

0/H. 380  
CHe

46760

**CONSEJO FEDERAL DE INVERSIONES**

**PROVINCIA DE MENDOZA**

***Estudio Factibilidad***

***Implementación Portal WAP***



**INFORME FINAL**

**Marzo 2008**

**ING. CATAPANO GILI, ANÍBAL DANIEL**



## AUTORIDADES

<b>PROVINCIA DE MENDOZA</b>	<b>CONSEJO FEDERAL DE INVERSIONES</b>
<p><b>Gobernador de la Provincia</b> Cont. Celso Alejandro JAQUE</p> <p><b>Subsecretaría de Gestión Pública</b> Lic. Diego Luis BOSSIO</p>	<p><b>Secretario General</b> Ing. Juan José CIÁCERA</p> <p><b>Directora de Coordinación</b> Ing. Marta Velazquez CAO</p> <p><b>Jefe de Área Red de Información</b> Lic. Edmundo Szterenlicht</p>





---

**Autor:**

Catapano Gili, Anibal Daniel.

**Colaboradores:**

Ing. Nora Alejandra Costa



## ÍNDICE GENERAL

<b>AUTORIDADES.....</b>	<b>2</b>
<b>ÍNDICE GENERAL .....</b>	<b>4</b>
<b>RESUMEN .....</b>	<b>8</b>
<b>DESARROLLO DEL ESTUDIO .....</b>	<b>10</b>
<b>INTRODUCCIÓN .....</b>	<b>12</b>
<b>ANTECEDENTES WAP .....</b>	<b>14</b>
<b>¿POR QUÉ USAR INTERNET?.....</b>	<b>18</b>
<b>ANÁLISIS DE LOS MODELOS DE APLICACIÓN.....</b>	<b>20</b>
MODELO DE INTERNET.....	20
<i>Tipos de Contenidos Existentes.....</i>	<i>22</i>
EL MODELO WAP .....	23
<b>COMPONENTES WAP.....</b>	<b>25</b>
PRINCIPALES CARACTERÍSTICAS.....	25
<b>ELEMENTOS WAP.....</b>	<b>26</b>
CLIENTE.....	27
PASARELA.....	28
SERVIDOR DE ORIGEN .....	29
SERVIDOR WTA .....	30
<b>ARQUITECTURA WAP .....</b>	<b>31</b>
CAPAS DEL PROTOCOLO .....	31
<i>Entorno Inalámbrico de Aplicación (WAE) .....</i>	<i>32</i>
<i>Modelo de Direccionamiento.....</i>	<i>36</i>
<i>Wireless Markup Language (WML) .....</i>	<i>36</i>
<i>WMLScript .....</i>	<i>37</i>
<i>Aplicaciones Telefónicas Inalámbricas (WTA).....</i>	<i>38</i>
PROTOCOLOS WAP.....	43



<i>Modo Sin Conexión</i> .....	43
<i>Modo Sin Conexión con Seguridad</i> .....	43
<i>Modo Conexión</i> .....	44
<i>Modo Conexión con Seguridad</i> .....	44
<b>CAUSAS DE WAP</b> .....	<b>44</b>
¿POR QUÉ UN NUEVO ESTÁNDAR? .....	44
ADAPTACIÓN A LOS LÍMITES DE LA RED INALÁMBRICA .....	45
<i>Bajo Ancho de Banda:</i> .....	45
<i>Alta Latencia:</i> .....	46
<i>Estabilidad sin Conexión / Disponibilidad Impredecible del Portador:</i> .....	48
ADAPTACIÓN A LOS LÍMITES DEL DISPOSITIVO INALÁMBRICO .....	49
<i>Pantalla Limitada:</i> .....	50
<i>Facilidades de Entrada Limitada:</i> .....	51
<i>Memoria y CPU Limitada:</i> .....	52
<i>Limitada Carga de Baterías:</i> .....	53
<b>DOCUMENTOS WML</b> .....	<b>53</b>
CARACTERÍSTICAS PRINCIPALES DEL LENGUAJE .....	54
SINTAXIS WML .....	55
<i>Entidades</i> .....	55
<i>Etiquetas</i> .....	56
<i>Elementos</i> .....	57
<i>Atributos</i> .....	57
<i>Comentarios</i> .....	58
<i>Variables</i> .....	58
<i>Sensibilidad de formato de letra</i> .....	60
<i>Sección Cdata</i> .....	60
ELEMENTOS DE WML.....	61
<i>Cartas y Barajas</i> .....	61



ATRIBUTOS COMUNES.....	64
<i>Cabecera del documento</i> .....	65
<i>Elemento wml</i> .....	67
<i>Elemento card</i> .....	67
<i>Elemento template</i> .....	69
<i>El elemento head</i> .....	70
<i>Elemento access</i> .....	70
<b>TÉCNICAS DE PROCESAMIENTO EN SERVIDOR WWW PARA GENERACIÓN DE CONTENIDOS WAP .....</b>	<b>71</b>
SERVER SIDE INCLUDES (SSI).....	73
LA INTERFAZ CGI (COMMON GATEWAY INTERFACE) .....	77
INTERFACES DE PROGRAMAS DE APLICACIÓN (APIS) PROPIETARIAS .....	78
INTERFAZ CGI ASÍNCRONA (FASTCGI).....	79
SERVLETS .....	80
<b>DESARROLLO DE APLICACIONES WAP .....</b>	<b>82</b>
CONSIDERACIONES GENERALES .....	82
ESTABLECIMIENTO WAP .....	83
<i>Navegadores WAP (WAP Browsers)</i> .....	84
<i>WAP Gateways</i> .....	86
<i>Servidor de Aplicaciones WAP</i> .....	88
<i>Recursos WAP</i> .....	95
<i>Herramientas para Desarrollo WAP</i> .....	97
<i>Nokia Mobile Internet Toolkit 4.0</i> .....	98
<b>SERVICIO DE MENSAJES CORTOS – SMS.....</b>	<b>105</b>
¿QUÉ ES SMS (SERVIVIO DE MENSAJES CORTOS)? .....	105
ADQUISICIÓN POR EL USUARIO.....	106
IMPORTANCIA DEL SERVICIO .....	107
FORMA DE FUNCIONAMIENTO .....	107



---

CRECIMIENTO DE LA UTILIZACIÓN DE SMS.....	108
CAMINOS DEL SMS .....	110
¿TAMAÑO DE UN SMS? .....	111
<b>PRUEBAS PILOTOS.....</b>	<b>112</b>
CONTENIDO ESTÁTICO.....	112
CONTENIDO DINÁMICO.....	113
CONTENIDO TRANSACCIONAL .....	114
PRUEBAS REALIZADAS .....	115
CONSULTA DEL PADRÓN ELECTORAL.....	115
<i>Estructura de la Página Generada de Acuerdo al Navegador.....</i>	<i>116</i>
VISUALIZACIÓN EN NAVEGADORES WEB Y NAVEGADOR WAP .....	118
<i>Internet Explorer.....</i>	<i>118</i>
<i>Navegador Celular Web.....</i>	<i>119</i>
<i>Navegador Celular WAP.....</i>	<i>121</i>
SECUENCIA DE NAVEGACIÓN WAP .....	122
<b>CONCLUSIONES .....</b>	<b>128</b>



---

## RESUMEN

Las personas que se desplazan necesitan servicios, informaciones y entretenimiento que puedan acompañar su ritmo. Con el acceso a servicios móviles, las decisiones e interacciones se dan en el mismo instante.

El valor de los servicios móviles para los usuarios finales es impulsado por tres elementos distintos: personalización, rapidez y conocimiento de la ubicación. La combinación eficaz de estos tres elementos añade más valor todavía.

El protocolo de aplicación inalámbrica (WAP) es un protocolo que ha conseguido establecer con Éxito, un estándar común para la forma de utilizar la tecnología inalámbrica para acceder a Internet. La tecnología WAP ha sido optimizada para el envío de informaciones a dispositivos de clientes finos, como los Teléfonos celulares.

Los servicios móviles impulsados por las diferentes compañías han recibido una amplia aceptación de los usuarios. En 2010 se calcula que el número de usuarios de WAP en Europa Occidental crecerá hasta superar los 300 millones. En parte, este crecimiento es impulsado por la introducción del Servicio General de Radio por Paquetes (GPRS), WAP 2.0, Bluetooth, el comercio móvil y la tecnología disponible 3G.



---

La reducción de los costos de acceso a internet y la utilización de mensajes de texto a través de dispositivos inalámbricos, incrementa considerablemente el número de usuarios que acceden a los contenidos.

Las pruebas pilotos realizadas, nos demuestran que los usuarios se encuentran dispuestos a la utilización de estos tipos de servicios, mientras los mismos le permitan acceder rápida y fácilmente a la información solicitada.

Además, para lograr una fuerte utilización de los servicios, los mismos deben ser de gran interés para los usuarios, aunque los mismos sean ofrecidos por un período corto de tiempo, tal es el caso de la consulta de la mesa de votación en el padrón electoral, en tiempos de elecciones o la consulta del estado de solicitud de un DNI.



---

## DESARROLLO DEL ESTUDIO

1. Análisis general de la necesidad de una nueva forma de acceder a los contenidos de Internet, por medio de dispositivos inalámbricos.
2. Descripción y Análisis de los modelos de Aplicación tanto Web como Wap
3. Identificación y Descripción de los componentes que integran la tecnología WAP y Análisis de sus principales características
4. Identificación y Descripción de los Elementos que intervienen en la tecnología de comunicación
5. Análisis y Descripción de la Arquitectura del protocolo WAP
6. Análisis de la necesidad de un nuevo estándar para acceder a Internet, que sea adaptado a los límites tanto de los dispositivos como de la red que interviene.
7. Análisis y Descripción del lenguaje utilizado en los documentos generados en el entorno WAP, principales características, sintaxis, elementos existentes, etc.
8. Análisis de técnicas de procesamiento utilizadas en servidores WEB, para transformar los contenidos existentes en contenidos aptos para WAP.





- 
9. Identificación, Análisis y Descripción de los principales aspectos a tener en cuenta, cuando se desarrollan aplicaciones destinadas a dispositivos inalámbricos.
  
  10. Análisis y Descripción del Servicio de Mensajes Cortos SMS.
  
  11. Análisis de la Prueba Piloto realizada y descripción de los procedimientos de generación de los contenidos obtenidos en la prueba.



---

## INTRODUCCIÓN

Hasta hace muy poco, la PC era la única forma de acceder a Internet, con la aparición de tecnologías para aplicaciones inalámbricas, la computadora deja de ser el único medio para acceder a la información contenida dentro de Internet y cada vez se agregan gradualmente, más dispositivos móviles como teléfonos móviles, asistentes personales (PDA), etc.

Varias iniciativas independientes de los fabricantes, buscaron el desarrollo de un ambiente que propiciase el acceso al contenido disponible en Internet, a partir de los terminales móviles.

Algunos sistemas propietarios llegaron a ser ofrecidos, sin embargo no estaban completamente alineados con los padrones y las tecnologías ya establecidas para Internet.

Esta situación llevó a las empresas Ericsson, Motorola, Nokia y Unwired Planet (hoy Phone.com/Openwave) a fundar, en el verano de 1997, el WAP Fórum con el objetivo inicial y definir, un conjunto de especificaciones para el desarrollo de aplicaciones para redes celulares.

Las especificaciones desarrolladas para la versión WAP 1.0, no tuvieron resultados satisfactorios, y la mayoría de los fabricantes esperó la versión WAP 1.1 para equipar a los terminales celulares con esa tecnología, siendo que los primeros terminales llegaron al mercado a mediados del 1999.



---

La versión WAP 1.2.1 fue finalizada en Junio de 2000, y ya introducía nuevas funcionalidades, tales como la tecnología PUSH.

La versión actual, WAP 2.0, fue finalizada y puesta a disposición en enero de 2002.

Actualmente WAP Fórum es parte de Open Mobile Alliance (OMA), que cuenta con más de 300 asociados entre operadores, suministradores de terminales y redes celulares, empresas de servicios de tecnología de información y proveedores de servicios y contenidos.

La tecnología WAP, creó una gran expectativa en el mercado por la promesa que hizo inicialmente de proveer medios para navegar por Internet a partir del terminal móvil, y se creó la tan soñada Internet móvil.

Sin embargo, esa expectativa no fue plenamente atendida, una vez que las limitaciones inherentes al terminal móvil, formato e interfaces, y a las redes celulares de la época, tornaron la navegación muy lenta y limitada.

Muchos sitios en Internet ya se encuentran habilitados para enviar información hacia aparatos inalámbricos, sin embargo, la principal carencia de esta Tecnología (WAP-WML) reside en la Interface Gráfica capaz de observarse en los dispositivos inalámbricos (Teléfonos Celulares, PDA's, etc).

Ante este problema han surgido nuevos mecanismos para enviar contenido hacia medios inalámbricos.



---

Entre estos encontramos al J2ME ("Java 2 Micro Edition"), el cual, viene a formar el último "Suite" desarrollado por Sun Microsystems, al lado del J2SE ("Java 2 Standard Edition") y J2EE ("Java 2 Enterprise Edition") el cual esta enfocado hacia aplicaciones inalámbricas.

Otro mecanismo es el BREW ("Binary Runtime Environment for Wireless"), el cual, es una creación de la empresa Qualcomm que intenta ofrecer la misma solución al mercado inalámbrico, el generar aplicaciones dinámicas **altamente gráficas en el Cliente (Teléfono Celular)**, esto a diferencia de WAP-WML donde un Servidor de Páginas genera un contenido muy restringido para el Cliente (Teléfono Celular).

## **ANTECEDENTES WAP**

WAP puede ser descrito a grandes rasgos, como un conjunto de protocolos que ha heredado las características y funcionalidades, tanto de los estándares de Internet como de los estándares para servicios inalámbricos, desarrollado por algunas de las compañías líderes del mundo en el negocio de las comunicaciones inalámbricas.

En 1995, Ericsson inició un proyecto cuyo propósito fue desarrollar un protocolo general, o más bien un concepto, para servicios de valor agregado en las redes móviles.



---

El protocolo fue llamado Intelligent Terminal Transfer Protocol (ITTP), y maneja la comunicación entre un nodo de servicio, donde la aplicación es implementada, y un teléfono móvil inteligente. La ambición fue hacer ITTP un estándar para los servicios de valor agregado en las redes móviles.

Durante 1996 y 1997 Unwired Planet, Nokia y otros, dieron conceptos adicionales en el área de los servicios de valor agregado en redes móviles.

Unwired Planet presentó HDML<sup>1</sup> y HDTP<sup>2</sup>. Tal como HTML<sup>3</sup> se utiliza en WWW, HDML es usado para describir contenidos e interfaces usuarias, pero optimizado para acceso a Internet inalámbrico, desde dispositivos de mano con pequeña pantalla y capacidad de entrada limitada.

Del mismo modo, HDTP, puede ser considerado como un equivalente inalámbrico del estándar de Internet HTTP<sup>4</sup>, por ejemplo: un protocolo de bajo peso para realizar transacciones entre cliente/servidor.

---

<sup>1</sup> Handheld Device Markup Language

<sup>2</sup> Handheld Device Transport Protocol

<sup>3</sup> HTML (Hyper Text Markup Language) es el lenguaje estándar de marcas en el texto usado para construir páginas *Web. Está basado en el lenguaje estándar de marcas de texto de IBM. En HTML, el texto se marca usando etiquetas delimitadas por símbolos. En* estos momentos el estándar es HTML 3.2

<sup>4</sup> HTTP (HyperText Transfer Protocol) es el protocolo básico que especifica las interacciones entre los browsers de Web y los servidores Web (más conocidos como servidores Web). Cuando se solicita una página Web realmente se utiliza un comando "Get" a un servidor HTTP para un ítem específico. Diseñados para servir al mayor número de



---

En marzo de 1997 Nokia presentó oficialmente el concepto de mensajes inteligentes, una tecnología de servicio de acceso a Internet especialmente diseñada para dispositivos de mano GSM<sup>5</sup>.

La comunicación entre un usuario móvil y la información de Internet, contenida en un servidor, utiliza SMS<sup>6</sup> y un lenguaje de marcas llamado TTML (Tagged Text Markup Language).

Tal como HDML, este lenguaje es adaptado para la comunicación inalámbrica, por ejemplo las conexiones de banda angosta.

Con una multitud de conceptos, había un riesgo sustancial de que el mercado podía ser fragmentado, un desarrollo del que ninguna de las compañías involucradas se beneficiaría. Por lo tanto, las compañías acordaron en llevar a delante una solución en común.

---

usuarios posibles al mismo tiempo. HTTP es una norma controlada por le Internet Engineering Task, o IETF.

<sup>5</sup> GSM o Global System Mobile communications. Sistema Global de comunicaciones Móviles. Sistema digital de telecomunicaciones principalmente usado para telefonía móvil. Existe compatibilidad entre redes por tanto un teléfono GSM puede funcionar teóricamente en todo el mundo. En EEUU se encuentra situado en la banda de los 1900 Mhz y es llamado DCS-1900

<sup>6</sup> SMS o Short Message Service. Servicio de Mensajes Cortos. Servicio de mensajería electrónica de texto entre teléfonos GSM. Permite enviar a un celular un mensaje de hasta 160 caracteres. Este servicio fue habilitado inicialmente en Europa.



---

El 26 de Junio de 1997 Ericsson, Motorola, Nokia y Unwired Planet tomaron la iniciativa de comenzar la creación rápida de un estándar, para hacer una realidad los servicios avanzados dentro del dominio inalámbrico.

En diciembre de 1997 WAP Forum fue creado formalmente, y después del lanzamiento de las especificaciones WAP 1.0 en abril de 1998, la asociación al WAP Forum fue abierta a todos.

En febrero de 1999 más de 90 compañías eran miembros del WAP Forum. Entre estas compañías se encuentran muchos de los fabricantes de terminales e infraestructura que lideran el mundo, compañías de software, operadores y proveedores de servicio.

Los fabricantes de dispositivos de mano dentro del WAP Forum, representan más del 90% del mercado del mundo a través de todas las tecnologías, y los operadores de la red representan cerca de 100 millones de suscriptores.

Los principales objetivos del WAP Forum son:

- Generación de un Estándar independiente de la red inalámbrica.
- Abierto a todos.
- Pensado como cuerpos de estándares apropiados.
- Aplicaciones de escalas a través de opciones de transporte (GSM, IS-95, IS-136, PDC, etc.)



- 
- Aplicaciones de escalas a través de tipos de dispositivos (teléfonos móviles, PDAs, etc.)
  - Extensible a través del tiempo, de nuevas redes y transportes (e. g. 3G systems)

## ¿POR QUÉ USAR INTERNET?

Durante los últimos años nos hemos acostumbrado a una amplia variedad de servicios ofrecidos por Internet y por WWW. No solamente los servicios en sí mismos nos atraen, sino también el camino conveniente para acceder a ellos a través de un navegador de Internet.

Nosotros podemos acceder a los mismos servicios en todo el mundo, siempre que tengamos acceso a una computadora y a Internet. Nadie puede negar que Internet haya obtenido un gran éxito, cerca de 200 millones de usuarios.

Los proveedores de servicio también se benefician por el paradigma WWW, a partir de que sus servicios, pueden ser independientemente desplegados de la localización de los usuarios.

Los servicios son creados y almacenados en un servidor, esto significa que se vuelven muy fáciles de cambiar de acuerdo a las necesidades de los usuarios.





---

Usando las herramientas que están fuera del dispositivo, los servicios son creados con un mínimo de esfuerzo, el cual, combinado con una forma rápida y conveniente de lanzarlos, permiten un mercado de extremadamente corto tiempo.

El tiempo reducido de desarrollo del servicio, por supuesto, también implica costos reducidos, comparados con el desarrollo de servicios convencionales en las redes inalámbricas.

Como los usuarios se vuelven más y más dependientes de los servicios ofrecidos en Internet, un defecto se vuelve evidentemente creciente - la necesidad de un cable para conectarse a Internet.

Este defecto hace que sea recordado especialmente por los millones de usuarios que consumen una cantidad sustancial de su tiempo en movimiento.

En los últimos años se han hecho intentos para hacer que este defecto desaparezca, pero no han hecho que Internet cruce el abismo como una plataforma de servicio inalámbrico, solamente los adaptadores primarios han aceptado las tecnologías provistas hasta ahora.

Una de las principales razones para esto, es la falta de un estándar ampliamente aceptado, un problema que ha sido atribuido al WAP Forum.

Un amplio respaldo a WAP, probablemente habilitará a Internet como un medio para proveer servicios a dispositivos inalámbricos dentro de un futuro inmediato. Esto incluye tanto servicios del WWW, como servicios telefónicos.



---

## **ANÁLISIS DE LOS MODELOS DE APLICACIÓN**

Para lograr entender los distintos elementos que intervienen tanto en el modelo utilizado por WWW y el modelo utilizado por WAP, en el primer informe parcial, se realizó una comparación entre el Modelo WWW y el Modelo WAP, donde se analizaron los diferentes elementos que intervienen dentro del modelo WAP destinados a la comunicación entre el teléfono y la red.

En este informe, se hará mención a la estructura utilizada en ambos modelos y se analizarán y desarrollarán los elementos existentes en el modelo WAP.

### ***MODELO DE INTERNET***

El modelo de Internet hace posible, para un cliente, obtener los servicios en un gran número de servidores de origen; cada uno dirigido por una única URL<sup>7</sup>.

