

UNIVERSIDAD NACIONAL DE INGENIERÍA

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA



ARQUITECTURA WAP Y APLICACIONES

INFORME DE SUFICIENCIA

PARA OPTAR EL TÍTULO PROFESIONAL DE:

INGENIERO ELECTRÓNICO

PRESENTADO POR:

HERBERT CASAVARDE WARTHON

**PROMOCIÓN
2002-I**

**LIMA – PERÚ
2006**

ARQUITECTURA WAP
Y
APLICACIONES

Dedico este trabajo a:
Mis inolvidables padre y madre, inspiración plena de lucha y
sacrificio,
Mis Hermanas: Zeyla, Karina y Betsy por el
apoyo incondicional en mi carrera;
Mi esposa Teresa y mi hijo Sebastian
esperanza de superación.

SUMARIO

Este informe sintetiza los requisitos necesarios para el manejo del protocolo WAP; formación objetivos y resultados.

El protocolo de aplicaciones inalámbricas es en particular un habilitador precoz para todo el potencial que permite ofrecer, como dar acceso libre a la industria emergente a la Internet móvil. Se espera que WAP sea la tecnología que pondrá en marcha la nueva industria para la navegación móvil.

Este informe muestra un resumen sobre protocolos, definiciones, tecnología usada para establecer conexiones con el protocolo WAP, las capas que forman el protocolo.

INDICE

PROLOGO

CAPÍTULO I: INTRODUCCION Y SITUACION ACTUAL DE WAP

1.1	Identificación y Descripción del Tema	02
1.2	Wap 2.0	03
1.3	Objetivos Generales y Específicos	05
1.3.1.	Objetivos Generales	05
1.3.2.	Objetivos Específicos	06

CAPÍTULO II: PLATAFORMA WAP

2.1	Plataforma WAP	07
2.2	La red WAP	08
2.3	Operación WEB vs WAP	10
2.3.1.	Operaciones WEB	10
2.3.2.	Operaciones WAP	11
2.4	WAP Push	12

CAPÍTULO III: ARQUITECTURA WAP

3.1	Arquitectura WAP	14
3.2	Capa de Aplicación WAE	15
3.2.1	WAE User Agents	16
3.2.2	Generador de contenidos	17
3.2.3	Codificación de contenidos	17
3.2.4	Wireless Telephony Application (WTA).	17
3.3	Capa de Sesión	18

3.4	Capa de Transacciones	20
3.5	Capa de Seguridad	21
3.6	Capa de Transporte	23
3.7	Protocolos y servicios más utilizados	27
3.7.1	USSD - Unstructures Supplementary Services Data	27
3.7.2	MMS	27
3.7.3	CSD (circuit switched data)	27
3.7.4	CDPD (cellular digital packet data)	28
3.7.5	GPRS General Packet Radio Service	29

CAPÍTULO IV: DESARROLLO Y APLICACIONES

4.1	Desarrollo de Aplicaciones WAP	32
4.2	Desarrollo de Página WEB	32
4.3.	Desarrollo de Página WAP	33
4.4.	Aplicaciones y servicios WAP	36
4.4.1	Aplicaciones Generales con Servidor WEB	36
4.4.2	Aplicaciones del servidor WTA	37
4.5.	Definición de Términos	37

CAPITULO V: INTRODUCCIÓN AL LENGUAJE WML

5.1	Características de WML.	40
5.2	Entorno de desarrollo.	41
5.3	Card y Deck	41
5.4	Diseño de páginas en WML .	41
5.5	El formato de los textos	42
5.6	La etiqueta card	44
5.7	La etiqueta template	46
5.8	La etiqueta access.	46
5.9	La etiqueta do	47
5.10	La etiqueta go	48
5.11	La etiqueta a	49

5.12	La etiqueta timer	50
5.13	La etiqueta img	51
5.14	La etiqueta table	52
5.15	La etiqueta input	54
5.16	La etiqueta select	55
5.17	La etiqueta option	56
5.18	Ejemplo de aplicación	58
CONCLUSIONES		61
ANEXO A		62
BIBLIOGRAFIA		68

PROLOGO

Este informe consta de cinco capítulos y un anexo, estructurados de la siguiente manera:

El Capítulo I, referido a la introducción y situación actual del protocolo WAP, como fue creado, cuales son los objetivos, que servicios puede brindar y que medios se necesitan para su uso.

El Capítulo II, referido a la Plataforma WAP, donde se describe sobre este tema, dando énfasis a la comunicación inalámbrica para luego dar a conocer los fundamentos, las redes WAP y sus operaciones WAP.

El Capítulo III, referido a la Arquitectura WAP, cuya función es proporcionar un entorno escalable y extensible para el desarrollo de aplicaciones para dispositivos de comunicación móvil, describiendo cada capa y su composición: la capa de aplicación, sesión, transacciones, seguridad y transporte.

El Capítulo IV, se refiere al desarrollo de aplicaciones y servicios WAP (las aplicaciones generales, así como el desarrollo de aplicaciones WEB y el desarrollo de aplicaciones WAP).

El Capítulo V, muestra una introducción al lenguaje WML, como se pueden realizar programas y herramientas que permitan mostrar las aplicaciones WAP.

CAPITULO I: INTRODUCCION Y SITUACION ACTUAL

1.1. Identificación y descripción del tema.

WAP, es un estándar global, no controlado por ninguna compañía en particular. Ericsson, Motorola y Unwires Planet fundaron el WAP FORUM en 1997 con el propósito de definir especificaciones para el desarrollo de aplicaciones en redes de comunicación inalámbricas.

Wireless Application Protocol (WAP) es el resultado de un trabajo continuo para definir especificaciones en el desarrollo de aplicaciones que operan sobre redes de comunicación inalámbricas.

WAP es un estándar global desarrollado para poder ofrecer los servicios de Internet a los usuarios móviles. A pesar que WAP está basado en la tecnología de Internet, WAP e Internet se encuentran lado a lado. Una persona o una empresa que tiene un Sitio de Internet puede hacer disponible la información para un usuario móvil mediante la transformación de páginas de Internet a página de WAP.

Hoy WAP, adapta la tecnología existente del Internet al ambiente móvil. Las tecnologías requeridas para construir el Internet Móvil están avanzando cada día y obteniendo un mayor reconocimiento, gracias a su alta velocidad en el envío de paquetes de datos, transmisión de radio de banda ancha, mayor número de terminales móviles avanzados, mucha capacidad y más calidad en la tecnología de redes.

WAP maneja los diferentes tipos de redes y estándares de terminales móviles, por lo que tiene el potencial de convertirse en un verdadero estándar global. Así que, sin importar las limitaciones de las tecnologías actuales de Internet móvil, los operadores

móviles, proveedores de contenidos, proveedores de servicios, desarrolladores de aplicaciones y empresas de todo tipo, reconocen el potencial de WAP y buscan entrar al mercado lo más oportunamente posible.

WAP se posiciona como la convergencia de 2 tecnologías rápidamente evolucionadas:

Telefonía móvil

Internet

Esta plataforma pretende ofrecer a los usuarios las mismas funcionalidades que se nos ofrecen en la actualidad en la Web, pero con las ventajas y características de un móvil.

Pero WAP no sólo consiste en hacer la Internet móvil. También habrá otros servicios como bajar agendas telefónicas o el manejo de las llamadas entrantes y salientes que hacen de la telefonía móvil un medio aún más fácil de usar.

Objetivos de WAP Forum

Brindar contenidos y servicios de datos de Internet en teléfonos celulares digitales y otros terminales inalámbricos.

Crear una especificación del protocolo global inalámbrica que trabajará entre diferentes tecnologías de red inalámbricas.

Habilitar la creación de contenidos y aplicaciones que escalen por una muy amplia gama de soporte de red y tipos de dispositivos.

Adoptar y extender estándares existentes y tecnología apropiada.

1.2 WAP 2.0

WAP Forum lanzó su primera especificación WAP 1.0 en 1998. WAP 2.0 es la siguiente generación de especificaciones que, como las versiones anteriores, marcó los esfuerzos continuos del Foro WAP para adoptar las más recientes normas y

protocolos de Internet. WAP 2.0 también se optimiza para el uso de anchos de banda más altos y la conexión de paquetes, basado en redes inalámbricas. Mientras mejora la utilización y el soporte en las capacidades de los últimas tecnologías de los dispositivos inalámbricos e Internet, WAP 2.0 también proporciona manejo para la compatibilidad en lo que refiere a aplicaciones y servicios que proporcionan las versiones anteriores de WAP.

Los elementos principales de la arquitectura WAP 2.0 son los siguientes:

Protocol Stack Support –además de WAP stack introducida en WAP 1, WAP 2.0 agrega soportes y servicios en una stack basada en la stack de la Internet estándar, incluyendo el soporte para TCP, TLS y HTTP. Al contener dos stacks, WAP 2.0 proporciona un modelo de conectividad en un amplio rango de redes y portadores inalámbricos.

WAP Application Environment-Normalmente visto como el “WAP Browser” (Navegador de WAP), el ambiente de la aplicación WAP 2.0 ha evolucionado para adoptar los estándares de la Internet Browser Markup Language. Esto ha llevado a la definición del XHTML Mobile Profile (XHTMLMP). XHTMLMP se basa en la plataforma de eXtensible HyperText Markup Language (XHTML) desarrollado por el W3C (World Wide Web Consortium) para reemplazar y mejorar el lenguaje HTML actualmente muy usado.

Servicios adicionales y Capacidades-Las especificaciones de WAP tuvieron artículos que no eran parte del “WAP Stack” ni del “WAP Browser”, pero ayudaron a enriquecer las especificaciones de WAP. Con WAP 2.0, hay un aumento considerable en el número de características disponibles para diseñadores, operadores y usuarios.

Objetivos de WAP 2.0

Con el lanzamiento de WAP 2.0, WAP FORUM tiene los siguientes objetivos:

Incrementar soporte para protocolos de comunicación de Internet. WAP2 provee soporte para protocolos como IP, TCP y HTTP. Por el incremento de estándares y protocolos de Internet en el entorno de las telecomunicaciones, las especificaciones WAP proveen un entorno que permite a los dispositivos utilizar tecnologías existentes en Internet, proveen un ambiente que permite a dispositivos inalámbricos utilizar tecnologías de Internet.

Continuar el trabajo de la versión WAP 1.x, permitiendo el uso de sus aplicaciones y servicios para operar en tecnologías existentes en la interfase aérea y las portadoras; esto incluye las nuevas tecnologías de alta velocidad conocidas como General Packet Radio Service (GPRS) y la 3G (tercera generación celular).

Proveer aplicaciones, las cuales entregan información y servicios interactivos para teléfonos móviles digitales, pagers, PDAs (Personal Digital Assisand) y otros dispositivos inalámbricos.

Administrar características singulares de dispositivos inalámbricos. Estos dispositivos tienen características como: pantallas pequeñas, limitada duración de batería y limitadas memorias ROM y RAM; también presentan consideraciones de interfase de usuario como navegación de una página a la vez.

Minimizar el uso de potencia de procesamiento de los dispositivos y optimización de los recursos de red para minimizar costos y máxima performance.

El Forum WAP trabaja con organizaciones como **World Wide Web Consortium** (W3C) y la **IETF** (Internet Engineering Task Force) para desarrollar especificaciones y así lograr los objetivos.

1.2. Objetivos generales y específicos

1.3.1. Objetivo general

Estudiar la arquitectura WAP para la implementación de soluciones móviles, mostrando las consideraciones de desarrollo en la práctica.

1.3.2. Objetivos específicos

Determinar que la navegación desde móviles permitirá, que los profesionales tengan acceso a datos empresariales, tales como correo electrónico, estado de la producción, listas de precios, y otras informaciones de importancia para hacer negocios, mientras se esté fuera de la oficina.

Mostrar los conceptos necesarios para entender todo lo relacionado con WAP, el desarrollo de aplicaciones en esta tecnología.

Resumen

WAP Forum fue fundado en 1997 por compañías como: Ericsson, Motorola, Nokia, Unwires Planet, para desarrollar un conjunto de protocolos y tecnologías, llegando a ser una asociación de industrias de construcción de dispositivos inalámbricos, proveedores de servicios, industrias de computadoras (IBM, Intel, Microsoft y Hewlett Packard).

WAP es un estándar global no controlado por ninguna compañía en particular. WAP es el resultado de la convergencia de dos tecnologías: Internet y Telefonía móvil.

WAP 2.0 fue lanzado al mercado en 1998, es el resultado de la evolución de las versiones anteriores de (WAP 1.x) para adoptar recientes normas y protocolos de Internet, así como para el soporte de anchos de banda más altos y conexión de paquetes basados en redes inalámbricas.

CAPITULO II: PLATAFORMA WAP

2.1. La plataforma WAP.

La plataforma WAP, se encarga de ofrecer un acceso inalámbrico seguro a un conjunto de servicios de Internet y a otras redes, a los usuarios que accedan mediante su teléfono móvil.

La navegación de un teléfono en la plataforma WAP es igual a la de un navegador Web: el usuario teclea para solicitar una URL. Pero, al contrario que los navegadores estándar que usan HTML para visualizar la información en la pantalla del ordenador, los teléfonos WAP utilizan WML, un lenguaje abierto desarrollado por el WAP Forum, que permite adaptarse a pequeños dispositivos de mano. Al igual que el HTML, WML se construye por medio de "tags" y permite la presentación de texto e imágenes, entrada de información y formularios.

El teléfono WAP utiliza las capacidades de información de conexiones inalámbricas convencionales para que el usuario realice peticiones al Gateway WAP. El Gateway WAP convierte éstas, en peticiones HTTP y las envía a través de Internet. Cuando el servicio requerido responde, el Gateway WAP vuelve a enviar la información al teléfono WAP.

