

**APLICACION DE SUPERVISION DENTRO DEL MODELO DE
GESTION WBEM A TRAVES DE TERMINALES MOVILES WAP**



**JULIANA MARIA CITELLY MARTINEZ
ROGER EDUARDO CARVAJAL MUÑOZ**

**UNIVERSIDAD DEL CAUCA
FACULTAD DE INGENIERIA ELECTRONICA Y
TELECOMUNICACIONES
GRUPO DE NUEVAS TECNOLOGÍAS EN TELECOMUNICACIONES
POPAYAN
2001**

**APLICACION DE SUPERVISION DENTRO DEL MODELO DE
GESTION WBEM A TRAVES DE TERMINALES MOVILES WAP**



**JULIANA MARIA CITELLY MARTINEZ
ROGER EDUARDO CARVAJAL MUÑOZ**

**Monografía presentada como requisito
para optar al título de Ingeniero en
Electrónica y Telecomunicaciones**

DIRECTOR

ING. GUEFRY LEIDER AGREDO MENDEZ

UNIVERSIDAD DEL CAUCA

FACULTAD DE INGENIERIA ELECTRONICA Y

TELECOMUNICACIONES

GRUPO DE NUEVAS TECNOLOGÍAS EN TELECOMUNICACIONES

POPAYAN

2001

TABLA DE CONTENIDO

A. INSTALACIÓN Y CONFIGURACIÓN BÁSICA DE LA APLICACION.....	1
A.1 El Servidor de Información de Internet 5 (IIS-Internet Information Server), un servidor incorporado	1
A.2 El proceso de instalación	1
A.3 Trás la instalación	4
A.4 Tipos de Extensiones Multipropósito de Correo Internet (MIME-Mail Internet Multypropose Extensión)	5
A.5 configuración de los tipos MIME	5
B. TECNOLOGIAS Y LENGUAJES UTILIZADOS EN LA APLICACION	12
B.1 WML.....	12
B.1.1 Sintaxis WML	12
B.1.2 Caracteres especiales	12
B.1.3 Etiquetas	13
B.1.4 Elementos	14
B.1.5 Atributos.....	14
B.1.6 Comentarios.....	15
B.1.7 Variables.....	15
B.1.8 Distinción mayúsculas/minúsculas	16
B.1.9 Estructura	16
B.1.9.1 Edición.....	16
B.1.9.2 Estructura básica de una página WML	16
B.1.9.3 WML: Imágenes	17
B.1.9.4 Enlaces.....	20
B.1.9.5 Tablas	21
B.1.9.6 Barajas y Cartas	22
B.1.9.7 Template	24
B.1.9.8 Tareas	25
B.1.9.9 Variables.....	29
B.1.9.10 Conversión al formato escape.....	31
B.2 PROGRAMACION ASP y WML	32
B.2.1 Introducción a la programación de ASP y WML	33
B.2.2 Introducción a ASP.....	33
B.2.2.1 Definición de ASP:	33
B.2.3 Desarrollo de aplicaciones en ASP.....	34
B.2.3.1 Fundamentos básicos de vbscript.....	35
B.2.3.2 Tipos de datos, variables y arrays en vbscript	36
B.2.3.3 Control de flujo en vbscript	37
B.2.4 Objetos integrados en ASP	38
B.2.4.1 Objeto Session	39

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

B.2.4.2 Objeto Application	39
B.2.4.3 Objeto Request.....	40
B.2.4.4 Objeto Response	41
B.2.4.5 Objeto Server	41
B.2.5 El archivo Global.asa	43
B.2.6 Cookies.....	45
B.3 IIS	46
B.3.1 Windows 2000, un servidor en Internet	46
B.3.2 Servicios Web.....	46
B.3.2.1 Servidores virtuales	47
B.3.2.2 Servidor FTP.....	48
B.3.2.3 Servicios SMTP y TNP	49
B.3.2.4 Active Server Pages	49
B.3.2.5 Administración mediante Web y administración delegada	50
B.3.2.6 Seguridad	50
B.4 Bases de Datos	51
B.4.1 Cómo son las bases de datos	51
B.4.2 Tablas	52
B.4.3 Campos clave.....	53
B.4.4 Creación de una base de datos.....	54
B.4.5 Relaciones entre datos.....	54
B.4.6 Modelos de bases de datos	55
B.4.7 El uso de las relaciones intertabla.....	56
D. CODIGO DE LA APLICACION	58
D.1 Visual Basic	58
D.2 ASP.....	77
GLOSARIO.....	98
ACRONIMOS	99

LISTA DE FIGURAS

Figura A.1 Instalación.....	2
Figura A.2 Componentes de IIS.	3
Figura A.3 El IIS.....	6
Figura A.4 Propiedades del IIS	7
Figura A.5 Asignación MIME.....	8
Figura A.6 Tipos de archivo.....	8
Figura A.7 Propiedades Principales.....	9
Figura A.8 Tipos de Archivos.	10
Figura A.9 Nombres de Archivo	10

LISTA DE TABLAS

Tabla 1. Tipos MIME	5
Tabla 2. Caracteres Especiales.	13
Tabla 3. Etiquetas WMI.	14
Tabla 4. Atributos de la etiqueta IMG	18
Tabla 5. Atributos de la etiqueta ANCHOR	20
Tabla 6. Atributos de la etiqueta TABLE	22
Tabla 7. Atributos de la etiqueta CARD	23
Tabla 8. Atributos de la etiqueta GO	26
Tabla 9. Atributos de la etiqueta PREV	27
Tabla 10. Formato Escape.....	32
Tabla 11. Caracteres Escape.....	32
Tabla 12. Características de ASP.....	34
Tabla 13. Características básicas de VBScript.....	36
Tabla 14. Métodos, Colecciones y Eventos	43
Tabla 15. Estructura archivo Global.asa	44

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

A. INSTALACIÓN Y CONFIGURACIÓN BÁSICA DE LA APLICACION

A.1 El Servidor de Información de Internet 5 (IIS-Internet Information Server), un servidor incorporado

A diferencia de su antecesor, IIS 5 se incorpora como una prestación más del sistema operativo Windows 2000 fundamental, pues se utiliza como base para un buen número de productos y servicios.

IIS 4 debe instalarse con posterioridad al sistema operativo Windows NT 4, denominado Windows NT Option Pack que, aunque sea gratuito, debía obtenerse independientemente. Con la adición del Option Pack a Windows NT 4, se constituyó un nuevo sistema operativo que disponía de una interesante y completa plataforma de servicios Internet. Windows 2000 ya proporciona esa plataforma.

De hecho, IIS 5 se incluye en la instalación por defecto de Windows 2000, en cualquiera de sus versiones si bien puede ser necesario modificar la configuración manualmente.

A.2 El proceso de instalación

¿Qué se necesita tener instalado en el sistema operativo para que la instalación de IIS 5 pueda llevarse a cabo? La siguiente es una lista de las opciones de configuración necesarias:

- Windows 2000 Professional, Server, Advanced Server o DataCenter.
- TCP/IP. No es necesario que la dirección IP esté fija.
- Sistema de archivos de Seguridad de windows NT (NTFS- NT File Security) si se desea seguridad orientada al archivo.
- Servidor de Dominio de Nombres (DNS-Domain Name Server) (Opcional).

Una vez se haya asegurado que los requisitos de software están cumplidos, se puede proceder a la instalación propiamente dicha. El proceso de instalación se inicia desde

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

"Agregar/Eliminar Programas" en el Panel de Control, concretamente en la pestaña de "Añadir/Eliminar Componentes" de Windows, tal como se muestra en la figura A.1

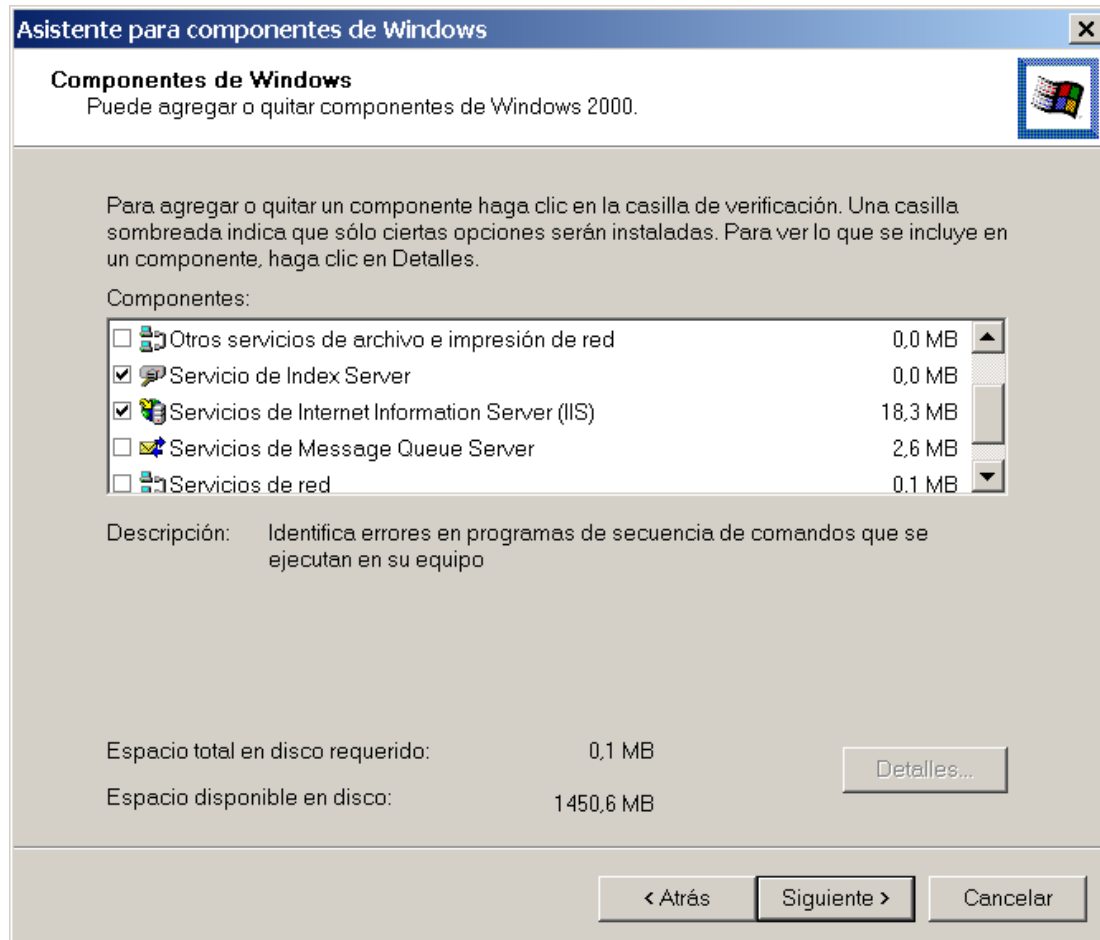


Figura A.1 Instalación

La modificación de las opciones disponibles en una instalación existente (por ejemplo, para añadir los servicios NNTP de noticias) se lleva a cabo exactamente por el mismo procedimiento.