El contenido almacenado en los servidores es de varios formatos, pero HTML es el predominante. HTML provee el desarrollo de contenidos, con un medio para describir la apariencia de un servicio en una estructura de

---

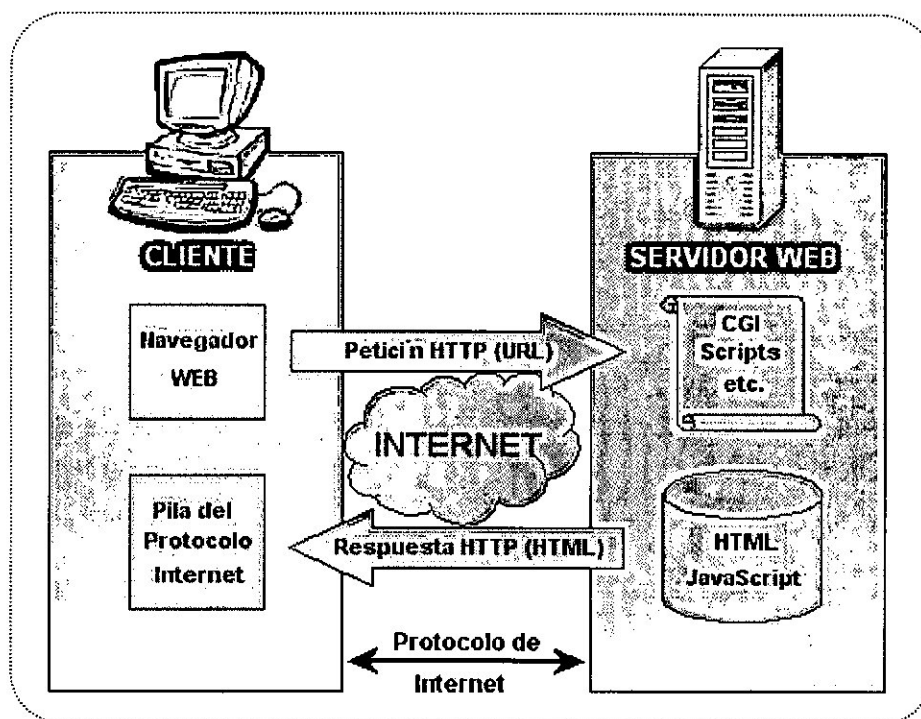
<sup>7</sup> El Universal Resource Locator identifica simplemente un archivo disponible en Internet. Una URL tal como <http://www.sei-europe.com/homepage.html> posee tres partes. La primera es el protocolo (http o ftp, dependiendo del servicio solicitado), seguido por dos barras y dos puntos (//:). El segundo componente puede ser el nombre del dominio ([www.sei-europe.com](http://www.sei-europe.com)) o la dirección IP del host, seguida de la barra inclinada (/). El tercer componente es el directorio y la ubicación de la página HTML deseada (homepage.html)



documento chato; por ejemplo el contenido entero de una página es mostrado simultáneamente.

Si se necesitan formas más avanzadas, como lógica de procesos, pueden utilizarse lenguajes de scripting como JavaScript o VBScript.

La figura muestra como un cliente de WWW requiere un recurso almacenado en un servidor web.



*Modelo Internet*

Como se mencionó anteriormente, un recurso en Internet es identificado por una única URL, la cual consiste en una secuencia de texto que constituye una dirección hacia ese recurso. En este caso el recurso es un documento HTML.



---

En Internet, se utilizan los protocolos de comunicaciones estándar, como HTTP y TCP/IP<sup>8</sup>.

### **Tipos de Contenidos Existentes**

El contenido dentro de un servidor puede ser estático o dinámico.

El contenido estático se produce una vez y no cambia, o se actualiza muy a menudo, por ejemplo la presentación de una compañía.

El contenido dinámico es necesario cuando la información suministrada por el servicio cambia a menudo, por ejemplo horarios, noticias, cantidades de almacenamiento e información de mercado.

Este se hace posible gracias a tecnologías tales como ASP<sup>9</sup>, CGI<sup>10</sup> y Servlets que permite generar dinámicamente el contenido de los documentos almacenados.

---

<sup>8</sup> TCP/IP Transfer Control Protocol / Internet Protocol.

<sup>9</sup> ASP o Active Server Page. Es el lado del server es un documento que contiene un script incorporado, el cual permite la generación de páginas dinámicas.

<sup>10</sup> CGI o Common Gateway Interface. Mecanismo estándar por la extensión de la funcionalidad del server web a través de ejecuciones de programas o script sobre el mismo server web en respuesta a las interrogaciones de los browsers web. El CGI viene comúnmente utilizado para la elaboración de módulos: los browsers envían los datos del módulo hacia un script CGI sobre el server que lo integra al database, restituyendo una página html con los resultados. La utilización de CGI da mayor dinamicidad e interactividad a las páginas.



---

## ***EL MODELO WAP***

WAP hace uso también del paradigma de Internet para proveer una plataforma flexible de servicio.

Para ajustar el acceso inalámbrico al espacio de información ofrecida por WWW, WAP está basado en la conocida tecnología de Internet, que ha sido mejorado para superar las limitaciones del medio inalámbrico.

Los servicios creados usando HTML, no se ajustarían muy bien en pequeños dispositivos manuales, ya que ellos son destinados para el uso en computadoras de escritorio con grandes pantallas.

Los dispositivos con un bajo ancho de banda inalámbrico, no servirían para enviar información más o menos extensa, en la que consisten a menudo las páginas de HTML.

No obstante un lenguaje de marcas adaptado a estas restricciones ha sido desarrollado y se ha denominado WML<sup>11</sup>.

WML ofrece un modelo de navegación diseñado para dispositivos con pantalla pequeña y facilidades limitadas de entrada (no poseen mouse y tienen un limitado teclado).

---

<sup>11</sup> Wireless Markup Language

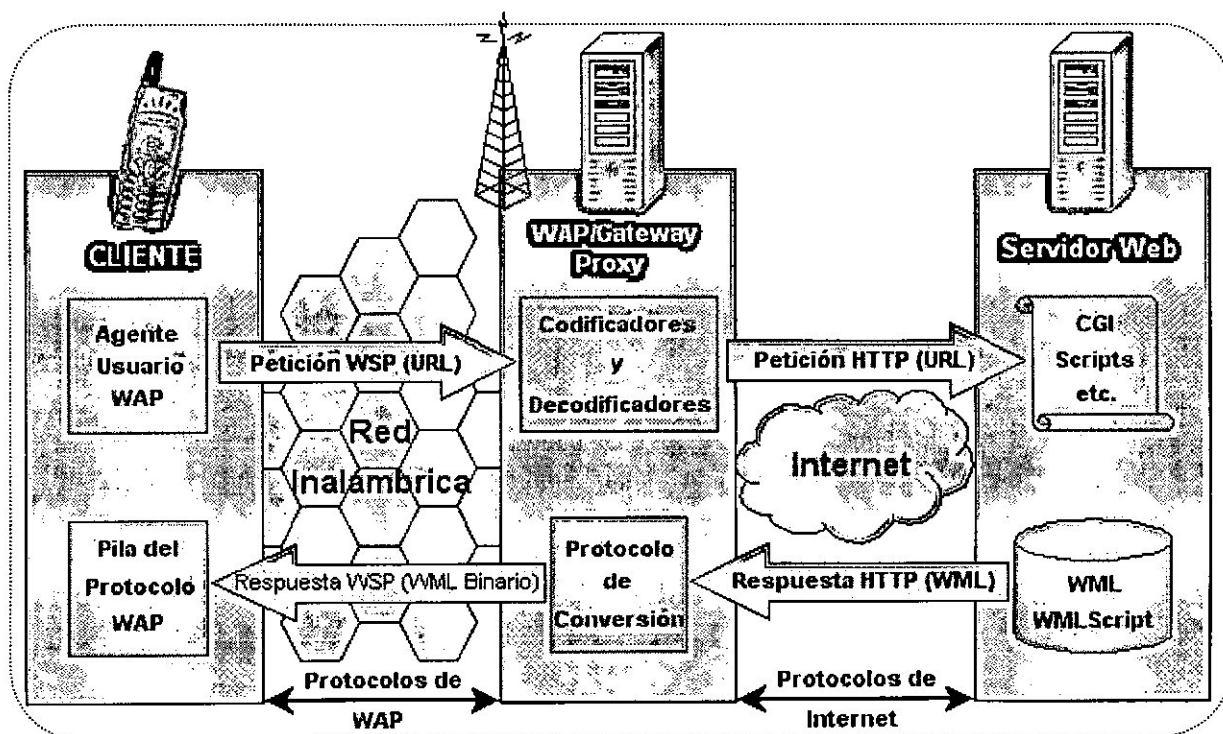


Para salvar la limitación de la amplitud de banda ancha en la red inalámbrica, WML puede ser codificado en un formato binario compacto. El codificado de WML es una de las tareas realizadas por el WAP Gateway/Proxy, el cual es la entidad que conecta el dominio inalámbrico con Internet.

WAP también provee un medio para soportar las tareas más avanzadas, comparable a las soluciones utilizadas en HTML, por ejemplo JavaScript<sup>12</sup>. En el caso de WAP la solución se llamada WMLScript.

La figura muestra el modelo de programación WAP.

Se notan similitudes con el modelo de Internet. Sin el WAP Gateway/Proxy, los dos modelos hubieran sido prácticamente idénticos.



<sup>12</sup> JavaScript es un lenguaje procedural desarrollado por Netscape y Sun Microsystems Inc. para las páginas Web.



---

Como se puede notar, el pedido que es mandado desde el cliente inalámbrico al WAP Gateway/Proxy usa el protocolo WSP (Wireless Session Protocol). WSP en su esencia, es una versión binaria de HTTP.

WAP es desarrollado a escala a través de un ancho campo de redes inalámbricas, como GSM, IS-95, IS-136 y PDC. Finalmente el stack del protocolo WAP, es designado en forma de capas, lo cual significa que se vuelve extensible para pruebas futuras.

## **COMPONENTES WAP**

Tal como se mencionó en el primer informe, WAP, es un estándar de facto para la presentación, envío de información y utilización de servicios adicionales de telefonía sobre dispositivos móviles y otros terminales inalámbricos.

### ***PRINCIPALES CARACTERÍSTICAS***

A diferencia de las tecnologías de Internet para PC, WAP esta pensado para dispositivos que tienen algunas limitaciones técnicas inherentes a la tecnología actual como son:

- Menor potencia de procesamiento
- Menor capacidad en Memoria (ROM-RAM)
- Restricciones de suministro de potencia
- Despliegues pequeños



- 
- Dispositivos de entrada diferentes

De igual forma, las redes inalámbricas, imponen limitaciones como:

- Menor ancho de banda
- Mayor latencia
- Menos estabilidad de la conexión
- Disponibilidad menos predecible.

## **ELEMENTOS WAP**

WAP es el protocolo puente, que permite adaptar los terminales inalámbricos de segunda generación a los nuevos estándares, en una comunicación vía radio entre un terminal móvil con tecnología WAP y un servidor externo.

Los elementos que intervienen son:

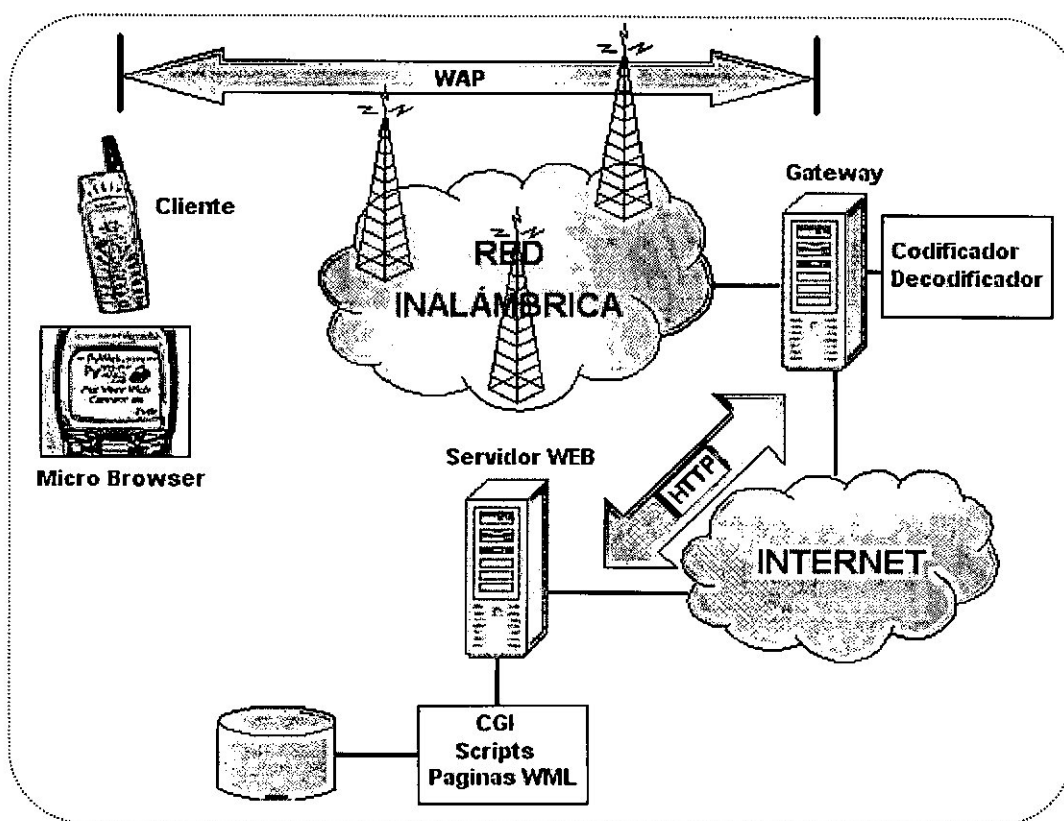
- Cliente
- Pasarela o Gateway
- Servidor de HTTP o WTA

Como podemos ver, en la figura, los tres protagonistas interactúan para poder establecer una conexión vía radio entre un móvil y un servidor web.





Los componentes involucrados en las aplicaciones WAP, son muy similares a los de WWW, excepto por la incorporación de una pasarela o Gateway utilizada para servir de intermediario entre el mundo inalámbrico e Internet.



*Componentes WAP*

## **CLIENTE**

Está constituido por un terminal inalámbrico, habilitado para usar el protocolo WAP, contiene un micro navegador, que sirve de interfaz de usuario, de manera análoga a los navegadores web.



---

Las especificaciones del micro navegador han sido diseñadas de tal forma, que el código resultante sea compacto y eficiente, para que provea una interfaz flexible y potente a los usuarios.

El cliente puede hacer uso de funciones WTA (Wireless Telephony Application) para acceder a funcionalidades de control de llamada, libreta de direcciones, mensajes etc. a través de códigos escritos en WMLScript.

El lenguaje usado para crear las interfaces de usuario en WAP, es denominado WML, el cual es una especialización de XML para sistemas móviles y provee un conjunto de etiquetas similar a HTML, pero muy reducido y adecuado al entorno inalámbrico.

Los documentos WML, son denominados Deck y están estructurados en unidades bien definidas llamadas Cards; también se dispone de WMLScript, un lenguaje estandarizado, para realizar guiones que corren en el cliente de manera similar a Java Script, y que permite realizar labores como validación de datos, realización de cálculos, etc.

### ***PASARELA***

Es la encargada de realizar el traslado de requerimientos WAP a requerimientos HTTP, hace uso de tecnología proxy estándar para web de modo de permitir la conexión de terminales inalámbricos al dominio web; típicamente tiene dos funciones:



- 
- Pasarela de Protocolo:

Permitiendo la traducción de WAP a HTTP

- Codificador / Decodificador:

Compactar el contenido WML para ser enviado por el medio inalámbrico.

Además, esta pasarela puede prestar servicios de seguridad para aplicaciones de comercio electrónico, realizar traducción de páginas HTML a WML, ofrecer servicios de DNS, caché para optimizar la respuesta a los móviles, y servir de interfaz al sistema del operador, para prestar servicios de localización, personalización, configuración de servicios etc.

### ***SERVIDOR DE ORIGEN***

Típicamente es un servidor web al que se ha modificado para servir contenidos WAP, recibe peticiones de páginas WML, imágenes wbmp, páginas compiladas.

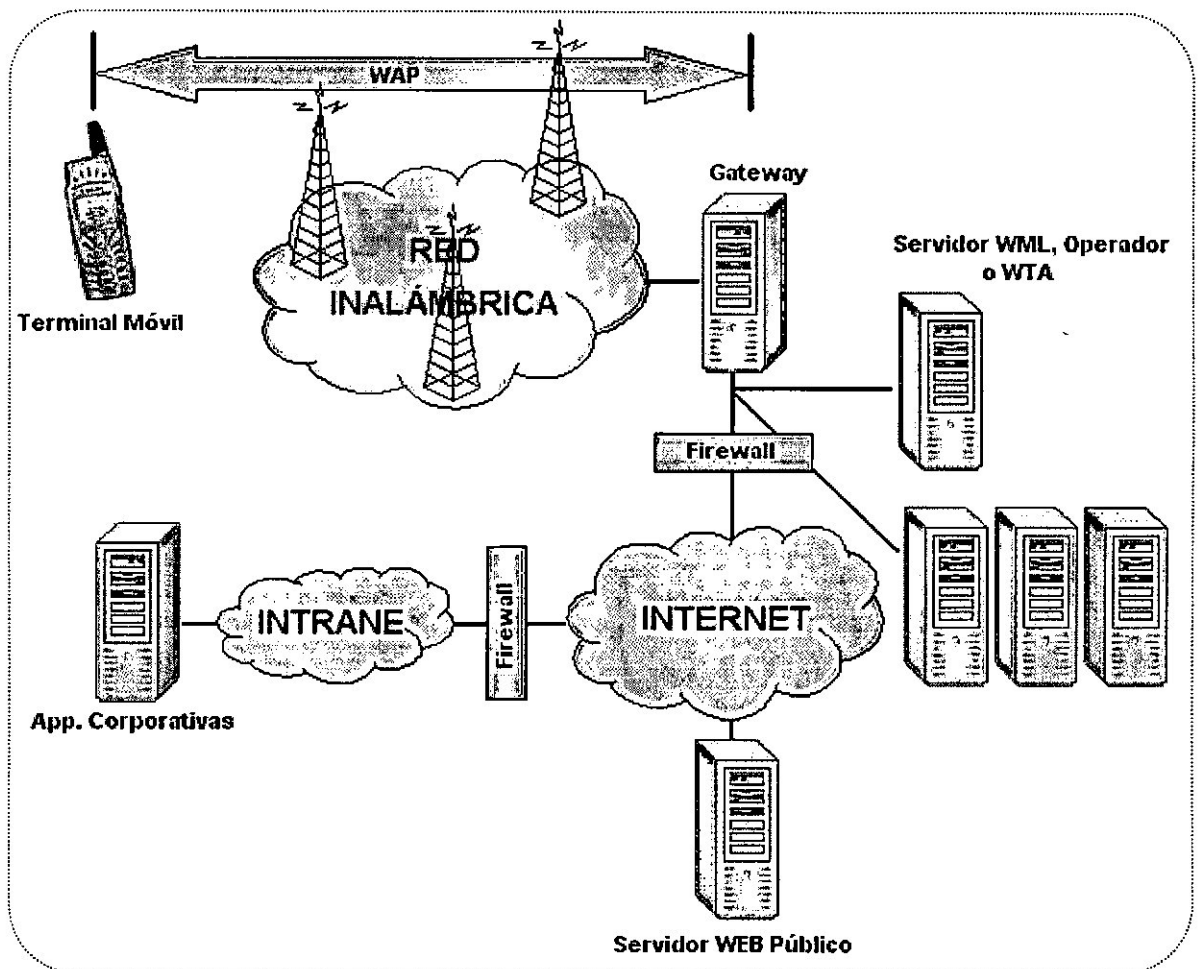
Haciendo uso de tecnologías para aplicaciones web como CGI, Servlets, ASP, PHP, etc. se puede generar contenido WML dinámico a partir de Bases de Datos, Servidores de Correo, etc.



## SERVIDOR WTA

Es un servidor que responde directamente a los requerimientos de un cliente y es usado para proveer acceso WAP, a las características de la infraestructura de red inalámbrica, como por ejemplo, configurar servicios, utilizar servicios de valor agregado etc.

La siguiente figura ilustra un ejemplo de implementación WAP, y se puede apreciar el alcance de la misma a nivel de Internet e Intranet.



*Ejemplo de Implementación WAP*



---

Es importante además tener en cuenta que respecto a la seguridad de los datos transportados, se debe integrar la seguridad de las redes inalámbricas basadas en WAP (WTSL) y las de redes TCP/IP (SSL)

## **ARQUITECTURA WAP**

La arquitectura WAP está pensada para proporcionar un entorno extensible, flexible y escalable para el desarrollo de aplicaciones destinadas a dispositivos de comunicación móvil.

Para ello, 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 interfaces muy bien definidas y especificadas.

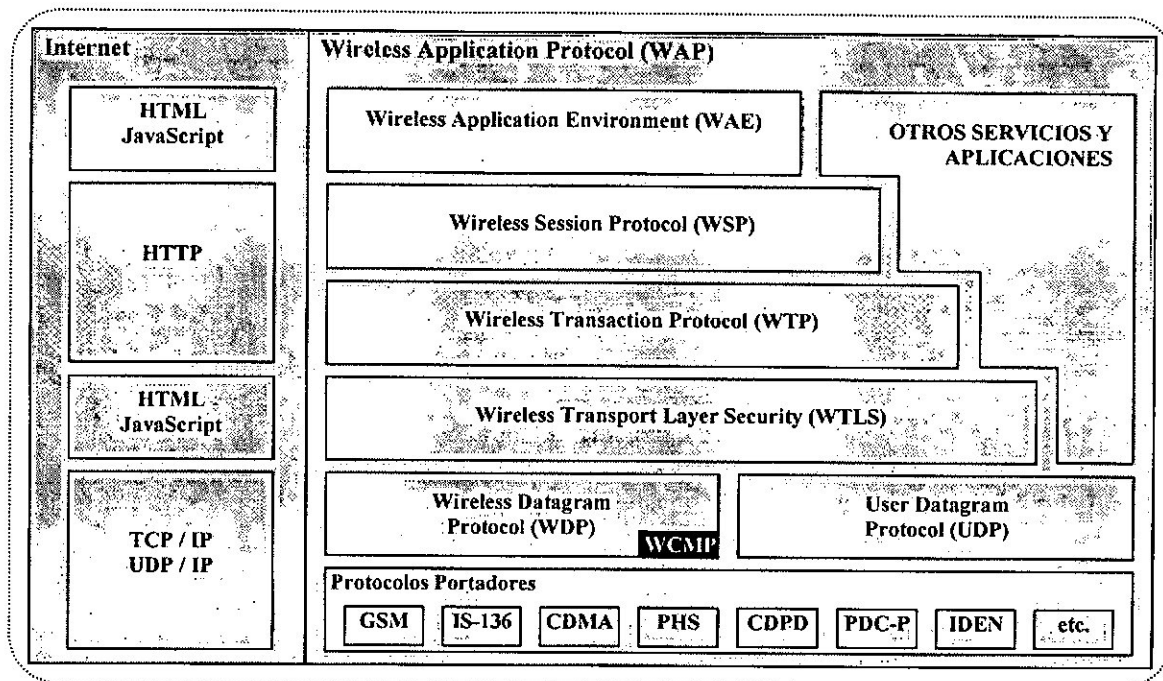
### ***CAPAS DEL PROTOCOLO***

Con el modelo OSI (Open System Interconnection model) en mente, el stack del protocolo WAP es dividido básicamente en cinco capas. Ellas son:

- Application Layer o Capa de Aplicación (WAE)
- Session Layer o Capa de Sesión (WSP)
- Transaction Layer o Capa de Transacciones (WTP)
- Security Layer o Capa de Seguridad (WTLS)
- Transport Layer o Capa de Transporte (WDP)



La figura muestra el stack del protocolo y como este se relacionan con los protocolos en Internet.