El Gateway WAP es el núcleo de la plataforma WAP. Su capacidad para actuar en esta clase de teléfonos como un proxy HTTP, permite a los suscriptores acceder a cualquier site WWW. Algunos proveedores de información ofrecen igualmente servicios WML y usan WML para aprovechar la interfaz del teléfono WAP. Estos servicios pueden además iniciar la comunicación "impulsando" la información al Gateway WAP, que como respuesta, transmite la misma a un teléfono WAP. Este

proceso se denomina notificación. Además de la translación HTML, la oferta de servicios del Gateway varía. Estos pueden ser un servicio de protección de información por medio del mantenimiento de una base de datos de teléfonos WAP y sus privilegios de acceso, un servicio de fax que permite a los usuarios de teléfonos WAP mandar por fax contenido de un site Web a una máquina de fax local, o servicios de correo, organizadores o directorios. Todos ellos dependen de beneficio de servicios que ofrezca cada Gateway.

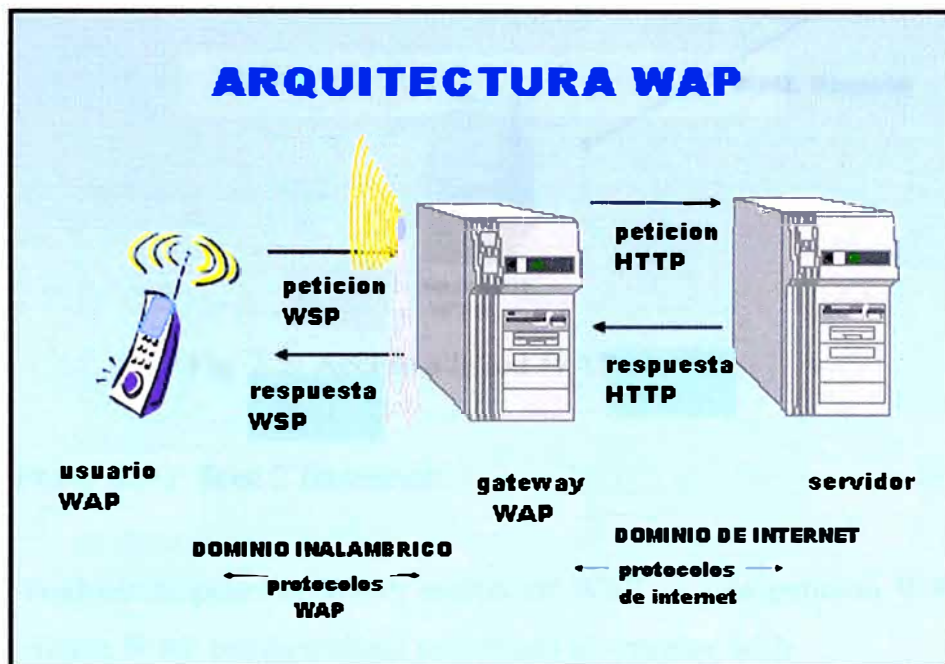


Fig. 2.1: Arquitectura WAP

2.2. La red WAP

La figura 2.2 representa cómo, un cliente WAP puede acceder a la red de datos comunicándose con dos servidores, el proxy WAP y el servidor WTA.

El servidor WTA (Wireless Telephony Application) es un entorno para aplicaciones de telefonía que permite a los operadores la integración de funciones de telefonía propio del dispositivo móvil con micronavegador incorporado.

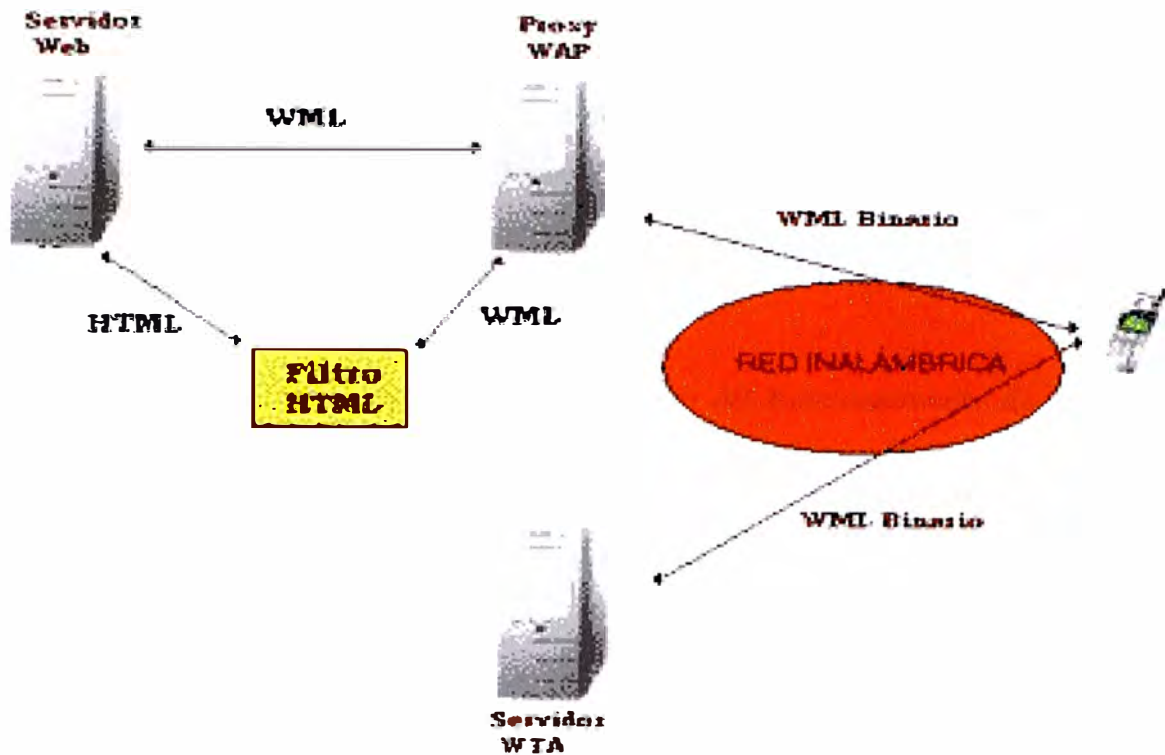


Fig. 2.2: Acceso a la red WAP

El Proxy WAP tiene 2 funciones:

1. Traducir la petición WAP, escrita en WML, a una petición WWW, para que el cliente WAP pueda realizar peticiones al servidor Web.
2. Codificar las respuestas del servidor a un formato binario de modo que sea entendible por el cliente WAP.

El Servidor WTA (Wireless Telephony Application) puede responder las peticiones WAP del cliente directamente; de este modo, ofrece acceso WAP a determinadas características de la infraestructura de comunicaciones del operador de red.

Por otro lado tenemos el Servidor Web, que se comunica con el Proxy WAP de dos modos posibles:

1. Si el servidor Web proporciona un contenido WAP, como por ejemplo WML o WMLS, entonces no se necesita ningún filtro HTML.

2. Si el servidor Web proporciona un servicio WWW, como HTML, entonces se usa un filtro HTML para traducir el contenido WWW en uno WAP. Como se ve en la figura 2.2, el filtro HTML, puede encargarse de traducir una respuesta HTML en una WML, y devolvérsela al Proxy WAP.

2.3. Operación WEB vs. operación WAP

A continuación, se muestra una comparación del funcionamiento de una operación WEB y WAP, dado que la arquitectura de la plataforma WAP está influida por la infraestructura y diseño de la WWW.

2.3.1. Operaciones WEB

1. El usuario abre el navegador y especifica la URL.
3. El navegador analiza la URL y envía una petición HTTP o HTTP segura (HTTPS) al servidor web.
4. El servidor Web analiza la petición y determina qué recuperar. Si la URL especifica un archivo estático como en este ejemplo, el servidor Web lo recupera. Si la URL especifica un programa CGI, el servidor Web inicia el programa.
5. El servidor Web coloca un encabezado HTTP o HTTPS en el archivo estático o programa CGI y lo manda de vuelta al navegador.
6. EL navegador interpreta la respuesta y despliega el contenido al usuario.

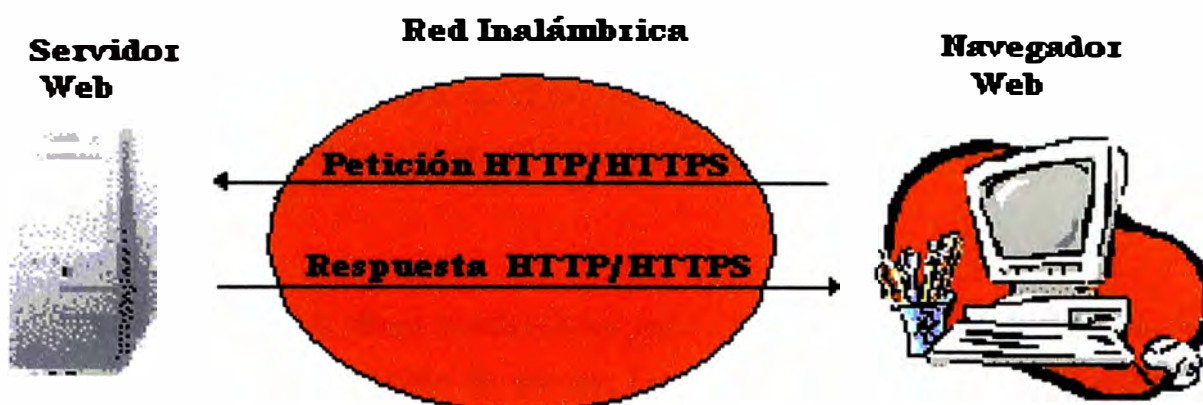


Fig. 2.3: Transacción http

2.3.2. Operaciones WAP

Las transacciones WAP utilizan el mismo modelo básico de WEB, siendo la principal diferencia que el teléfono y el Gateway WAP sustituyen en conjunto al navegador Web.

1. El usuario utiliza un teléfono WAP para solicitar una URL
2. El navegador WAP crea una petición que contiene la URL e información que identifique al suscriptor y las envía al Gateway WAP.
3. El Gateway WAP interpreta la petición, genera una petición convencional HTTP o HTTPS y la envía al servidor Web.
4. El servidor Web interpreta a su vez la petición y determina que recuperar. Si la URL especifica un archivo estático, el servidor Web lo recupera. Si la URL especifica un programa CGI, el servidor Web inicia el programa.
5. El servidor Web coloca un encabezado HTTP o HTTPS en el archivo estático o programa CGI y lo manda de nuevo al Gateway WAP.
6. El Gateway WAP interpreta la respuesta, valida el WML, genera una respuesta (quitando el encabezado HTTP o HTTPS) y lo envía al teléfono WAP.
7. El navegador WAP interpreta la respuesta y despliega el contenido al usuario.

2.4. WAP Push

WAP Push es una función en el Gateway WAP, la cual habilita aplicaciones en el Servidor Origen para enviar información al usuario WAP sin una previa acción requerida.

Los mensajes Push abren posibilidades para nuevas aplicaciones, donde los clientes pueden recibir información de interés. Las aplicaciones Push incluyen por ejemplo notificación de acción (“tu acción ha excedido el valor máximo, deseas enviarlo”) o notificación de transporte (“retardo causado por congestión”). Las aplicaciones Push necesitan implementar el PAP (Push Access Protocol). Esto significa que la aplicación incluye un iniciador Push (PI) que es responsable de la comunicación con el Push Proxy Gateway (PPG). El PI transmite el contenido del Push y entrega instrucciones al PPG usando PAP. El PPG entonces, usa el protocolo Push Over-Air (OTA) para entregar el contenido WAP al terminal o móvil.

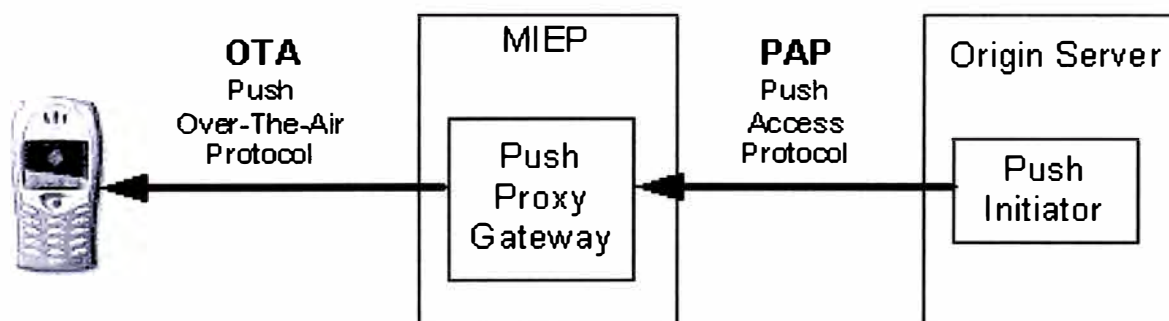


Fig.2.4: WAP PUSH

Un PI puede direccionar a un WAP móvil por su MSISDN (Número de móvil), el ID de usuario (Usuario WAP) o dirección IP.

Push puede hacer esto de dos maneras:

- **Push sobre IP:** usualmente para aplicaciones Push con requerimientos en tiempo real; por ejemplo, aplicaciones basadas en Standard Wireless Village.
- **Push a través de MSN:** Necesario cuando se activa a un WAP móvil que no tiene conexión IP.

Resumen

La plataforma WAP se encarga de ofrecer un acceso inalámbrico a un conjunto de servicios de Internet y otras redes.

La navegación de un usuario móvil es similar a la de un navegador WAP, el cual utiliza el lenguaje WML con las restricciones como; pantallas pequeñas, teclados limitados, memorias pequeñas, etc.

Un cliente WAP puede acceder a la red a través de dos servidores: el Proxy WAP y el Servidor WTA.

El servidor WTA (Wireless Telephony Application) permite a los operadores la integración de funciones de telefonía móvil con el micro navegador.

El Proxy WAP traduce una petición escrita en WML a una petición WWW, también codifica las respuestas del servidor a un formato binario de modo que sea entendible por el cliente WAP.

CAPITULO III ARQUITECTURA WAP

3.1 Arquitectura WAP

La arquitectura WAP está definida para proporcionar un entorno escalable y extensible para el desarrollo de aplicaciones para dispositivos de comunicación móvil.

Para ello, se define una estructura en capas, en la cual cada capa es accesible por la capa superior, así como por otros servicios y aplicaciones a través de un conjunto de interfases muy bien definidas y especificadas.

