar da imagem.

2.10.2 WMLScript

A *WMLScript* [WAP 00b] é uma linguagem de *script* procedimental, semelhante ao *JavaScript*, e é utilizada em conjunto com arquivos no formato WML. Ela estende as capacidades de navegação e apresentação do WML com características comportamentais, suporte para um comportamento mais avançado da interface de usuário, adiciona recursos inteligentes ao dispositivo móvel, provê um mecanismo inteligente para acessar o dispositivo e seus periféricos e reduz as necessidades de tráfego ao servidor de origem. A *WMLScript* é baseada num subconjunto de instruções da linguagem de *script* *JavaScript* a qual é bastante utilizada na WWW. Tem a finalidade de criar um meio padrão para adicionar lógica procedimental aos *decks* WML. Em geral, *WMLScript* são funções criadas por provedores de conteúdo pra, por exemplo, verificar se todos os campos de entrada de um *card* estão preenchidos. Essas funções podem ser invocadas por nomes, aceitam parâmetros e retornam valores. Com a utilização desses *scripts*, seleções do usuário (ou entradas) podem ser manipuladas e enviadas a *cards* carregados, o que elimina a excessiva troca de dados entre o cliente e o servidor remoto.

Algumas das capacidades do *WMLScript*:

- Habilidade para validar os dados de entrada do usuário antes de enviar ao servidor de conteúdo;
- Capacidade para acessar as facilidades do dispositivo e seus periféricos;
- Habilidade de interação com o usuário sem causar sobre carga de tráfego ao servidor de origem (ex. mostrar uma mensagem de erro).

Além destas capacidades, *WMLScript* apresenta também outras características interessantes, como a possibilidade de o código *WMLScript* ser compilado para diminuir a quantidade de *bytes* a serem transferidos do servidor ao cliente, e por ser o *WMLScript* uma linguagem baseada em eventos, existe a possibilidade de invocar um *script* para interagir com o usuário ou reagir a eventos de ambiente.

2.10.3 Acesso a Conteúdo WML e WMLScript

A Figura 12, a seguir, apresenta as diferentes partes da arquitetura lógica assumida por um agente de usuário WML.

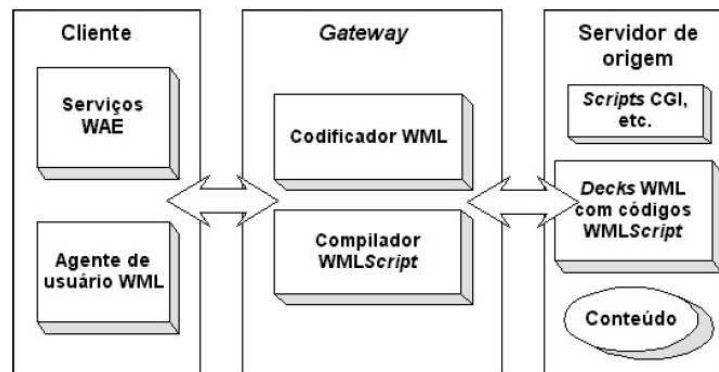


Figura 12 – Arquitetura lógica do agente de usuário WML

Servidores de origem provê serviços de aplicação ao usuário final. O serviço de interação entre o usuário final e o servidor de origem é empacotado como *scripts* e *decks* WML. Serviços podem contar com *decks* e *scripts* que estão armazenados estaticamente no servidor de origem, ou podem contar com conteúdo produzido dinamicamente por aplicações no servidor de origem.

Alguns estágios estão envolvidos quando agentes de usuário WML e servidores de origem trocam informações no formato WML e WMLScript. Em especial, um usuário, desejando acessar um serviço qualquer, envia uma requisição ao servidor de origem utilizando um agente de usuário WML. O agente de usuário requisita o serviço ao servidor de origem em nome do usuário utilizando, algumas operações do esquema URL (ex. método de requisição GET do protocolo HTTP).

O servidor de origem então valida a requisição do usuário e como resposta envia o conteúdo de um *deck*. Presumidamente este *deck* está inicialmente em um formato textual. No seu caminho de ida ao cliente, os *decks* textuais passam por um *gateway* onde são convertidos para formatos mais adaptados para transmissões sem fio e dispositivos de processamento limitado. Em princípio, uma vez que o *gateway* recebe o *deck* do servidor original, o *gateway* realiza todas as conversões necessárias entre o formato de texto e binário. Um codificador WML

contido no *gateway* converte cada *deck* WML para seu respectivo formato binário. O conteúdo codificado é enviado ao cliente para ser visualizado e interpretado. As requisições por conteúdo *WMLScript* são tratadas por um compilador *WMLScript* que compila o *script* em um *bytecode* (otimizando para baixa largura de banda) e envia ao cliente para que seja interpretado e executado.

Não é obrigatória a presença de um *gateway* nessa estrutura, podendo os dispositivos de codificação e compilação estarem disponíveis de outra forma. Pode-se, por exemplo, adicionar estas duas funções ao próprio servidor de origem, onde o conteúdo já se encontra codificado em *bytecode*, o que elimina a necessidade de qualquer conversão.