*Arquitectura WAP*

### Entorno Inalámbrico de Aplicación (WAE)

La capa superior de la pila de protocolo WAP, compuesta por el Entorno Inalámbrico de Aplicación (WAE), provee un ambiente que hace posible un amplio rango de aplicaciones, utilizadas en los dispositivos inalámbricos.

WAE es un entorno de aplicación de propósito general basado en la combinación del *World Wide Web* y tecnologías de Comunicaciones Móviles.

WAE se encuentra compuesta por los siguientes elementos:



- Modelo de direccionamiento:

Una sintaxis adecuada para nombrar recursos almacenados en servidores.

- Wireless Markup Language (WML):

Un lenguaje de marcas ligero, designado para superar las restricciones de un entorno inalámbrico, con bajo ancho de banda y dispositivos portátiles pequeños.

- WMLScript.
- Aplicaciones Telefónicas Inalámbricas (WTA, WTAI):

Una interfaz de entorno de trabajo y de programación para servicios telefónicos.

Con el propósito de hacer uso de las características mencionadas anteriormente, WAP asume que dos usuarios pueden estar disponibles en el dispositivo inalámbrico.

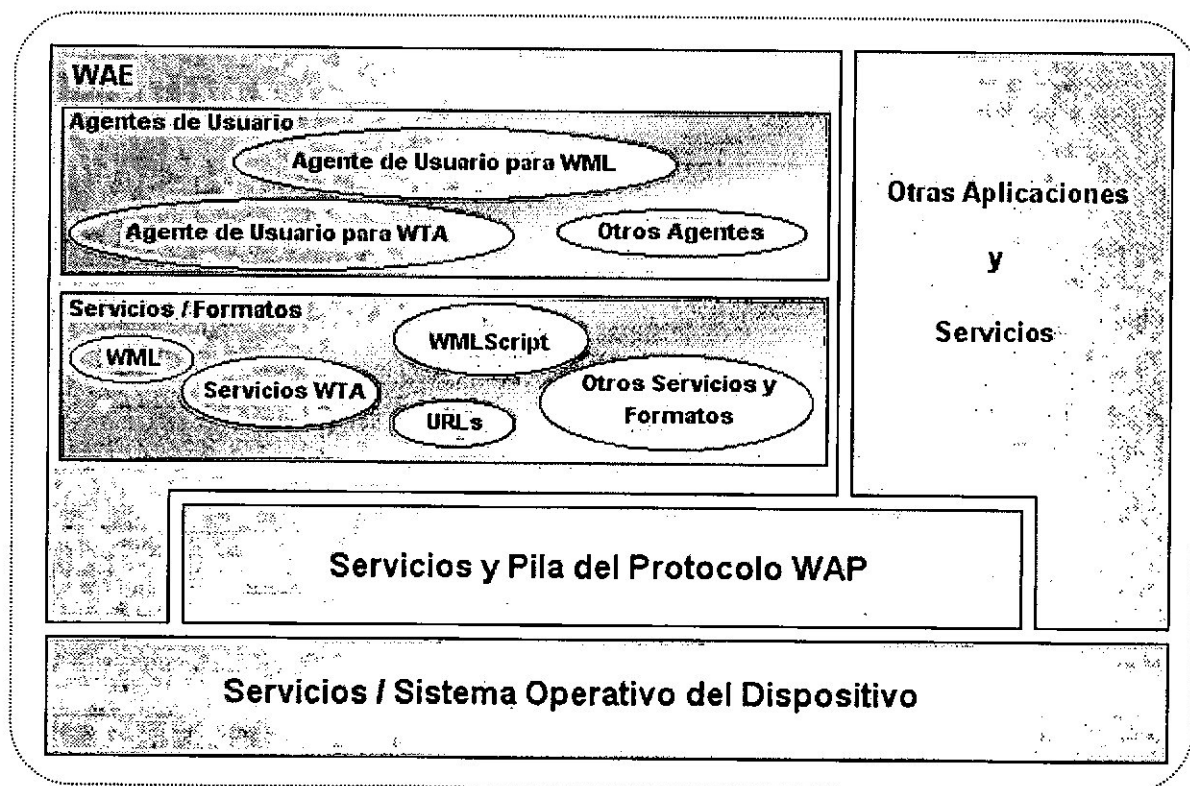
Un Agente de Usuario está dentro de este contexto, típicamente en una aplicación dentro del dispositivo, que interpreta el contenido en una manera bien definida y que manipula la interacción del usuario cuando sea necesario.



En el caso de las conexiones por medio de cables, un Agente de Usuario se refiere típicamente a un navegador; por ejemplo Microsoft Internet Explorer o Netscape, usados en computadoras de escritorio.

Basándonos en el Modelo de Funcionamiento de WAP, el cual se describió anteriormente, vemos que el entorno Inalámbrico de Aplicación se divide en dos partes, dos capas lógicas:

- Los **Agentes de Usuario**, que incluyen aquellos elementos como navegadores, agendas telefónicas, editores de mensajes, etc.
- Los **Servicios y Formatos**, que incluyen todos aquellos elementos y formatos comunes, accesibles a los Agentes de Usuario, tales como WML, WMLScript, formatos de imagen, etc.



*Componentes del Cliente WAE*





---

Como se puede observar en la figura anterior, dentro de WAE se separan los Servicios de Agentes de Usuario, lo que proporciona flexibilidad para combinar varios Servicios dentro de un único Agente de Usuario, o para distribuir los Servicios entre varios Agentes de Usuario.

Los dos Agentes de Usuario más importantes son el Agente de Usuario para WML y el Agente de Usuario para WTA.

El Agente de Usuario para WML es el Agente de Usuario fundamental en la arquitectura del Entorno Inalámbrico de Aplicación. A pesar de su importancia, este Agente de Usuario no está definido formalmente dentro de esta arquitectura, ya que sus características y capacidades se dejan en manos de los encargados de su implementación.

El único requisito de funcionalidad que debe cumplir este Agente de Usuario, es el proporcionar un sistema intérprete a los lenguajes WML y WMLScript, de forma que se permita la navegación desde el terminal móvil.

Por otra parte, el Agente de Usuario para WTA permite a los autores acceder e interactuar con las características de los teléfonos móviles (p. e. Control de Llamada), así como otras aplicaciones supuestas en los teléfonos, tales como agendas de teléfono y aplicaciones de calendario.

La redacción del Agente de Usuario WML podría ser un poco confusa desde este agente, ya que no solo interpreta WML, sino que también WMLScript, etc. La relación debe reflejar que este Agente de Usuario, en contraste con el usuario WTA, no es capaz de manipular servicios telefónicos.



---

## **Modelo de Direccionamiento**

WAP utiliza el mismo modelo de direccionamiento que se utiliza en Internet, el cual es el Uniform Resource Locators (URL). Una URL identifica únicamente a un recurso, por ejemplo un documento WML sobre un servidor, que puede ser recuperado usando los conocidos protocolos.

En adición a las URLs, WAP además utiliza Uniform Resource Identifiers (URI). Una URI es usada para direccionar recursos que no son accesibles necesariamente utilizando los conocidos protocolos. Un ejemplo de uso de una URI es el acceso local de las funciones telefónicas de un dispositivo inalámbrico.

## **Wireless Markup Language (WML)**

El Wireless Markup Language de WAP, tal como se ha mencionado anteriormente, es usado analógicamente como HTML en WWW. WML está basado en XML (Extensible Markup Language).

WML usa un desk/card para especificar un servicio. Una card es típicamente una unidad de interacción con el usuario, que es, cualquier presentación de información o requerimiento de información desde un usuario.

Una colección de card se llama desk, las cuales usualmente constituyen un servicio. Ese acercamiento asegura que una cantidad adecuada de información es desplegada para el usuario simultáneamente desde páginas internas.



---

Las características de WML incluyen:

- Variables
- Características para dar formato al texto
- Soporte para imágenes
- Soporte para botones
- Control de navegación
- Control de la navegación histórica
- Soporte para el manejo de eventos (por ejemplo: servicios telefónicos)
- Diferentes tipos de interacciones del usuario, por ejemplo selección de listas y entrada de campos

WML puede ser binariamente codificado por el WAP Gateway/Proxy, con el propósito de preservar el ancho de banda en los dominios inalámbricos.

### **WMLScript**

WMLScript está basado en ECMAScript, el mismo lenguaje de script, en el cual se basa JavaScript. Este puede ser usado para incrementar los servicios descritos en WML, en la forma que esté para alguna extensión de



---

servicios inteligentes, por ejemplo lógica de procedimientos, lazos, expresiones condicionales, y funciones computacionales.

WMLScript puede ser usado por ejemplo para validar las entradas de los usuarios. Gracias a esto WML no necesita suministrar algún mecanismo para archivar la validación de la entrada de usuario, basta con un paso de ida y vuelta hacia el servidor, para determinar si la entrada del usuario es válida o no, o para verificar la disponibilidad del script.

Brindar acceso a funciones locales en un dispositivo inalámbrico, es otra área donde WMLScript es usado, por ejemplo acceso a las funciones relacionadas con telefonía.

WMLScript también soporta librerías WMLScript. Estas librerías contienen funciones que extienden las funcionalidades básicas de WMLScript. Estas proveen una manera para realizar futuras expansiones de funciones sin tener que cambiar el núcleo de WMLScript.

Tal como WML, WMLScript puede ser codificado binariamente por el WAP Getway/Proxy, para minimizar la cantidad de datos enviados a través del aire.

### **Aplicaciones Telefónicas Inalámbricas (WTA)**

El ambiente de las aplicaciones telefónicas inalámbricas (WTA) suministra un medio para crear servicios telefónicos usados por WAP. Como se mencionó, WTA utiliza un Agente de Usuario separado del Agente de Usuario



---

común de WML. El Agente de Usuario WTA está basado en el Agente de Usuario de WML, pero es ampliado con funcionalidades que encuentran requerimientos especiales para servicios telefónicos.

Esas funcionalidades incluyen:

- **Interfaces de Aplicación Telefónica Inalámbrica (WTAI)**

Interfaz hacia un conjunto de funciones, relacionadas con funciones dentro de un teléfono móvil, que pueden ser invocadas desde WML y/o WMLScript.

Esas funcionalidades incluyen por ejemplo: administración de llamadas, manejo de mensajes de texto y controles de guía telefónica.

WTAI se divide dentro de tres categorías: *Funciones comunes de red, funciones específicas de red y funciones públicas.*

Las funciones comunes son habilitadas en todos los tipos de redes, mientras que las funciones específicas son únicas en un cierto tipo de red.

En contraste a las otras dos librerías de funciones, la librería de funciones públicas puede ser invocada también desde el Agente de Usuario de WML. Actualmente, la librería de funciones públicas solamente contiene una función para establecer llamadas, la cual debe ser reconocida por el usuario antes de llevarla a cabo.



---

- **Repositorio**

Muchos servicios WTA aplican requerimientos sobre el manejo en tiempo real, esto implica que no es factible recuperar el contenido desde un servidor, lo que involucra un cierto retraso.

El repositorio hace esto posible, permitiendo almacenar servicios WTA persistentemente dentro del dispositivo, para habilitarles el acceso fuera del acceso a la red.

- **Manipulación de Eventos**

Los eventos típicos dentro de una red móvil son: entrada de llamadas, llamadas desconectadas, y llamadas de respuesta. Con el propósito de crear servicios telefónicos, debe ser posible manejar estos eventos.

Los eventos para la manipulación dentro del WTA, permiten almacenar servicios WTA en el repositorio, para ser iniciados en respuesta a tales eventos. Los eventos también pueden ser destinados a una cierta acción dentro de WML, para hacer posible el manejo de eventos dentro de un servicio.

- **Indicador de Servicios WTA**

Es un contenido típico, que permite a los usuarios ser notificados acerca de eventos de diferentes tipos, por ejemplo correos nuevos de voz,



---

y determinar la posibilidad de comenzar el servicio apropiado para manejar el evento.

En su forma más básica, el indicador de servicios WTA, permite enviar una URL y un mensaje para un dispositivo inalámbrico. El mensaje es visualizado por el usuario, el cual es interrogado si quiere iniciar el servicio indicado por la URL inmediatamente, o si quiere posponer la Indicación de Servicio para manejarlo luego. La Indicación de Servicios WTA debe ser entregado a un dispositivo que utiliza push.

WTAI permite el acceso a funciones a las cuales no es adecuado permitir un acceso común a ellas, exceptuando las librerías de funciones públicas. Por ejemplo, el establecimiento de llamadas y la manipulación de la guía telefónica, sin el conocimiento del usuario, pueden implicar costos indeseados y una violación a la integridad del usuario, si las correspondientes funciones son utilizadas en forma inapropiada. Las otras funciones provistas por el esqueleto de WTA pueden ser considerado de la misma manera.

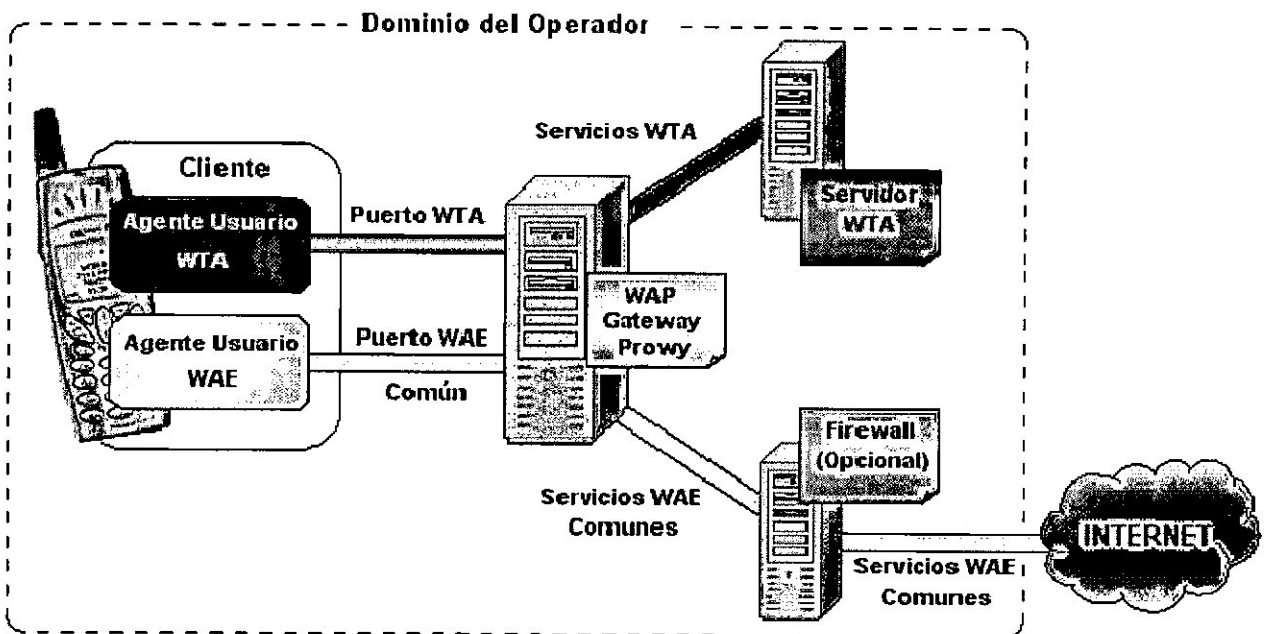
El esqueleto de WTA depende de un Agente de Usuario WTA dedicado, capaz de llevar a cabo estas funciones que no son proporcionadas por el Agente de Usuario común de WML.

Solamente un proveedor confiable de contenidos debe ser capaz de hacer disponible, este contenido para el Agente de Usuario WTA, por ejemplo el operador o un proveedor de contenido fiable por el operador.

De esta manera debe ser posible distinguir, entre servidores que son autorizados para abastecer al Agente de Usuario, con servicios que contienen estas funciones, y aquellos que no lo son.

Para lograr esto el Agente de Usuario WTA recupera estos servicios desde el dominio WTA, el cual, en contraste con Internet, es controlado por el operador de la red.

La figura siguiente, muestra como los servicios WTA y otros servicios son separados uno del otro usando el control de acceso WTA, basado en el número de puertos:



*Control de Acceso WTA*

El servidor WTA puede ser un servidor común de web usado para el alojamiento de contenido. Este puede ser capaz también de comunicarse con otras entidades, tal como nodos-IN o sistemas de correo de voz, para proveer funcionalidades extendidas, relacionadas con la telefonía.





---

La comunicación con tales entidades, puede ser controlada por aplicaciones sobre el servidor WTA, que pueden ser referencia dentro de un servicio WTA usando URLs.

## ***PROTOS WAP***

El conjunto de protocolos WAP contiene cuatro protocolos para el manejo de la comunicación entre los clientes y el WAP Gateway/Proxy. Estos protocolos son modelados después de los protocolos usados en Internet, y pueden ser usados en cuatro diferentes configuraciones:

### **Modo Sin Conexión**

Esta configuración, utiliza solamente WSP por encima de WDP. Esta ofrece un servicio de datagrama, esto significa que los mensajes enviados no son aceptados, y por lo tanto no se ofrece una garantía de entrega. La funcionalidad puede ser como un simple modelo "send-and-forget".

### **Modo Sin Conexión con Seguridad**

En complemento a lo dicho anteriormente, WTLS es usado para proveer autenticación, encriptación, etc.



---

## **Modo Conexión**

El modo conexión utiliza WTP junto con WSP y WDP. WTP ofrece transmisiones seguras, esto significa que los mensajes enviados deben ser aceptados y puede ser retransmitido si se pierde. Esto también utiliza un modo de WSP que maneja sesiones de conexión permanente.

## **Modo Conexión con Seguridad**

En complemento a lo dicho anteriormente, WTLS es utilizado para proveer, autenticación, encriptación, etc.

## **CAUSAS DE WAP**

Durante los últimos años, tanto Internet y la comunicación inalámbrica de voz han experimentado una amplia y rápida aceptación. La unificación de estas dos tecnologías, la Internet inalámbrica, no obstante no ha disfrutado el mismo desarrollo, aunque, Internet mantiene un medio para un rápido desarrollo de servicios, corto tiempo de mercado, facilidad de uso, y una conveniente manejabilidad.

### ***¿POR QUÉ UN NUEVO ESTÁNDAR?***

Una pregunta legítima es por qué el uso de las capacidades de datos inalámbricas, en este contexto, el acceso inalámbrico a Internet, no ha seguido las tendencias de la comunicación inalámbrica de voz ni la de Internet.



Así como con muchas otras cosas en la vida, las expectativas son muchas. Cualquiera que ha intentado acceder a Internet usando una computadora portátil y un teléfono celular, sabe que las expectativas que tenemos creadas utilizando Internet en la oficina o en la casa, no se cumple; de hecho normalmente es una experiencia bastante pesada.

WAP resuelve este problema siendo diseñado para superar las restricciones de un ambiente inalámbrico. Ambas limitaciones en la red y en el cliente se tiene en cuenta.

### ***ADAPTACIÓN A LOS LÍMITES DE LA RED INALÁMBRICA***

WAP escala a través de un amplio rango de redes inalámbricas y portadores. Por lo tanto se diseña para permitir el acceso a los servicios vía Internet usando SMS simple, así como también las redes de paquetes de datos como el Servicio GPRS (General Packet Radio Service).

Dentro de los problemas más importantes de las redes inalámbricas encontramos:

#### **Bajo Ancho de Banda:**

El problema con el pobre rendimiento, cuando se utilizan portadores inalámbricos con bajo ancho de banda, se hace válido especialmente si el usuario no es consciente del servicio al que accede.



---

El servicio no debe consumir mucho ancho de banda, pero si debe ser conveniente para el acceso inalámbrico.

La porción más grande de la masa del mercado, así como muchos usuarios avanzados no son conscientes de esto. Y además, los usuarios simplemente no deben tener que preocuparse sobre cómo se diseñan los servicios, a los que ellos acceden, para tener sus expectativas cumplidas.

WAP resuelve este problema minimizando el tráfico sobre la interfaz aérea. WML y WMLScript son codificados en forma binaria, dentro de una forma compacta, cuando se envían sobre el aire para minimizar el número de bits y bytes transmitidos.

WSP, es el equivalente de WAP para HTTP en Internet, también es binario por la misma razón. Es más, soporta sesiones duraderas que pueden suspenderse y luego reanudarse, y ocultamiento de encabezados; esto permite ahorrar, desde el establecimiento de la sesión, un valioso ancho de banda.

El Protocolo de Transacción Inalámbrica (WTP), es la analogía de TCP en Internet, no sólo está diseñado para minimizar la cantidad de datos en cada transacción, sino también el número de transacciones.

### **Alta Latencia:**

Las redes inalámbricas poseen una alta latencia comparada con las redes que utilizan cables. Esta restricción es relevante en todas las redes



---

inalámbricas de hoy, incluso para aquéllas que suministran un gran ancho de banda.

Esto es superado en WAP minimizando la cantidad de roundtrips entre el dispositivo inalámbrico y la red inalámbrica. También utiliza un modelo de requerimiento / respuesta asíncrono.

En las redes con cables, la baja latencia implica que los requerimientos y respuestas pueden ser manejados sincrónicamente, de modo que el tiempo entre ellos no afecte la experiencia del usuario.

En las redes inalámbricas con alta latencia, esto no es un acercamiento factible, especialmente cuando se utilizan proveedores con alta latencia como SMS.

Este problema está entre otros rasgos, dirigido por WSP, permitiendo manejar los requerimientos y respuestas asincrónicamente, es decir, un nuevo requerimiento puede ser enviado antes de contestar a un requerimiento que ha sido recibido anteriormente.