Las capas de la arquitectura WAP se recogen en el siguiente diagrama (fig.3.1):

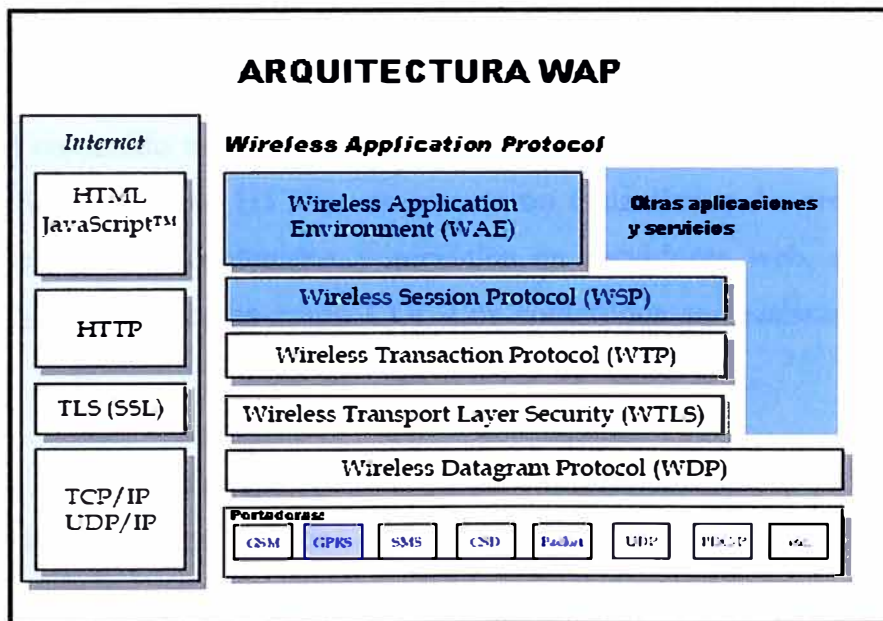


Fig. 3.1: Capas de la arquitectura WAP

A continuación se pasará a describir cada una de las capas

3.2. Capa de Aplicación (WAE):

WAE (Wireless Application Environment), es un entorno de aplicación de propósito general, basado fundamentalmente en la combinación del World Wide Web y tecnologías de Comunicaciones Móviles.

La arquitectura WAE incluye todos los elementos de la arquitectura WAP, relacionados a las especificaciones de aplicación y ejecución.

La arquitectura WAE está predominantemente enfocada a los aspectos de la arquitectura del sistema WAP en el lado cliente, es decir artículos que relacionan al usuario. Específicamente, la arquitectura WAE se define principalmente en términos de redes de computadoras, esquemas, formatos, lenguajes de programación y los servicios compartidos.

El modelo WAE

WAE adopta un modelo, que estrechamente sigue al modelo WWW. Todo el contenido se especifica en formatos, que son similares a los formatos de la Internet estándar. El contenido es transportado y optimizado usando protocolos estándares en el dominio WWW (como HTTP), en el entorno inalámbrico. La arquitectura WAE permite presentar los contenidos y servicios en servidores web, que pueden ser incorporados en tecnologías como CGI. Los contenidos se localizan usando URLs (WWW).

Extensiones WAE se agregan para soportar los servicios de la red móvil, como control de llamada y mensajería. Además soporte para bajo ancho de banda y alta latencia.

WAE asume las funcionalidades del Gateway, en lo que se refiere a la codificación y decodificación de data transferida desde el móvil hacia el cliente. El propósito de

codificar el contenido entregado al cliente, es minimizar el tamaño de los datos enviados al cliente sobre la red inalámbrica; así como, para minimizar la energía computacional requerida por el cliente para procesar esos datos. La funcionalidad del Gateway puede agregarse en los servidores origen o puede localizarse en los Gateways dedicados, como se ilustra en la figura 3.2.

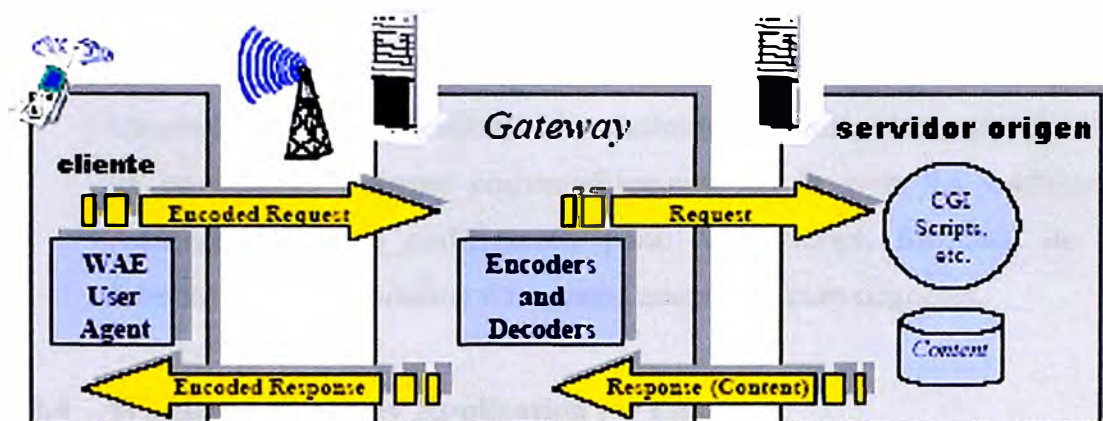


Fig. 3.2: Modelo lógico WAE.

Entre los elementos de WAE se tienen:

WAE User Agents

Generador de contenidos:

Codificación de contenidos:

Wireless Telephony Application (WTA):

3.2.1 WAE User Agents.

Los WAE User Agents son servicios de WAE. El software del dispositivo lado cliente, proporciona funciones específicas en el usuario (ej. contenidos en el display). Los WAE User Agents son integrados en la arquitectura WAP (como los navegadores). Ellos interpretan los contenidos de la red que se refieren a una URL. WAE incluye contenidos para estándares como Wireless Markup Language (WML) y el compilador Wireless Markup Language Script (WMLScript).

3.2.2 Generador de contenidos.

Las aplicaciones (o servicios) en los servidores de origen (ej, CGI scripts) elaboran formatos estándares en respuesta a las peticiones del agente usuario en el terminal móvil.

3.2.3 Codificación de contenidos.

Un conjunto de codificadores bien definidos, permite al usuario WAE (como un navegador) navegar convenientemente en la web. La codificación de contenidos incluye codificación para: WMLScript, formatos de imagen, formatos para calendarios y formatos adaptados para negocios.

3.2.4 Wireless Telephony Application (WTA).

Conjunto de especificaciones de telefonía para llamadas y control de mecanismos para servicios avanzados de la red móvil.

Los resultados de la arquitectura WAE son:

La influencia de Internet. El modelo toma las ventajas, la tecnología e infraestructura desarrollada para Internet.

La ventaja de ser cliente pequeño; el despliegue de servicio tiene significativamente más bajo costo por dispositivo, debido a la naturaleza independiente de WAE y el manejo centralizado de los servicios en los servidores origen.

Proporciona al usuario final servicios avanzados de la red móvil a través de operadores de telefonía.

Mantiene una plataforma abierta, extensible para desarrollar servicios inalámbricos.

Típicamente, un usuario inicia un petición de datos. Sin embargo no todos los datos serán entregados. En muchos casos el servidor origen entrega

información dependiendo de las características del dispositivo. Por ejemplo el servidor origen entregará información al terminal, sin que este, hiciera el pedido, como se muestra en la figura 3.3.

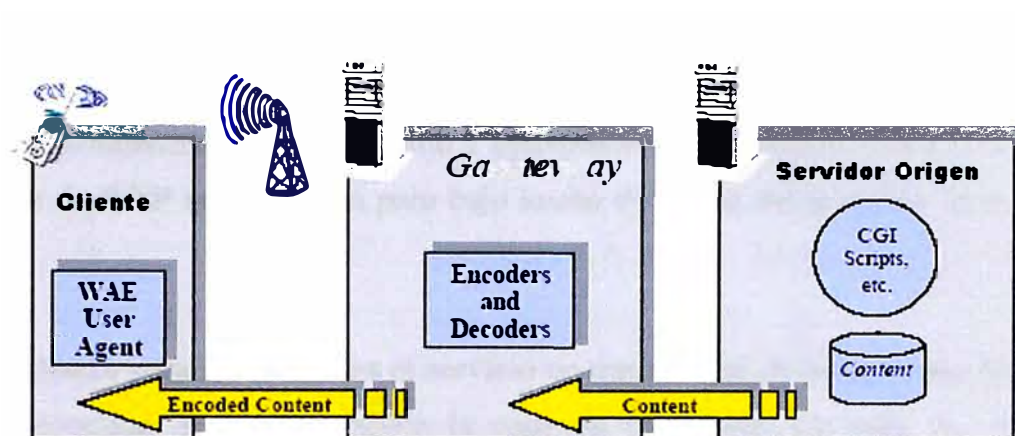


Fig. 3.3: El modelo WAP Push.

Este entorno incluye un micronavegador, que posee las siguientes funcionalidades:

El lenguaje WML.

El lenguaje WMLS.

WTA (Wireless Telephony Applications), el entorno para aplicaciones y servicios de telefonía.

WTAI (Wireless Telephony Application Interface), es una interfaz utilizada en los terminales móviles para operaciones locales de control de llamadas (recepción, inicio y terminación) y acceso a listines telefónicos.

3.3. Capa de Sesión (WSP – Wireless Session Protocol)

La capa de Sesión; este protocolo de la arquitectura WAP se llama Protocolo de Sesión Inalámbrico o WSP. WSP proporciona interfaz consistente con la capa superior (la capa de aplicación WAP), para dos servicios. El primero es un servicio de conexión que opera sobre la capa de transacción WTP (Wireless Transaction

Protocol), y el segundo es un servicio de no conexión que opera sobre un servicio de transporte de datagramas segura o no segura.

Los Protocolos de Sesión Inalámbricos actualmente ofrecen muchos servicios satisfactorios para aplicaciones de navegación (WSP). WSP proporciona funcionalidades HTTP 1.1 e incorpora nuevas características como las sesiones persistentes, capacidad de negociación y suspensión de una sesión. Los protocolos en la familia de WSP se optimizan para bajo ancho de banda del portador con relativa latencia

WSP se diseña para funcionar en el servicio de transacción de datagramas. Se asume que la seguridad es optativa sobre la capa de transporte. La capa de seguridad conserva el servicio de transporte. Las entidades de transacción, sesión y manejo de aplicaciones son asumidas para proveer soporte adicional, para proporcionar el soporte que se exige al establecer contextos de seguridad y las conexiones seguras. Este apoyo no se proporciona directamente por los protocolos de WSP. El WSP no requiere una capa de seguridad; sin embargo, aplicaciones que usan WSP pueden requerirlo.

Funciones de WSP

Esta capa proporciona las siguientes funcionalidades:

Establecimiento y liberación de conexiones entre cliente y servidor

Intercambio de información entre cliente y servidor.

Negociación de las características del protocolo.

Suspensión y reanudación de la sesión.

Este protocolo proporciona a la Capa de Aplicación (WAE) interfaz con dos servicios de sesión:

Un servicio orientado a conexión que funciona por encima de la Capa de Transacciones (WTP).

Un servicio no orientado a conexión que funciona por encima de la Capa de Transporte (WDP), y que proporciona servicio de datagramas seguro o no seguro.

El servicio de no conexión es muy conveniente, cuando las aplicaciones no necesitan entregar datos exactos y no se necesita de la confirmación.

Funcionamiento básico de WSP

La esencia de WSP es el diseño de una forma binaria para HTTP, consecuentemente las peticiones al servidor y las respuestas que van al cliente pueden incluir cabeceras y datos. Soportan todos los métodos definidos por HTTP 1.1. Además, la capacidad de negociación puede ser usada para concordar en un conjunto de métodos requeridos, para la compatibilidad con las aplicaciones de HTTP 1.1.

3.4. Capa de Transacciones (WTP - Wireless Transaction Protocol)

El protocolo de transacción se define para proveer los servicios necesarios para aplicaciones interactivas de navegación (peticiones y respuestas). Durante una sesión de navegación, el cliente hace la petición de información a un servidor, que puede ser fijo o móvil, y el servidor responde con la información. El objetivo del protocolo es entregar la transacción (petición/respuesta) fiablemente mientras se equilibra fiabilidad requerida para la aplicación.

WTP corre sobre un servicio de datagramas y opcionales servicios de seguridad. WTP se ha definido como una liviana transacción de protocolo orientado convenientemente para la aplicación en pequeños clientes (estaciones móviles) y opera eficazmente sobre datagramas de redes inalámbricas.

Entre los beneficios de usar WTP se tiene:

Mejora de la fiabilidad sobre servicios de datagramas. WTP releva la capa superior de re-transmisión y reconocimientos que son necesarios si se usan los servicios de datagramas.

Mejora de la eficiencia sobre servicios de conexión orientados.

Proporciona los servicios necesarios para soportar las transacciones. Estos servicios pueden ser de tres clases:

- Peticiones inseguras de un solo camino.
- Peticiones seguras de un solo camino.
- Transacciones seguras de dos caminos.

También proporciona seguridad en las transacciones.

Características de WTP

La confiabilidad es lograda a través del uso de identificadores, reconocimientos, borrado de duplicados y retransmisiones.

Opcionalmente fiabiliza la conexión usuario a usuario, el usuario de WTP confirma cada mensaje recibido.

El protocolo proporciona los mecanismos para minimizar el número de transacciones.

Aborto de transacción excedente, vaciando datos no enviados por el cliente y servidor. La interrupción puede ser activada por el usuario que cancela un servicio.

Informa el éxito y el fracaso de mensajes. Si un pedido no pudo concretarse, un mensaje de la interrupción se devolverá en lugar del resultado.

3.5. Capa de Seguridad (WTLS)

El protocolo de capa de Seguridad en la arquitectura WAP se llama la Capa de Seguridad de Transporte Inalámbrica, WTLS (Wireless Transport Layer Security). La capa WTLS opera sobre el protocolo de la capa de transporte.