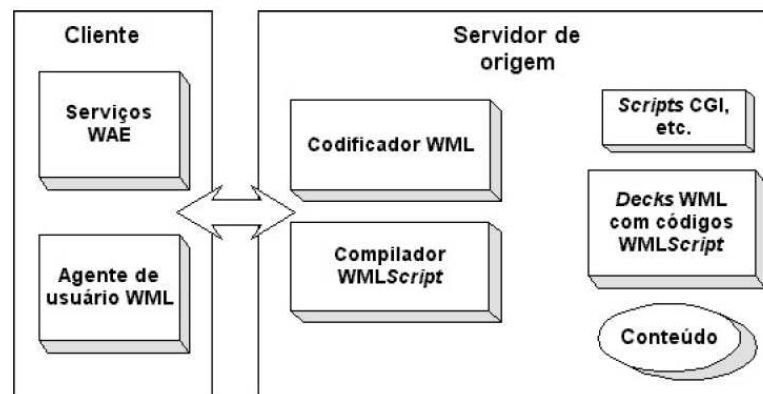


Figura 13 – Arquitetura lógica do agente de usuário WML(sem utilização do *gateway*)

2.10.4 Segurança e Controle de Acesso

A camada de aplicação WAE influencia o protocolo de segurança WTLS onde serviços requerem trocas de informações autenticadas ou seguras define a biblioteca de interface para *WMLScript* e provê funcionalidade de criptografia aos agentes de usuários WAE [WAP 99a]. Adicionalmente, , WML e *WMLScript* incluem instruções de controle de acesso que comunica ao cliente as restrições no acesso baseado em URL. Essas construções permitem que autores de conteúdo WML e *WMLScript* garantam acesso público a determinado conteúdo (ex. um *deck* ou *script* que pode ser referenciado por outro conteúdo) ou restringir o acesso ao conteúdo pelo ajuste de *decks* ou *scripts* “confiáveis”. O protocolo WAE também suporta a autenticação básica

do protocolo HTTP v1.1 onde um servidor pode autenticar o usuário baseado em um nome de usuário e senha.

REFERÊNCIAS BIBLIOGRÁFICAS

- CARVALHO, Allan; ROCHA Leonardo M. **Aprenda WAP**. Endereço eletrônico: <http://www.aprendawap.com.br>. Data da consulta: 20/04/2005.
- Criando Aplicativos WAP**. Data da Consulta: 15/05/2005.
- DEMÉTRIO, Rinaldo. **A Tecnologia WAP: aprenda a criar sites para celulares com linguagem WML**. São Paulo: Érica, 2000.
- DEERTONI, Roberto. **Cinco passos para segurança em WAP**. Endereço Eletrônico: http://www.modulo.com.br/empresa3/noticias/artigo_entrevista/a-5passos.htm. Data da Consulta: 28/06/2005.
- DIAS, Adilson de Souza. **WAP – wireless application protocol: a internet sem fios**. Rio de Janeiro: Ciência Moderna, 2000.
- INFO ABRIL. **WAP Push controla ambulância em SP**. Endereço eletrônico: <http://info.abril.com.br/aberto/infonews/052004/18052004-10.shl> . Data da Consulta: 28/06/2005.
- OPENWAVE. **Openwave WAP Push Library, Developer's Guide Java Edition**. Endereço eletrônico: <http://phone.com>. Data da Consulta: 06/05/2005.
- OPENWAVE. **Push Backward Compatibility**. Endereço eletrônico: <http://phone.com>. Data da Consulta: 05/05/2005.
- OPENWAVE. **The Value of WAP Push**. Endereço Eletrônico: <http://www.openwave.com>. Data da Consulta: 18/05/2005.
- OPENWAVE. **WAP Push, UP, Notify, and SMS Features and Benefits Comparison**. Endereço eletrônico: <http://www.openwave.com>. Data da Consulta: 19/05/2005.
- OPENWAVE. **Comparison of WAP Push and Short Message Service (SMS)**. Endereço eletrônico: <http://www.openwave.com>. Data da Consulta: 19/05/2005.
- PRAIANET. **PHP e WML – Tutorial**. Endereço eletrônico: <http://www.praianet.com/tutoriais/php/wml/phpwmltutorial.php> . Data de Consulta: 20/05/2005.
- TESTECCELL. **WAP - Wireless Application Protocol**. Endereço eletrônico: <http://www.testecell.hpg.ig.com.br/wap.htm> . Data da Consulta: 18/05/2005.
- WIRELESSBRASIL. **WAP Push – Wireless Application Protocol Push**. Endereço eletrônico:

http://geocities.yahoo.com.br/wirelessbrasil/virgilio_fiorese/valor_adicionado_04.html. Data da Consulta: 20/05/2005.

WAPBRASIL. **WAP Brasil**. Endereço eletrônico: <http://www.wapbr.org>. Data da Consulta: 25/06/2005.

WIRELESSBRASIL. **Wireless Application Protocol**. Endereço eletrônico: http://www.wirelessbrasil.org/wirelessbr/colaboradores/liliam_maluta/monitor_uti_17.html. Data da Consulta: 28/06/2005.

XML. **XML - Extensible Markup Language**. Endereço eletrônico: http://www.gta.ufrj.br/grad/00_1/miguel/. Data da Consulta: 25/05/2005.