El ambiente de aplicación en WAP, usa el concepto de scripting, denotando que los roundtrips entre un cliente y un servidor, pueden ser evitados cuando viene por ejemplo la aprobación de las entradas del usuario.

El ambiente de Aplicación de la Telefonía Inalámbrica resuelve la latencia y el bajo ancho de banda, introduciendo un almacenamiento, el cual es un recipiente de almacenamiento persistente, utilizado para alojar servicios que



---

deben ser empezados en respuesta a un evento producido en la red móvil por ejemplo una llamada entrante.

Dado que estos servicios están disponibles inmediatamente, no se necesita ningún roundtrips al servidor de WTA y por eso es posible el manejo en tiempo real.

Comparado con TCP, WTP necesita un número más pequeño de transacciones para cada método invocado, es decir controla el número de roundtrips.

### **Estabilidad sin Conexión / Disponibilidad Impredecible del Portador:**

El acceso de una red por medio de un cable, proporciona una conexión más fiable a la red. Ése no es el caso de las redes inalámbricas, dónde los portadores podrían ser inaccesibles por períodos largos o cortos de tiempo debido al desvanecimiento, perdida del alcance de radio, o capacidad deficiente.

Como se mencionó, se asume que las sesiones soportadas por WSP son duraderas. El problema mencionado está dirigido en WSP permitiendo reasumir las sesiones perdidas, incluso cuando dinámicamente se asignó una dirección IP utilizada.



La capa de la transacción en WAP, WTP, se ha conservado simple comparada a TCP usada en la Internet con cables. Dado que ninguna conexión es establecida, los efectos de la pérdida de portador y los tiempos de inactividad se minimiza.

La naturaleza de una conexión inalámbrica implica que se pierdan a menudo pequeños segmentos de un mensaje. WTP soporta la retransmisión selectiva de datos, denotando que los segmentos perdidos son los únicos retransmitidos, no el mensaje entero como en TCP.

### ***ADAPTACIÓN A LOS LÍMITES DEL DISPOSITIVO INALÁMBRICO***

WAP está destinado a dispositivos portátiles de varios tipos. Los servicios deben ser accesibles de un PC Portátil así como desde un teléfono pequeño. WAP resuelve este hecho, teniendo en cuenta los siguientes problemas:



*Limitaciones de un dispositivo Inalámbrico*



---

### **Pantalla Limitada:**

Cuando accedemos a un servicio desde una computadora de escritorio, el tamaño de la pantalla no limita la experiencia del usuario. Los dispositivos inalámbricos también podrían tener “grandes” pantallas, por ejemplo un PDA (Personal Digital Assistant).

Pero muchos dispositivos tendrán pantallas más pequeñas, por ejemplo los teléfonos móviles para proporcionar una portabilidad más grande.

No importa que tan buenas sean estas pantallas en el futuro, el tamaño de la mano humana siempre limitará el tamaño de estos dispositivos.

Intentemos imaginar que la experiencia sería semejante a cuando accedemos a un servicio diseñado para un dispositivo con una gran pantalla, pero accediendo desde un pequeño teléfono.

El resultado en la mayoría de los casos sería muy malo, el rendimiento del servicio estará lejos de lo que se pensó originalmente.

La información que el usuario quería realmente, probablemente estaría inundada por información indeseada, debido a la baja claridad que la pequeña pantalla de tales dispositivos puede ofrecer.

En lugar de utilizar la estructura plana proporcionada en los documentos HTML, WML estructura sus documentos en decks (barajas) y cards (cartas).





---

Una carta es una unidad simple de interacción con el usuario final, por ejemplo un texto en pantalla, una lista de selección, un campo de entrada, o una combinación de ellos.

Una carta es típicamente pequeña, lo suficiente para ser visualizada en una pantalla pequeña. Cuando un servicio es ejecutado el usuario navega a través de una serie de cartas. La serie de cartas utilizadas para hacer un servicio, se reúnen en una baraja.

### **Facilidades de Entrada Limitada:**

Los dispositivos inalámbricos no poseen a menudo las mismas facilidades de entrada, como sus equivalentes que utilizan cables, es decir, les falta el teclado y el Mouse, por lo tanto poseen menos interfaces.

WML también resuelve este problema. Los elementos utilizados en WML penden ser implementados fácilmente, de esta manera hacen muy humildes los requerimientos en el teclado.

El uso de barajas y cartas proporcionan un modelo de navegación, que requiere una mínima navegación de paginas intermedias, dado que el usuario es guiado como en un canal por una serie de cartas, en lugar de tener que desplazase de arriba abajo en una página grande.



---

Además, WML soporta botones suave que proporcionar a los desarrolladores un recurso para combinar las acciones deseadas, con un proveedor específico.

### **Memoria y CPU Limitada:**

Normalmente, no se equipa a los dispositivos inalámbricos con las mismas cantidades de memoria y CPU, comparados con las computadoras de escritorio. La restricción de memoria es válida tanto para la memoria RAM, así como para la memoria ROM.

Aunque las tendencias indican que en el futuro se podrá disponer de más memoria y CPUs más poderoso, pero las diferencias relativas probablemente permanecerán.

WAP soluciona estas restricciones definiendo un stack de protocolo liviano, adaptado a sus propósitos. El juego limitado de funcionalidades, proporcionadas por WML y WMLScript, hacen esto posible implementando navegadores que generan pequeñas demandas de CPU y recursos de ROM.

Cuando las demandas se originan en la RAM, la codificación binaria de WML y WMLScript ayudan a la conservación de la memoria RAM tan pequeña como sea posible.



---

### **Limitada Carga de Baterías:**

El obstáculo en los dispositivos de comunicación inalámbricos, hoy es el tiempo de operación, es decir la carga de las baterías restringen su uso.

Aunque las baterías son cada vez mejores, y las interfaces de radio se mejoren para consumir menos energía, todavía hay mucho que lograr en esta área.

El acceso a los servicios inalámbricos aumentará la utilización de portadores (interfaz de radio), por lo que aumenta también el consumo de baterías.

Este problema se resuelve minimizando el ancho de banda y manteniendo la utilización de los portadores tan bajo como sea posible.

## **DOCUMENTOS WML**

En el primer informe parcial, se hizo referencia al lenguaje utilizado para el desarrollo de aplicaciones WAP, el mismo es WML, es un lenguaje de marcas basado en el lenguaje de marcación extensible (XML, *Extensible Markup Language*) y fue desarrollado para especificar contenidos e interfaces de usuario con terminales de banda estrecha, tales como teléfonos móviles e intérpretes de páginas.



---

## **CARACTERÍSTICAS PRINCIPALES DEL LENGUAJE**

Las características del lenguaje WML pueden agruparse en cuatro áreas principales:

- WML ofrece un soporte de texto e imagen y tiene una amplia variedad de formatos y comandos.
- Las cartas WML se agrupan en barajas. Una baraja WML, es similar a una página HTML identificada por un URL (*Uniform Resource Locator*) y es la unidad básica de transmisión de contenidos.
- WML ofrece soporte para gestión de navegación entre cartas y barajas, e incluye comandos para su manejo. Estos pueden usarse para navegar o ejecutar "scripts". WML también provee de conexiones de anclaje, similares a las usadas en el HTML versión 4.
- Se pueden establecer parámetros para todas las barajas de WML usando un modelo establecido. Se pueden usar variables en lugar de cadenas y sustituirse en el tiempo de ejecución. Esta forma de establecer parámetros permite que los recursos de la red sean usados de forma eficiente.



---

Toda la información de WML, se transmite en formato codificado por la red inalámbrica.

## **SINTAXIS WML**

WML hereda la mayoría de sus construcciones sintácticas de XML<sup>13</sup>.

### **Entidades**

El texto WML puede contener entidades de caracteres numéricos o alfanuméricos, las cuales especifican caracteres específicos en el juego de caracteres de documento.

Estas entidades especiales, se utilizan para especificar caracteres que deben ser solventados en WML o que pueden ser difíciles de introducir en un editor de texto. Por ejemplo, el ampersand (&), se representa por la entidad de nombre &#amp;. Todas las entidades empiezan con un ampersand y terminan con un punto y coma.

WML es un lenguaje XML. Esto implica que los caracteres ampersand y “menor que” deben ser simulados cuando se utilicen en datos de texto, es decir, esos caracteres pueden aparecer en su forma literal solo cuando se utilicen como delimitadores de marca, dentro de un comentario, etc.

---

<sup>13</sup> Para más información sobre la sintaxis de XML, ver en *XML Specification*.

### Ejemplo

```

<card id="card1">
  <p>
    Ampersand = &amp;
  <br/>
    Quote = &quot; <br/>
    Less than = &lt; <br/>
  </p>
</card>
    
```

Interfaz de usuario en el terminal  
de destino

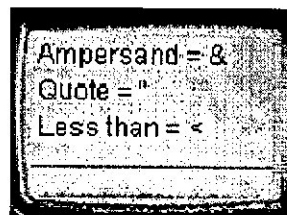


Figura 6.: Ejemplo de Entidades

### Etiquetas

Una etiqueta (*tag*) es un descriptor de elementos del lenguaje. Una etiqueta describe un elemento y contiene un nombre de tipo de elemento y un identificador único. Una etiqueta podría también incluir atributos describiendo otras propiedades.

El código WML consiste en un conjunto de etiquetas, cada una de ellas encerrada entre un par de corchetes, < y >.

<tag>	Comienzo del elemento. La etiqueta de comienzo puede contener atributos.
</tag>	Etiqueta que indica el fin de un elemento.
<tag/>	Indica un elemento vacío, por ejemplo  , que indica un salto de línea.



---

## Elementos

Los elementos especifican todas las marcas e información estructural de una baraja de WML. Los elementos pueden contener una etiqueta de comienzo, contenido, otros elementos y una etiqueta de final. Los elementos tienen una o dos estructuras:

`<tag> content </tag> ó`

`<tag/>`

Los elementos que incluyen contenido u otros elementos están identificados por una etiqueta de comienzo `<tag>` y una etiqueta de finalización `</tag>`. Una etiqueta de elemento vacío `<tag/>` identifica elementos sin contenido.

## Atributos

Los atributos WML especifican información adicional para un elemento. Los atributos se especifican siempre en la etiqueta de comienzo de un elemento.

Por ejemplo:

`<tag attr="value"/>`

Los nombres de los atributos deber estar en minúsculas.

WML requiere que todos los valores del atributo sean puestos entre comillas, usando tanto dobles comillas ( " ) como comilla simple ( ' ). Las



---

comillas simples pueden incluirse dentro del valor de atributo cuando el valor se delimita por comillas dobles y viceversa. Las entidades de caracteres pueden incluirse en un valor de un atributo.

Algunos atributos son obligatorios. Por ejemplo, el elemento *go* requiere el atributo *href*:

```
<go href="http://www.acme.com"/>
```

### **Comentarios**

Los comentarios en WML siguen el estilo de comentario de XML y tienen la siguiente sintaxis:

```
<!-- a comment -->
```

Los comentarios son una ayuda para el programador del documento WML y no se muestran al usuario en el terminal de destino. Hay que señalar que los comentarios WML no pueden ser almacenados.

### **Variables**

Los parámetros pueden ser incluidos en las cartas y barajas de WML usando variables. Para sustituir una variable, por su valor, dentro de una carta o baraja, se utilizan las siguientes sintaxis:

```
$identifier
```

```
$(identifier)
```





---

`$(identifier:conversion)`

Se requieren paréntesis si el espacio en blanco no indica el final de una variable, es decir si el nombre de la variable lo conforman varias palabras separadas por espacios en blanco. La sintaxis de variable tiene la mayor prioridad en WML, esto es, en cualquier lugar donde la sintaxis de variable sea legítima, un carácter "sin escapar" '\$' conlleva a una sustitución de una variable. Las referencias de variables son legítimas en cualquier *PCDATA* y en cualquier valor de atributo que sea de tipo entero *vdata*.

Ejemplo:

Una secuencia de dos signos de pesos (\$\$) representa un carácter de signo de peso sencillo.

El código WML podría tomar la siguiente forma:

This is a \$\$ character.

The value is \$(amount)\$\$.

En el terminal del usuario, esto se mostraría como:

This is a \$ character.

The value is 5000\$.



---

## Sensibilidad de formato de letra

El lenguaje XML es un lenguaje sensible al formato de letra (mayúsculas - minúsculas) y WML ha heredado dicha característica. Cuando se analiza una baraja de WML, no se realiza aceptando el doble formato como uno, es decir, reconoce como diferentes lo que está en mayúsculas y lo que está en minúsculas. Esto implica que todas las etiquetas, atributos y contenidos de WML son sensibles al formato de letra. Además cualquier valor de atributo enumerado es sensible al formato.

### Ejemplo

Los siguientes valores de atributo son todos diferentes:

id="Card1"

id="card1"

id="CARD1"

## Sección Cdata

Las secciones *Cdata* se utilizan para "escapar" bloques de texto y son legítimas en cualquier sección *pcdata*, por ejemplo, dentro de un elemento. Las secciones *Cdata* empiezan con una cadena "`<![CDATA [`" y terminan con una cadena "`]] >`".

Por ejemplo:

```
<![CDATA [This is <b> a test.]]>
```



Cualquier texto dentro de una sección cdata se trata como texto literal y no será procesado. Las secciones cdata son útiles en cualquier lugar donde sea conveniente el texto literal.

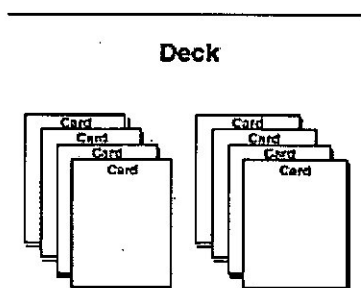
## ***ELEMENTOS DE WML***

### **Cartas y Barajas**

Todo el código WML se organiza dentro de una colección de cartas y barajas. Las **cartas** especifican una o más unidades de interacción con el usuario, por ejemplo un menú de opciones, una pantalla de texto o un campo de entrada de texto. Lógicamente, un usuario navega a través de una serie de cartas de WML, revisando el contenido de cada una de ellas, introduciendo la información requerida, realizando elecciones y moviéndose a otra carta.

Las cartas están agrupadas dentro de barajas. Una baraja es la unidad más pequeña WML que un servidor puede enviar al terminal del usuario.

La siguiente figura ilustra el significado de carta y baraja:



*Cartas (cards) y barajas (decks) en el lenguaje WML.*



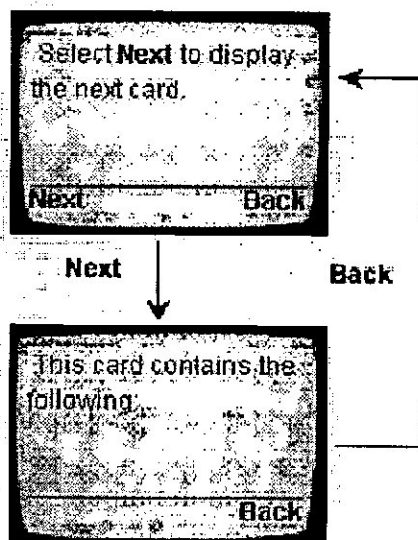
En este ejemplo de WML, se introduce una baraja WML simple conteniendo dos cartas. Cuando el usuario presiona la tecla ACCEPT (Aceptar), rotulada "Next" (Siguiente), el usuario navega a la segunda carta de la baraja y muestra su contenido.

```

<?xml version="1.0"?>                                <!-- 1 -->
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">          <!-- 2 -->
<wml>                                                <!-- 3 -->
<card id="First_Card">                               <!-- 4 -->
<do type="accept" label="Next">                     <!-- 5 -->
<go href="#Second_Card"/>                           <!-- 6 -->
</do>                                                <!-- 7 -->
<p>                                                  <!-- 8 -->
Select <b>Next</b> to display the next card.         <!-- 9 -->
</p>                                                <!-- 10 -->
</card>                                              <!-- 11 -->
<card id="Second_Card">                              <!-- 12 -->
<p>                                                  <!-- 13 -->
This card contains the following:...                 <!-- 14 -->
</p>                                                <!-- 15 -->
</card>                                              <!-- 16 -->
</wml>                                              <!-- 17 -->

```

La siguiente figura ilustra la interfaz de usuario genera por la baraja anterior:



*Baraja que contiene dos cartas.*

Pequeña explicación de este ejemplo:

1. Las dos primeras líneas definen la cabecera del documento, que identifica el subjuego XML. Esta cabecera debe incluirse al comienzo de cada baraja de WML, es decir, antes de cada etiqueta <wml>.
2. La tercera línea define el encabezado de la baraja WML. Todas las barajas WML deben empezar con una etiqueta <wml> y terminar con una etiqueta </wml>.
3. La cuarta línea de la baraja, especifica el encabezado e identificador de la primera carta. Análogamente a las barajas, las



---

cartas también requieren etiquetas de comienzo y de finalización, `<card>` y `</card>`.

4. La mayoría de los elementos de WML permiten especificar atributos. Los atributos se introducen en la forma `attribute=value` (atributo=valor), donde *attribute* es el nombre de atributo y *value* es un valor alfabético o numérico que toma el atributo.
5. La quinta línea define una acción, la cual especifica lo que el terminal del usuario debiera hacer al presionar una determinada tecla de función. El atributo *type* identifica la tecla (*accept*) y el atributo *label* asigna un rotulo (*Next*) a la tecla especificada.
6. La sexta línea indica una acción de relación entre cartas, páginas webs, etc. El atributo *href* indica el destino URI donde dicha acción nos va a llevar (en este ejemplo éste será la carta "Second-Card").

### **ATRIBUTOS COMUNES**

Todos los elementos de WML tienen dos atributos centrales, *id* y *class*, que pueden usarse para tareas como transformaciones en el servidor.

El atributo *id* proporciona a un elemento un nombre único dentro de una baraja sencilla. El atributo *class* afilia un elemento con una o más clases.



---

A los elementos múltiples se les puede dar el mismo nombre de clase. Todos los elementos de una baraja sencilla con un nombre de clase común se consideran parte de la misma clase.

Los nombres de clase son sensibles al formato de letra, se distingue entre mayúsculas y minúsculas. Un elemento puede pertenecer a varias clases si tiene definidos varios nombres de clases en su atributo *class*.

Si dentro de un atributo incluimos varios nombres de clases, estos se deben separar con un espacio en blanco. Los nombres de clase redundantes, así como espacios en blanco no significativos entre nombres de clase deben eliminarse. El terminal WML debería ignorar este atributo.

Todos los elementos que contengan texto pueden contener el atributo *xml:lang*. Este atributo especifica el lenguaje natural de un elemento o sus atributos.

El atributo indica al terminal del usuario el lenguaje utilizado para texto, tales como un contenido del elemento y valores del atributo, que puede ser mostrado al usuario.

## **Cabecera del documento**

Una baraja de WML válida, es un documento XML válido y por ello debe contener una declaración XML y una declaración de tipo de documento. A continuación se expone un encabezado típico de un programa WML:



---

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

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

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

Se puede guardar la declaración de tipo de documento en un archivo, permitiéndose así una traducción a *bytecodes* más rápida.

Hay que distinguir los siguientes identificadores de documento:

- El identificador público SGML es "-//WAPFORUM//DTD WML 1.1//EN".
- El identificador tipos de medios de difusión de WML es:
  - En forma textual: "text/vnd.wap.wml".
  - En forma de signo: "application/vnd.wap.wmlc".

Estos tipos no están todavía registrados con el IANA y son por tanto tipos de medios de difusión experimentales.





---

## Elemento wml

El elemento *wml* define una baraja y encierra toda la información y las cartas de la baraja.

### Elementos Contenidos

- head ?
- template?
- card +

## Elemento card

Una baraja de WML contiene una colección de cartas. Hay diferentes tipos de cartas, cada una especificando un modo diferente de interacción con el usuario.

El elemento *card* es un contenedor para texto y elementos de entrada, lo suficientemente flexible para permitir la presentación en una amplia variedad de dispositivos, con una gran variedad de características de entrada y de presentación.

El elemento *card* indica la disposición general y los campos de entrada requeridos, mientras da una considerable libertad para la disposición instrumentalizada y los esquemas de entrada del terminal del usuario.



---

Por ejemplo, un card puede ser presentada como una página sencilla en un dispositivo de pantalla grande o como una serie de páginas más pequeñas en un aparato de pantalla pequeña.

Un card puede contener marcas, campos de entrada y elementos que indican la estructura de la carta. Hay que señalar que el orden de los elementos en la carta es significativo. Se puede usar un *id* de una carta como un anclaje de fragmento.

### ***Elementos contenidos***

- *onevent* \*
- *timer* ?
- *do* \*
- *p* \*

Si un elemento card contiene elementos *onevent*, estos deben estar primero. Si un card contiene un elemento *timer*, el timer debe seguir a cualquier elemento *onevent* y preceder a los elementos *do* o elementos *p*.



---

## **Elemento template**

El elemento *template* (patrón), declara un patrón para las cartas de la baraja. Las acciones especificadas en el elemento *template* (por ejemplo, *do* o *onevent*) se aplican a todas las cartas de la baraja.

Especificar un suceso en el elemento *template*, es equivalente a especificarlo en cada carta. De todos modos, un elemento *card* puede supeditar las acciones especificadas en el elemento *template*.

En particular, pueden señalarse las siguientes reglas de supeditación:

- Los elementos *do* especificados en el *template*, puede ser supeditados en cartas individuales si ambos elementos tienen el mismo valor de atributo *name*.
- Los enlaces de sucesos intrínsecos especificados en el *template* pueden ser supeditados por enlaces de sucesos de una carta.

## **Elementos contenidos**

- *do* \*
- *onevent* \*



---

## **El elemento head**

El elemento head contiene información relativa a la baraja como un todo, incluyendo los meta-datos y los elementos de control de acceso.

Se debe incluir uno de los siguientes elementos al menos una vez dentro de un elemento head:

- access
- meta

### ***Elementos contenidos***

- (access | meta) +

## **Elemento access**

El elemento Access, especifica la información del control de acceso para la baraja entera. Hay que advertir que es un error sintáctico en WML, que una baraja contenga más de un elemento access.

Si una baraja no incluye un elemento access, el control de acceso está inhabilitado. Cuando el control de acceso está inhabilitado, las cartas de cualquier baraja pueden acceder a esta baraja.



---

La colocación por omisión del control de acceso permite acceder a cualquiera de las URI (Uniform Resource Identifier) del mismo dominio.