La Capa Inalámbrica de Seguridad de transporte es un protocolo basado en el estándar SSL, utilizado en el entorno Web, para la seguridad en la transferencia de datos, esta capa proporciona a las capas de nivel superior de WAP una interfaz de servicio de transporte seguro, que lo resguarde de una interfaz de transporte inferior.

La meta primaria de la capa WTLS es proporcionar privacidad, integridad de los datos y autenticación en la comunicación entre dos aplicaciones. El protocolo WTLS es optimizado para portadores de red de bajo ancho con relativa latencia.

La comunicación entre capas es lograda por medio de servicios primitivos. Los servicios primitivos representan de una manera abstracta, el cambio lógico de información y control entre la capa de seguridad y las capas adyacentes.

Los servicios primitivos es un conjunto de comandos y sus respectivas peticiones asociadas con los servicios de otras capas. La sintaxis general de una primitiva es:

X-Service.type (parametros)

Donde X designa el servicio de la capa del cual proviene. En este caso si X fuera "SEC" es para la capa de seguridad.

Tiempo de secuencia de los caracteres

El comportamiento de los servicios primitivos se ilustra en la figura 3.4, usando una secuencia de características.

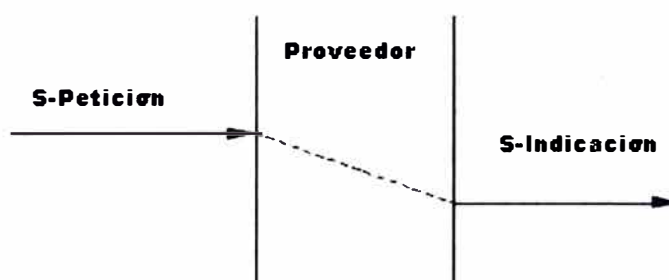


Fig. 3.4 Servicio no confirmado.

La figura 3.4 muestra un simple servicio no confirmado, el cual es invocado usando una petición primitiva el cual resulta en indicación primitiva par. La línea punteada representa la propagación durante un período de tiempo a través del proveedor indicado por la división vertical entre las dos flechas que representan las primitivas.

Las Funciones de esta Capa son las siguientes:

Integridad de los Datos: se asegura que la información intercambiada entre el terminal y el servidor de aplicaciones, no haya sido modificada.

Privacidad de los Datos: se asegura que la información intercambiada entre el terminal y el servidor de aplicaciones, no pueda ser capturada ni entendida por elementos externos a la comunicación.

Autenticación: se ofrecen servicios para determinar la autenticidad del terminal y del servidor de aplicaciones.

También puede ser utilizado para el establecimiento de una comunicación segura entre terminales.

3.6. Capa de Transporte (WDP)

El protocolo de capa de Transporte en la arquitectura de WAP está formada por el Protocolo de la Transacción Inalámbrica (WTP - Wireless Transaction Protocol) y el Protocolo Inalámbrico de Datagramas (WDP – Wireless Datagram Protocol). La capa WDP opera sobre los datos que los servicios del portador son capaces de soportar en todas las redes. Como un servicio general de datagramas, WDP ofrece un servicio consistente a los protocolos de la capas superiores de WAP (las capas de Seguridad, Transacción y Sesión) y tiene una comunicación transparente con los servicios disponibles del portador.

Los protocolos en la familia WAP son diseñados para su uso en portadores de banda corta en las telecomunicaciones inalámbricas.

Puesto que el protocolo WDP provee una interfase común con los protocolos de las capas superiores (las capas de Seguridad, Transacción y Sesión), ellas son capaces de funcionar independientemente de la red inalámbrica subyacente. Esto es logrado adaptando la capa de transporte para especificar rasgos del portador subyacente.

Los servicios ofrecidos por WDP incluyen aplicaciones de direccionamiento de número de puertos, segmentación y reensamblaje para detección de error. Estos servicios permiten a las aplicaciones operar transparentemente sobre los diferentes servicios disponibles del portador.

El modelo para el protocolo Wireless Datagram Protocol es mostrado en la figura 3.5:

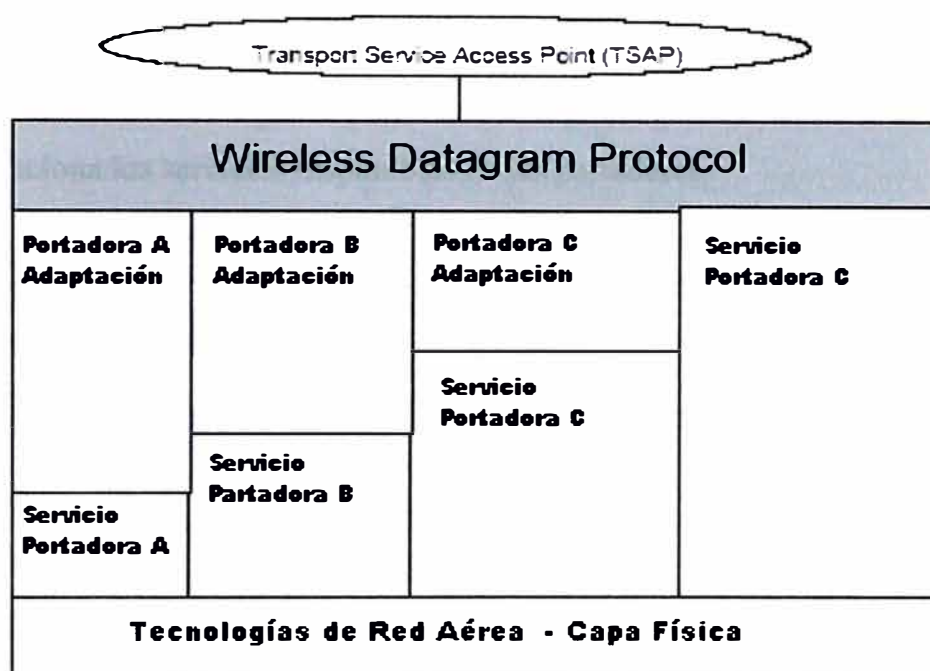


Fig 3.5: Datagrama de la Arquitectura WDP

WDP ofrece un servicio consistente al punto de acceso del servicio de transporte (Transport Service Access Point) sobre la capa superior de WAP. Este servicio permite a las aplicaciones operar sobre los servicios disponibles de los diferentes portadores. La diferencia de alturas de cada uno de los servicios del portador mostradas en la figura 3.5, ilustran la diferencia en las funciones proporcionadas por los portadores y así la diferencia necesaria en el protocolo WDP para operar sobre aquellas portadoras para mantener el mismo servicio que ofrece al Transport Service Access que es logrado por una adaptación de portadoras.

WDP puede trabajar con portadores, con características diferentes. Para perfeccionar el protocolo con respecto al uso de memoria y eficacia de radio transmisión, la actuación del protocolo sobre cada portador puede variar. Sin embargo, el servicio de WDP y el servicio de las primitivas permanecerán iguales, proporcionando una consistente interfase a las capas más altas

Descripción del protocolo WDP

La capa WDP opera sobre los servicios de datos de las portadoras de varios tipos de redes. Como un servicio general de datagramas, WDP ofrece un servicio consistente a los protocolos de las capas superiores de WAP (Seguridad, Transacción y Sesión) e interrelaciona los servicios disponibles de las portadoras.

WDP soporta muchas comunicaciones simultáneas sobre las capas superiores. El número de puerto identifica la entidad de la capa superior de WDP. Esta puede ser otra capa como el Wireless Transaction Protocol (WTP) o el Wireless Session Protocol (WSP) o una aplicación como el correo electrónico. Para re-uso de los elementos de los portadores subyacentes, WDP puede ser implementado para soportar múltiples portadoras y aún puede ser perfeccionada para el funcionamiento eficaz dentro de los recursos limitados de un dispositivo móvil.

La figura 3.6 muestra el modelo general de la arquitectura WAP y cómo WDP encaja en esa arquitectura.

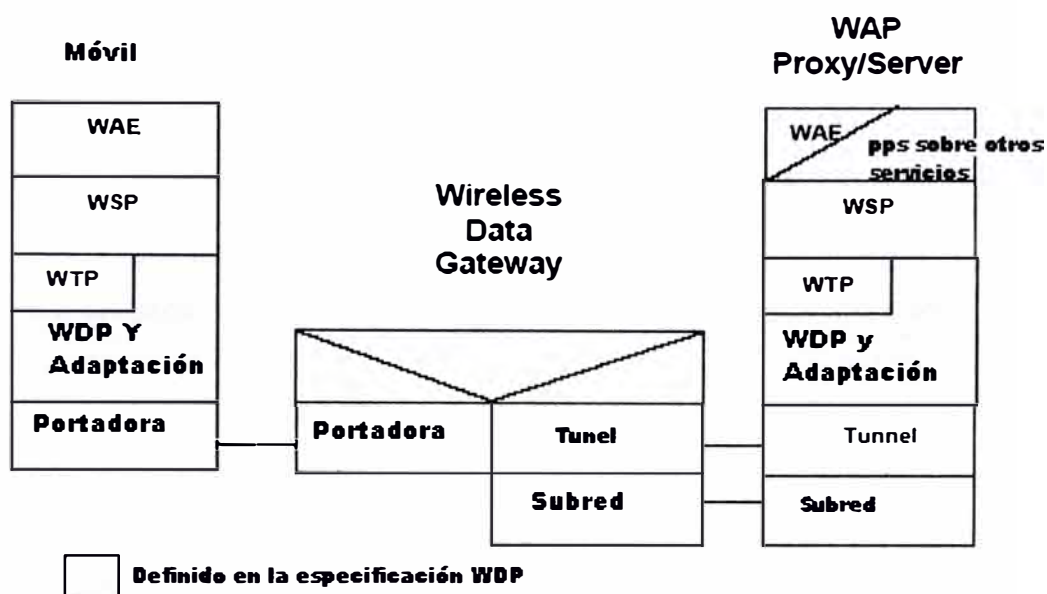


Fig. 3.6: Arquitectura WDP

En la figura 3.6, las áreas sombreadas son las capas del protocolo, en las cuales la WDP es específicamente aplicable. La Capa de Adaptación es la capa del protocolo WDP en la cual el protocolo WDP actúa en una portadora específica. La Capa de adaptación es diferente para cada portador y actúa con las capacidades y características específicas del servicio del portador.

La capa del Portador es el servicio de las portadoras como GSM SMS, USSD, ANSI-136 R-Data o CDMA Packet Data. En el Gateway termina la capa de adaptación y pasan los paquetes WDP hacia el WAP Proxy/Server vía un protocolo Tunel que es la interfase entre el Gateway que soporta al servicio del portador y al WAP Proxy/Server. Por ejemplo, si el portador fuera GSM SMS, el Gateway sería un GSM SMSC y soportaría las especificaciones del protocolo (el protocolo Tunel) para relacionar SMSC a otros servidores. La Subred es cualquier tecnología común de una red de computadoras que puede usarse para conectar dos dispositivos, ejemplos de esto son redes basadas en TCP/IP o X.25, o LANs TCP/IP que operan sobre Ethernet. El Proxy/Server WAP puede ofrecer la aplicación que contenga o pueda actuar como un Gateway entre el protocolo inalámbrico WTP y la Internet alámbrica.

3.7. Protocolos y servicios más utilizados.

3.7.1 USSD - Unstructures Supplementary Services Data

USSD - Unstructures Supplementary Services Data, es un protocolo de transmisión de mensajes que forma parte del sistema GSM de telefonía.

Se trata de una utilidad basada en **sesiones transaccionales** en las que no existe riesgo de pérdidas ni hay duplicidades; y, además, la transmisión se realiza en tiempo real. Otra de sus grandes ventajas es que permite el acceso y la comunicación en zonas de **mínima cobertura**, ya que las comunicaciones a través del protocolo USSD pueden realizarse en condiciones en las que no es posible enviar mensajes SMS o realizar llamadas.

De forma más sencilla, se define como un servicio de mensajería móvil que, a diferencia del SMS, abre una **sesión 'head - tail'** (*extremo a extremo*), que permite diferentes operaciones y que no se corta hasta que éstas finalicen. Así, permiten, por ejemplo, realizar transacciones monetarias en segundos permitiendo el pago a través del móvil.

3.7.2 MMS

MMS (*Multimedia Messaging Service*), permite a los usuarios de teléfonos móviles, con capacidades MMS, enviar mensajes que combinen texto, imágenes, gráficos y sonido en un único mensaje, rico en contenidos, directamente a otro terminal MMS, a direcciones de e-mail o incluso a teléfonos móviles no MMS.

3.7.3 CSD (Circuit Switched Data)

Así como las comunicaciones móviles están incorporándose rápidamente dentro de la vida diaria, la demanda al acceso de datos móviles y fax está experimentando un crecimiento de la misma forma que la voz. La tendencia de los datos móviles están destinados a seguir el mismo patrón de los datos

alámbricos el cual actualmente comprende el 25% del tráfico en las redes de telefonía fija y más del 50% de las ganancias en algunos casos.

La tecnología de Datos en Circuitos conmutados (CSD – Circuit Switched Data) permite a los operadores inalámbricos entrar a este mercado lucrativo, rentable y de rápido crecimiento. La función permite a los abonados móviles transferir datos sobre las redes inalámbricas usando un Canal de Tráfico Digital (DTC - Digital Traffic Channel) y una Unidad de Interconexión o Interworking Unit (IWU).

Además con el empleo del Sistema Integrado de Acceso (IAS) el abonado móvil podrá acceder a las redes de Internet o intranet a través de una conexión IP sin tener que pasar por la red de telefonía.