Una vez que se haya accedido al cuadro de diálogo del Windows Components se marca la casilla de Internet Information Services y se pulsa el botón Detalles, para escoger las opciones y funcionalidades a incorporar, tal como se muestra en la figura A.2

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

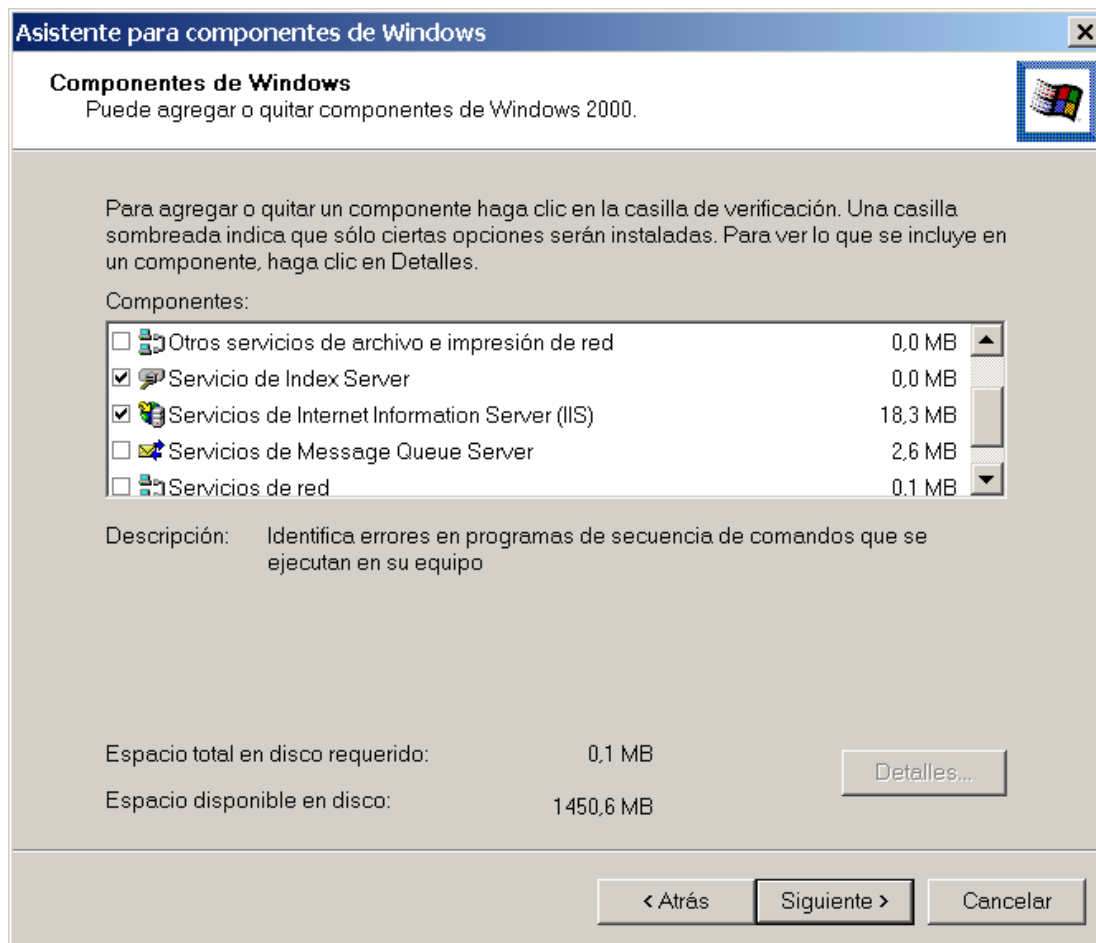


Figura A.2 Componentes de IIS.

En una instalación por defecto se seleccionan automáticamente las siguientes opciones:

- Servicios Web
- Herramientas de administración (MMC y HTML)
- Servicio de correo simple (SMTP-Simple Mail Transfer Protocol)
- Documentación del producto
- Extensiones de FrontPage 2000
- Soporte al desarrollo con Visual InterDev

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

Quedan como prestaciones opcionales los servicios FTP y NNTP (News). Estos últimos son imprescindibles, por ejemplo, para la instalación de Exchange 2000. De hecho, Internet Information Server es necesario para la totalidad de los .Net Enterprise Servers incluyendo SQL Server 2000 quien lo necesita para su soporte a XML y para el acceso a través de URL a los contenidos de las bases de datos. Por ejemplo, y sin ir más lejos, la instalación de Exchange 2000 requiere que se disponga del servidor SMTP de IIS 5.

En el caso de que se esté instalando Windows 2000, llevando a cabo una actualización de una instalación anterior, los servicios Internet sólo se instalarán automáticamente si Internet Information Server estaba incluido en el sistema antiguo. En cualquier caso, se puede solicitar igualmente la instalación bajo demanda.

A.3 Trás la instalación

Una vez se haya completado la instalación, Windows 2000 dispondrá de un elemento más dentro de las herramientas administrativas, es decir, de un acceso directo a la consola MMC de administración de los servicios Internet.

El proceso de instalación habrá creado un par de sedes web y, en función de los servicios escogidos, también creará los servidores virtuales de transferencia de archivos (FTP-File Transfer Protocol), SMTP y NNTP.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

A.4 Tipos de Extenciones Multipropósito de Correo Internet (MIME-Mail Internet Multypropose Extensión)

Para publicar paginas WML en internet se puede utilizar cualquier servidor. Tan sólo es necesario configurar adecuadamente los tipos MIME¹ en el servidor o incluir un fichero que haga lo propio.

	tipo MIME	extensión
código WML	text/vnd.wap.wml	.wml
imagen WML (extensión .wbmp)	image/vnd.wap.wbmp	.wbmp
código WML Script	text/vnd.wap.wmlscript	.wmls
código WML compilado	text/vnd.wap.wmlc	.wmlc
código WML Script compilado	text/vnd.wap.wmlscriptc	.wmlsc

Tabla 1. Tipos MIME

A.5 configuración de los tipos MIME

Para configurar el servidor solo es necesario configurar el IIS que viene dentro de Windows 2000.

Como se va a trabajar con paginas WML es necesario configurar los tipos MIME. Para ello se abre Herramientas Administrativas en el Panel de Control. Se escoge Administrador de Servicios Internet, tal y como se muestra en la figura A.3

¹ Tipos MIME: Extensiones multipropósito de correo Internet (MIME) es un estándar para Internet que proporciona a los exploradores Web la posibilidad de mostrar los archivos en varios formatos. Los tipos de archivo registrados que se instalan de forma predeterminada en Windows 2000 se enumeran en el cuadro de diálogo Tipos de archivo de la hoja de propiedades Sevicios de Internet Information Server.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

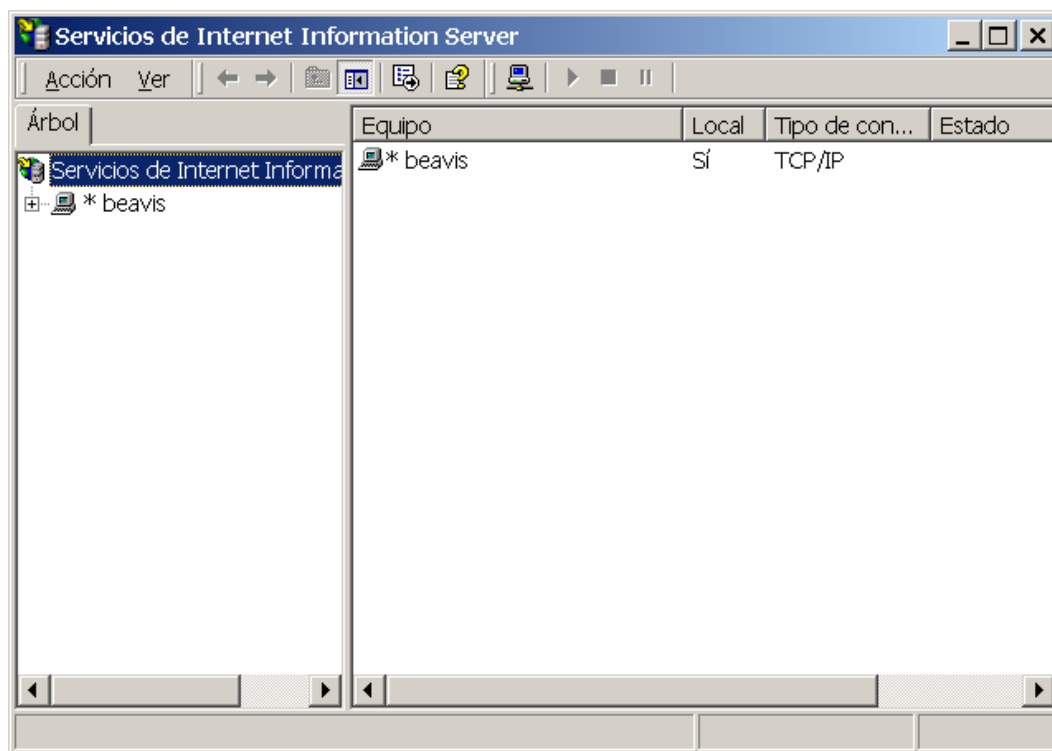


Figura A.3 El IIS

Posteriormente se selecciona el equipo que será el servidor y se da click derecho sobre él, luego se abren las propiedades, tal como se muestra en la figura A.4.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