El terminal del usuario utiliza una pareja de prefijos para comparar los URI de las barajas que están intentando acceder a una baraja con valores de atributo que se definen.

## **TÉCNICAS DE PROCESAMIENTO EN SERVIDOR WWW PARA GENERACIÓN DE CONTENIDOS WAP**

Como se puede apreciar, en este mismo documento cuando se analiza el modelo www y el modelo WAP, el modelo general de operación cliente/servidor de WAP es muy similar al usado en el dominio WWW, y, de hecho, aquél ha sido específicamente diseñado para que sea posible aprovechar la infraestructura tecnológica existente en la Web para el aporte de contenidos.

Obsérvese además que en tanto se accede en última instancia a un servidor Web estándar, los contenidos WAP devueltos pueden ser estáticos (tienen existencia previa en un sistema de archivos local o remoto accesible por el servidor) o generados dinámicamente usando tecnologías de servidor suficientemente probadas en la Web, sin que el cliente WAP pueda establecer distinción alguna sobre el mecanismo de producción de contenidos usado realmente (tal como sucedía en los clientes Web).



---

En general, nunca se debería confiar en la efectividad de procesos de filtrado y conversión automática de contenidos Web a formato WAP, debido a que estos filtros poco pueden hacer para adaptar al 100% contenidos que no tienen en cuenta –en origen- las características físicas especiales de los terminales móviles que realmente se van a usar como clientes.

Significa esto que es siempre preferible aportar contenidos directamente en formato WAP (WML, WMLScript..etc).

Por otra parte, y como uno de los puntos más importantes que contribuirán al éxito de WAP, es preciso entender que se trata de una tecnología en la que el usuario final es un usuario registrado y conocido por el operador de red celular.

Esto es esencial en operaciones de comercio electrónico, máxime si se tiene en cuenta que al propio mecanismo de seguridad de las redes GSM se le añade el ofrecido por los servicios de seguridad de WTLS.

Quiere todo esto significar que en el caso de WAP, cobra más importancia que nunca la necesidad de personalizar los contenidos entregados al cliente, y, por tanto, al usuario final. Por ello, el modelo de desarrollo a utilizar, sin duda alguna, será la entrega de contenidos WAP generados dinámicamente en el servidor Web.

Así pues, es necesario conocer cuáles son las técnicas de procesamiento en servidor disponibles en el dominio Web para su utilización en este nuevo ámbito, y conocer cómo se adaptan al mismo.



---

Además, dado que nos encontramos en un entorno donde las redes inalámbricas introducen tiempos de latencia elevados, el rendimiento del mecanismo de generación dinámica es crítico para no añadir más penalizaciones en el tiempo de descarga.

Por todo lo expuesto, a continuación se ofrece una visión general que refleja el "estado del arte" referido a la totalidad de técnicas de procesamiento en el servidor existentes actualmente, contemplando la problemática de su adaptación a la generación de contenidos WAP, y realizando una comparación final de posibilidades de desarrollo y rendimientos.

Actualmente existen cinco mecanismos que permiten a los Servidores Web ofrecer contenidos dinámicos y procesar consultas de datos casi en tiempo real:

### ***SERVER SIDE INCLUDES (SSI)***

Se trata de un mecanismo que permite integrar comandos especiales en documentos (originalmente HTML), de modo que los contenidos de la página resultante sean diferentes cada vez que un cliente acceda a la misma.

En este caso, el Servidor HTTP debe analizar los documentos antes de entregarlos, para poder así interpretar los posibles comandos SSI e incrustar dinámicamente bloques de datos en el flujo de datos correspondiente al documento finalmente enviado al cliente.



---

SSI es una técnica de primera generación, por lo que adolece de bastantes inconvenientes, entre los que destacan el bajo rendimiento y el carácter propietario del conjunto de comandos disponibles.

Esta técnica se utilizaba en el dominio WWW en aquellos casos en los que se deseaba implementar un mínimo comportamiento dinámico en servidores HTTP para sistemas Unix.

Tomando como base este mismo modelo de operación, han surgido técnicas de segunda generación, como es el caso de ASP (Active Server Pages, de Microsoft) y PHP (Personal Home Pages), que ofrecen posibilidades muy superiores en lo que a generación dinámica de contenidos se refiere, y que, en su conjunto, se podrían considerar mecanismos SSI Mejorados.

ASP es hoy día la principal tecnología ofrecida por Microsoft para la generación dinámica de páginas Web, y es directamente reutilizable en el contexto WAP. Los servidores Web Internet Information Server y Personal Web Server ofrecen soporte nativo de páginas ASP.

Básicamente, una Página ASP no es más que un archivo de texto con extensión .ASP que contiene una combinación de códigos de un lenguaje de descripción de contenidos (como HTML o WML) y sentencias en un lenguaje de script para servidores (En este caso concreto, se pueden usar JavaScript y VBScript).

Así las cosas, salvo en lo referente a la mayor flexibilidad que proporciona un lenguaje de scripts, este modelo en sí no parece diferir demasiado del





modelo básico SSI. Sin embargo, la verdadera importancia de esta tecnología radica en el hecho de que ASP permite la integración de componentes COM (*Component Object Model*, Modelo de Objetos para Componentes) en el servidor.

De esta manera, un script ASP avanzado puede limitarse a crear un flujo de control, desde el que se realiza la invocación de Componentes Activos de Servidor que implementan la lógica la negocio y la conectividad con diversas fuentes de datos (locales o remotas).

El propio servidor IIS 4.0 proporciona varios componentes útiles en este sentido.

Veamos ahora algunas notas de interés para la generación de contenidos WML con ejemplos de implementación usando ASP:

A. En primer lugar, se debe identificar en la cabecera HTTP de respuesta el tipo MIME del objeto devuelto:

```
<!--#include file="conn.asp">
<%
Response.ContentType= "text/vnd.wap.wml" %>
```

B. En segundo lugar, la aplicación debe identificar el tipo de documento XML devuelto (recuerde que WML se basa en XML):

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



```
<wml>
```

```
...
```

C. Típicamente, el código ASP puede utilizar el modelo ADO (ActiveX Data Objects) para el acceso a elementos de una base de datos, de modo que se pueda ofrecer una carta WML que contenga, por ejemplo, un elemento SELECT con una lista de opciones:

```

<card id="cartal" title="Seleccione producto">
<%
sqlQuery = "SELECT [Cod_Prod], [nombre] FROM Productos"
set rsProductos = conn.execute(sqlQuery)
%>
<select name='prod'>
<%
do while not rsProductos.eof
response.write("<option value='" & rsProductos("Cod_Prod") & "'>" &
rsProductos("nombre")
rsProductos.MoveNext
loop
%>
</select>

```

D. Se debe evitar el uso del objeto Session, aunque sea necesario repetir consultas al SGBD, pues su funcionamiento depende del uso de cookies, que, pese a formar parte de la especificación, no siempre están soportadas en terminales móviles WAP de primera generación

E. En ocasiones, aún a riesgo de comprometer el principio de independencia del dispositivo contemplado por WML, es preciso realizar ajustes finos en la presentación para terminales específicos:



```
<%  
if InStr(Request.ServerVariables ("HTTP_USER_AGENT"),"MotorolaTimePort") then ...  
%>
```

## **LA INTERFAZ CGI (COMMON GATEWAY INTERFACE)**

La interfaz CGI, profusamente usada en el dominio WWW, posibilita la ejecución de aplicaciones externas en el servidor, cuya misión consiste en procesar la información suministrada por el usuario y generar al vuelo un documento WML (u objeto Wap, en general) que se devuelve al gateway WAP como respuesta, para su reenvío al cliente.

Estas aplicaciones pueden encargarse de forma autónoma de la gestión de los datos (*procesamiento directo*), o bien actuar como pasarela *middleware* hacia otras aplicaciones especializadas (*procesamiento indirecto*).

El principal problema de las aplicaciones CGI es su pobre eficiencia en situaciones de carga elevada en el servidor.

En efecto, cada vez que un cliente Web (el gateway WAP) hace una referencia a un programa CGI, se crea un proceso totalmente nuevo en el servidor para su ejecución: el Sistema Operativo debe inicializar un espacio de direcciones, leer el ejecutable de disco, cargar librerías dinámicas, iniciar la ejecución del programa, inicializar sus variables, y, finalmente, leer y procesar los datos enviados por el cliente Web.

Si, además, el programa CGI está codificado en un lenguaje interpretado (por ejemplo, Perl), la situación se agrava aún más, ya que a todo lo anterior se



ha de añadir una importante penalización de tiempo debida a la carga del intérprete y al proceso de análisis léxico y sintáctico del archivo fuente.

Para la codificación de aplicaciones Wap basadas en CGI, existen librerías de código Perl que simplifican enormemente el desarrollo al ofrecer funciones específicas para generar contenidos Wap. Por ejemplo en la librería `wmlib.pl`, la función `&PrintHeader` genera la siguiente salida WML:

```
Context-type:                application/x-wap.wml\n\n
<?xml                        version="1.0"?>
<!DOCTYPE WML PUBLIC "-//WAPFORUM//DTD WML 1.0//EN"
"http://www.wapforum.org/DTD/wml.xml">
```

Como se puede observar, en el ejemplo, lo primero que se hace es lo que toda aplicación de servidor tiene por obligación: identificar el tipo MIME devuelto.

En realidad este tipo se entrega al gateway, por lo que cabe usar tipos MIME propietarios para activar funciones específicas del gateway.

Por ejemplo, en el caso del servidor UP.Link, se utiliza el tipo `multipart/x-up-fax` para activar el envío de los contenidos a un fax designado por el usuario/abonado.

### ***INTERFACES DE PROGRAMAS DE APLICACIÓN (APIs) PROPIETARIAS***

En este caso se trata de desarrollar aplicaciones de funciones similares a las de los programas CGI, pero accediendo directamente al código nativo de



---

los servidores. De este modo se incrementa el rendimiento de las aplicaciones y se reduce considerablemente la carga de los servidores.

Las APIs de referencia son las desarrolladas por Microsoft (ISAPI) y por Netscape (NSAPI), y son conceptualmente similares. Las aplicaciones xSAPI se implementan como librerías dinámicas (DLLs), con lo que se evita la creación de un proceso específico en el servidor para atender cada una de las peticiones de los clientes Web.

El uso de APIs en el servidor reduce notablemente la sobrecarga asociada a la ejecución de CGIs, pero no sin un coste: el código de una aplicación WAP implementada con xSAPI está directamente enlazado con el del servidor, por lo que cualquier bug podría, potencialmente, afectar al comportamiento del propio servidor. Por otra parte, no hay que olvidar que el código creado es propietario para cada servidor, por lo que se pierde la portabilidad de las aplicaciones WAP.

### ***INTERFAZ CGI ASÍNCRONA (FASTCGI)***

Se trata de una técnica reciente, y aún en evolución en el dominio WWW, que trata de combinar las ventajas de la interfaz CGI (simplicidad, aislamiento de procesos, solución estándar independiente del lenguaje y de la plataforma) con las derivadas de la utilización de APIs en el servidor (eficiencia y extensiones otras operaciones en el servidor).

La interfaz FastCGI es conceptualmente similar a la interfaz CGI, con dos importantes diferencias:



---

a. Los procesos FastCGI son persistentes: no se destruyen tras finalizar el procesamiento de una petición, sino que entran en estado de espera de nuevas peticiones.

b. El protocolo FastCGI multiplexa la información de entorno, la entrada estándar, la salida estándar y la salida de errores usando una única conexión bidireccional. En el caso de ejecución local se usa un pipe full-duplex, y en el caso de ejecución remota, el servidor utiliza una conexión TCP.

## **SERVLETS**

Los servlets son programas codificados en Java que se ejecutan, como alternativa directa a CGI, en servidores HTTP especialmente diseñados para ello.

Se trata, pues, de componentes de servidor, independientes de la plataforma, codificados en Java, y que permiten modificar dinámicamente las funciones del servidor.

Así, los servlets ofrecen un entorno de desarrollo general para la creación de servicios basados en el paradigma solicitud/respuesta.

La utilización de servlets constituye una de las últimas tendencias en la programación de aplicaciones Web para servidores, y tienen una aceptación que crece día a día, y en consecuencia, son candidatos idóneos para el desarrollo de soluciones WAP en servidores HTTP.



---

Existe una única máquina virtual Java ejecutándose en el servidor HTTP, y los servlets sólo se cargan una vez, bajo demanda.

Un servlet no se vuelve a cargar mientras no sufra ninguna modificación, y aún siendo éste el caso, no es necesario reiniciar el servidor.

Además, dado el carácter estándar del propio lenguaje Java, los servlets son directamente portables entre plataformas (independencia del Sistema Operativo y del tipo de servidor, en tanto en cuanto ofrezca soporte de Servlets)

A nivel de rendimiento, una de las principales características de los servlets es el hecho de que no precisan de la creación de un nuevo proceso para cada petición.

En la mayoría de los sistemas, los servlets se ejecutarán en paralelo, dentro del mismo proceso del servidor. En estos casos, los servlets tienen una gran ventaja de rendimiento frente a las aplicaciones CGI e incluso aplicaciones FastCGI. Además, puesto que los servlets residen en memoria, posibilitan la compartición de información estática o persistente entre invocaciones.

Ya en el apartado de características avanzadas, un servlet puede reenviar solicitudes a otros servidores, lo que permite equilibrar la carga (*load balancing*) o utilizar servidores específicos para la generación de determinados contenidos (*service request dispatching*).



---

## **DESARROLLO DE APLICACIONES WAP**

Durante el segundo informe parcial, se analizaron las diferentes tipos de herramientas existentes, destinadas al desarrollo de aplicaciones para dispositivos inalámbricos, además, se analizaron los tipos de dispositivos inalámbricos existentes que posibilitan el acceso a los contenidos WAP.

Debido a que en ese informe se especificaron y desarrollaron las características generales y particulares de cada tipo distinto de herramientas y dispositivo, es que en este se hará referencia a los aspectos no abarcados por el segundo informe parcial.

### **CONSIDERACIONES GENERALES**

Antes de poder determinar los aspectos y software necesarios para desarrollar aplicaciones para un dispositivo inalámbrico, debemos enfocar nuestra atención en los pasos que sigue una conexión inalámbrica:

- 1- El usuario selecciona en su terminal WAP una dirección, o sea un URL.
- 2- El terminal móvil WAP envía esta petición del URL al Gateway WAP utilizando el protocolo WAP.
- 3- El gateway convierte esta petición WML y/o WMLScript WAP, a WML y/o WMLScript HTTP convencional y envía la petición de búsqueda del URL al servidor web.





- 4- El servidor web evalúa la petición HTTP y determina que tipo de petición es; si el URL es un simple archivo, el servidor web le adjuntará un encabezado HTTP y ya estará en web, y si la petición es una aplicación Script o un CGI, el servidor web abrirá la aplicación.
- 5- El servidor web devolverá una respuesta en WML, WMLScript con encabezado HTTP.
- 6- El gateway verificará el encabezado HTTP y los contenidos WML, así como las aplicaciones Script las convertirá en forma binaria, creando una respuesta para el terminal WAP en WML y/o WMLScript con encabezado WAP y lo enviará al terminal móvil.
- 7- El terminal WAP recibirá la respuesta con el protocolo WAP, evaluará los contenidos WML y/o WMLScript y los visualizará según la configuración del terminal.

De acuerdo a esto podemos ver que, en una petición WAP, básicamente interactúan tres elementos los cuales son un teléfono WAP, un GATEWAY WAP y un servidor web.

### ***ESTABLECIMIENTO WAP***

Para desarrollar productos para WAP, existe una amplia variedad de diferentes productos de software disponibles que facilitan el desarrollo y alojamiento de aplicaciones WAP.

Entre los productos disponibles encontramos:



---

## **Navegadores WAP (WAP Browsers)**

El navegador WAP funciona en los dispositivos WAP y muestran el contenido que reciben. Este también provee la interfaz donde el usuario puede navegar la aplicación WAP.

El navegador puede ser construido dentro de un teléfono o un dispositivo móvil, o dentro de las tarjetas SIM que contienen los dispositivos. Los navegadores WAP también se encuentran disponibles directamente en PC o a través de un navegador Web a través de un emulador.

Frecuentemente un navegador WAP para PC se incluye en el software SDK (Software Development Kit o Equipo de desarrollo de software) disponibles desde varias compañías y pueden ser una herramienta importante para realizar las pruebas del software, particularmente si emula un dispositivo real.

De alguna manera esto puede ser más fácil que utilizar un dispositivo real, permitiéndole probar su aplicación durante todo su desarrollo. Sin embargo, es importante recordar que el mejor emulador no garantiza la emulación exacta del dispositivo de mano, ya que estos emuladores son simplemente una emulación de cómo los dispositivos de mano trabajan.

Para garantizar las pruebas de la aplicación, es necesario finalizar las pruebas en el dispositivo que se utilizará, de esta manera se puede asegurar que la aplicación trabajará utilizando la configuración exacta del usuario.



---

Actualmente es bastante complicado, si no imposible, proveer una única página WML cuya presentación sea adecuada en todos los navegadores.

Mientras que esto debe ser mejorado en futuros desarrollos, las implementaciones actuales tendrán que ser provistas teniendo en cuenta estas restricciones.

Existen distintas maneras para tratar el problema, pero en general pueden ser divididos dentro de soluciones estáticas y dinámicas. Una manera estática de proporcionar el contenido para los diferentes navegadores es proveer archivos separados para cada uno.

Esto será posible cuando la aplicación se encuentra pensada para un limitado número de dispositivos, pero con el rápido crecimiento del mercado de los dispositivos WAP, esta solución no es conveniente ya que se pondrá inmanejable rápidamente.

Anticipadamente vemos que como el contenido HTML, y en especial con la experiencia y los recursos disponibles, el mayor contenido WML debe ser proveído dinámicamente. Para generar el contenido apropiado en el servidor, se puede manejar una variedad de dispositivos diferentes.

A fin de generar dinámicamente una aplicación para un dispositivo específico, se debe analizar la cabecera de la petición hecha por el dispositivo, para encontrar el nombre del navegador que utiliza dicho dispositivo.



---

Dentro de Internet existen varias fuentes que nos permiten obtener información sobre las distintas capacidades de los navegadores utilizados.

Los navegadores incorporados dentro de los dispositivos reales poseen distintas características. Por ejemplo el dispositivo Nokia 7110 fue el primero en concentrar el mercado y su navegador es muy limitado.

Este navegador implementa las especificaciones mínimas a fin de ser considerado un dispositivo WAP, y en algunas áreas no se consideran todas las especificaciones requeridas para ser un dispositivo WAP. Sin embargo, los teléfonos que utilizan el navegador de Phone.com tienen más capacidades no especificadas.

Desafortunadamente, en lugar de llevar a cabo todas las especificaciones de WAP, Phone.com ha optado por implementar sus propias extensiones que no forman parte de las especificaciones de WAP 1.0. Estas extensiones están causando problemas de incompatibilidad para los desarrolladores.

## **WAP Gateways**

Básicamente el concepto de un Gateway WAP es una pieza de software que actúa como un traductor entre Internet y un dispositivo móvil, permitiendo la comunicación entre ambos.

Este software se coloca entre una red que soporta WAP y una red de paquetes IP, como es Internet. Este actúa como un intermediario que convierte



---

entre los protocolos de la red de paquetes y los protocolos en la red WAP (WSP, WTP, WTLS y WDP).

Un Gateway WAP puede ser implementado como un simple ordenador o un grupo de servidores para equilibrio de carga. Sin embargo, independientemente de la implementación, este puede ser considerado, desde la perspectiva del usuario móvil, como una simple caja.

Dentro de las funciones del Gateway WAP encontramos:

- Implementación de las capas de la pila del protocolo WAP
- Control de Acceso
- Conversión de los protocolos WSP ↔ http
- Resolución de Nombres de dominio
- Conversión de HTML ↔ WML
- Codificación del contenido WML
- Compilación de WMLScript
- Seguridad
- Provee una zona de almacenamiento para el contenido frecuentemente accedido.



---

## **Servidor de Aplicaciones WAP**

Es la parte más importante de una aplicación WAP, es donde y como va ha ser almacenada. Es muy facial realizarlo en un servidor web estándar. Después del Gateway, el navegador realiza el requerimiento en el dominio de Internet, y la salida es compilada en XML. Un servidor estándar de web puede proporcionar este resultado.

Atendiendo al plano en el que nos movemos podemos tener diferentes nociones de lo que es un servidor web:

### **Informático:**

Computadora conectada a una red que pone sus recursos a disposición del resto de los integrantes de la red. Suele utilizarse para mantener datos centralizados o para gestionar recursos compartidos. Internet es en último término un conjunto de servidores que proporcionan servicios de transferencia de ficheros, correo electrónico o páginas WEB, entre otros.

### **Empresarial o corporativo:**

Es el conjunto de información, servicios y herramientas que definen la presencia de la empresa en Internet. Es un medio de comunicación con sus clientes y prospectos. También puede convertirse en un canal comercial a través del cual proporcionar productos y servicios.

### **Tecnológico:**



---

Es un conjunto de páginas HTML ("Hypertext Markup Language") cuyos contenidos están relacionados entre si mediante Enlaces (hiperenlaces o vínculos). Estas páginas son almacenadas en un equipo servidor, gestionadas por un software especializado que atiende las peticiones de los usuarios y envía las páginas solicitadas.

Los servidores Web permiten servir contenido sobre Internet utilizando HTML (Hyper Text Markup Language). El servidor Web acepta los requerimientos de los navegadores como Netscape o Internet Explorer y se encargan de retornarles los documentos apropiados de HTML.

Varios servidores utilizan las tecnologías del ide para aumentar su poder y llevarlo más allá de su habilidad de entregar las páginas de HTML normales; éstos incluyen las escrituras de CGI, seguridad por medio de SSL, y los ASPs (Active Server Pages).

Como vemos un servidor Web se encuentra preparado para servir documentos en un formato HTML, por lo tanto debemos especificarle que debe servir páginas \*.wml como páginas WAP, de lo contrario el servidor responderá como que el tipo de fichero es de texto.