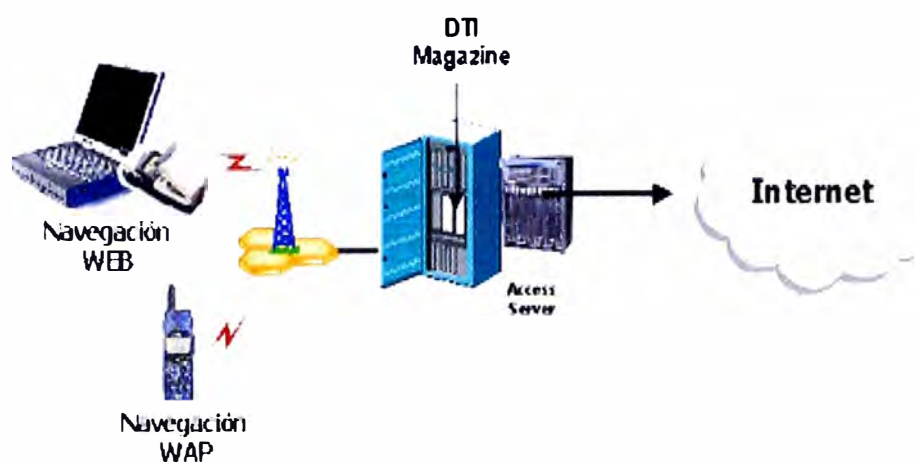


Fig. 3.7. Tecnología CSD.

3.7.4 CDPD (Cellular Digital Packet Data)

CDPD es una especificación abierta y soporta roaming y operación similar a través de redes CDPD de otros operadores. CDPD usa la misma frecuencia y ancho de banda de canales AMPS/TDMA. Por lo tanto, la planeación celular, sitios y líneas de transmisión pueden ser compartidos con el sistema celular.

CDPD provee un servicio seguro ya que incluye autenticación de los móviles y encriptación automática de datos sobre la Interfaz de aire. CDPD optimiza el espectro ya que puede dar servicio a varios subscriptores mediante un solo canal de paquetes de datos compartidos.

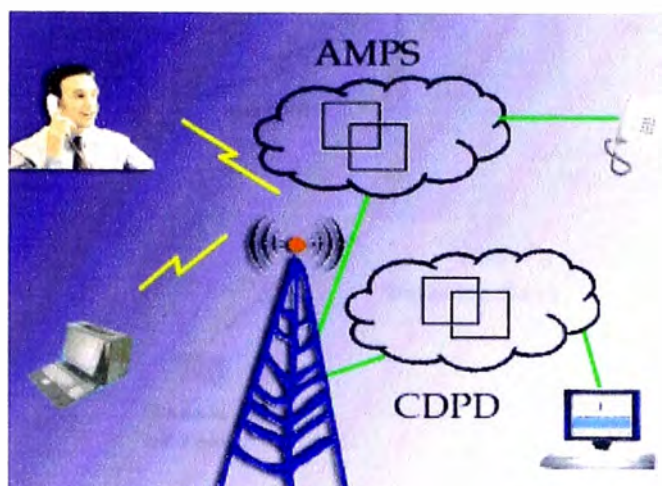


Fig. 3.8. Tecnología CDPD

3.7.5 GPRS General Packet Radio Service

La tecnología GPRS transfiere datos en paquetes a alta velocidad, utilizando uno o varios canales de la red GSM. Los usuarios comparten estos canales por lo que las velocidades de transmisión varían dependiendo del número de usuarios simultáneos conectados y de la carga de la red.

Las principales ventajas de GPRS son:

Siempre está conectado (always on).

Se puede recibir llamadas mientras se está conectado.

El cobro se realiza con base en la cantidad de información enviada/recibida

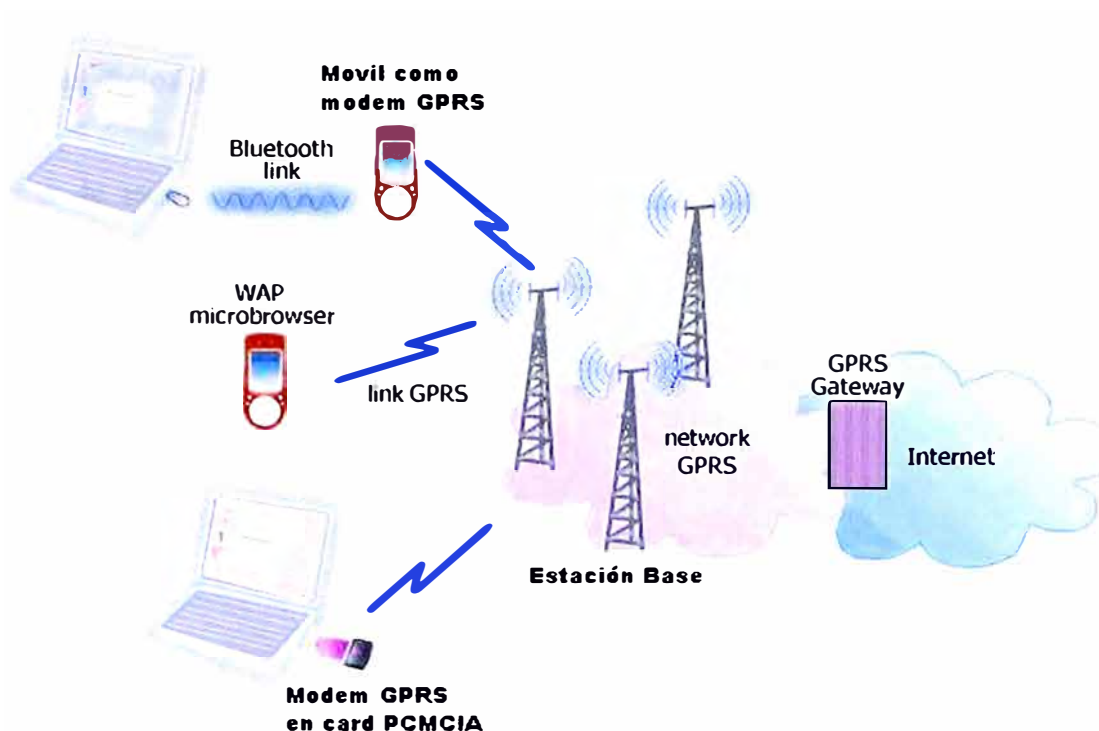


Fig. 3.9. GPRS General Packet Radio Service

Resumen

La arquitectura WAP está definida para proporcionar un entorno escalable y extensible para el desarrollo de aplicaciones para dispositivos de comunicación móvil. Para ello, se define una estructura de capas, en la cual cada capa es accesible por la capa superior.

La capa de aplicación WAE se encarga del enlace entre la WWW y las tecnologías de comunicaciones móviles; incluye un micronavegador que trabaja con el lenguaje WML.

La capa de Sesión (WSP) proporciona funciones como: establecimiento y liberación de conexiones, intercambio de información entre cliente y servidor; también establece la suspensión y reanudación de una sesión.

La capa de transacción (WTP) proporciona servicios para soportar transacciones como peticiones seguras e inseguras de un solo camino, transacciones seguras de dos caminos.

La capa de seguridad (WTLS) se encarga de la seguridad en la transferencia de datos, privacidad de los datos y autenticación.

La capa de transporte (WDP) permite la comunicación con los protocolos de las portadoras CDMA, SMS, GSD. Opera sobre los datos que los servicios del portador son capaces de soportar en todas las redes.

CAPITULO IV

DESARROLLO Y APLICACIONES

4.1. Desarrollo de aplicaciones WAP:

El desarrollo de aplicaciones WAP es muy similar a aplicaciones WEB.

Esta sección se ha dividido en tres partes:

Desarrollo de páginas WEB.

Desarrollo de páginas WAP.

Ejemplo práctico del desarrollo de una aplicación WAP.

4.2. Desarrollo de página WEB.

Para el desarrollo de páginas Web, es necesario estudiar un lenguaje de programación, en este caso, el lenguaje es HTML para páginas estáticas; y para páginas dinámicas asp, php, etc.

Para empezar con la programación de la página, se necesita un editor para escribir el código HTML; en la actualidad existen muchos programas que facilitan la construcción de una página como: DreamWeaver, FrontPage, etc., los cuales permiten diseñar una página Web de forma gráfica.

Una vez que la página esté codificada, para poder ver como quedaría si realmente estuviera en un servidor, se puede usar los browsers o navegadores, Microsoft Internet Explorer, Netscape Communicator, que permiten ver la página aunque esté ubicada en el disco duro; lo que no se puede hacer es ejecutar las páginas dinámicas, y para hacer esto si es necesario ubicarla en el servidor .

Cuando ya esté lista, y se hayan hecho las pruebas pertinentes de cómo queda con el navegador, entonces se puede llevar la página Web a un servidor, y una vez ubicada, se podrá acceder a la página con un ordenador que tenga acceso a Internet.

4.3. Desarrollo de página WAP

Para el desarrollo de una página WAP, primero debe familiarizarse con el lenguaje de desarrollo para páginas Web, en el cual se usó el HTML, asp, JavaScripts. En el caso de páginas WAP el lenguaje de desarrollo es el WML para las páginas estáticas, y el WMLS para las paginas dinámicas.

Se necesita cualquier editor de textos, para escribir el código de la página Wap, aunque es recomendable, como primer paso, utilizar un developer toolkit, es decir, un paquete de herramientas para desarrollar aplicaciones WAP. En la actualidad se tiene varios tipos, entre ellos:

Nokia Wap Toolkit.

UP.SDK, Kit de Ericsson.

Estas herramientas para desarrollar aplicaciones WAP, son muy útiles, ya que contienen editores, compiladores y ejemplos de aplicaciones. Además, éstos paquetes, contienen también el agente que interpreta éste lenguaje, de modo que además de servir como editores, también sirven de navegadores.

Una vez que se tiene lista la página codificada, se puede ver como quedaría si realmente estuviera en un servidor, para ello se tiene los browsers o navegadores que vienen incluidos en los “kits de desarrollo”.

Estos navegadores adoptan el aspecto de un teléfono móvil y simulan como quedaría una aplicación si realmente estuviera en un servidor y se estuviese accediendo desde un teléfono.

Estos navegadores poseen una ventaja, permiten ejecutar el código dinámico, pero sólo el escrito en WML. En este punto hay que mencionar que para realizar las páginas dinámicas, se puede usar el WMLS, pero lo que aún no se proporciona a este código es el acceso a Base de Datos, lo que si proporcionan otros lenguajes como el JavaScript, o el uso de Servlets. Entonces, lo que se hace es invocar desde el código escrito en WML y WMLS al código dinámico. Una vez que se incluye en la aplicación otro código dinámico que no sea WMLS, entonces ya no se puede simular sin tenerlo ubicado en el servidor.

Los navegadores, vienen incluidos dentro de los Kits de desarrollo.

Cuando ya esté lista la aplicación WAP, se debe llevar a un servidor, en el servidor existen dos posibilidades para acceder a la aplicación:

1. Que se acceda desde un teléfono móvil con la tecnología WAP; cada teléfono móvil posee su propia configuración para poder insertar la dirección de la página WAP.
2. Si no se tuviera un teléfono con acceso a WAP, se puede acceder desde una PC, gracias a los navegadores o browsers que se comentó en el punto anterior.

El único problema que se puede presentar al desarrollar una aplicación para WAP, es el tema de los servidores, a continuación se han plasmado las diferentes preguntas que pueden surgir y se ha dado respuesta a cada una de ellas:

¿Hace falta un servidor especial que soporte las páginas WAP?
En principio, basta con un servidor normal que sirva páginas de Internet, lo único que se tendrá que hacer es configurar los MIME Types.

¿Qué son los MIME Types y cómo se configuran?. Al acceder a una página web usando un navegador de una PC, el servidor indica al navegador qué clase de documento está sirviendo, función que se realiza usando los caracteres MIME.

Si no se configura los MIME Types característicos de WAP, el servidor no podrá indicar al navegador nada sobre el tipo de documento. Algunos códigos MIME types se presentan a continuación:

- Para el código WML text/vnd.wap.wml .wml.
- Para las imágenes wml (extensión wbmp) image/vnd.wap.wbmp .wbmp.
- Para el WML Script text/vnd.wap.wmlscript .wmls.
- Para el WML compilado text/vnd.wap.wmlc .wmlc.
- Para el WML Script compilado text/vnd.wap.wmlscriptc .wmlsc.

¿Cómo se puede establecer un servicio propio de operador WAP? Las soluciones que ofrece el mercado son muchas, desde el sofisticado Cisco Access Servers (AS5200) hasta las más sencillas y baratas soluciones basadas en software como Microsoft, NT Remote Access Service y Novell's Netware Connect.

Las instrucciones para la instalación se pueden encontrar en "Configuring your NT server to accept remote connections" en la que explica cómo configurarlo en una máquina con Microsoft NT 4.0 y un módem con una capacidad igual o mayor que 9600 bits/segundo. Aunque las instrucciones serán similares para otras máquinas.

Después de la instalación será necesario establecer un Gateway personal WAP y un servidor HTTP o usar un WAP Gateway y un servidor HTTP de uso público. (Ver anexo A)

¿Cómo se puede montar un Gateway privado y quién proporciona el número?

Existen varias posibilidades:

1. Construirlo siguiendo cada una de las especificaciones del WAP Forum, aunque esta tarea es muy difícil y pesada.

2. Comprarlo, aunque esta opción no es nada barata, y sólo es recomendable cuando se va a hacer gran uso de él.
3. La forma sencilla sería, montar en un ordenador un RAS (Remote Access Server), que puede ser Windows NT con un MODEM. Entonces configurar un nuevo Gateway (o acceso) en el móvil WAP, con el número de teléfono (que debe ser confidencial para mayor seguridad), y la contraseña (seguridad adicional). Siempre que se conecte con ese acceso, se puede tener acceso a la Intranet de una empresa de forma segura porque es una conexión punto a punto, que no pasa por Internet. Además del RAS, se tiene que montar el Gateway WAP, que puede ejecutarse en la misma máquina del RAS, pero es un servicio separado. En principio los elementos necesarios serian : móvil - red GSM - servidor RAS - servidor WAP - internet o Intranet - servidor Web. (Ver anexo A)

4.4. Aplicaciones y servicios WAP

Se han dividido las aplicaciones que se han ido recopilando en dos tipos, por un lado las aplicaciones generales que se pueden hacer para un Servidor Web normal que permita WML y WMLS. y por otro las aplicaciones específicas que se pueden encontrar para el Servidor WTA.