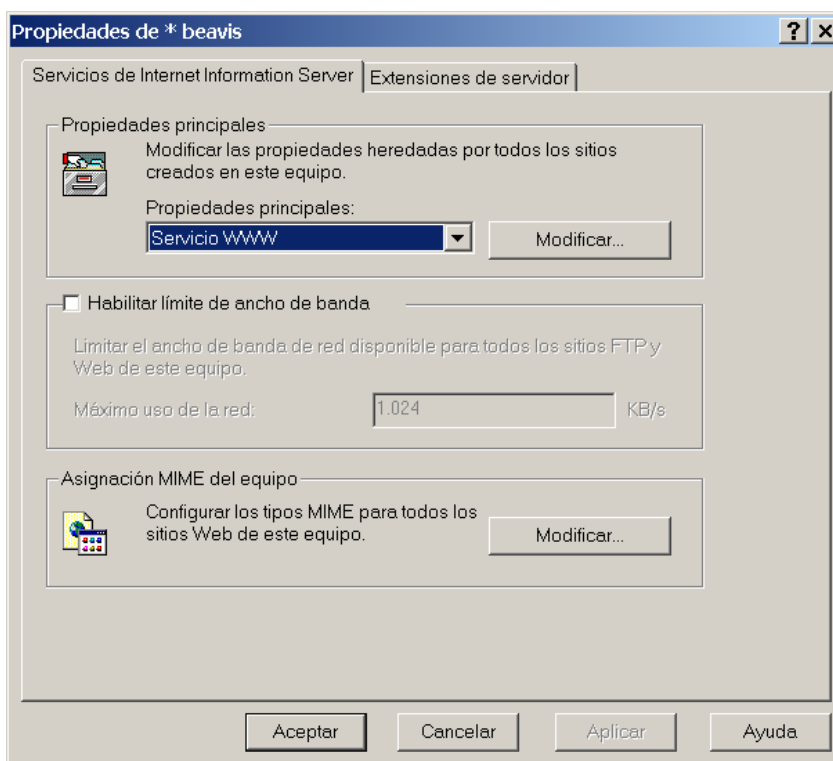


Figura A.4 Propiedades del IIS

Para configurar los tipos MIME se deben realizar dos pasos que son:

1. Se da click en Modificar en la Asignación MIME del equipo, tal como lo muestra la figura A.5

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

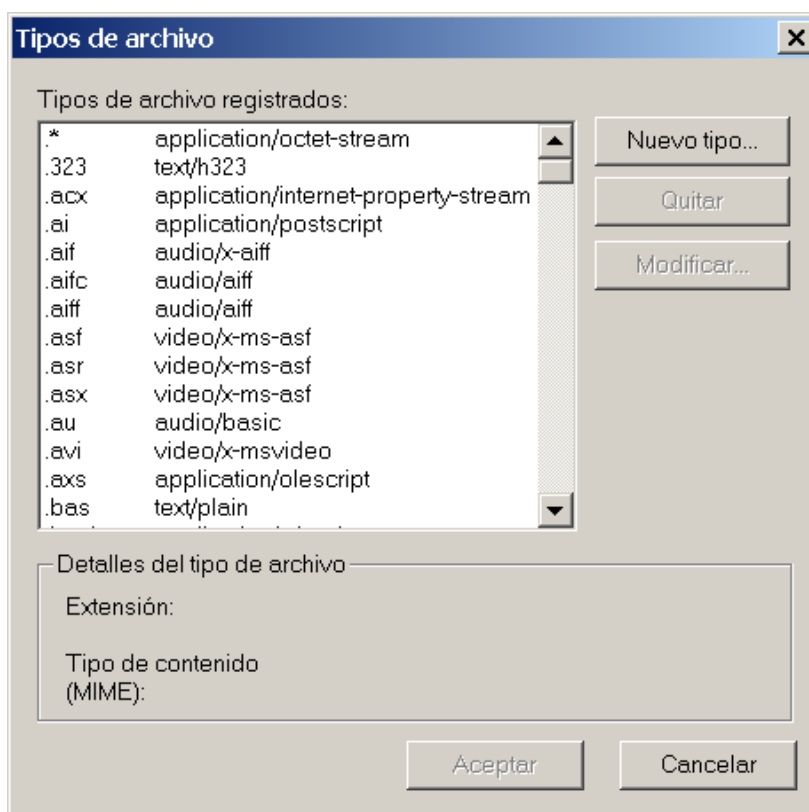


Figura A.5 Asignación MIME

Después se da click sobre Nuevo Tipo, desplegándose una ventana como la presentada por la figura A.6.

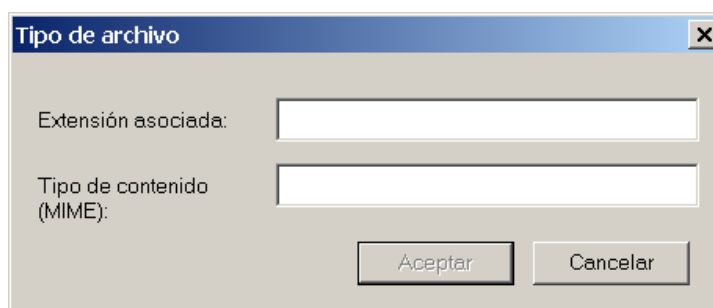


Figura A.6 Tipos de archivo

Y se colocan los cinco tipos MIME descritos anteriormente.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

- De igual forma se hace click en modificar dentro de Propiedades principales y se escoge la pestaña Encabezados http, tal como se muestra en la figura A.7

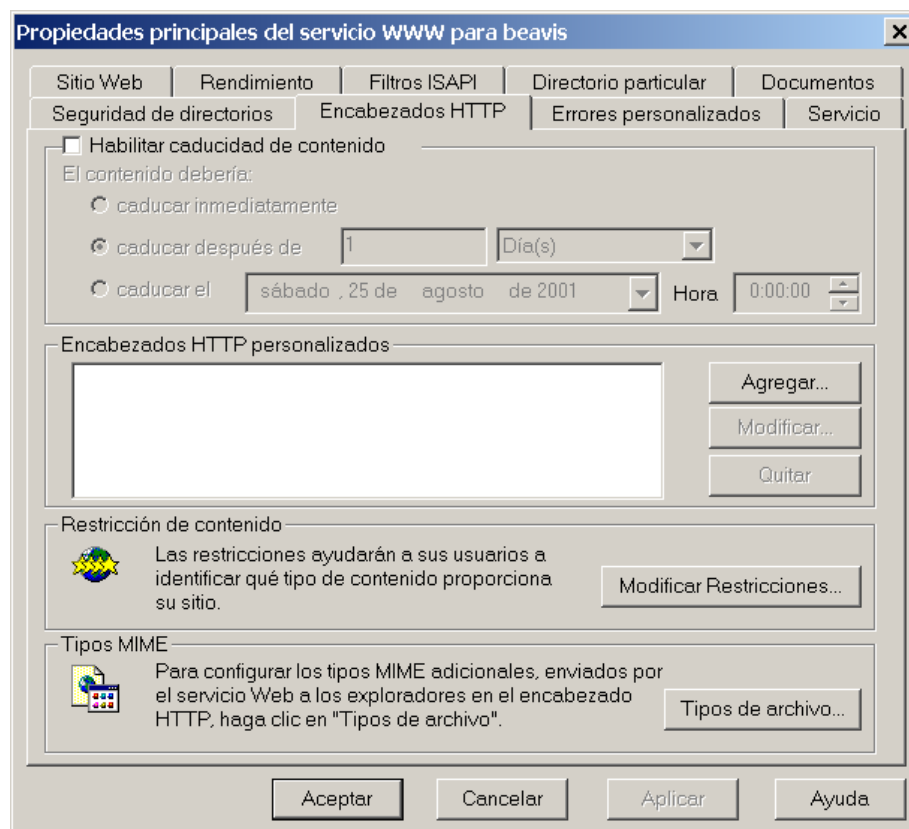


Figura A.7 Propiedades Principales

Luego se oprime el botón Tipos de Archivo en Tipos MIME, tal como lo muestra la figura A.8.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

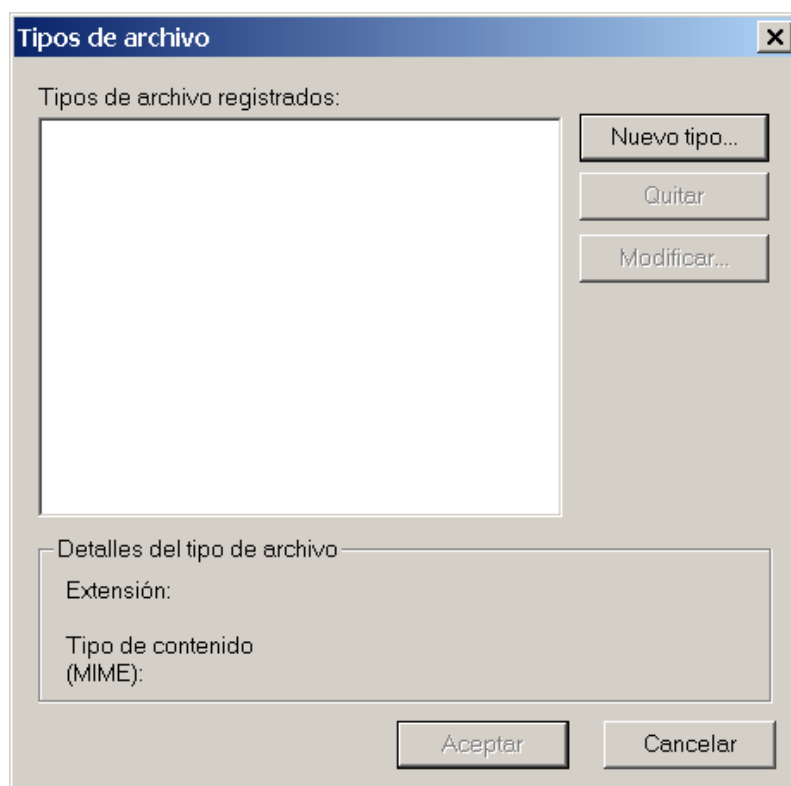


Figura A.8 Tipos de Archivos.