Para especificarle al servidor que las páginas de extensión wml debe servir las con páginas WAP, lo que se necesita hacer es agregar los siguientes tipos de MIME a los que el servidor ya conoce:



Contenido	Tipo MIME	Extensión
Código WML	text/vnd.wap.wml	.wml
WML compilado	image/vnd.wap.wbmp	wbmp
WML Script	text/vnd.wap.wmlscript	.wmls
WML Sript compilado	application/vnd.wap.wmlc	.wmlc
Wireless Bitmap	application/vnd.wap.wmlscriptc	.wmlsc

MIME (Multipurpose Internet Mail Extencions) es una especificación del formato de los datos que pueden ser enviados sobre internet. Este formato es originalmente diseñado para incluir imágenes, sonidos, animaciones y otro tipo de documentos sin mensajes de mail de Internet.

Un MIME Type es del formato type / subtype . El World Wide Web usa mayormente los siguientes types:

- Text
- Image
- Audio
- video





- y application (aquí existen otros dos tipos como son multiparte y mensaje que casi nunca son usados.

El subtype especifica un formato particular del archivo, por ejemplo el MIME type de un archivo HTML es text/html, en caso similar el MIME type de los archivos de imagen GIF es image/gif .

Cuando el servidor envía datos en respuesta a un requerimiento recibido, este envía un MIME type con la respuesta. Este MIME type además puede ser colocado explícitamente por la aplicación.

Normalmente, la extensión del archivo de un archivo requerido es asociada con un MIME type y así el servidor automáticamente publica el correcto MIME type.

Entonces, cuando el navegador recibe información del servidor, este verifica su MIME type para saber que hacer con esta. Si, por ejemplo, los datos recibidos contienen un MIME type de "image/vnd.wap.wbmp", entonces conoce que debe mostrar una imagen.

Los MIME types aceptados por el navegador pueden también ser descubiertos dinámicamente por el servidor, utilizando información incluida en los requerimientos de datos. Una aplicación puede colocar el MIME type explícitamente, determinando el MIME type soportado por el dispositivo.



---

Como el contenido HTML estático, el contenido estático WML posee un limitado ámbito para proporcionar servicios. Para las aplicaciones más complejas, la generación de páginas dinámicas es una necesidad.

Esto esencialmente significa desarrollar una aplicación que genere páginas dinámicamente desde una base de datos, la cual tuviera un mantenimiento mínimo.

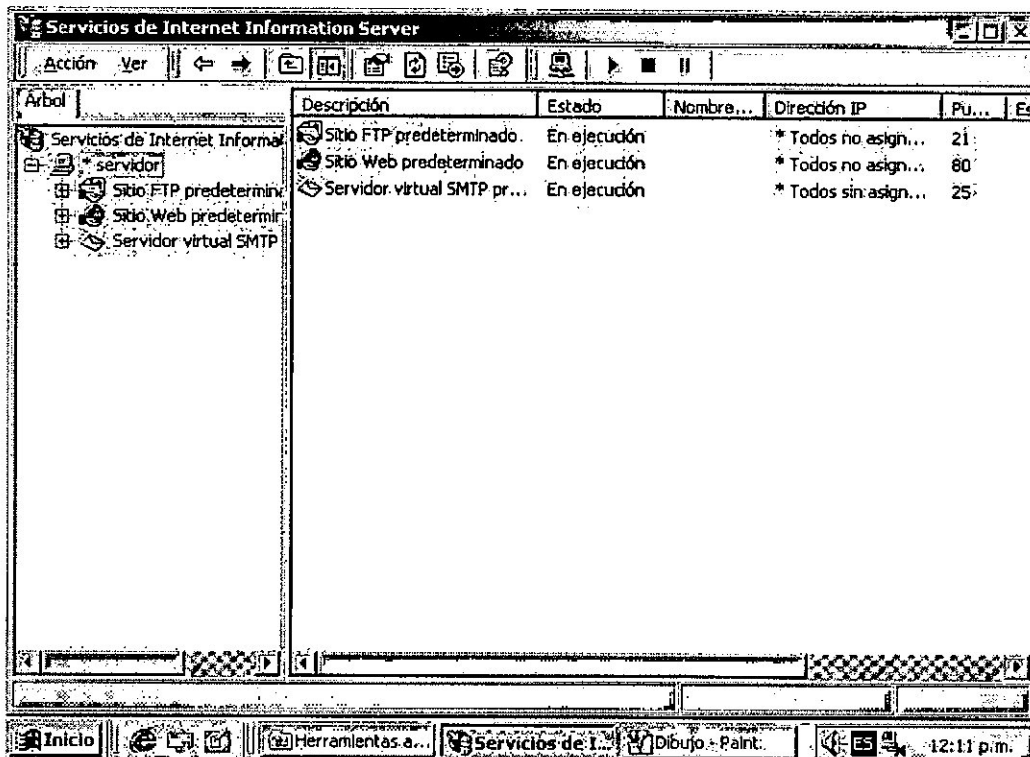
Estas aplicaciones son hospedadas típicamente en un servidor de aplicaciones. Para incluir el soporte WAP, lo que se necesita es configurar el servidor de aplicación para que utilice los MIME types anteriores.

### ***Configuración MIME types En un servidor IIS de Microsoft:***

IIS 5.0 es proveído básicamente con Windows 2000 y IIS 4.0 es un servicio opcional en Windows NT 4.0.

Con el propósito de configurar los MIME types para IIS, se necesita utilizar el Servicio de Información de Internet agregado dentro del MMC (Microsoft Management Console). Dicha consola puede abrirse desde el menú de Inicio bajo la carpeta Herramientas de Administración dentro del menú de Programas, o simplemente desde el Panel de Control:

- En la consola del servidor, abrir Management console
- Desde aquí se pueden definir los caracteres MIME para que sean válidos tanto para todo el servidor como para directorios separados



Pantalla de Configuración del Servidor IIS de Microsoft

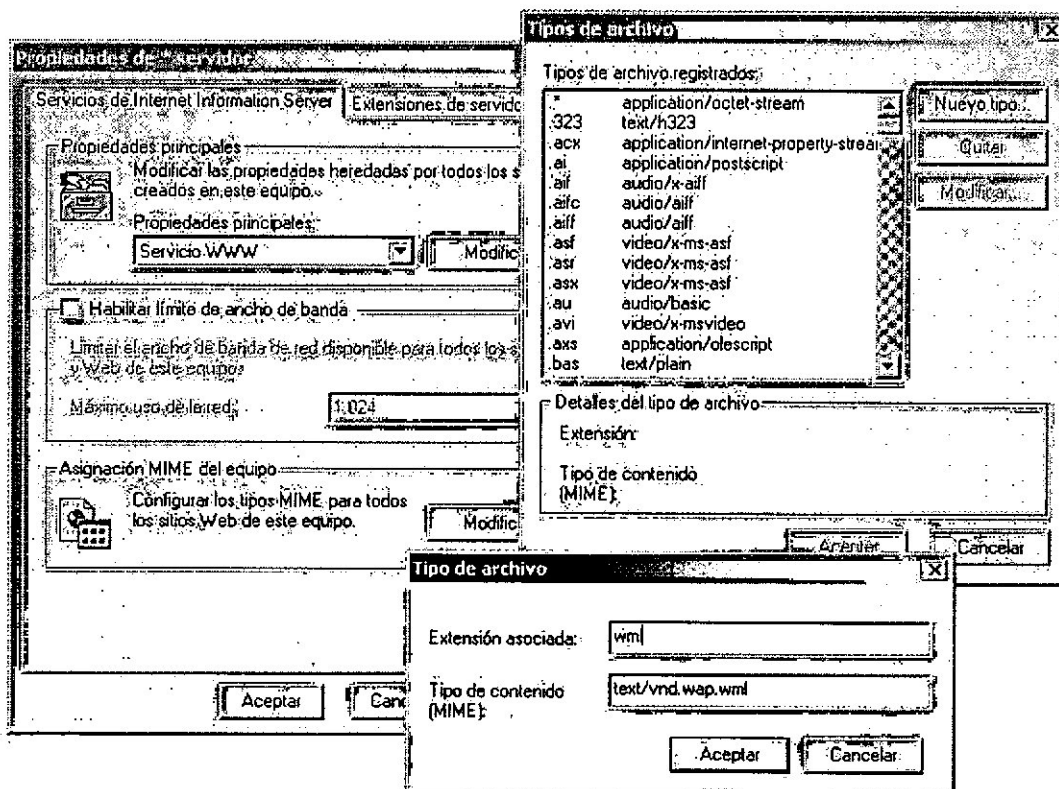


Figura 8.: Configuración MIME Types en un Servidor IIS de Microsoft



- 
- Para añadir el carácter MIME a un directorio: pulsar el botón derecho del ratón sobre el directorio en el que se quiera añadir el carácter MIME
  - Seleccionar la etiqueta HTTP headers
  - Pulsar en el botón File types abajo a la derecha
  - Pulsar en New Type y suministrar la clase de extensión y contenido
  - Añadir un carácter MIME a un servidor completo: pulsa con el botón derecho del ratón en el servidor
  - Pulsar en el botón File types y seguir las instrucciones de arriba
  - Pulsar en New type y definir el tipo de extensión y contenido.

### ***Servidor Apache***

Apache es un servidor de aplicación de dominio público el cual está originalmente escrito para UNIX, pero actualmente se encuentra disponible para otras aplicaciones.

Un camino para agregar los MIME type es editar el archivo srm.conf (normalmente en /etc/http/conf), luego localizar la sección AddType y añadir lo siguiente al archivo (con un editor de texto normal):



- 
- AddType text/vnd.wap.wml. .wml
  - AddType image/vnd.wap.wbmp. .wbmp
  - AddType text/vnd.wap.wmlscript .wmls
  - AddType application/vnd.wap.wmlscriptc .wmlsc
  - AddType application/vnd.wap.wmlc .wmlc

Guardar el archivo y reiniciar.

Otro camino para poder configurar los MIME type en un servidor Apache, incluso cuando el acceso al servidor se deba realizar por medio de una cuenta FTP (File Transfer Protocol), es agregar un archivo .htaccess dentro de cada directorio utilizado para almacenar archivos WML.

El archivo .htaccess puede ser ubicado en uno o varios de los directorios de un servidor virtual. Generalmente este archivo tiene como función controlar el acceso a las paginas ubicadas en un directorio, aunque también como en este caso se puede usar para agregar MIME types.

## Recursos WAP

Hasta el momento, WAP es relativamente una nueva tecnología, pero está causando ciertamente un considerable entusiasmo tanto en la industria de las telecomunicaciones como en la industria de las Tecnologías de Información.

Como esta tecnología se encuentra en continuo desarrollo, nombraremos los sitios con recursos más relevantes dentro de Internet:



- 
- Ericsson, Nokia y Pone.com, todos tienen buenas áreas de desarrollo, pero cada una posee distintos recursos independientes para desarrollo. El Foro de Wap es el sitio oficial donde se encuentran las especificaciones y las últimas noticias sobre WAP, y se encuentran disponibles en <http://www.wapforum.com>
  - AnywhereYouGo.com asegura ser "100% desarrollo de aplicaciones inalámbricas y despliegue" y ya ha estado corriendo con éxito durante algún tiempo bajo su nombre anterior de Waptastic. Su sección de WAP es de gran amplitud, incluyendo abundante información para desarrolladores, así como también vínculos hacia las últimas noticias, se encuentra disponible en <http://www.AnywhereYouGo.com>
  - Otro sitio muy popular de recursos WAP es Gelon.net. Este sitio posee foros de discusión, una sección del diseñador y un juego más comprensivo de vínculos disponible a otros recursos en el Web. El sitio se puede encontrar en <http://www.gelon.net>
  - Yourwap.com es otro sitio popular, el cual es el más orientado al cliente. Este incluye una combinación de WAP y navegador web, hasta ahora el más fiable. Este sitio puede encontrarse en <http://www.yourwap.com>
  - Existe también una biblioteca de archivos WBMP (Gíreles Bitmap) disponibles en <http://www.hicon.nl>



- 
- A un sitio bueno para conseguir un contacto con desarrolladores y discusiones relacionadas con los problemas de WAP es el WAP Group, que puede ser encontrado en <http://www.thewapgroup.com>

## **Herramientas para Desarrollo WAP**

El desarrollo de un sitio WAP puede ser realizado en un simple editor de texto, el uso de las herramientas de desarrollo permiten disminuir el tiempo de desarrollo significativamente.

Se verán dos tipos de herramientas de desarrollo SDKs (Software Development Kits) o ADKs (Application Development Kits). La mayoría de estas herramientas de desarrollo incluyen un emulador que rápidamente simula la salida del código.

Esto es importante para tener acceso al emulador o al simulador para todos los tipos de dispositivos que un usuario puede tener. Estos dispositivos varían en el tamaño de la pantalla y en el panel de control.

Esta es la razón por la que estas herramientas de desarrollo, son creadas para que tenga el potencial de emular diferentes configuraciones.

Durante el segundo informe parcial, se relevaron, analizaron y describieron las herramientas de desarrollo de utilización masiva, por lo que en este informe, solo se describirá la herramienta utilizada para la realización de las pruebas pilotos llevadas a cabo.



---

## **Nokia Mobile Internet Toolkit 4.0**

Es un de herramientas proporcionado por Nokia, destinada al desarrollo de aplicaciones móviles sin la necesidad de un teléfono celular o acceso a la infraestructura del portador.

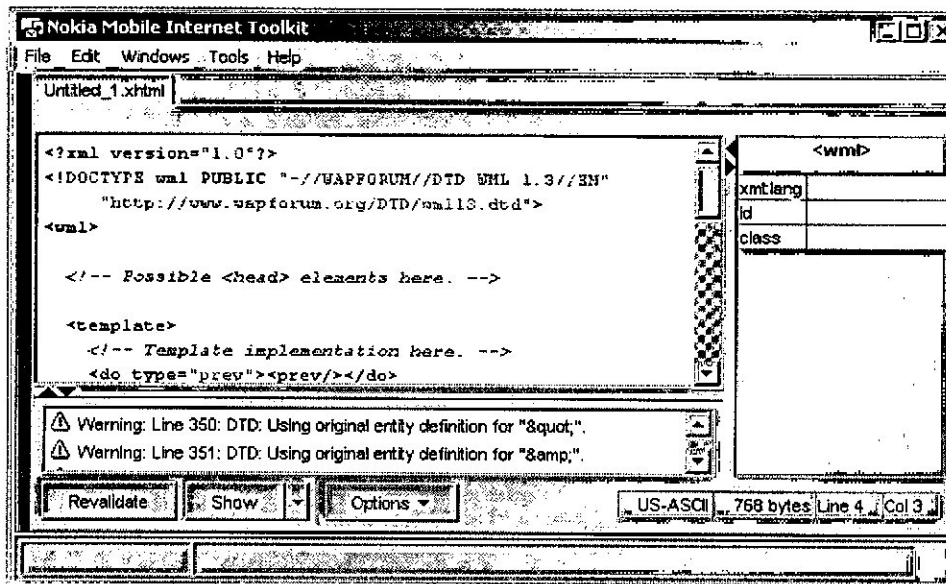
Este conjunto de herramientas viene con un emulador de referencia que soporta completamente las especificaciones WAP 2.0, incluyendo la próxima generación de tecnologías como XHTML y CSS.

Además, se encuentran disponibles en forma separada un conjunto de simuladores de teléfonos Nokia. Estos simuladores están basados en los dispositivos Nokia comercialmente disponibles, y posibilitan la visualización previa de las diferentes aplicaciones, incluyendo aquellos basados en WAP y MMS.

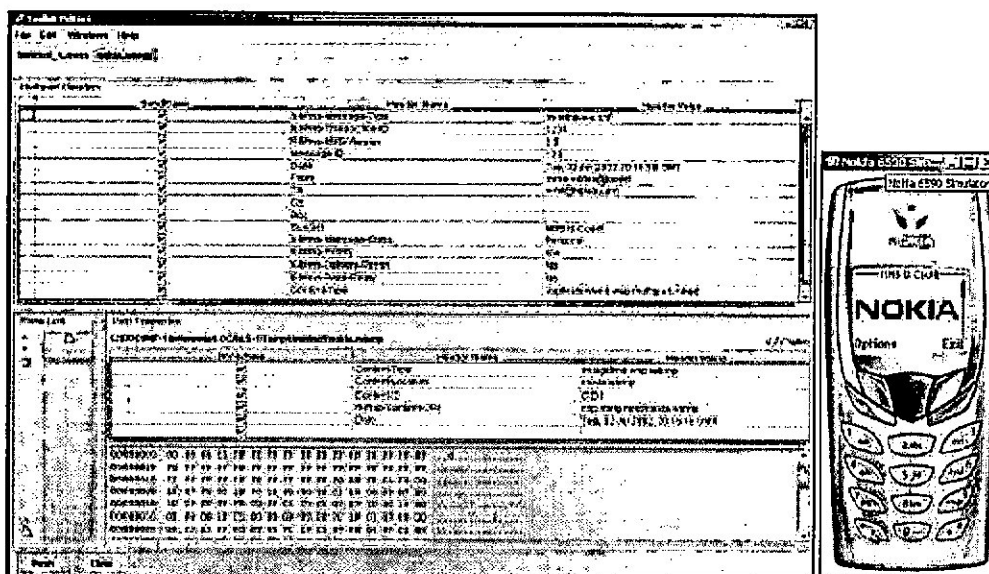
La prueba y demostración de las aplicaciones de internet para dispositivos móvil, son simples y confiables desde que el desarrollador puede navegar y solicitar las URLs desde cualquier servidor web dentro de Internet, o utilizar una versión limitada de Nokia Active Server, construido dentro del conjunto de herramientas.

Además, las páginas pueden ser almacenadas y cargadas directamente desde el archivo de sistema del PC.

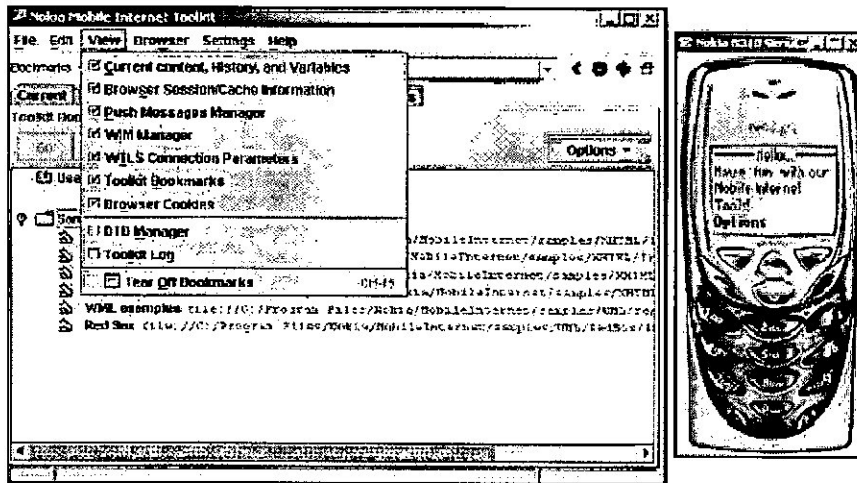




Editor Nokia Mobile Internet Toolkit



Editor Mensajes Multimedia Nokia Mobile Internet Toolkit



**Aspectos Claves**

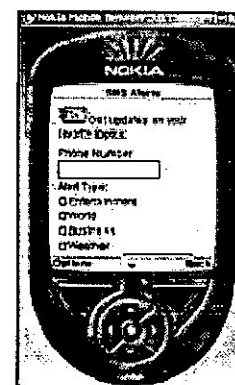
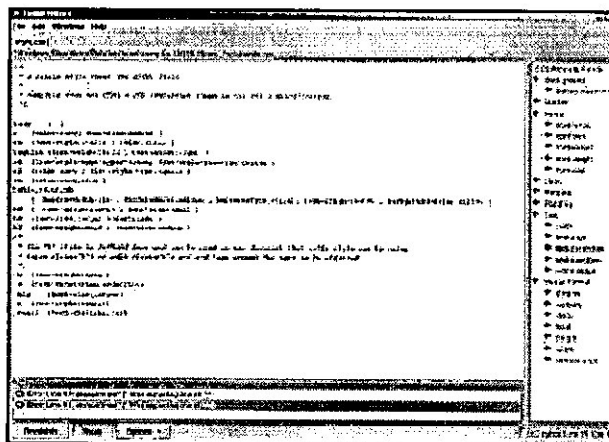
- El perfil XHTML móvil y el editor WAP CSS, facilitan la creación de servicios con interfaces gráfica de usuario a través de una variedad de plataformas.
- El soporte MMS habilita la creación, prueba y la visualización previa de mensajes MMS, incluyendo la habilidad de configurar un archivo SMIL y modificar la cabecera de los mensajes y el contenido.
- Funcionalidades extendidas con la adhesión a las especificaciones wap 2.0
- La referencia del emulador del navegador móvil Nokia, posee una pantalla ajustable, posibilitando la simulación del tamaño de la pantalla del dispositivo o del prototipo.

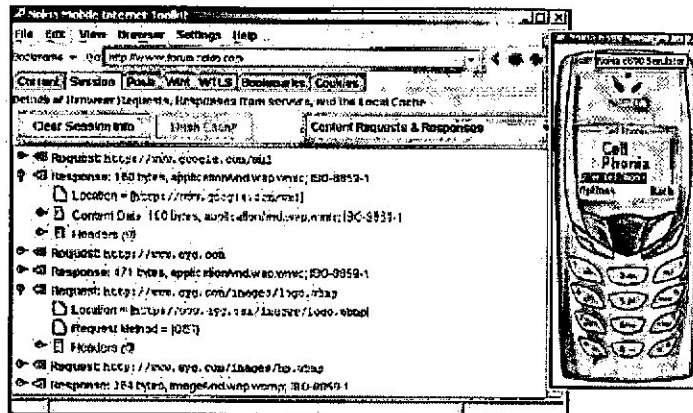


- 
- Los emuladores de los teléfonos móviles pueden ser descargados separadamente desde las herramientas, a fin de que los nuevos emuladores de los dispositivos puedan ser agregados cuando se encuentren disponibles.
  - Los editores validan dinámicamente el contenido de acuerdo al DTD (Documento de Tipo de Definiciones) con mensajes de error.
  - Un administrador DTD facilita la importación de DTDs adicionales para usar en los editores.
  - Una vista a los cookie permite la habilitación y des habilitación de estos, el despliegue de información detallada tal como la expiración y el dominio, así como también el control del flujo de la lista de cookies.
  - EL soporte de impresión le permite a los usuarios imprimir directamente el texto desde el editor.
  - La funcionalidad WTAI hace posible la creación de aplicaciones que agregan funcionalidades telefónicas a las páginas de WAP:
    - Hacer una llamada directamente desde un servicio.
    - Agregar una entrada dentro de guía del teléfono.
    - Enviar tono de discado durante la llamada.