4.4.1 Aplicaciones Generales con Servidor WEB que acepte_wml, wmls

En general las aplicaciones y servicios que se pueden ofrecer, son del mismo tipo que las aplicaciones Web, por lo que nos resultarán conocidas:

- Consulta de todo tipo de información y noticias.
- Últimas noticias de política, sociedad, economía, cultura.
- Noticias y novedades del mundo del deporte.
- Programación diaria de las cadenas de televisión .
- Cartelera de cines.
- Consulta de documentos remotos.
- Servicios de banca.

- Comercio electrónico.
- Acceso a aplicaciones corporativas residentes en redes locales o intranets.
- Servicio de e-mail, noticias y otros servicios que ofrecen los operadores de telefonía.
- Acceso a todo tipo aplicaciones corporativas residentes en redes locales o intranets, como por ejemplo:

Envío de faxes y e-mail al servidor de la Intranet .

Envío de e-mails a los móviles de la compañía.

Agenda.

Acceso a la base de datos de una compañía.

4.4.2. Aplicaciones del Servidor WTA

Cabe resaltar que el servidor WTA (Wireless Telephony Application) da acceso WAP con las facilidades proporcionadas por la infraestructura de telecomunicaciones del proveedor de acceso a la red. Las posibles aplicaciones que se pueden ofrecer con este servidor son las siguientes:

- Factura de teléfono
- Servicios ofrecidos
- Buzones de sugerencias
- Mensajes cortos entre móviles.
- Acceso a los datos de una Intranet local.

4.5. Definición de términos

A continuación se definen los términos más usados al utilizar tecnología WAP.

- **PDA** Siglas para Personal Digital Assistant. (Asistente Personal Digital). Se refiere a aquellos dispositivos de mano que pueden combinar ordenador, teléfono, fax y conexión a Internet, como por ejemplo, una agenda electrónica tipo Palm V.

- **A (User Agent)** Siglas de "User Agent". Se refiere al software capaz de interpretar códigos como el WML o WMLScript, que está instalado en un dispositivo WAP.
- **WAP Gateway** Dispositivo de dos direcciones a través del cual se convierte el contenido que hay en el WAP Server al formato WML que pueda entender un teléfono WAP. Igualmente, a través de él y desde el punto de vista del servidor de HTTP, el WAP Gateway envía información acerca del dispositivo WAP por medio de los encabezamientos (headers) HTTP.
- **WAP Server** Básicamente es un servidor HTTP, es decir un servidor de Internet como otro cualquiera. En cambio, Nokia llama WAP Server a un conjunto de dos productos: un Gateway WAP y un servidor HTTP (servidor de contenidos). Pero un WAP Server es un simple servidor HTTP.
- **WMLScript** Lenguaje de programación con base en JavaScript, aunque menos potente, dirigida a los dispositivos WAP.
- **WTAI** Son las siglas de Wireless Telephony Application Interface. Es una interfaz utilizada en los terminales móviles para operaciones locales de control de llamadas (recepción, iniciación y terminación) y de acceso a listines telefónicos.
- **ML** Extensible Markup Language. Partiendo de este lenguaje se ha formado el WML, por tanto el WML tiene su origen en este lenguaje.

Resumen

Para realizar una aplicación WAP se debe tener conocimiento del lenguaje de desarrollo para páginas WEB (HTML, USP, JavaScript). El lenguaje utilizado para desarrollar páginas WAP es el WML para páginas estáticas Y WMLS para páginas dinámicas.

También es necesario tener herramientas como el developer toolkit para desarrollar aplicaciones WAP como el Nokia WAP Toolkit, o el UP.SDK de Ericsson, los cuales contienen editores, compiladores y pueden servir como navegadores.

Una aplicación WAP para poder ser vista, debe ser contenida en un servidor; los contenidos accesados son similares a las aplicaciones WEB, que pueden brindar muchos servicios de ventajas de importancia como: consultas de todo tipo de información, servicios de banca, comercio electrónico, e-mails, noticias, etc.

El servidor WTA ofrece el acceso a WAP con las facilidades proporcionadas por la infraestructura de telecomunicaciones del proveedor de acceso a la red.

CAPITULO V

INTRODUCCION AL LENGUAJE WML

5.1. Características de WML

El lenguaje WML (*Wireless Markup Language*) esta basado en el lenguaje XML (Extensible Markup Language) y fue desarrollado para especificación de contenidos visualizables desde un terminal WAP (un terminal dotado con un micro-navegador WAP).

El alcance de este dispositivo tiene capacidad limitada, o son diseñados para propósitos especiales. Un teléfono móvil típicamente tiene un teclado con pocas funciones específicas. Dispositivos más sofisticados pueden tener funciones para programar software, pero no tienen por ejemplo un mouse.

Las características principales pueden ser agrupadas en cuatro

- WML ofrece soporte para texto e imagen, y tiene una variedad de formatos y comandos.
- Los cards (cartas) WML son agrupadas en decks (barajas). Una card WML es similar a una página WEB y se identifica por una URL que es la unidad de contenido de transmisión.
- WML ofrece soporte para la navegación entre cards y decks, e incluye comandos para su manejo. Pueden ser usados para navegación y ejecución de scripts. WML también soporta fijación de links, similar al usado en HTML versión 4.
- Las variables pueden ser usadas en lugar de strings y son sustituidas en rutinas. Esto permite una mayor eficiencia.

Las páginas WML podrán ser ubicadas en cualquier servidor *web* convencional, siempre que éste, haya sido correctamente configurado para servir las.

5.2. Entorno de desarrollo

Para poder desarrollar páginas WAP se necesita un editor de texto convencional (por ejemplo, el NotePad de Windows).

Sin embargo, para probar las páginas no se puede utilizar un navegador de Internet. Se necesita un *software* específico que emule el comportamiento del micro-navegador de un terminal móvil (es decir, que simule ser un cliente WAP).

5.3. Card y Deck

En el diseño de páginas WML la unidad de desarrollo es la *card* (carta). Una página puede contener varias *cards*. Sin embargo, en la pantalla del terminal WAP se mostrará siempre una sola *card*. El usuario podrá navegar de una *card* a otra, para visualizar los contenidos de la página.

Un conjunto de *cards* constituirán una *deck*. La *deck* es la unidad mínima de transmisión entre el servidor y el terminal móvil. Cuando el terminal recibe la *deck* completa, mostrará en su pantalla el primer *card*. De este modo, la navegación se realiza siempre entre las diferentes *cards* que componen la *deck*, hasta que se carga una nueva *deck*.

5.4. Diseño de páginas en WML

Toda página WML debe comenzar con la siguiente cabecera:


```
<?xml version="1.0"?>
```

```
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
```

```
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

El resto del código va entre las etiquetas <wml> y </wml>.

5.5. El formato de los textos

Para conseguir una presentación aceptable de los datos, se debe tener en cuenta el formato del texto, ya que se proporciona un pequeño espacio en el navegador WML.

El texto debe estar dentro de una etiqueta `<p> Texto... </p>`

Tabla 5.1: Modificadores de texto

Etiqueta	Uso
	Texto con énfasis
	Texto con mucho énfasis
<i>	Texto con cursiva
	Texto con negrilla
<u>	Texto con subrayado
<big>	Texto con letra grande
<small>	Texto con letra pequeña

La alineación también es importante, y esta se consigue utilizando la etiqueta `<p align="..."> Texto </p>`.

En la tabla 5.2, se muestra su uso.

Tabla 5.2: Alineación de texto

Alineación a la izquierda (Por defecto)	Alineación al centro	Alineación a la derecha
<code><p align="left"></code> El resultado final <code>
</code> es el siguiente <code></p></code>	<code><p align="center"></code> El resultado final <code>
</code> es el siguiente <code></p></code>	<code><p align="right"></code> El resultado final <code>
</code> es el siguiente <code></p></code>
El resultado final es el siguiente	El resultado final es el siguiente	El resultado final es el siguiente

*La etiqueta `
` se utiliza para hacer un salto de línea al igual que en HTML, hay ciertos caracteres que no se pueden escribir literalmente (ver tabla 5.3).*

Tabla 5.3: Caracteres especiales.

Caracteres especiales	
á	<code>&#xE1;</code>
é	<code>&#xE9;</code>
í	<code>&#xED;</code>
ó	<code>&#xF3;</code>
ú	<code>&#xFA;</code>
ñ	<code>&#xF1;</code>
&	<code>&#38</code>
cedilla	<code>&#xE7;</code>
"	<code>&quot;</code>
abrir comillas	<code>&#xAB;</code>
cerrar comillas	<code>&#xBB;</code>
apóstrofe	<code>&#39;</code>
<	<code>&#60;</code>
>	<code>&#62;</code>
nbsp	<code>&#160;</code>
\$	<code>\$\$</code>

5.6 . La etiqueta card

Una WML deck consiste de una o más <cards>, cada una de las cuales especifican una sola interacción entre el usuario y el dispositivo. Los dispositivos muestran una card a la vez.

Sintaxis:

```
<wml>
  <card id="name" title="label" newcontext="boolean"
  ordered="true" onenterforward="url" onenterbackward="url"
  ontimer="url">
    content
  </card>
</wml>
```

Donde *content* representa la card WML

Los atributos de esta etiqueta son:

Tabla 5.4. Atributos de la etiqueta card.

Etiqueta	Explicación	Valores que puede tomar
<i>id</i>	Nombre de la card.	Texto
<i>title</i>	Título, mostrado en la parte superior del navegador al entrar en la card	Texto
<i>newcontext</i>	Borra el historial de navegación. Resetea todas las variables. Sólo actúa al realizar el evento <i>go</i> .	True ó false (valor por defecto)
<i>onenterforward</i>	Llamada a una dirección (ir adelante), se usa con la etiqueta <i>go</i> .	Href. URL o dirección de otro card.
<i>onenterbackward</i>	Llamada a una dirección (ir atrás), se usa con la etiqueta <i>go</i> .	href. URL o dirección de otro card.

<i>ontimer</i>	Llamada a una dirección (reloj), con la etiqueta <i>timer</i> .	Href. URL o dirección de otro card.
----------------	---	-------------------------------------

El siguiente programa define una deck, mostrada en la figura 5.1.

```

<wml>
<card>
<do type="accept">
<go href="#card2"/>
</do>
<p>
Press OK to display the next screen.
</p>
</card>
<card id="card2">
<p>
This screen displays Card 2.
</p>
</card>
</wml>

```

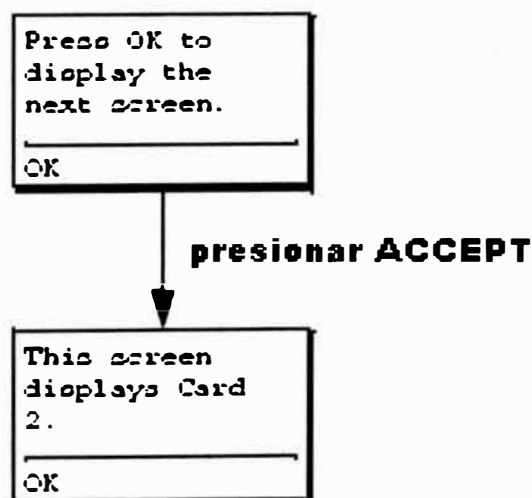


Fig. 5.1. Navegación de una card dentro de una deck.

5.6. La etiqueta `template`

Una WML deck puede contener una etiqueta `<template>` que define los valores que se aplican a las cards en una deck. Los atributos de esta etiqueta son:

Tabla 5.5: Atributos de la etiqueta `template`

Etiqueta	Explicación	Valores que puede tomar
<i>id</i>	Nombre de la plantilla.	Texto
<i>onenterforward</i>	Llamada a una dirección (ir adelante), con la etiqueta <i>go</i> .	Href. URL o dirección de otro card.
<i>onenterbackward</i>	Llamada a una dirección (atrás), con la etiqueta <i>go</i> .	href. URL o dirección de otro card.
<i>ontimer</i>	Llamada a una dirección (reloj) con la etiqueta <i>timer</i> .	Href. URL o dirección de otro card.

Ejemplo:

```
<template id="Template_1">
  <do type="options" name="hazlo" label="Default">
    Código eliminado...
  </do>
</template>
<card id="Card 1">
  <do type="options" name="hazlo">
    Código eliminado...
  </do>
</card>
```

5.7. La etiqueta `access`

La etiqueta `<access>` especifica la información de control de acceso para un deck WML. Una deck puede tener una sola etiqueta `<access>`.

Syntax

```
<head>
```

```
<access domain="domain" path="path" />
```

```
</head>
```

Los atributos de esta etiqueta son:

Tabla 5.6: Atributos de la etiqueta access

Etiqueta	Explicación	Valores que puede tomar
<i>id</i>	Nombre de la etiqueta.	Texto
<i>domain</i>	Dominio de acceso	Href
<i>path</i>	Dirección de acceso	Href

5.8. La etiqueta do

La etiqueta <do> asocia una tarea con un elemento dentro de la interfase de usuario (como teclado, comando de voz).

Sintaxis.