Y se selecciona Nuevo tipo, tal como se muestra en la figura A.9

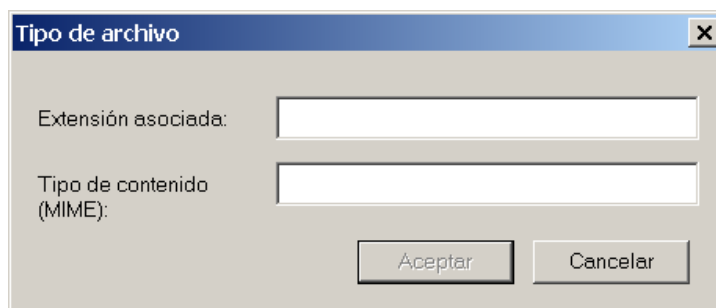


Figura A.9 Nombres de Archivo

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

De igual manera se colocan los cinco tipos MIME de los que se habló anteriormente.

NOTA: Las asignaciones MIME se pueden configurar en el equipo, en los sitios Web, en los directorios virtuales, en los directorios o en los archivos. Para configurar asignaciones MIME en el equipo, se usa la hoja de propiedades Servicios de Internet Information Server. Para configurar las asignaciones MIME en otros lugares, se utiliza la hoja de propiedades Encabezados HTTP.

Las asignaciones MIME configuradas en el equipo no suplantarán automáticamente las establecidas en niveles inferiores.

B. TECNOLOGIAS Y LENGUAJES UTILIZADOS EN LA APLICACION

B.1 WML

WML son las siglas de *Wireless Markup Lenguaje*. Es el lenguaje empleado para escribir páginas para cualquier dispositivo que utilice tecnología WAP (*Wireless Application Protocol*), como algunos Asistentes Digitales Personales (PDAs-*Personal Digital Assitant*) y algunos teléfonos móviles.

La unidad básica de WML es la carta. Las cartas se agrupan en barajas. Una baraja es la mínima unidad WML que un servidor Web puede enviar a un agente². Cuando un agente recibe una baraja, se muestra la primera de sus cartas, aunque puede ocurrir con cualquier otra que se desee.

WML diferencia entre mayúsculas y minúsculas. También convierte cualquier número contiguo de espacios en blanco, nueva línea, retorno de carro y tabuladores a un único espacio en blanco.

B.1.1 Sintaxis WML

WML hereda la mayoría de las construcciones sintácticas de XML, así como su mapa de caracteres.

B.1.2 Caracteres especiales

En un texto WML hay ciertos caracteres que no pueden escribirse tal cual porque WML los reserva para su código. Estos pueden mostrarse especificando un nombre o un número. Por ejemplo, el *ampersand* (&) se representa mediante el nombre "**&**". Todas los caracteres

² Agente: Es el actor que hace las peticiones de supervisión.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

especiales comienzan con un ampersand y terminan con un punto y coma. Otra razón para introducirlas de forma especial es evitar su interpretación por parte del navegador. En la siguiente tabla se puede ver algunas de las más importantes:

carácter	nombre	número	descripción
"	"	"	comillas
&	&	&	ampersand
'	'	'	apóstrofe
<	<	<	menor que
>	>	>	mayor que
	 	 	espacio en blanco
-	­	­	guión
\$	\$\$		dólar

Tabla 2. Caracteres Especiales.

B.1.3 Etiquetas

Una etiqueta es un descriptor del lenguaje asociado a un elemento a identificar único. También puede incluir atributos que describan otras propiedades y modifique su comportamiento.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

B.1.4 Elementos

Especifican toda la información de aspecto y estructura de una página WML. Los elementos pueden tener una etiqueta de inicio, contenido y otros elementos y una etiqueta de fin. Pueden tener una de las dos siguientes estructuras:

```
<etiqueta> contenido </etiqueta>
```

```
<etiqueta/>
```

La primera de ellas se utiliza cuando puede existir contenido. La segunda forma es para etiquetas vacías, es decir, sin contenido. Ambos tipos pueden poseer atributos. A continuación veremos la lista de etiquetas **WML** :

a	fieldset	postfield	small
anchor	go	pre	strong
access	head	prev	table
b	i	onevent	td
big	img	optgroup	template
br	input	option	timer
card	meta	refresh	tr
do	noop	Select	u
em	p	setvar	wml

Tabla 3. Etiquetas WML.

B.1.5 Atributos

Sirven para especificar información adicional sobre un elemento. Siempre se introducen en la etiqueta de inicio y con el siguiente formato:

```
<etiqueta atributo="valor">
```

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

```
    contenido
</etiqueta>

<etiqueta atributo="valor"/>
```

En el nombre de atributos se diferencia entre mayúsculas y minúsculas. Todos los atributos deben encerrarse entre comillas dobles o simples. Pueden emplearse comillas simples dentro del valor del atributo cuando este se encuentra encerrado entre comillas dobles y viceversa. Algunos atributos son obligatorios, como por ejemplo “href” en “go”.

B.1.6 Comentarios

Los comentarios en WML son iguales a los XML y HTML. No pueden anidarse.

```
<!-- comentario -->
```

B.1.7 Variables

Las cartas y las barajas WML pueden parametrizarse utilizando variables. Para sustituir una variable por su valor se utiliza la siguiente sintaxis:

```
$identificador
```

```
$(identificador)
```

```
$(identificador:conversion)
```

Los paréntesis son obligatorios si un espacio en blanco no indica el final del nombre de la variable. La interpretación de variables tiene la prioridad más alta dentro de WML.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

B.1.8 Distinción mayúsculas/minúsculas

Como se ha mencionado anteriormente, WML es un lenguaje que distingue entre mayúsculas y minúsculas. Todos los nombres de etiquetas, atributos y valores de atributos deben escribirse correctamente (en mayúsculas o minúsculas) para que funcionen.

B.1.9 Estructura

B.1.9.1 Edición

Las páginas WML pueden crearse con cualquier editor de texto normal. También se puede utilizar el que se usa habitualmente para HTML. Además existen algunos específicos para WML, la mayoría de pago, y algunos emuladores de teléfono móvil traen el suyo propio. La extensión empleada por este tipo de archivos es **.wml**.

B.1.9.2 Estructura básica de una página WML

Lo primero que se debe incluir en cualquier página WML es el siguiente encabezado que define que versión del lenguaje se está utilizando:

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

Tras esta cabecera, todo el código de la página debe englobarse entre las etiquetas **<wml>** y **</wml>**.

Este es un ejemplo de baraja de una sola carta:

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

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

```
<wml>
  <card>
    <p>
      Mas sencillo imposible
    </p>
  </card>
</wml>
```

B.1.9.3 WML: Imágenes

□ **img**

img es la etiqueta que nos permite introducir imágenes. Es una etiqueta vacía, sin contenido. Su forma de uso es:

```

```

En el anterior ejemplo se ha usado dos atributos de **img**, **src** y **alt**. **src** indica la dirección de la imagen a mostrar, es un atributo obligatorio. Con **alt** podemos mostrar un texto alternativo en caso de que la imagen indicada en **src** no esté disponible, es opcional. **img** tiene muchos atributos.

atributo	acción	obligatorio
Src	Especifica la dirección de la imagen.	si
Alt	Muestra un texto alternativo en caso de que la imagen no esté disponible.	no
localsrc=nombreadarchivo	Imagen local que se mostrará si esta disponible.	no
hspace=longitud vspace=longitud	Cantidad de espacio en blanco que se insertará horizontal o verticalmente. Su valor por defecto es 0. Puede expresarse mediante un valor porcentual.	no
align=(top middle botto	Alineamiento de la imagen respecto al texto.	no

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

m)		
height=longitud width=longitud	Da una idea al dispositivo del tamaño de la imagen (para mientras poder seguir descargando el resto del texto).	no
xml:lang=	Especifica el lenguaje formal o natural de un elemento o sus atributos.	no
Id	Proporciona un nombre único dentro de una baraja.	no
Class	Afilia un elemento a una o más clases.	no

Tabla 4. Atributos de la etiqueta IMG

Aquí se puede ver un ejemplo de uso de img:

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

<wml>
  <card id="imagenes" title="imagenes">
    <p>
      cabra 
      <br/>
      cerdo 
      <br/>
      conejo 
      <br/>
      oso 
      <br/>
      reno 
      <br/>
    </p>
  </card>
</wml>
```


Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

□ Imágenes en movimiento

La única forma de crear imágenes en movimiento es utilizar contadores de tiempo (timer) para mostrar sucesivamente las diferentes imágenes. Por ejemplo:

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

<wml>

  <card id="img1" ontimer="#img2">
    <timer value="3"/>
    <p>
      
    </p>
  </card>

  <card id="img2" ontimer="#img3">
    <timer value="3"/>
    <p>
      
    </p>
  </card>

  <card id="img3" ontimer="#img4">
    <timer value="3"/>
    <p>
      
    </p>
  </card>

  <card id="img4" ontimer="#img1">
    <timer value="1"/>
    <p>
      
    </p>
  </card>

</wml>
```

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

B.1.9.4 Enlaces

Las etiquetas **anchor** ya especifican el inicio de un enlace a otra página. El final de un enlace es especificado por otras etiquetas como card. Es un error anidar enlaces. Pueden utilizarse en cualquier parte excepto dentro de una etiqueta option.