- Los desarrolladores pueden desarrollar y verificar aplicaciones seguras utilizando WTLS (Clase III) y WIM (Gíreles Identity Module). La funcionalidad WIM permite el acceso al hardware basado en WIM (como llevado a cabo en una tarjeta inteligente externa o una tarjeta SIM) así como también un software emulador de WIM (llamado softID) para dispositivos que no poseen el soporte de hardware.
- Un editor WMLScript permite la creación de páginas WAP Interactivas.
- El soporte Push permite el envío de datos automáticamente a los teléfonos móviles a intervalos regulares, como la actualización de noticias cada hora, o cuando se provoca un evento.
- Los servicios basados en la personalización y la localización pueden ser creados utilizando MSISDN, localización GSM y editor de localización de cabeceras GPS.

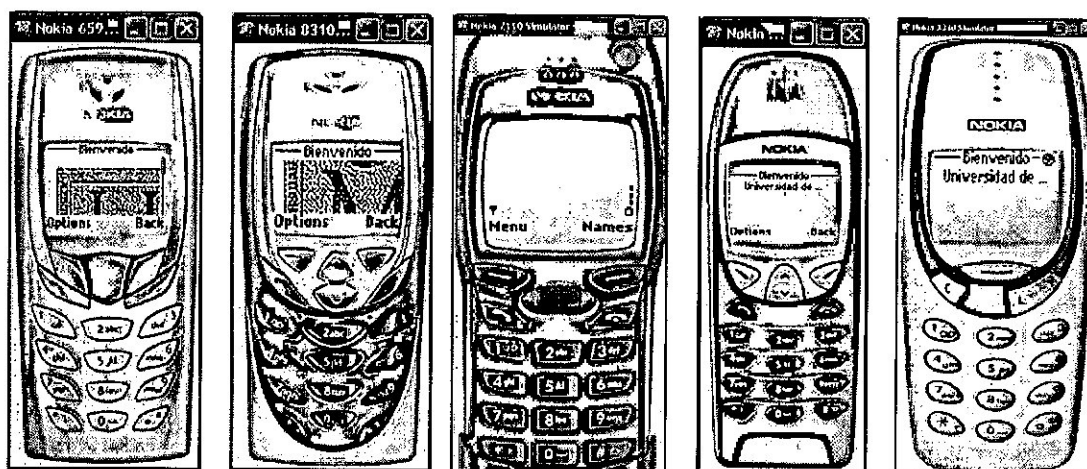




### Componentes

- Perfil móvil XHTML, Editores WAP CSS, WML, WMLScript, WBMP y Push.
- Soporte para WTLS (Clase III) y WIM.
- Editor de cabeceras para MSISDN, Localización GPS y GSM.
- Compilador WAP 2.0, Navegador Nokia Mobile 3.0.1, emulador de Aplicación de Referencias basado en el apoyo del perfil móvil XHTML, WAP CSS, CHTML y WML 1.x, adicionalmente un ajuste de pantalla.
- Disponibilidad de descarga de los simuladores de teléfonos:
  - Nokia 6590 (para XHTML y MMS)
  - Nokia 8310/8390
  - Nokia 3330/3395

- Nokia 6210
- Nokia 7110
- Herramienta de corrección de errores
- Ejemplos de aplicación y Java Servlets



*Figura 9.: Emuladores de Teléfonos Disponibles en el Mercado*

### **Disponibilidad y Precios**

El Nokia Mobile Internet Toolkit, los simuladores de los teléfonos y la documentación se encuentran disponibles en ambos foros de Nokia (<http://forum.nokia.com>) y (<http://americas.forum.nokia.com>). El conjunto de herramientas y la documentación son ofrecidas gratuitamente a cambio de una registración en el foro de desarrolladores de Nokia.



---

## Requerimientos del Sistema

- Instalación de Java Runtime Environment (JRE) 1.3.1
- Procesador Pentium 300Mhz o mejor (recomendado 350Mhz)
- 128 Mb RAM (Recomendado 256Mb)
- Windows NT 4.0 con SP6, Windows 98/2000 (2000W/SP2), Windows XP
- 64k de color y resolución de 800x600
- 50Mb de espacio libre en disco (recomendado 70Mb)
- Mouse y teclado.

## SERVICIO DE MENSAJES CORTOS – SMS

Durante el Tercer Informe Parcial, se describieron completamente las características y aplicaciones de este servicio, por lo que en este informe se realizará una breve referencia.

### ***¿QUÉ ES SMS (SERVIVIO DE MENSAJES CORTOS?).***

SMS significa Servicio de mensajes cortos. Es una tecnología que permite el envío y recepción de mensajes entre teléfonos celulares. SMS apareció por primera vez en Europa en 1992.



---

Tal como el sugiere el nombre de "Servicio de mensajes cortos", los datos que pueden ser enviados por un mensaje SMS es limitado, lo cual es muy útil para aplicaciones de control industrial a distancia.

Una gran ventaja del SMS es que lo podemos encontrar en todos los teléfonos celulares GSM. Casi todos los planes de suscripciones proporcionados por las compañías incluyen servicio de mensajería SMS y son de bajo costo.

A diferencia de los SMS, las tecnologías móviles como WAP y Java móviles no son compatibles en muchos modelos de teléfonos celulares.

SMS fue diseñado originariamente como parte del estándar de telefonía móvil digital GSM, pero en la actualidad está disponible en una amplia variedad de redes, incluyendo las redes 3G.

### ***ADQUISICIÓN POR EL USUARIO***

¿Porqué los mensajes SMS tienen tanto éxito en todo el mundo?

SMS es un éxito en todo el mundo. El número de mensajes SMS intercambiados todos los días es enorme. La mensajería SMS es hoy una de las fuentes de ingreso más importantes de las prestadoras de servicios de telefonía móvil.





---

## ***IMPORTANCIA DEL SERVICIO***

¿Qué es tan especial en los mensajes de SMS que lo hace tan popular en todo el mundo?

### **Algunas de las razones son:**

- Los mensajes SMS pueden ser enviados y leídos a cualquier hora.
- Hoy en día, casi todas las personas tienen un teléfono celular y que llevan la mayor parte del tiempo.
- Con un teléfono móvil, puede enviar y leer mensajes SMS en cualquier momento, no importa que esté en su oficina, en un ómnibus o en el hogar.
- Puede recibir datos o alarmas en cualquier momento y lugar.

## ***FORMA DE FUNCIONAMIENTO***

Tras el envío de un mensaje, este no es enviado directamente al destinatario sino para un centro de mensajes, que lo almacena y envía posteriormente.

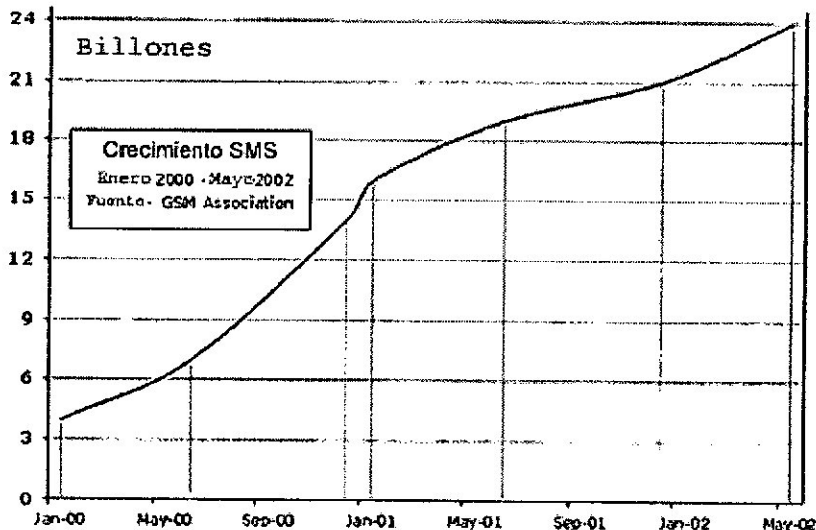
Este centro hace también la cobranza posterior del servicio. El centro de mensajes reencamina después el mensaje para el destinatario, cuando el móvil esté ligado a la red.



De esta manera y al contrario de los servicios de "pager" es posible tener la certeza que el mensaje llegó a su destino, porque el centro de mensajes puede notificar el remitente en caso de que la operación falle.

Mientras que la voz, faxes e Internet son enviados por un canal de radio, los mensajes son enviados utilizando otra frecuencia. Esto garantiza que, en caso de existir una sobrecarga en la utilización de la red (que provoca que los usuarios no consigan línea para llamar), los usuarios del SMS poco serán afectados.

### **CRECIMIENTO DE LA UTILIZACIÓN DE SMS**



Más que un servicio el SMS es actualmente el modelo de negocio que posee las previsiones de crecimiento más elevadas a corto plazo dentro del sector de las comunicaciones móviles inalámbricas.



---

Abreviatura de la expresión inglesa «Short Message Service» (Servicio de Mensajes Cortos), el SMS se limitaba originalmente a la posibilidad de enviar a través de las redes GSM mensajes de texto con una extensión de hasta 160 caracteres.

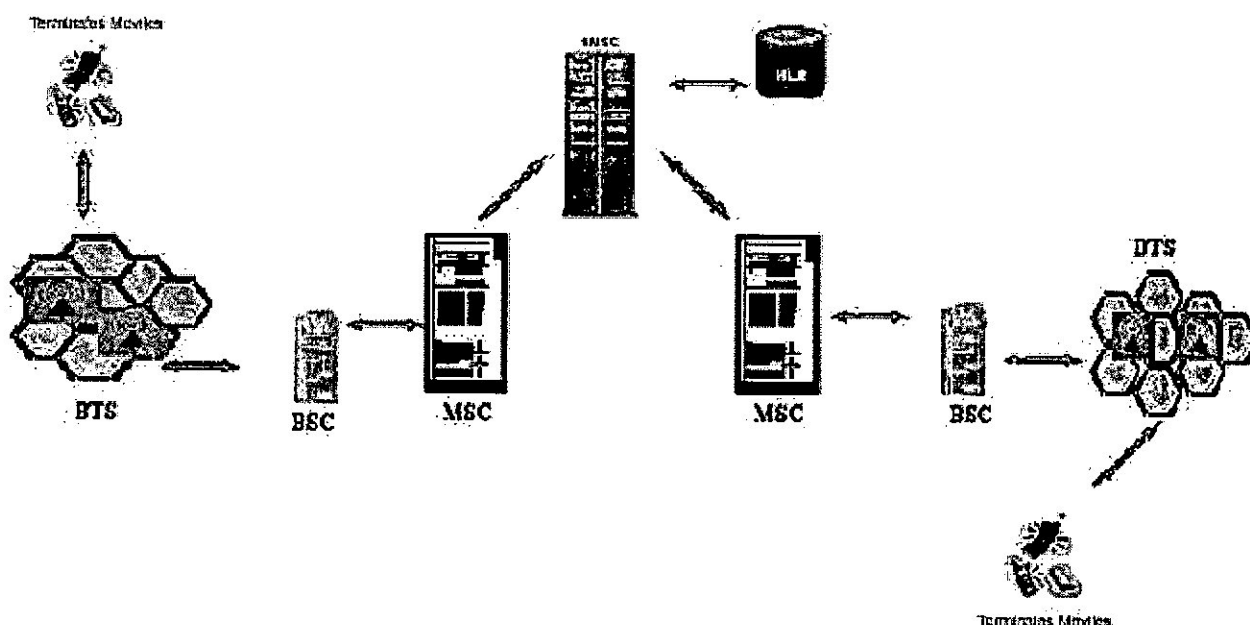
El servicio se desarrolló predominantemente en Europa. El crecimiento de su uso fue exponencial siendo estimados 20 millones de envíos mensuales a escala mundial en el inicio de 2001 existiendo estimativas de que este valor ascenderá a 70 millones en el plazo de 4 años.

Del sencillo envío de texto y funcionalidades de alerta para la existencia de mensajes de voz - y del sistema complementario de apoyo al cliente con el envío de los valores de facturación, saldo etc... - el SMS evoluciona progresivamente, gracias a los fabricantes, para la posibilidad de envío de logotipos, imágenes y el gran éxito de las melodías personalizadas.

De la misma forma que la llegada del SMS eclipsó los servicios de *paging* se prevé que la llegada de la tercera generación móvil y la evolución del concepto para el EMS (Enhanced Messaging Service - Servicio de Mensajes Mejorado) y posteriormente para el MMS (Multimedia Messaging Service - Servicio de Mensajes Multimedia) abra nuevas perspectivas de utilización y exploración comercial de nuevos servicios.

## CAMINOS DEL SMS

En condiciones normales, un mensaje de texto, tarda cerca de cinco segundos en recorrer el camino desde el teléfono móvil emisor hasta el receptor.



Teniendo como destino otro terminal, el «apeadero» siguiente a la estación base (BTS) más próxima del terminal emisor, esta, a su vez, encamina el mensaje hacia una estación de control denominada BSC (Base Station Controller).

La siguiente parada la compone el conmutador, designado por MSC (Mobile Switching Center), que a su vez transfiere la información al STP (Signalling Transfer Point), que se encarga de transmitirla al SMSC (Short Message Service Center) o lo que es lo mismo, el centro de control global de procesamiento de mensajes de cada operador.



---

Es en este punto que el mensaje recorrió medio camino, haciendo enseguida el recorrido inverso hasta llegar a la BTS (estación base) más próxima del receptor, que se encarga de descargarla en dicho terminal.

Y esto solo es cuando se trata de mensajes enviadas a un teléfono adscrito a la misma red.

Ahora, cuando el envío es realizado a un suscriptor de una red diferente, es el SMSC el que envía la información al GMSC (Gateway MSC) que la transmite a su vez a su congénere de la red de destino, a través de conexiones dedicadas a ello y directas, y que se encarga de encaminar el mensaje por el recorrido inverso hasta el destinatario.

Y todo esto en sólo cinco segundos, realizando exactamente el mismo recorrido que hace una llamada de voz cuando es establecida entre dos usuarios de redes móviles.

### ***¿TAMAÑO DE UN SMS?***

La única diferencia entre las llamadas de voz para las transferencias de datos es que esta última, utiliza un protocolo existente en el sistema GSM, (Mobile Application Part-MAP) que, a través de la arquitectura de red, permite el envío de datos.

Entre ellos, un subgrupo de la tabla ASCII, que no es más que el conjunto de caracteres ofrecido para la escritura del mensaje.



---

Considerando que se estipuló que un mensaje no podría contener más de 160 caracteres - lo que se traduce en unos 140 bytes, a los cuales hay que añadirles 30 bytes más por el espacio que ocupa la cabecera

En el caso de los MMS (mensajes multimedia), necesariamente más pesados, debido a la integración del sonido, imágenes o animaciones, el límite de los cuales está establecido en 30 kb.

## **PRUEBAS PILOTOS**

De acuerdo a los contenidos existentes en la guía de trámites, se ofrecen tres tipos de información distinta, información estática, información dinámica e información transaccional.

### ***CONTENIDO ESTÁTICO***

El contenido estático se encuentra dado por toda aquella información que no varía.

Esta información, se encuentra agrupada en páginas html, y se producen variaciones en el contenido en forma esporádica.

Dentro de esta información encontramos como por ejemplo:

- Ubicación de los centros de informes
- Gran parte de la información Útil



- 
- Prestadores de Salud Laboral para Empleados Públicos en caso de accidente de trabajo
  - Prestadores en caso de Emergencias para Empleados Públicos en caso de accidente de trabajo
  - Listado de Farmacias y Farmacias de Turno.
  - Etc.

### ***CONTENIDO DINÁMICO***

El contenido dinámico se encuentra dado, por toda aquella información que es obtenida y generada por medio de consultas realizadas a las bases de datos.

Esta información, esta sujeta al resultado de la consulta realizada, por lo que no es fija y su frecuencia de variación es muy alta.

Dentro de esta información encontramos como por ejemplo:

- La información de realización de los trámites contenidos en la Guía de Trámites.
- Foros
- Mapa del Sitio
- Etc.



---

## **CONTENIDO TRANSACCIONAL**

El contenido transaccional, se encuentra dado por toda aquella información, que es requerida por un formulario y almacenada en una base de datos, ya sea para la realización de un trámite online o la devolución de un resultado a la persona que realiza la transacción.

Este tipo de información, requiere que el usuario incorpore datos, los mismos sea verificados, almacenados y luego procesados para generara un resultado.

Dentro de esta información encontramos como por ejemplo:

- Los trámites On-lines como por ejemplo, el pedido de Partidas de Nacimientos, Matrimonio o Defunción.
- Consulta de estado de partidas
- Consulta del Padrón Electoral.
- Generación de consultas, quejas y reclamos
- Etc.





---

Ya que esta se encuentre en una página estática o sea obtenida por medio de una consulta en la base de datos.

### ***PRUEBAS REALIZADAS***

Para la realización de las pruebas, se habilitó una dirección especial, la misma fue denominada [ewap.mendoza.gov.ar](http://ewap.mendoza.gov.ar).

Sobre esta dirección, se realizaron las configuraciones adecuadas y fueron habilitados los servicios correspondientes para indicarle al servidor que debe brindar no solo páginas con contenido WWW, sino que debe dar además, contenidos WAP, dependiendo del cliente que realice la consulta.

### ***CONSULTA DEL PADRÓN ELECTORAL***

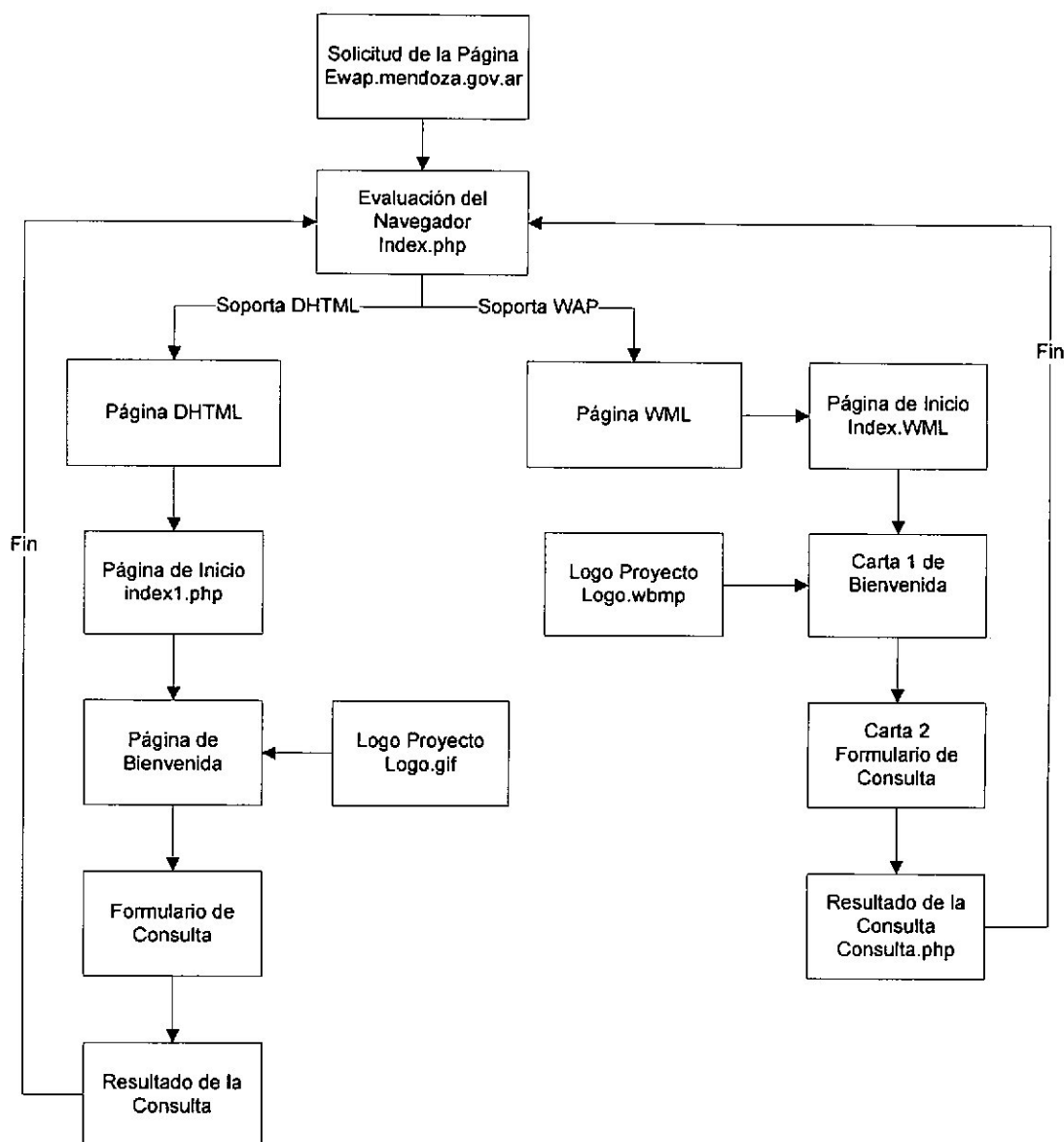
Se generaron pruebas pilotos para comprobar las características de los diferentes tipos de contenidos, y como estos eran visualizados por los diferentes tipos de dispositivos inalámbricos existentes.

Se eligió la consulta del padrón, debido a que es un servicio donde se podían combinar los diferentes tipos de contenidos antes relevados.



## Estructura de la Página Generada de Acuerdo al Navegador

En la siguiente imagen, se ilustran como de acuerdo a la solicitud realizada por el navegador al servidor, este evalúa el tipo de contenido soportado por este, y le presenta el contenido adecuado.



Esta selección, es realizada dentro del archivo index.php.



Dentro del Archivo, encontramos la siguiente sección de código:

```
if (strpos($_SERVER["HTTP_ACCEPT"], "text/vnd.wap.wml"))
{ // Si el Navegador soporta WML
$url = "index1.wml";
}
else
{
$url = "index1.php";
}
header("Location: $url"); exit;
```

Cuya principal función es verificar si el navegador soporta WML, de ser así, es re enviado a la página index.wml, de lo contrario, se encuentra ante un navegador WEB, el que es re dirigido a la dirección index1.php.