```
<do type="type" label="label" name="name" optional="boolean">
```

```
task
```

```
</do>
```

Los atributos de esta etiqueta son:

Tabla 5.7: Atributos de la etiqueta do.

Etiqueta	Explicación	Valores que puede tomar
<i>name</i>	Nombre del objeto.	Texto
<i>label</i>	Texto que aparece al pulsar el botón options en el navegador para iniciar el evento	Texto
<i>type</i>	Define el tipo de evento que se va a	<i>Accept</i> Aceptar

	realizar	<i>prev</i> Ir atrás <i>reset</i> Resetear <i>options</i> Operaciones opcionales <i>delete</i> Borra un objeto
--	----------	--

5.9. La etiqueta go

La etiqueta `<go>` asocia una tarea con un elemento dentro de la interfaz del usuario (por ejemplo, una tecla funcional, el botón de gráfico). Cuando el usuario invoca el mecanismo de interfaz de usuario, el dispositivo realiza la tarea asociada a la etiqueta `<do>`.

La etiqueta `<go>` se ha extendido para soportar especificaciones de imágenes. Esto se especifica con la etiqueta ``.

Por ejemplo:

```
<do type="accept">
  <go href="/foo"/>
  
</do>
```

Sintaxis:

```
<do type="type" label="label" name="name" optional="boolean">
  task
</do>
```

Los atributos de esta etiqueta son:

Tabla 5.8: Atributos de la etiqueta go.

Etiqueta	Explicación	Valores que puede tomar
<i>href</i>	Dirección a la que se quiere ir	Href. Url o nombre de otra card
<i>Meted</i>	Utilizado para enviar datos al servidor.	post o get

Nota: esta etiqueta siempre acaba con una barra `<go href="..." />`

5.10. La etiqueta a

Esta etiqueta es una sintaxis corta de la etiqueta anchor. Se usa `<a>` en lugar de `<anchor>`, y sólo puede usarse para definir una etiqueta `<go>`, tarea que requiere una especificación URL.

Sintaxis:

```
<a href="url" <!-- required --> title="label">
...any valid combination of <text>, <br/> and <img> elements
</a>
```

Los atributos de esta etiqueta son:

Tabla 5.9: Atributos de la etiqueta `<a>`.

Etiqueta	Explicación	Valores que puede tomar
<i>title</i>	El texto que sale al entrar en las opciones	Texto
<i>href</i>	Dirección a la que se accederá	Href. URL u otra card

5.11. La etiqueta timer

La etiqueta `<timer>` provee un método para invocar automáticamente una tarea después de algún período de tiempo. Se puede asociar sólo una tarea para cada timer, y se puede definir sólo un timer para cada card.

Sintaxis:

```
<timer name="variable" value="value"/>
```

Los atributos de esta etiqueta son:

Tabla 5.10: Atributos de la etiqueta timer

Etiqueta	Explicación	Valores que puede tomar
id	identificador del objeto. Usado para referirse a ella	Texto
value	Tiempo que debe pasar para que se inicialice el evento. Se expresa en décimas de segundo	Número

El siguiente ejemplo ilustra cómo un timer puede iniciar y reusar un contador. El dispositivo restablece el timer a un valor del tiempo variable cada vez que el usuario navega en la card. Si tiempo no tiene el valor, el dispositivo pone el timer en 5 segundos. Cuando el timer expira, el dispositivo muestra automáticamente la segunda card en la deck.

Ejemplo:

```
<wml>
<card ontimer="#card2">
<timer name="time" value="50"/>
<p>
Hello, Unwired World!
</p>
</card>
<card id="card2">
```

```

...
...
</card>
</wml>

```

5.12. La etiqueta img

La etiqueta `` instruye al dispositivo para mostrar una imagen. se debe tener en cuenta que no todos los dispositivos pueden mostrar imágenes.

Sintaxis:

```



```

Los atributos de esta etiqueta son:

Tabla 5.11: Atributos de la etiqueta img

Etiqueta	Explicación	Valores que puede tomar
src	Lugar donde se encuentra la imagen	Href. acepta ficheros .wbmp ,
alt	Texto alternativo. Se recomienda usarlo siempre	Texto
Vspace y hspace	Espacio en blanco entre la imagen y el resto de contenidos.	Número o porcentaje
align	Alineación del texto con respecto a la imagen	top, middle o bottom
height y width	Alto y ancho de la imagen (por este orden)	Número
id	Identificador del objeto. Usado para referirse a ella	Texto

El siguiente programa muestra la figura 5.2:

```
<wml>
<card>
<p>
This is a smileyface:
<br/>
</p>
</card>
</wml>
```

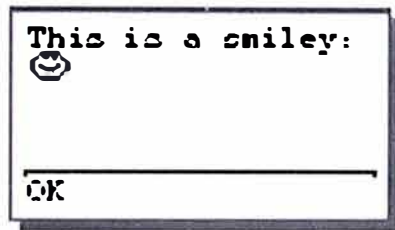


Fig. 5.2 : Acción de la etiqueta img

5.13. La etiqueta table

La etiqueta `<table>` permite especificar formatos de tablas. Las tablas WML son similares a las tablas HTML. Se usa junto con `<tr>` y `<td>`, los elementos creados son alineados en filas y columnas.

Al definir una tabla, se puede definir el número de columnas, seguido por el contenido. El contenido puede incluir filas y columnas vacías.

Sintaxis:

```
<table title="name" align="left|right|center" columns="number
of columns">
...row and data declarations
</table>
```

Los atributos de esta etiqueta son:

Tabla 5.12: Atributos de la etiqueta table.

Etiqueta	Explicación	Valores que puede tomar
title	Título de este elemento	Texto
Columns	Número de columnas que se van a representar. Se recomienda usarlo siempre. El valor 0 es un error en WML	Número
align	Alineación del texto dentro de su celda	<i>L</i> = Izquierda <i>C</i> = Centro <i>R</i> = Derecha
id	identificador del objeto. Usado para referirse a él	Texto
<tr>	Definición de salto de línea. Se cierra con </tr>	Ninguno
<td>	Definición de salto de columna. Se cierra con </td>	Ninguno

Ejemplo:

```
<table columns=2>
<tr>
<td>Pais</td><td>l/m2</td>
</tr>
<tr>
<td>Suiza</td><td>1,3</td>
</tr>
<tr>
<td>Francia</td><td>2,5</td>
</tr>
</table>
```

mostrará el siguiente resultado

Pais
l/m2
Suiza
1,3
Francia
2,5

5.14. La etiqueta input

La etiqueta `<input>` permite al usuario ingresar texto al dispositivo, el cual le asignará a una variable específica.

Sintaxis:

text

```
<input name="variable" title="label" type="type"
value="value" format="specifier"
emptyok="boolean" size="n" maxlength="n" tabindex="n"/>
```

Donde *text* representa el texto y/o imagen que el dispositivo mostrará para pedir al usuario una entrada.

Los atributos de esta etiqueta son:

Tabla 5.13: Atributos de la etiqueta table.

Etiqueta	Explicación	Valores que puede tomar
<i>name</i>	Nombre de la variable que va a tomar los datos que introduzca el usuario (Obligatorio)	Texto
<i>value</i>	Valor que aparece en el campo input por defecto.	Texto
<i>type</i>	Forma en la que se introducen los datos	<i>text</i> Texto normal

		<i>password</i> Aparecen asteriscos en vez del texto tecleado.
<i>format</i>	Formato en que se acepta la introducción de datos	El formato por defecto es <i>*M</i> . <i>A</i> = Cualquier caracter no numérico en mayúsculas <i>a</i> = Cualquier caracter no numérico en minúsculas
<i>emptyok</i>	True si no es obligatorio introducir este dato.	True o false(Por defecto si el campo tiene formato)
<i>size</i>	Especifica el ancho en caracteres del input	Número
<i>maxlength</i>	Especifica el número máximo de caracteres a introducir	Número
<i>title</i>	Título usado en la presentación del objeto	Texto
<i>tabindex</i>	Especifica el orden en que se accede a los distintos objetos dentro de una card.	Número

5.15. La etiqueta select

La etiqueta <select> especifica una lista de opciones que el usuario puede escoger. Se puede especificar una singular o múltiple elección de etiquetas <select>.

Sintaxis.

text

```
<select title="label" multiple="boolean" name="variable"
value="default" iname="index_var" ivalue="default"
tabindex="n">
```

content

```
</select>
```

Donde el *text* representa el texto y/o imagen que el dispositivo mostrará para pedir al usuario la selección, y *content* representa la lista de artículos para elegir.

Los atributos de esta etiqueta son:

Tabla 5.14: Atributos de la etiqueta select.

Etiqueta	Explicación	Valores que puede tomar
<i>name</i>	Nombre de la variable que va a tomar los datos que introduzca el usuario(Obligatorio)	Texto
<i>value</i>	Valor que tiene la variable por defecto.	Texto
<i>multiple</i>	Posibilidad de seleccionar más de una opción. Si es así el valor de la variable será ambos valores separados por “ ; “	True ó False(por defecto)
<i>ivalue</i>	Determina el index de la opción se quiere esté seleccionada por defecto. El primer item es 0, el segundo 1....	Número
<i>title</i>	Título usado en la presentación del objeto	Texto

5.16. La etiqueta option

La etiqueta <option> especifica una opción particular dentro de una etiqueta <select>. La etiqueta <option> se ha extendido para permitir una imagen como una elección.

Sintaxis.

```
<option title="label" value="value" onpick="url">
  content
</option>
```

Donde content representa el texto que el dispositivo mostrará para realizar una selección particular de un item.

Los atributos de esta etiqueta son:

Tabla 5.15: Atributos de la etiqueta option.

Etiqueta	Explicación	Valores que puede tomar
<i>value</i>	Valor que toma la variable al seleccionar este item	Texto
<i>title</i>	Título usado en la presentación del objeto	Texto
Onpick	Evento ocurrido al seleccionar o deseleccionar un item	Href. URL o dirección de otra card

Ejemplo:

```
<select name="mascota"
value="N" ivalue=1>
<option
value="N">Ninguno</option>
<option
value="P">Perro<</option>
<option
value="G">Gato<</option>
</select>
```

Explicación:

Se tiene 3 opciones, y por defecto ninguna seleccionada (value=1). Sólo se puede elegir una de ellas. Si se elige Perro, la variable mascota es P, gato es G y si no se elige nada N (value="N").

5.18. Ejemplo de aplicación

Un ejemplo de aplicación puede ser un banco (por ejemplo BANCO RIO) en el cual se pueda conocer en el momento que se desee, información de cuentas y servicios.

Para acceder a este servicio sólo debe contar con un teléfono celular WAP (un teléfono celular con microbrowser) y desde el portal (menú) WAP desde dispositivo móvil, en el cual se selecciona la opción Bancos, luego Banco RIO y allí podrá seleccionar las consultas que se desee.



Fig. 5.3: Acceso a un banco

Cada vez que se acceda al servicio, el teléfono celular se comunica con Banco RIO, obteniendo la respuesta a la consulta efectuada.

Todos los resultados se presentan en forma de texto en la pantalla del celular.

Se puede consultar en un teléfono WAP, el saldo de cualquier cuenta: cuenta única, cuenta corriente y caja de ahorros.

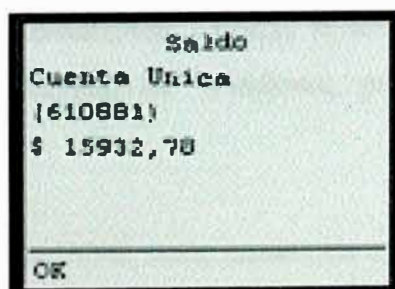


Fig. 5.4: Vista de saldo de una cuenta.

Se puede disponer en el teléfono WAP, la comunicación inmediata con un servicio de banca telefónica, por ejemplo RIOLine, donde un operador puede ayudarle si tiene consultas adicionales y en operaciones bancarias urgentes

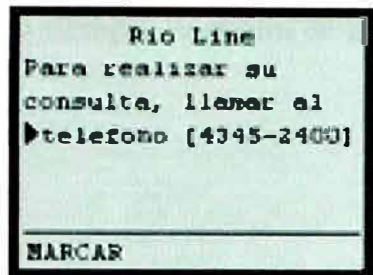


Fig. 5.5 Consultas a un banco

Resumen

El lenguaje WML (Wireless Markup Lenguaje) basado en XML (eXtensible Markup Lenguaje) está desarrollado para la visualización de contenidos desde un terminal móvil (micronavegador) con las restricciones que puede ofrecer como: pantallas chicas, teclados limitados, memorias pequeñas.

WML ofrece soporte para texto e imagen con una variedad de formatos y comandos similares al HTTP.

Las páginas visualizadas se llaman cards que son agrupadas en decks, una card se identifica por una URL.

Las páginas WML pueden ser ubicadas desde cualquier servidor WEB convencional, siempre que haya sido configurado para servidor.

Para poder desarrollar páginas WAP se necesita de un editor de texto convencional como el Notepad de Windows, para visualizar el contenido se necesita de un emulador.

Toda páginas WML debe comenzar con la siguiente cabecera:

```
<?xml version="1.0"?>  
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"  
"http://www.wapforum.org/DTD/wml_1.1.xml">
```

El contenido siempre va dentro de las etiquetas <wml> y </wml>.

CONCLUSIONES

1. Las tecnologías para dispositivos de mano son una de varias tecnologías clave para la Internet móvil. De las tecnologías que compiten, WAP aparece como la solución dominante para sistemas de tercera generación. Al pasar el tiempo, WAP mantendrá sus principios y evolucionará para adoptar las características de sistemas inalámbricos de futuras generaciones.
2. Ciertas características claves que han sido observadas en nuevos modelos emergentes son las siguientes:
 - Varias nuevas clases de servicio acompañarán al crecimiento de la Internet móvil.
 - La Internet móvil mejora mucho el potencial del comercio electrónico móvil.
3. A medida que toman forma nuevos modelos comerciales, se están encontrando los operadores de redes en una posición clave para servir de centros de pago y de transacción en la Internet móvil, de portales de la Internet móvil, y de sitios para comercio electrónico basado en la Internet móvil.
4. La instalación de soluciones WAP de punto a punto debe ser bienvenida y dar lugar a una penetración en el mercado rápida y a gran escala.
5. La plataforma WAP se encarga de ofrecer un acceso inalámbrico a un conjunto de servicios de Internet por otras redes; la navegación de un usuario móvil es similar a la de un navegador WEB, el cual, utilizará el lenguaje WML con las restricciones

del dispositivo móvil como: pantallas pequeñas, teclados limitados, pequeñas memorias, etc.

6. La arquitectura WAP, está definida para proporcionar un entorno escalable y extensible para el desarrollo de aplicaciones para dispositivos de comunicación móvil. Para ello, se define una estructura de capas, en la cual cada capa es accesible por la capa superior.
7. El desarrollo de aplicaciones WAP necesita del lenguaje WML, el cual, ofrece soporte para texto e imágenes.
8. La importancia del protocolo WAP es que puede brindar un sin número de servicios de tal importancia que pueden ser accesados desde cualquier punto. Los servicios brindados son por ejemplo: acceso a datos de todo tipo de información, servicios de banca, comercio electrónico, e-mails, noticias, etc.
9. WAP es por eso, más que otras tecnologías inalámbricas; es el catalizador de la Internet móvil. WAP va a llegar a ser un artículo práctico entre los usuarios finales móviles, cambiando de modo significativo la manera de enfocar y realizar numerosas tareas diarias.

ANEXO A: CONFIGURACION DE UN SERVIDOR NT PARA QUE ACEPTE CONEXIONES DESDE MOVILES

1. Instalar el Remote Access Service

Desde el panel de control las opciones de Network , se pulsa en la pestaña de servicios y se escoge añadir. Bajar en la lista y seleccionar Remote Access Service. pulsar OK y el servicio será instalado.

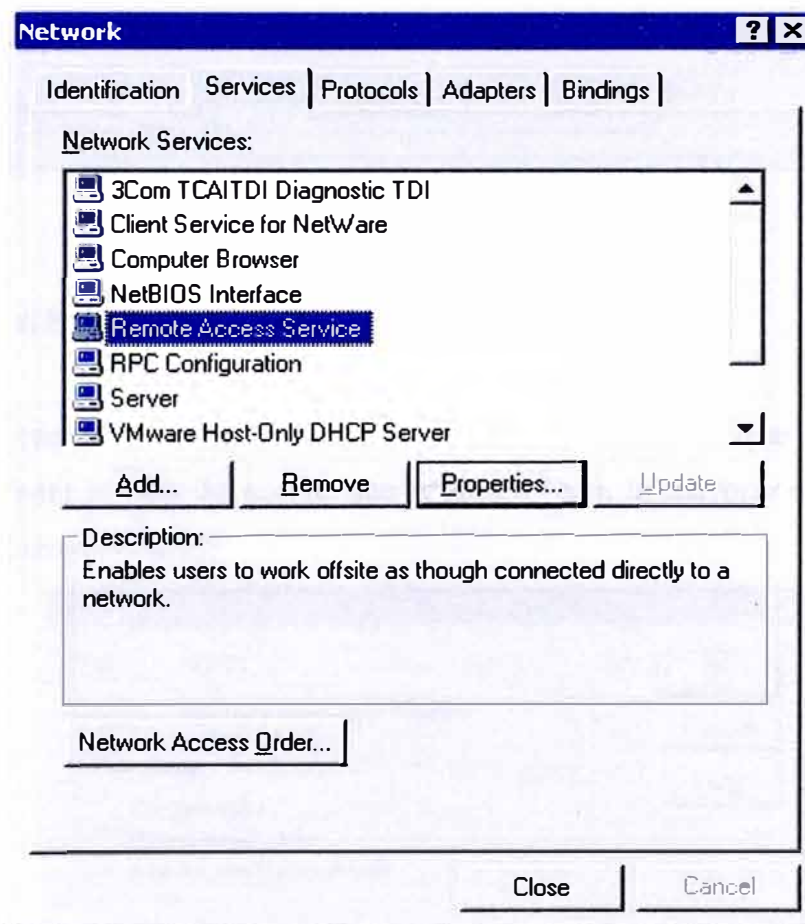


Fig. A1: Instalar el Remote Access Service.

2. Configurar el Modem

Si no se tiene instalado un modem en un servidor, el programa de configuración de remote access pedirá que se instale uno. Si no se dispone de los drivers para su

instalación, simplemente se instala como un standard 28,800 modem (actualmente las conexiones GSM sólo soportan velocidades de 9,6kbps)

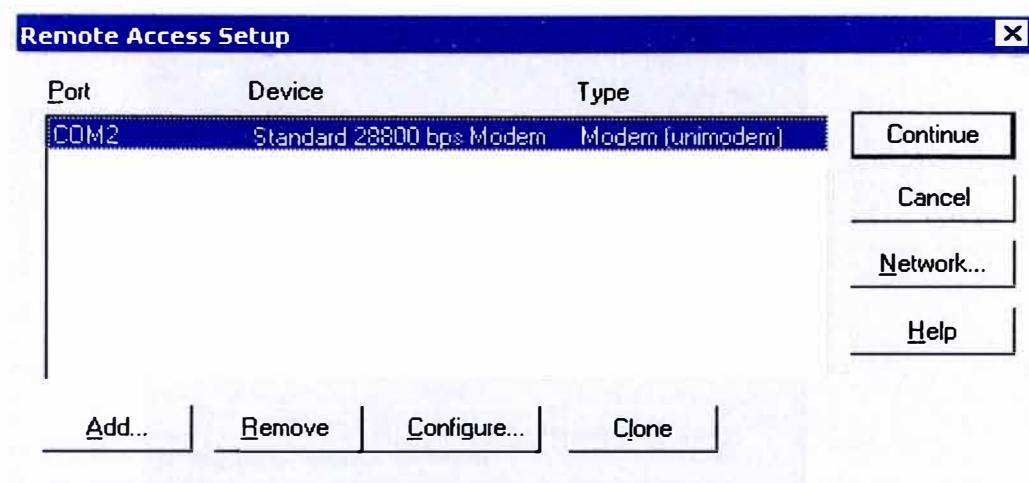


Fig. A2: Configurar el MODEM

3. Configurar el Puerto

Pulsar en Configure en la parte inferior y asegurarse de que el modem esté configurado para el tipo de acceso que se desea (para la mayoría de los propósitos valdrá: Receive calls only)

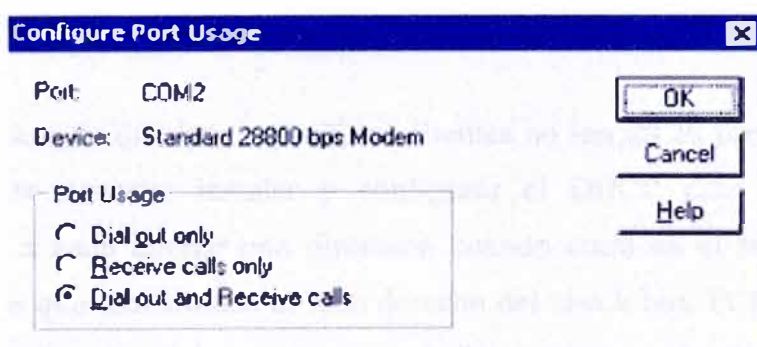


Fig. A3: Configurar el Puerto

4. Configurar los protocolos de red

Elegir OK en el anterior cuadro de diálogo. Esto llevará al cuadro del paso 2 hacer clic en network. Se debe asegurar que el protocolo TCP/IP esté seleccionado en Server Settings. También asegurarse de que el tipo de autenticación seleccionado es

el correcto (usar la opción de clear text option, a no ser que requiera autenticación encriptada de Microsoft)

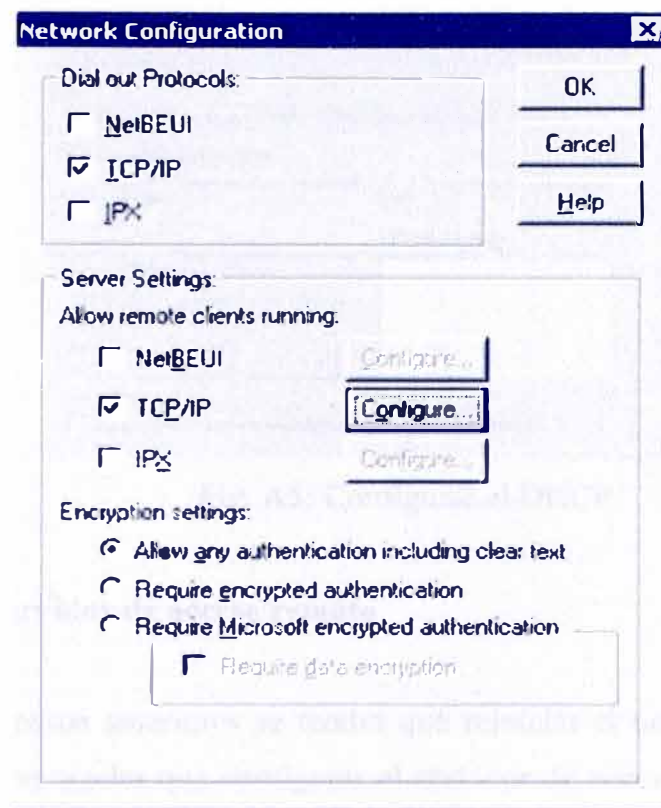


Fig. A4: Configurar los protocolos de red

5. Configurar el DHCP

Situarse en el caso de que los dispositivos clientes no tengan su propia dirección IP. Por lo tanto, se necesita instalar y configurar el DHCP para que se asigne dinámicamente a cada cliente una dirección cuando entre en el sistema. Pulsar el botón Configure que está situado al lado derecho del check box TCP/IP para acceder al siguiente cuadro de dialogo. Si no necesita un rango de IP fijas para cada dispositivo se puede aceptar los valores por defecto que aparecen en esta pantalla (después se debe poner a que máquinas se quiere dejar acceder). Si desea usar un rango fijo de direcciones IP se debe seleccionar Use static address pool y definir su rango de direcciones.

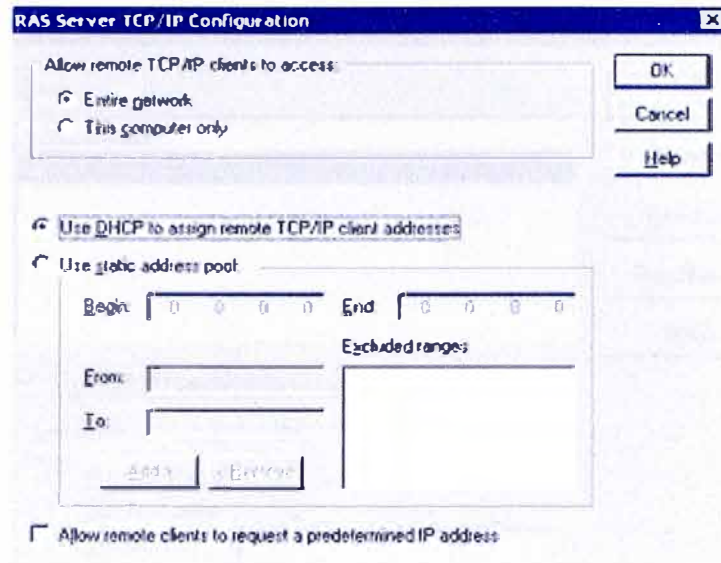


Fig. A5: Configurar el DHCP

6. Configurar el servidor de acceso remoto

Después de los pasos anteriores se tendrá que reiniciar el ordenador. Una vez que haya reiniciado, se tendrá que configurar el servidor de acceso remoto. Pulsar Start luego Programs - Administrative Tools-Remote Access Admin. Si se quiere ejecutar siempre en el arranque se puede automatizar, dirigiéndose a: Control Panel-Services-Remote Access Server-Startip, verificar automatic.

Una vez que esté funcionando se necesita añadir los usuarios con permisos de llamada al sistema. Si no dispone de los usuarios a los que quiere dar acceso al sistema, se podrá modificar posteriormente a través del NT user manager.

Una vez que los haya añadido, se deberá verificar en cada usuario la casilla de Grant dial-in permission para darles permisos de llamada.

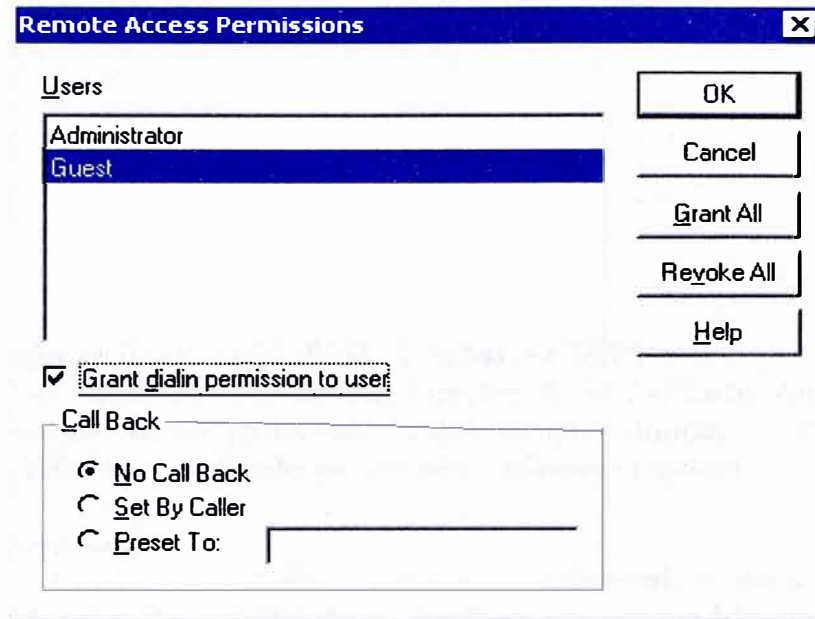


Fig. A6: Configurar el servidor de acceso remoto

BIBLIOGRAFIA

1. **DESARROLLO WAP CON WML Y WML SCRIPT**
Forta, B. / Lauver, K. D. / Fonte Paul /J uncker, R. M. / O'Leary Amy / Mandel, Ronan / Bromby Dylan **Editorial:** Anaya, Grupo Editorial **Páginas:** 576
Año: 2001 **Edición:** 1 **Editado en:** España **Idioma:** Español
2. **WAP. Guía esencial**
Hougland, Damon / Zafar, Khurram **Editorial:** Pearson educación
Páginas: 308 **Año:** 2002 **Edición:** 1 **Editado en:** España **Idioma:** Español
3. **WAP. Guía esencial**
Autor: Hougland, Damon / Zafar, Khurram
4. **WIRELESS WEB DEVELOPMENT WITH PHP AND WAP**
Autor: Ray Rischpater, Boulder Creek, CA, USA
5. **WAP. Guía práctica**
Autor: Dornan, Andy
6. **WAP PROFESIONAL**
Autor: Arehart, Charles
7. **INSIDE WAP. Programming applications wit WML and WMLScript**
Autor: Niskanen, Peca
8. **WAP SERVIETS**
Autor: Cook, John L.
9. **DESARROLLO WAP CON WML Y WML SCRIPT**
Autor: Forta, B. / Lauver, K. D. / Fonte Paul /J uncker, R. M. / O'Leary Amy / Mandel, Ronan / Bromby Dylan
10. **WAP PROFESIONAL**
Autor: Arehart, Charles **Editorial:** Inforbook's **Editado en:** España
Idioma: Español **Año:** 2001 **Edición:** 1