Un enlace debe tener asociada una acción (¡y sólo una!) que especifique que acción realizar cuando el enlace sea seleccionado. La acción debe ser una de:

- go
- prev
- refresh

a es una forma abreviada de la etiqueta **anchor** que está asociada siempre a la etiqueta go sin variables ni atributos. Veamos dos formas equivalentes de escribir un enlace con a y con anchor:

```
<anchor> sigueme
  <go href="destino.wml">
</anchor>
```

Es equivalente a:

```
<a href="destino.wml">sigueme</a>
```

□ Atributos de anchor:

atributo	acción	obligatorio
title="titulo"	Breve descripción del enlace.	no
accesskey="#"	Asigna la tecla "#" a el enlace, es decir, el enlace puede activarse presionando "#".	no

Tabla 5. Atributos de la etiqueta ANCHOR

Un ejemplo, un poco más completo de anchor:

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

```

<?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="options" name="prev" label="anterior">
      <prev/>
    </do>
  </template>

  <card id="elige" title="animales">
    <p>
      <anchor title="araña">araña
        <go href="#arana"/>

      </anchor>
      <anchor title="gato">gato
        <go href="#gato"/>
      </anchor>
    </p>
  </card>

  <card id="arana" title="araña">
    <p>
      
    </p>
  </card>

  <card id="gato" title="gato">
    <p>
      
    </p>
  </card>
</wml>

```

B.1.9.5 Tablas

Aunque la etiqueta **table** está recogida en la especificación WML 1.1, hay algunos móviles que no la implementan, como es el caso de Nokia 7110. En su lugar los datos son mostrados en una tabla de una única columna. Para explicar como construir tablas, ver el siguiente ejemplo:

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

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

<wml>
  <card>
    <p>
      <table columns="3">
        <tr><td>1,1</td><td>1,2</td></tr>
        <tr><td>2,1</td><td>2,2</td></tr>
      </table>
    </p>
  </card>
</wml>
```

tr es la etiqueta que permite especificar filas y **td** la que permite especificar columnas. Los atributos de table son los siguientes:

atributo	Acción	obligatorio
title="titulo"	Título de la tabla	no
align="C L R"	Especifica el alineamiento de la tabla	no
columns="numero"	Especifica el número de columnas	si

Tabla 6. Atributos de la etiqueta TABLE

B.1.9.6 Barajas y Cartas

□ card

La etiqueta card es la unidad básica de navegación dentro de una baraja. Es posible moverse de una carta a otra dentro de una baraja o a otras barajas. A no ser que se especifique otra cosa, siempre se mostrará la primera carta (card). Un ejemplo simple de su uso puede ser:

```
<card id="carta1" title="Titulo">
  <p>
    Texto <u>informativo</u>
```

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

```
</p>
</card>
```

He aquí una descripción detallada de los atributos de card:

nombre	Acción	obligatorio
id="nombre"	Especifica un nombre único dentro de la baraja.	no
title="titulo"	Muestra un titulo en la carta.	no
newcontext="false"	Reinicia el contexto del navegador al entrar en la carta si especificamos true.	no
ordered=boolean	Da una indicación acerca de la organización del contenido de la carta.	no
onenterforward="href"	Salta a dirección especificada. Ocurre cuando se entra en una carta que utiliza go.	no
onenterforward="href"	Salta a dirección especificada. Ocurre cuando se entra en una carta que utiliza prev.	no
ontimer="href"	Salta a dirección especificada tras una cantidad de tiempo especificada.	no
xml:lang	Especifica el lenguaje formal o natural en que el documento está escrito.	no
class	Afilia un elemento a una o más clases.	no

Tabla 7. Atributos de la etiqueta CARD

He aquí un ejemplo muy simple con 3 cartas y utiliza enlaces para pasar de unas a otras.

```
<wml>

  <card id="carta1" title="Título 1">
    <p><a href="#carta2">contenido carta 1</a></p>
  </card>

  <card id="carta2" title="Título 2">
```

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

```

    <p><a href="#carta3">contenido carta 2</a></p>
  </card>

  <card id="carta3" title="Título 3">
    <p><a href="#carta1">contenido carta 3</a></p>
  </card>

</wml>

```

B.1.9.7 Template

Esta etiqueta permite utilizar otras etiquetas que sirven para especificar tareas que puedan ejecutarse en cualquier carta de la baraja. Las tareas especificadas por el patrón se ejecutarán a no ser que se evite volviendo a especificar otra acción para la misma tarea dentro de la carta.

En un patrón se pueden incluir las siguientes tareas:

- do
- onevent
- onenterforward
- onenterbackward
- ontimer

En el siguiente ejemplo se hace uso de un patrón en el que definiremos una acción asociada a una baraja con 3 cartas. La acción es volver a la carta anterior, pero en la segunda carta deshabilitaremos dicha acción:

```

<?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="options" name="prev" label="anterior">

```

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

```

    <prev/>
  </do>
</template>

<card id="carta1">
  <do type="accept" name="next" label="siguiente">
    <go href="#carta2"/>
  </do>
  <p>Contenido de la primera carta</p>
</card>

<card id="carta2">
  <do type="accept" name="next" label="siguiente">
    <go href="#carta3"/>
  </do>
  <do type="options" name="prev">
    <noop/>
  </do>
  <p>Contenido de la segunda carta</p>
</card>

<card id="carta3">
  <do type="accept" name="next" label="options">
    <go href="#carta1"/>
  </do>
  <p>Contenido de la tercera carta</p>
</card>
</wml>

```

B.1.9.8 Tareas

Con la etiqueta **do** podemos especificar tareas que los navegadores realizarán al presionar ciertas teclas o navegar a través de cartas y barajas. A continuación se enumeran las acciones más habituales que se pueden realizar.

□ go

Produce un salto hacia otra página wap o hacia otra carta. En **href** se indica el nombre de otra carta, baraja o dirección wap a mostrar. Veamos un ejemplo:

```
<go href="http://www.ucauca.edu.co"/>
```

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

Estos son los atributos de "go":

atributo	Acción	obligatorio
Ref.	Dirección a la que saltar.	si
senreferer=boolean	Si es true, debe enviarse la dirección de la carta al servidor	no
method="post get"	Especifica el método de transmisión de datos.	no
enctype="tipo de contenido"	Cuando method="post" podemos utilizar application/x-www-form-urlencoded (por defecto) o multipart/form-data (wap 1.2). Si method="get" sólo podremos emplear el primero.	no
Id	Proporciona un nombre único dentro de una baraja.	no
Class	Afilia un elemento a una o más clases.	no

Tabla 8. Atributos de la etiqueta GO

Un ejemplo sencillo:

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

<wml>
  <card>
    <do type="accept">
      <go href="http://www.ucauca.edu.co"/>
    </do>
    <p>Ejemplo de go...</p>
  </card>
</wml>
```

□ prev

La etiqueta **prev** indica al navegador que vuelva a la dirección anterior que tenga en la pila. Como efecto lateral elimina la carta actual de la pila de cartas visitadas. Sus únicos atributos son:

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

atributo	acción	obligatorio
Id	Proporciona un nombre único dentro de una baraja.	no
Class	Afilia un elemento a una o más clases.	no

Tabla 9. Atributos de la etiqueta PREV

A continuación se verá un ejemplo que hace uso de prev par volver a una carta anteriormente visitada:

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

<wml>
  <card id="carta1" ontimer="#carta2">
    <timer name="t1" value="10"/>
    <p>esta es la primera carta</p>
  </card>
  <card id="carta2">
    <do type="options" label="anterior">
      <prev/>
    </do>
    <p>esta es la segunda carta</p>
  </card>
</wml>
```

□ refresh

Esta etiqueta provoca una actualización del contexto del navegador, especificado por la etiqueta setvar. Como efecto lateral del cambio de estado puede cambiar el estado de la pantalla durante el procesamiento de refresh.

□ noop

Esta etiqueta especifica que no debe hacerse nada. Suele emplearse para evitar acciones por defecto habilitadas para toda una baraja mediante un patrón (template). En el siguiente

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

ejemplo se utiliza para deshabilitar la opción de volver, especificada en el patrón, en la segunda carta:

```
<?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="options" name="prev" label="anterior">
      <prev/>
    </do>
  </template>

  <card id="carta1">
    <do type="accept" name="next" label="siguiente">
      <go href="#carta2"/>
    </do>
    <p>Contenido de la primera carta</p>
  </card>

  <card id="carta2">
    <do type="accept" name="next" label="siguiente">
      <go href="#carta3"/>
    </do>
    <do type="options" name="prev">
      <noop/>
    </do>
    <p>Contenido de la segunda carta</p>
  </card>

  <card id="carta3">
    <do type="accept" name="next" label="options">
      <go href="#carta1"/>
    </do>
    <p>Contenido de la tercera carta</p>
  </card>
</wml>
```

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

B.1.9.9 Variables

Una de las grandes diferencias entre el wml y el html es que con el wml se puede definir variables en las cartas, asignarles valores y presentarlos en la pantalla, incluso utilizar las variables en expresiones (programa.pl?f=\$(mivar), etc.)

La mayor ventaja de todo esto es que se puede conservar información en el paso de una carta a otra y así poder dividir el contenido en varios pasos.

Las variables son cadenas de texto (que distingue mayúsculas y minúsculas, como siempre en wml) a la que se le asigna un valor (secuencia de caracteres) o ningún valor.

El nombre de la variable puede empezar por el guión bajo: "_" o una letra US-ASCII, seguida de una o más letras US-ASCII, números o el guión bajo. Ejemplos de nombres de variables válidos: NOMBRE_Usuario _mivar1 X700II

□ Creación e inicialización de variables

Hay varias formas posibles:

- Con **setvar**

```
<setvar name="nombre" value="Pedro"/>
```

Pedro es el valor de la variable nombre. Con setvar se crea la variable y a la vez se le asigna un valor.

- Con **input**

Con input se puede crear (declarar) la variable, asignarle un valor y también devolverle el valor original.

Input se utiliza para la entrada de datos y esos datos (valores) se asignarán a una variable.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

- **Con select**
Con select se puede crear (declarar) la variable, asignarla un valor y también devolverla el valor original. Select permite seleccionar al usuario una o más valores entre una lista de opciones que se asignarán a una variable.
- **Con postfield**
Con postfield se puede crear (declarar) la variable, asignarla un valor y también devolverla el valor original.

□ Referenciar variables

Podemos incluir el valor de una variable dentro de un documento wml, tanto para que aparezca en la pantalla como para que se envíe a un programa.

Hay tres formas de referenciar la variable:

- **\$nombrevariable**
se utiliza cuando no hay ambigüedad con el nombre de la variable dentro del contexto.
- **\$(nombrevariable)**
Cuando puede existir ambigüedad con el nombre de la variable dentro del contexto.
- **\$(nombrevariable:conversión)** Se explica más abajo.

Cómo el wml se reserva el uso del signo de pesos, para que queramos que aparezca en la pantalla este signo, hay que escribirlo dos veces (\$\$). Por ejemplo tenemos esta variable: \$moneda y queremos presentar su valor en la pantalla, así: \$15, escribiremos:
Saldo actual: \$\$\$moneda

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

B.1.9.10 Conversión al formato escape

En su día se acordó una sustitución de algunos caracteres propios de los URLs para que el servidor no los confundiera (reglas del formato escape, RFC2396). Estas reglas facilitan un mecanismo para poder incluir en una línea URL, esos caracteres. Aunque estas reglas fueron creadas para referenciar URLs, se pueden aplicar las mismas reglas para referenciar variables.

En este ejemplo se convierte el texto en formato escape

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

<wml>

  <card id="carta1" title="conversión">
    <p>
      Teclea un texto con caracteres especiales:
    </p>
    <p><input type="text" name="texto"/></p>
    <do type="accept">
      <go href="#carta2"/>
    </do>
  </card>
  <card id="carta2" title="conversion">
    <p>
      texto normal: $(texto)
      <br/>
      texto en formato escape: $(texto:e)
    </p>
  </card>

</wml>
```

Cuando se introduce el valor de una variable dentro de una baraja, se puede definir el formato (escape, unescape o no escape) así:

\$(var:e)	traduce al formato escape
\$(var:E)	
\$(var:escape)	

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

\$(var:u) \$(var:U) \$(var:unesec)	traduce del formato escape a texto US-ASCII
\$(var:n) \$(var:N) \$(var:noesc)	hacen que el valor de la variable no se traduzca al formato escape

Tabla 10. Formato Escape

El WML siempre aplica el formato escape cuando se trata de atributos que trabajan con URLs. Por eso muchas veces se puede asumir que el WML convertirá al formato escape cuando lo tiene que hacer. De todas formas, conviene incluirlo si se tienen mínimas dudas de lo que ocurrirá.