De este modo, el servidor es el que determina el tipo de contenido que será enviado y visualizado por el navegador consultante.



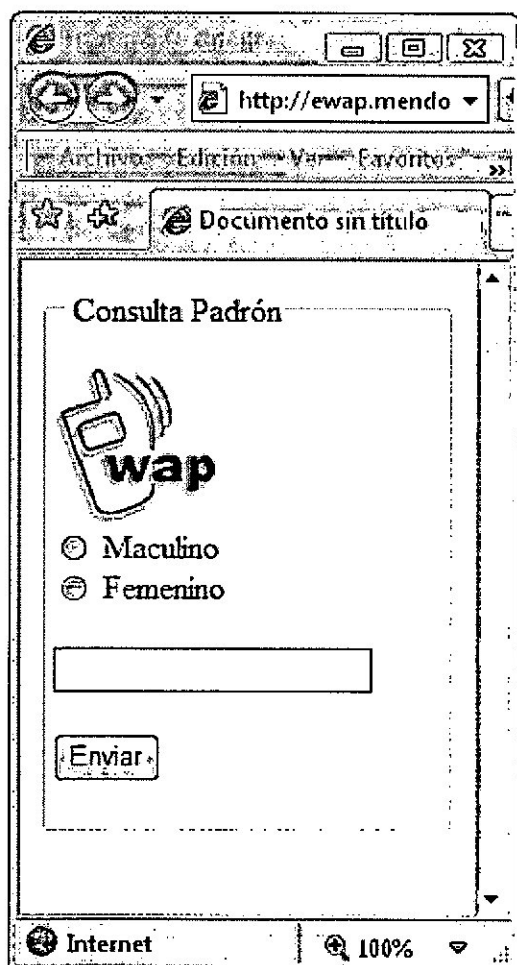
## VISUALIZACIÓN EN NAVEGADORES WEB Y NAVEGADOR WAP

Las siguientes imágenes corresponden a las diferentes secciones visualizadas durante la consulta del padrón por los diferentes navegadores.

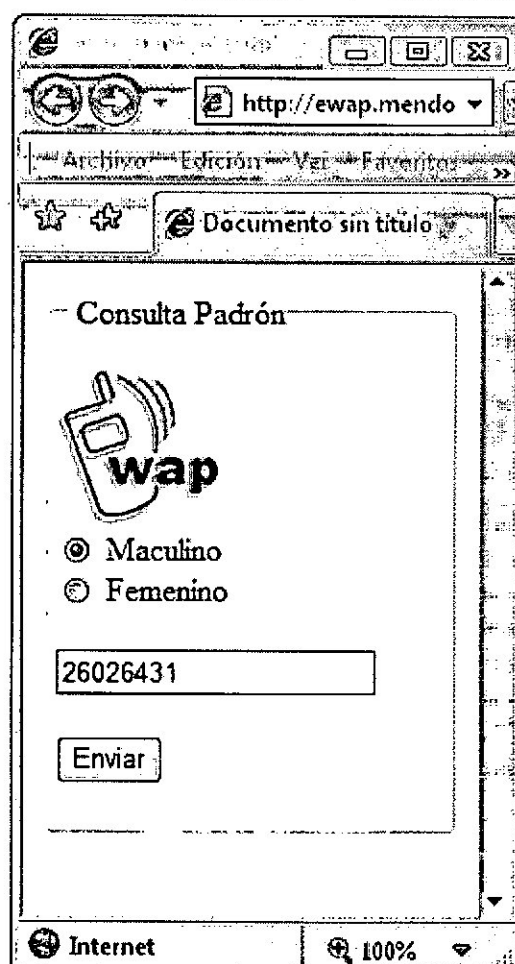
### Internet Explorer

Cuando el servidor web detecta que el usuario corresponde a un navegador web común, un browser de una PC, le presenta las siguientes páginas.

**Pantalla de Inicio**

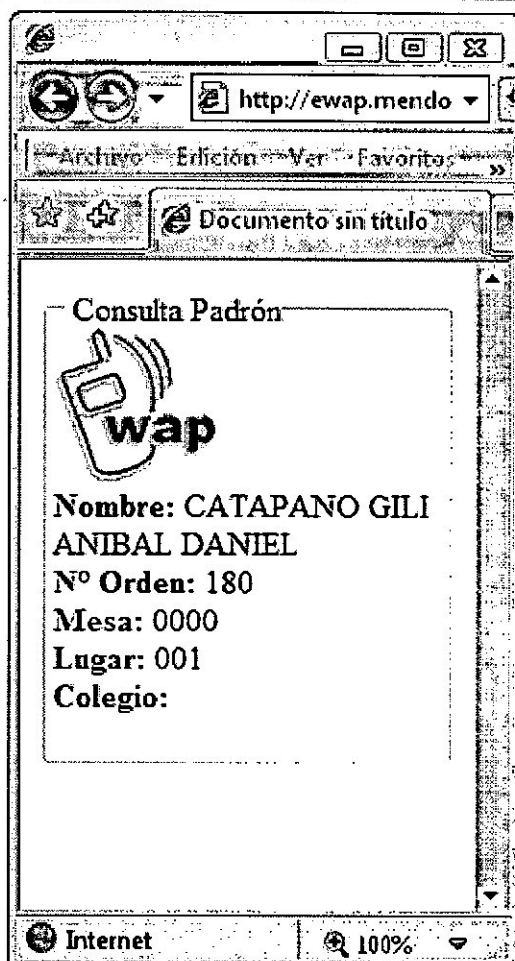


**Formulario de Consulta**





## Resultado de la Consulta



## Navegador Celular Web

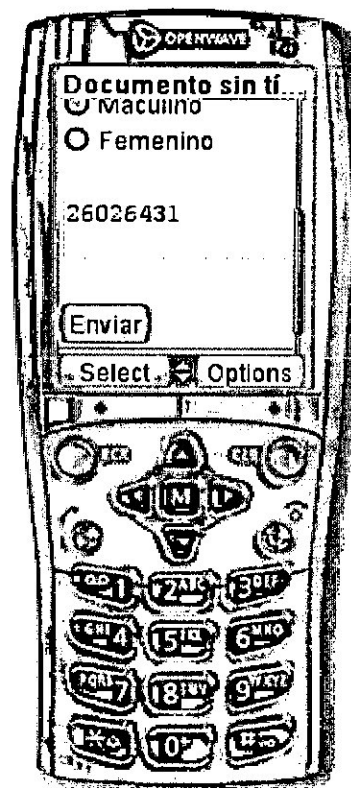
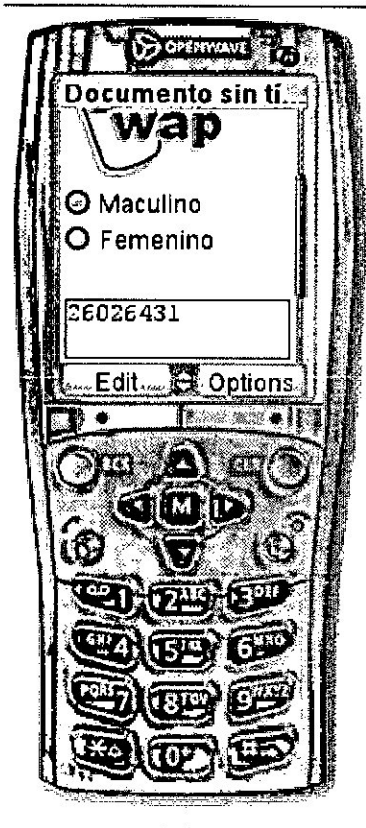
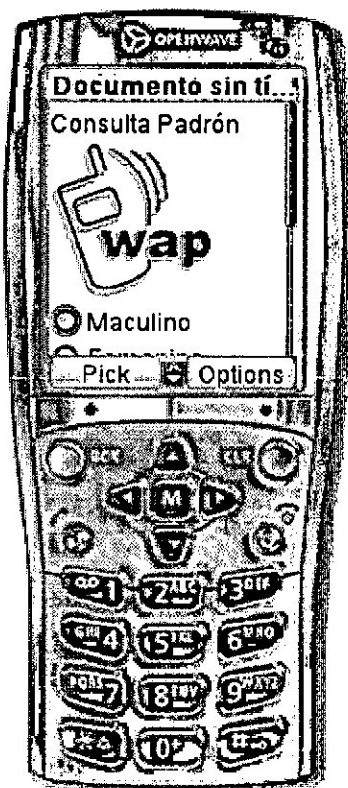
Cuando el servidor web detecta que el usuario corresponde a un navegador de celular, y además este soporta páginas web, le presenta la



siguiente secuencia de páginas, una secuencia muy parecida a la visualizada anteriormente en un navegador web de una máquina de escritorio.

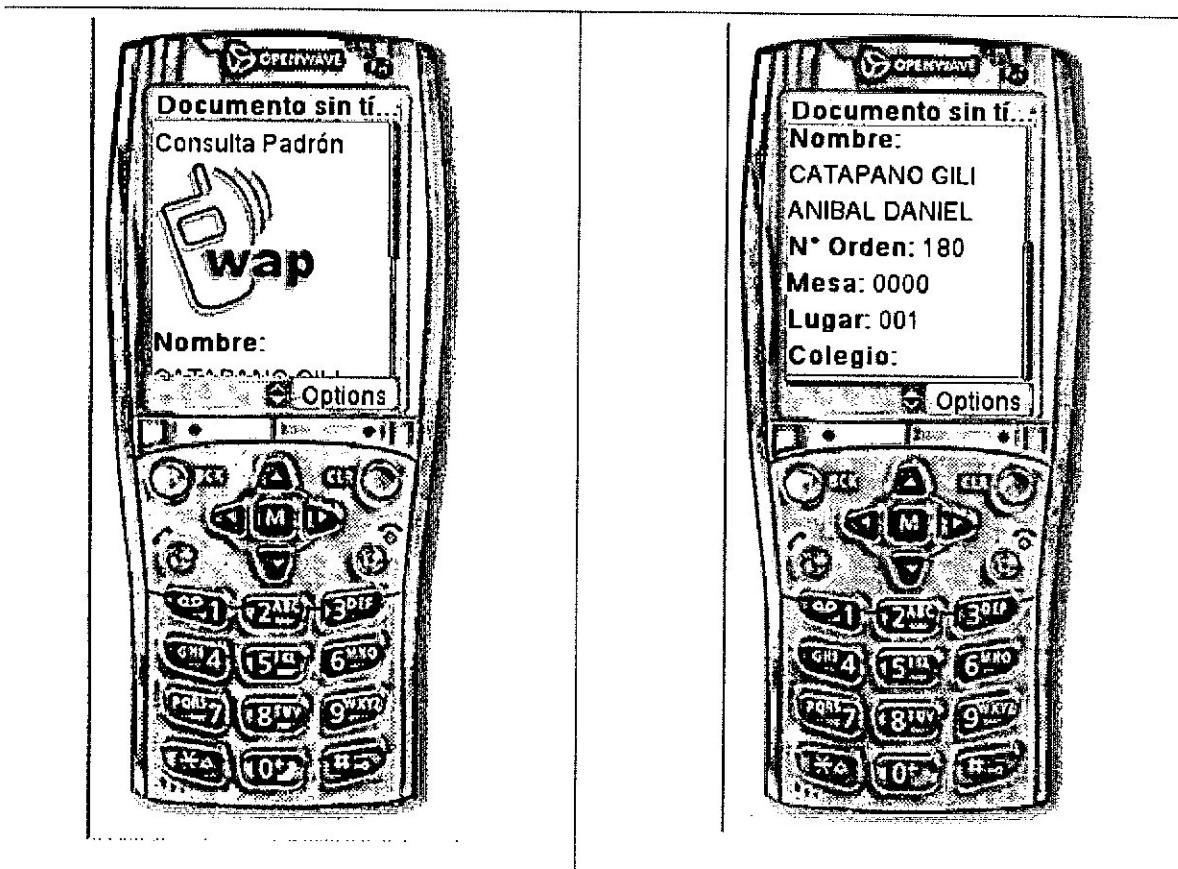
Pantalla de Inicio

Formulario de Consulta





## Resultado de la Consulta



### Navegador Celular WAP

Cuando el servidor web detecta que el usuario corresponde a un navegador de celular, y además solo soporta páginas WAP, le presenta la siguiente secuencia de páginas.



Pantalla de Inicio	Formulario de Consulta	Resultado de la Consulta

### SECUENCIA DE NAVEGACIÓN WAP

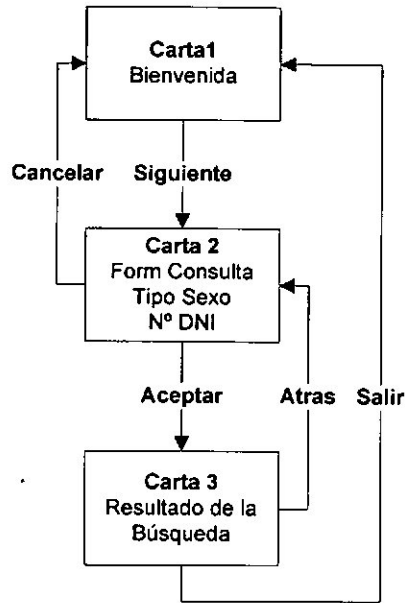
Debido a las limitaciones propias de los dispositivos inalámbricos, descritas y analizadas a lo largo de los distintos informes presentados, es que al realizar una aplicación WAP, a diferencia de una aplicación WEB, se debe analizar la estructura de navegación del sitio o de la aplicación desarrollada.

Se debe tener presente que la estructura a desarrollar, no debe superar un cierto número de niveles aproximadamente 3 o 4, ya que de ser los niveles



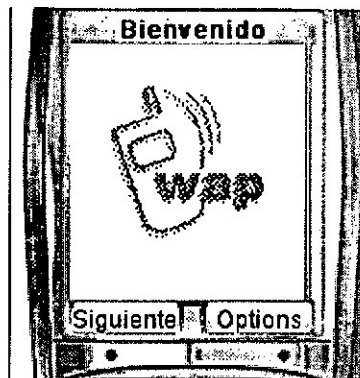


más extensos, el usuario perdería el hilo de la navegación provocando que no encuentre la información deseada.



Tal como vemos, el diagrama del circuito consta de solo 3 bloques, esto hace que la aplicación sea sencilla y de fácil navegación para el usuario.

La primer carta, contiene solo información estática de bienvenida, mostrándole al usuario el logo del proyecto en formato wbmp.





Al presionar el usuario el botón asignado a la acción Siguiete, este es dirigido al formulario de consulta, donde el usuario debe ingresar el número de documento y el sexo de la persona a consultar en el padrón.

Una vez finalizado el ingreso de los datos correspondientes, por medio de la presión del botón Done, se envía la consulta al servidor, convirtiendo este tipo de comunicación en transaccional, por que el usuario envía un pedido concreto y el servidor le devuelve información sobre el dato solicitado.

Las dos Cartas anteriormente descritas, perteneces a una Baraja que se encuentra en el mismo archivo, y son presentadas juntas o independientes de acuerdo a las características del dispositivo que genere la consulta.

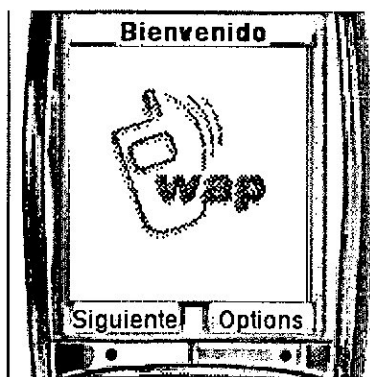
```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC
//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>

  <template>
    <do
      label="Atras"><prev/></do>
      type="prev"
    </template>
```



Carta N° 1

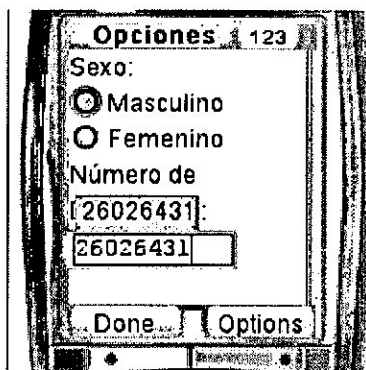
Logo de Bienvenida



```
<card id="carta1" title="Bienvenido"
newcontext="true">
  <do type="accept" label="Siguiete">
    <go href="#carta2"/>
  </do>
  <p align="center" mode="nowrap" ><br />
    
  </p>
</card>
```

Carta N° 2

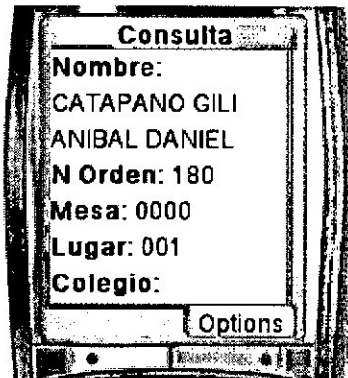
Formulario de Consulta



```
<card id="carta2" title="Opciones"
newcontext="true">
  <do type="accept" label="Consultar">
    <go href="consulta.php" method ="get">
      <postfield name="sexo"
value="$(sexo)"/>
      <postfield name="numero"
value="$(numero)"/>
    </go>
  </do>
  <p align="left" mode="nowrap">
    Sexo:<br/>
    <select title="Sexo" name="sexo">
<option value="M">Masculino</option>
<option value="F">Femenino</option>
    </select>
  </p>
  <p>
N&#xFA;mero de Documento:<br/>
<input name="numero" type="text"
maxlength="9" size="10" format="9N"/>
  </p>
</card>
</wml>
```



Analizada y procesada la información ingresada por el usuario, en el servidor, este realiza el envío del resultado de la consulta en una nueva Carta perteneciente a otra Baraja.



Luego al presionar sobre Options y seleccionando Atrás o Salir regresa a las secciones anteriores descriptas.

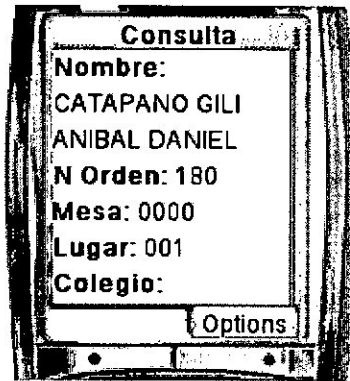
```
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-
//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">
<wml>

  <template>
    <do type="prev" label="Atras">
<prev/></do>
    <do type="" label="MENU">
      <go href="index.wml#carta2"/>
    </do>
  </template>
```



## Carta N° 1

## Resultado de la Consulta



```

<card id="carta1" title="Consulta">
  <do type="accept" label="Inicio">
    <go href="index.wml#carta2"/>
  </do>
  <p align="left" mode="nowrap" ><br /> </p>
  <p align="left" mode="nowrap" >
    <b>Nombre:</b> XXXXXX</p>
  <p align="left" mode="nowrap" >
    <b>N Orden:</b> 180</p>
  <p align="left" mode="nowrap" >
    <b>Mesa:</b> 2</p>
  <p align="left" mode="nowrap" >
    <b>Lugar:</b> Capital</p>
  <p align="left" mode="nowrap" >
    <b>Colegio:</b> Don Bosco</p>
</card>
</wml>

```

En el código anterior, no se incorporó toda la lógica de controles realizados sobre la información ingresada por el usuario, ni la lógica de búsqueda de la información solicitada, solo se muestra el resultado obtenido de la consulta realizada.



---

## CONCLUSIONES

La tecnología WAP y SMS, suponen un importantísimo avance en lo que se refiere al acceso ubicuo a servicios por vía telemática, aprovechando la aceptación generalizada del uso cotidiano de dispositivos móviles, que, en el caso concreto de terminales telefónicos, ya se cifran en torno a los 300 millones a nivel mundial.

Uno de los aspectos más notables de WAP, es la posibilidad de reutilizar tecnologías maduras y suficientemente probadas como las existentes en el dominio WWW, para ofrecer dichos servicios y resolver el problema de la generación dinámica de contenidos, logrando de este modo un grado de personalización no contemplado hasta el momento.

Además, por ser WAP una tecnología independiente del portador físico de acceso a la red inalámbrica, queda garantizada su vigencia para un futuro próximo, en el que existirá una evolución de las redes GSM actuales a redes GPRS y, posteriormente, UMTS.

Esta evolución no sólo supondrá un beneficio directo para las aplicaciones WAP, por el importante aumento de ancho de banda que trae consigo, sino que además existirá integración con otros servicios como es el caso de GPS, lo que permitirá una personalización de contenidos sin precedentes: basada en la ubicación actual del terminal móvil, y, por ende, del usuario.

En cualquier caso, es preciso tener en cuenta que en el desarrollo de soluciones WAP, se deberán extremar los controles de calidad, pues los



---

usuarios las perciben como servicios de telecomunicaciones, y por ello, el grado de exigencia de disponibilidad y fiabilidad es mucho más alto que, por poner un ejemplo, en el ámbito de los servicios tradicionales de Internet, donde el usuario suele ser más tolerante.

Debido al elevado coste del servicio, el tiempo de conexión se prevé como un obstáculo importante. La localización del servicio que está buscando el usuario debe ser fácil y rápida.

En efecto, la eficacia del WAP depende del tiempo que se gasta, de los precios que debemos pagar por ello, y por esto se intenta que la localización de los contenidos sea lo más intuitiva posible y que, a la vez, dichos contenidos se personalicen al máximo, pero no se trata de una tarea nada fácil.

El usuario consultará el servicio WAP cuando sepa los servicios existentes y le interesen dichos servicios que deben ser muy concretos, en todo caso, pueden ser generales pero a la vez específicos sobre un tema.

Tal es el caso de la consulta de la mesa de votación en periodos de elecciones.

Para lograr un verdadero interés y utilización de los servicios ofrecidos para dispositivos inalámbricos por parte de los usuarios, se deben tener presente cuatro principios básicos.



- 
1. Que sea ágil, y que los contenidos sean sencillos, por ejemplo: "Usted Vota en ...".
  
  2. Que esté personalizado y que todos los servicios se acomoden a las preferencias de los usuarios: "cocina de mercado, extranjera, etc."
  
  3. Que sean servicios inmediatos y que puedan obtenerse con un simple clic: "Teléfonos Útiles".
  
  4. Que la oferta sea lo más local posible: "estamos a 100 metros de usted"  
Siempre será más fácil recomendar un restaurante cercano que otro a donde haya que desplazarse.