Aquí se muestra una lista de los caracteres de escape (todos empieza por %):

carácter	formato escape	carácter	formato escape	carácter	formato escape
;	%3b	/	%2f	?	%3f
:	%3a	@	%40	&	%26
=	%3d	+	%2b	\$	%24
,	%2c	espacio	%20	{	%7b
}	%7d		%7c	\	%55c
^	%5e	[%5b]	%5d
`	%27	<	%3c	>	%3e
#	%23				

Tabla 11. Caracteres Escape

B.2 PROGRAMACION ASP y WML

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

B.2.1 Introducción a la programación de ASP y WML

La programación de páginas WML más interactivas se puede realizar gracias a la utilización de una tecnología fácil y que ha demostrado ser lo suficientemente potente y versátil para ser utilizada dentro de los desarrollos móviles, estamos hablando de ASP o Active Server Pages.

B.2.2 Introducción a ASP

B.2.2.1 Definición de ASP:

ASP es un entorno para crear y ejecutar aplicaciones dinámicas e interactivas en la Web es decir, mediante el uso de ASP se pasa a mostrar en los navegadores distinto contenido en función de las acciones/preferencias del usuario, el ejemplo más claro de uso se tiene en las consultas a bases de datos donde en función de los datos que el usuario selecciona en su terminal obtiene unas respuestas o por poner otro ejemplo una aplicación de comercio electrónico en la que se implemente un carrito de la compra y se vaya llenando con los productos que se va a seleccionar de las diferentes listas de productos que se muestran.

La potencia que tiene es enorme aunque su mecanismo de funcionamiento es básicamente muy sencillo: las páginas ASP comienzan a ejecutarse cuando un usuario solicita un archivo .asp al servidor Web a través del navegador, seguidamente el propio servidor web llama a una parte de su arquitectura que implementa ASP, en ese momento es cuando se lee el archivo solicitado, se ejecuta las secuencias de comandos contenidas en el archivo y se envía los resultados al explorador del cliente. Todas las secuencias de comandos se ejecutan en el servidor, y no en el cliente, de este modo se descarga al navegador cliente de hacer trabajo innecesario liberando el canal de comunicación de datos superfluos e innecesarios para el usuario (fundamental en las comunicaciones en general pero en telefonía móvil mucho más), además dado que todo se realiza en el lado servidor (que generalmente suele ser un equipo potente) se mejora el rendimiento y se consigue el propósito inicial, obtener páginas dinámicas en el terminal del usuario. En la tabla 12 se tiene un pequeño resumen de las características fundamentales de las páginas ASP.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

- Los archivos se identifican por .ASP
- Las secuencias de comandos se ejecutan en el servidor
- Las secuencias de comandos nunca se visualizan en el explorador del cliente solo sus resultados
- Los archivos .ASP son archivos normales de texto que contienen código ASP y se comportan como archivos WML normales cuando los devuelve el servidor Web.
- Pueden mezclar en una misma página diferentes secuencias de comandos de lenguajes distintos por ejemplo VBScript y Jscript (aunque esta práctica no es muy recomendable)
- Mejora del rendimiento y eficiencia de la red
- Aumento de la interactividad de las páginas
- Fácil de aprender

Tabla 12. Características de ASP

B.2.3 Desarrollo de aplicaciones en ASP

El lenguaje de secuencia de comandos que se va a utilizar es VBScript debido a su sencillez y potencia.

Una pagina ASP mezcla en el mismo archivo secuencias de comandos con código WML standard. Las secuencias de comandos se distinguen del resto del texto del archivo mediante delimitadores que vienen representados por los símbolos "<% " y "%>" para enmarcar las secuencias de comandos.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

Por ejemplo en el siguiente programa devuelve la hora actual haciendo uso de una función de Vbscript llamada Now que da la hora:

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

<wml>

    <card id="Principal" title="Principal">
        <p><b><u>EJEMPLO 1 </u></b></p>
        <p>Son las: <%= Now () %></p>
    </card>
</wml>
```

Dentro de los delimitadores ASP se puede incluir cualquier instrucción, expresión, procedimiento u operador válido para el lenguaje de programación que se esté utilizando.

B.2.3.1 Fundamentos básicos de vbscript

ASP viene de forma nativa con dos motores de secuencia de comandos: Microsoft Visual Basic Scriptig Edition (VBScript) y Microsoft Jscript aunque se puede instalar y utilizar motores de otros lenguajes, por ejemplo Perl. El mecanismo para establecer el lenguaje principal de secuencia de comandos depende de si lo que se quiere es activarlo en una única página o en toda la aplicación, en caso de ser en toda la aplicación se puede hacer configurando el IIS para uso de ASP o si lo que se quiere es establecer el lenguaje principal de secuencia de comandos en una única página, hay que agregar la directiva <%@ LANGUAGE%> al principio del archivo .asp con una sintaxis similar a la siguiente:

```
<%@ LANGUAGE=Lenguaje_secuencia_comandos %>
```

en donde Lenguaje_secuencia_comandos es el lenguaje principal de secuencia de comandos que se va a establecer en esa página concreta. Es importante señalar que el valor de la página

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

(el asignado por la directiva Language) invalida el valor global dado a todas las páginas de la aplicación. En la tabla 13 se resumen algunas de las características básicas de VBScript.

- VBScript acepta comentarios marcados con apóstrofes que se eliminan al procesarse la secuencia de comandos y no se envían al explorador.
- VBScript no distingue entre mayúsculas y minúsculas.
- Es muy similar a Visual Basic por lo que su aprendizaje es muy sencillo.
- Solo tiene un tipo de datos, el tipo Variant aunque existen subtipos implícitos dentro de él para manipular la información de manera más precisa.
- No es necesario declarar las variables que se usen (si conveniente).
- Restricciones de los nombres de variables:
 - Debe comenzar con un carácter alfabético.
 - No puede contener un punto.
 - No debe superar los 255 caracteres

Tabla 13. Características básicas de VBScript

B.2.3.2 Tipos de datos, variables y arrays en vbscript

Vbscript solo tiene un tipo de datos llamado Variant, este tipo es un es una clase especial de datos que puede contener diferentes tipos de información, de modo que su comportamiento varía en función de la información que se encuentra almacenada en él, solo hay una excepción a esta regla y es cuando se fuerza de manera explícita que los números se comporten como cadenas poniéndolos entre comillas (" ").

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

Un aspecto importante que existe en VBScript es que las variables no necesitan de declaración explícita para ser utilizada, aunque se recomienda su declaración para prevenir posibles errores.

Para declarar una variable se utiliza una sintaxis similar a la siguiente:

```
<% Dim Mivariable %>
```

El operador de asignación es el = y puede usarse para dar a la variable valores numéricos (MiVariable = 4), de cadena (MiVariable = "Hola") y fecha (MiVariable=#12-1-1999#)

Para declarar una variable de tipo arreglo solo se tiene que posponer al nombre de la variable unos paréntesis dentro de los cuales se sitúa el número de elementos de los que constará el array, en caso de tener múltiples dimensiones se separan por comas (en VbScript el límite está en 60 dimensiones distintas un número lo suficientemente alto como para poder trabajar con comodidad).

B.2.3.3 Control de flujo en vbscript

En VBScript los elementos de control de flujo para los programas son los típicos de cualquier lenguaje de programación, se tiene:

- Sentencias condicionales: representadas por el conjunto de expresiones siguientes:

```
IF Condición THEN
```

```
.... Acciones
```

```
ELSE
```

```
.... Acciones
```

```
END IF
```

Si se cumple la condición el flujo del programa nos llevará por la primera rama del IF y sino por la segunda.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

- Bucles: Se tienen dos fundamentalmente, el bucle DO WHILE y el bucle FOR, con una sintaxis como sigue:
DO WHILE Condición
.... Acciones
LOOP
- En este caso las acciones que hay dentro del bucle se ejecutan mientras la condición sea verdadera.

```
FOR Variable=ValorInicio TO ValorFin  
..... Acciones  
NEXT
```

Y en este otro caso las acciones se ejecutan hasta que la Variable alcanza el valor de ValorFin, en cada iteración se incrementa su valor en 1 comenzado en el valor marcado por ValorInicio. Un tipo de bucle FOR especial en VBScript es el que permite recorrer colecciones (objetos contenedores de otros objetos), la sintaxis es similar:

```
FOR EACH Elemento IN Colección  
.... Acciones  
NEXT
```

Este bucle recorre todos los elementos que haya dentro de la colección indicada. Los bucles y las sentencias condicionales pueden anidarse unas con otras tantas veces como se quiera.

B.2.4 Objetos integrados en ASP

Dentro de ASP existen una serie de objetos integrados que aportan funcionalidad añadida y que pueden ser utilizados dentro de estos desarrollos. Estos objetos son los siguientes:

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

B.2.4.1 Objeto Session

El objeto Session permite almacenar la información necesaria para una determinada sesión de usuario. Las variables almacenadas en el objeto Session no se descartan cuando el usuario pasa de una página a otra dentro de la aplicación, si no que dichas variables persisten durante todo el tiempo que el usuario tiene acceso a las páginas de la aplicación. También puede utilizar los métodos de Session para terminar explícitamente una sesión y establecer el periodo de tiempo de espera de inactividad de las sesiones. Las variables de Session de un cliente solo pueden ser accedidas por ese cliente. El servidor crea automáticamente el objeto Session cuando un usuario que no tenga actualmente una sesión solicita una página Web de la aplicación.

El mecanismo para crear variables dentro del objeto Session es sencillo y sigue un esquema como el siguiente:

```
<% Session("Nomvariable")= valor %>
<% Session ("Marca")="FORD" %>
```

pudiendo recuperar el valor almacenado en dicha variable del modo siguiente:

```
<% MarcaF = Session("Marca") %>
```

Generalmente las variables de este objeto se declaran dentro del archivo Global.ASA. El método más utilizado de este objeto es Abandom que destruye todos los objetos y variables almacenados en el objeto Session (<% Session.Abandom %>)

B.2.4.2 Objeto Application

Se utiliza para compartir información entre todos los usuarios de una aplicación. Este objeto dispone de dos métodos básicos que se utilizan para asegurar la integridad cuando múltiples usuarios están accediendo a sus propiedades. Estos métodos son Lock y Unlock. El método Lock bloquea el objeto Application de manera que sólo un cliente puede modificar/acceder a las propiedades de este objeto al mismo tiempo. El método Unlock tiene el efecto contrario y

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

desbloquea el objeto Application para que pueda ser modificado por otro cliente después de haberse bloqueado mediante el método Lock. Si no se llama a este método de forma explícita, el servidor Web desbloquea el objeto Application cuando el archivo .asp termina o transcurre su tiempo de espera. Un ejemplo muy claro de la utilización de Application es suponer que se define un propiedad llamada Nvisitas (declarada dentro del archivo Global.asa, las variables del objeto Application se declaran del mismo modo que el objeto Session) que contiene el número de visitantes de la página que se quiere. Cuando se realice dicha actualización se hace de manera exclusiva por cada usuario de manera que quede al finalizar el proceso la propiedad en un estado consistente. Para asegurar esto, se puede hacer uso de Lock y Unlock del siguiente modo:

```
<% Application.Lock<br>  
Application("Nvisitas") = Application("Nvisitas")+1<br>  
Application.Unlock %>
```

B.2.4.3 Objeto Request

Sirve para tener acceso a la información que se pasa en las peticiones HTTP, como son los parámetros que se pasan desde los documentos WML mediante los métodos POST y GET de HTTP o las cookies. El objeto Request se caracteriza por disponer de varias colecciones que se utilizan para acceder a la información pasada por el terminal de manera que dependiendo de la forma en que enviemos los datos al servidor tendremos que utilizar unas u otras. Las más útiles son:

- **FORM** que recupera datos enviados desde un formulario mediante el método POST.
- **QUERYSTRING** recupera datos enviados como cadena de consulta HTTP.
- **COOKIES** recupera los valores de las Cookies.

B.2.4.4 Objeto Response

El objeto Response se utiliza para controlar la información que se envía al usuario. Esto incluye el envío de información directamente al terminal, la redirección del explorador a otra dirección URL o el establecimiento de valores de las cookies (más adelante se hablará con más detalle sobre este tema). Entre los métodos más interesantes del objeto Response están los siguientes:

- **Write:** Escribe una cadena de resultado en el terminal del cliente. Cuando se hace uso de la expresión `<%=variable%>` para devolver el valor de una variable se está usando el método `Response.Write` implícitamente.

Por ejemplo si se hace algo similar a lo siguiente:

```
<% response.write "WML/ASP " %>
```

se obtendría en el display de el terminal el mensaje WML/ASP.

- **Redirect:** Este método hace que el terminal se conecte con una dirección diferente. Su uso y sintaxis son también muy sencillos:

```
<% response.redirect <a href://www.ucauca.edu.co>www.ucauca.edu.co </a>%>
```

nos mandaría a la página de la Universidad del Cauca.

B.2.4.5 Objeto Server

Da acceso a los métodos y las propiedades del servidor. De este objeto conviene destacar lo siguiente:

Con respecto a las propiedades se tiene a `ScriptTimeout` que especifica la cantidad máxima de tiempo que puede tardar la ejecución de una secuencia de comandos, conviene tener este parámetro muy en cuenta ya que muchos errores que se pueden obtener en ASP se deben a falta de tiempo en la ejecución del script (pocos recursos, acceso a base de datos muy

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

complicados etc) y que pueden fallar ante un script aparentemente correcto. El tiempo asignado a esta propiedad viene dado en segundos, siendo su sintaxis similar a:

Server.ScriptTimeout = nº de segundos

El método más comúnmente utilizado con el objeto Server es el método CreateObject que sirve para crear una instancia de un componente ActiveX en el servidor. Su sintaxis es la siguiente:

Server.CreateObject (Objeto)

Donde Objeto es el tipo de instancia que se quiere crear, por ejemplo para una conexión a una base de datos se puede utilizar objetos ADO del modo siguiente:

Server.CreateObject (ADO.Connection)

En la tabla siguiente se tiene una relación de todos los métodos, colecciones y eventos que forman parte de los objetos anteriormente explicados y que puede servir como referencia:

OBJETO	Application	Session	Server	Request	Response
colecciones	Contents Satic Objects	Contents Static Objects	Script Timeout	Client Certificate Cookies Form Query String Server variables	Cookies
Métodos	Lock unlock	Abandon	Create Object	Binary Read	Add Header Append To Log

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

			Map path URL Encode		Binary Write Clear End Flush Redirect Write
Eventos	Application_On Start Application_On End	Session_On Start Session_On End	-	-	-
Propiedades	-	Code Page LCID Session ID Time Out	-	Total Byte	Buffer Control Charset Content Type Expires Expires Asolute Is Client Conected PICS Status

Tabla 14. Métodos, Colecciones y Eventos

B.2.5 El archivo Global.asa

Dentro de ASP existe un archivo que se sitúa dentro del directorio raíz del sitio que está desarrollando y cuyo nombre es Global.asa, la importancia de este archivo radica en que permite la automatización de los cuatro eventos básicos del servidor sobre los objetos Application y Session. En la tabla 15 se tiene la estructura que presenta este archivo.

```
<SCRIPT LANGUAGE="VBScript" RUNAT="Server">
Sub Application_OnStart
```

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

```
End Sub
Sub Application_OnEnd

End Sub
Sub Session_OnStart

End Sub
Sub Session_OnEnd

End Sub
</SCRIPT>
```

Tabla 15. Estructura archivo Global.asa

En el archivo existen dos eventos (OnStart y OnEnd) asociados a dos objetos distintos (Application y Session), el objeto Session dentro del archivo Global.asa se utiliza para establecer los parámetros que serán particulares para cada sesión de usuario que se conecte al servidor, por otro lado el objeto Application se usa para establecer los parámetros que serán globales, es decir, compartidos para todas las sesiones que se inicien en el servidor. Concretamente la ejecución de cada evento se produce en los siguientes instantes:

- Application_OnStart: Se ejecuta antes de que se cree nueva sesión por parte del primer cliente que se conecta al servidor.
- Application_OnEnd: Se ejecuta justo cuando la aplicación termina.
- Session_OnStart: Se ejecuta cuando el servidor crea una nueva sesión de manera que los comandos se ejecutan antes de enviar la página solicitada al cliente.
- Session_OnEnd: Se ejecuta cuando se abandona o se supera el tiempo de espera de una sesión.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

B.2.6 Cookies

Este mecanismo permite guardar información relativa a un usuario a lo largo de sus distintos accesos a las páginas, no es muy recomendable basar estos desarrollos en ellas ya que no todos los dispositivos las soportan bien pero a veces resultan útiles. Las cookies se almacenan en los terminales de los clientes y se transmiten en las cabeceras cuando se realiza la comunicación http. En ASP las cookies se implementan como colecciones y se utilizan mediante los objetos integrados Request y Response. La sintaxis de manipulación de cookies es la siguiente:

```
Response.Cookies(cookie)[(clave)].atributo] = valor
```

```
Request.Cookies(cookie)(clave)
```

Ejemplos:

Enviar una cookie simple:

```
<% Response.Cookies ("IDUsuario")="12" %>
```

Recuperar el valor de esa cookie :

```
<% ID=Request.Cookies ("IDUsuario") %>
```

Este es el código anterior dentro de un programa WML:

```
< % Response.ContentType = "text/vnd.wap.wml" %>
<% Response.Cookies ("IDUsuario")="12" %>
<?xml version="1.0"?>
<!DOCTYPE wml PUBLIC "-//WAPFORUM//DTD WML 1.1//EN"
"http://www.wapforum.org/DTD/wml_1.1.xml">

<wml>

    <card id="Principal" title="Principal">>
        <p><b><u>EJEMPLO 3 </u></b></p>
        <% DIM ID
            ID=Request.Cookies ("IDUsuario") %>
        <p> El ID de usuario es: <%= ID %> </p>
    </card>
</wml>
```

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

En el ejemplo anterior se envía al terminal del cliente una cookie llamada IDUsuario con un número identificativo, y en el cuerpo del programa se retorna dicha cookie para mostrarla en el terminal, un uso típico de las cookies podría ser a continuación examinar en una base de datos las preferencias sobre nuestro sitio que dicho usuario tenía y proceder a enviárselas. Es importante fijarnos en las cookies que están almacenadas en el terminal para ver que es lo que está pasando.

B.3 IIS

B.3.1 Windows 2000, un servidor en Internet

Windows 2000 incorpora, en cualquiera de sus ediciones (incluyendo el exitoso Windows 2000 Professional), y sin ningún coste adicional, un servidor Internet integrado: Internet Information Services 5.0 (IIS 5.0), heredero directo de la versión 4.0 que se instalaba en Windows NT 4.0 dentro del Option Pack. El nombre del producto es, como ya se ha comentado IIS. Sin embargo, en los menús que dan acceso a la consola de administración, el término que se encuentra es Internet Services.

IIS 5.0 es mucho más que un servidor web. De hecho, bajo IIS se refugian prácticamente todas las prestaciones relacionadas con Internet de Windows 2000, desde los servicios de correo SMTP y de noticias NNTP, a los servicios Web y FTP, pasando incluso, de manera tangencial pero irrenunciable, por los servicios de Terminal.

B.3.2 Servicios Web

Evidentemente, la primera prestación de IIS 5, es la de poder albergar sitios web. Eso no resulta especialmente novedoso, ya que esta prestación ya era ofrecida por la versión anterior. Sin embargo, existen algunas novedades o mejoras que se enumeran a continuación:

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

B.3.2.1 Servidores virtuales

Si utilizamos el servidor web para dar servicios Internet, es decir, si somos un proveedor de servicios Internet (ISP), o simplemente, si se desea en un entorno Intranet hospedar más de un sitio Web, se necesita que IIS 5 permita albergar servidores virtuales, es decir, diferentes sitios Web. IIS 5.0 permite definir servidores virtuales que compartan una misma dirección IP, o bien que utilicen IP diferentes. Cuando una petición Web llega desde un navegador, contiene un conjunto de información identificativa del servidor al que desea conectarse y de la página que se desea ver. Esta información contiene fundamentalmente tres partes: la dirección IP (tras la resolución de nombres DNS), el puerto de comunicación (port), y una tercera información que contiene la cadena de caracteres que el usuario ha escrito en el explorador para designar al dominio en el que desea encontrar la página, algo así como www.ucauca.edu.co. Esta tercera información se denomina "host header name" (nombre o culto de host) y es precisamente lo que utiliza IIS 5. Para saber a cual de los servidores virtuales que comparten una dirección IP se envía la petición.

Cada servidor virtual aparece como un nodo en "Microsoft Management Console" que se administra de manera separada. Cuando se instala el producto se nos crean dos servidores virtuales, el "Default Web Site" (sitio web por defecto) y el "Administration Web Site", (sitio web de Administración) que, como su nombre indica, está destinado a la administración del servidor a través del web y que se halla protegido al escuchar en un puerto diferente del 80. Cada uno de estos servidores virtuales puede administrarse separadamente en todos sus extremos. En particular, podemos indicar las tres informaciones identificativas del sitio Web.

- **Soporte para WebDAV:** WebDAV es una extensión al protocolo HTTP 1.1 que sigue un lema muy explícito: "makes the Web Writable". Es decir, se trata de un estándar que permite ejecutar acciones de gestión de archivos tales como escribir, copiar, eliminar o modificar, todo ello sobre una conexión HTTP. WebDAV es clave en la estrategia de convertir el web en una plataforma de desarrollo de aplicaciones. Exchange 2000, SharePoint Portal 2001 y BizTalk Server 2000 utilizan WebDAV de manera exhaustiva. Por ejemplo, BizTalk Server utiliza WebDAV como estándar para acceder y manipular la totalidad de los archivos que emplea, y siempre

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

accediendo a los receptáculos de información mediante conexiones HTTP. Los servicios de Internet de Windows 2000 lo incorporan como una novedad respecto a la versión anterior.

- **Compresión del protocolo HTTP:** también es novedad la posibilidad de que IIS 5 comprima los documentos previos a su envío al cliente, con el consiguiente ahorro de ancho de banda. Para que esta opción sea útil, evidentemente, es preciso que el cliente admita esta posibilidad. Por otra parte, no es recomendable que se emplee la compresión sin antes someter al servidor a una prueba de estrés. La sobrecarga de procesador a la que puede ser sometido el servidor puede ser inaceptable en servidores con muchos impactos. Sin embargo, puede ser una buena solución para pequeños sitios Web departamentales en servidores en extremos remotos de una WAN, por ejemplo.
- **Aislamiento de Web Sites, aplicaciones y servicios:** En IIS 4 era posible un mínimo aislamiento entre las distintas aplicaciones albergadas en un servidor. En IIS 5 esta posibilidad se ha extendido en múltiples direcciones. En primer lugar, cada aplicación web (un directorio a partir del cual hemos definido una aplicación), puede ejecutarse dentro del mismo proceso de IIS (Inetinfo), en un proceso externo (DLLHOST) o en un tercer proceso compartido por varias aplicaciones, opción esta última nueva en la versión 5. Por otra parte, es posible limitar el tiempo de proceso que va a utilizar una aplicación, para que no afecte al resto de las hospedadas en el servidor. Finalmente, una novedad más: IIS 5 permite monitorear (mediante anotaciones extra en los archivos de log) cual es la utilización de CPU que está haciendo cada una de los sitios Web hospedados en el servidor.

B.3.2.2 Servidor FTP

A pesar de que muchos están esperando su desaparición, el servicio FTP sigue siendo necesario e Internet Information Server 5 le da soporte. De hecho, el servicio ha sido mejorado, ya que IIS 5 incluye, por fin, la funcionalidad de que las descargas de archivos

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

desde un servidor FTP Microsoft puedan reanudarse en el punto de interrupción en caso de caída repentina de la conexión, sin necesidad de reiniciar el proceso.

B.3.2.3 Servicios SMTP y TNP

En Windows 2000, el servicio SMTP forma parte de los servicios incorporados al sistema operativo en IIS 5, igual que NNTP, si bien este último, a diferencia del primero, no se instala por defecto. Exchange 2000 utiliza estos dos servicios, de tal forma que los requiere para proceder a la instalación. No hay conflicto, sino dependencia.

B.3.2.4 Active Server Pages

Active Server Pages (ASP) es un entorno de scripting que Internet Information Server proporciona para la creación de aplicaciones web. En definitiva, una página de servidor activo es un archivo con extensión **.asp** que contiene una combinación de código HTML y scripts que se ejecutarán en el servidor. Para que este proceso pueda llevarse a cabo es necesario que el servidor disponga de una extensión a tal efecto. Cuando un usuario solicita, en su navegador, la visión de una ASP, el servidor llama a la extensión para procesar la página. Esta extensión ejecuta los scripts de servidor, se comunica con los componentes, construye un documento HTML puro y lo envía por la red al cliente. De este modo el cliente recibe código HTML puro, sin que quede constancia alguna del script de servidor que lo ha generado.

ASP ha tenido una popularidad extraordinaria, y la sigue teniendo, aunque en los próximos meses se van a ver extendidas por una nueva plataforma ASP+, en el ámbito de .NET. Sin embargo, se han cometido no pocos excesos con ellas y se ha incurrido en prácticas muy poco eficientes tendentes a enredar el código ASP. Sin embargo, el desarrollo de aplicaciones web comienza a estar maduro para una evolución hacia técnicas de desarrollo como ASP+, que emplean los mismos paradigmas, pero haciendo un uso más racional de los conceptos de los componentes y el desarrollo orientado a objetos.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

IIS 5 introduce algunas novedades en ASP como: el control de redireccionamiento entre páginas, mejoras en el control de errores, soporte para los Windows Script Components (pseudo componentes COM escritos en realidad como archivo de script), encriptación del código de script en el servidor para prever accesos indeseados a la lógica del servidor, así como algunas mejoras de rendimiento.

B.3.2.5 Administración mediante Web y administración delegada

En IIS 5 se potencia la posibilidad de administración remota a través del WWW de cada uno de los servidores haciendo uso del navegador. Esto permite, en primer lugar, resolver problemas que surjan en momentos en los que no se esté físicamente en contacto con la máquina. Además, la administración basada en web va a permitir descargar en los Webmasters de cada uno de los servidores virtuales las tareas de administración de sus sedes, sin que tengan que acudir a nuestras instalaciones, algo que normalmente no suele ser deseable y que debe evitarse.

B.3.2.6 Seguridad

Internet Information Server 4.0 se ha ganado una mala fama en términos de seguridad. Ya sea por la adicción de los hackers contra los productos Microsoft o por su propia vulnerabilidad, no han sido pocos los agujeros de seguridad que se han encontrado en los servidores web Microsoft. En cualquier caso se enumeraran algunos de los aspectos más reseñables:

- Nuevos métodos de autenticación con encriptación hash
- Mejoras a los servicios de Certificación ya incluidos en Windows NT Option Pack.
- Nuevos servicios de encriptación de 128 bits.
- Autenticación Kerberos integrada con Windows 2000

B.4 Bases de Datos

Las bases de datos relacionales, se empezaron a utilizar cuando existió la necesidad de crear un enlace entre datos de distintas bases, ya fuere por la necesidad de disminuir el espacio que ocupasen dos bases de datos con relación entre ellas o bien por la necesaria relación de datos entre dichas bases.

Aunque el campo principal de actuación y utilidad de las bases de datos relacionales es la gestión, también se podrá si fuera necesario adaptarlo a las necesidades propias de cada usuario.

B.4.1 Cómo son las bases de datos

Podemos definir a una base de datos, como un archivo en el cual se almacena información de cualquier tipo.

En dicho archivo la información se almacena en campos o delimitadores, o sea, podemos almacenar el Nombre y el Apellido de las personas de modo separado, de esta forma se puede sacar del archivo todos los nombres o todos los apellidos, tanto de forma separada como de forma conjunta.

Normalmente el numero de campos que se puede tener en una base varía según las necesidades que tengamos en cuanto a separación de datos, de forma que después se pueda sacar la información de forma ordenada y separada, aunque el resto de la información siga almacenada y guardada en la base de datos.

Aunque en realidad se ha de tener en cuenta, que una base de datos, tal y como se conoce, no es solo el archivo en donde se guarda los datos, si no que en dicho archivo se encuentra la estructura de los datos, o sea, para saber que longitud tiene cada campo, se debe saber como se llama el campo y que longitud en caracteres tiene, así como el tipo de datos que se almacenan en dicho campo. Se puede guardar desde letras a números o incluso otros datos

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

más sofisticados, dependiendo de la estructura de la base y del sistema que se tenga para saber cual es dicha estructura.

En realidad aparte de los datos que son almacenados en el archivo, también se tiene un Gestor de datos, en los cuales se informa el tipo de campo, los campos y la longitud de cada campo. Dicho gestor nos permite saber que hay en cada registro (un registro es una suma de campos, por ejemplo Tenemos a José Antonio LOPEZ, José Antonio lo guardamos en el campo Nombre y LOPEZ en el campo Apellidos, cada registro es cada persona que se almacena en la base, o sea una persona es un registro y cada registro esta constituido por los campos Nombre y Apellido).

Normalmente cuando se habla de base de datos, el archivo que contiene los datos y el gestor, actualmente se les denomina Tabla, lo único que cambia es que en un mismo archivo se puede almacenar varias tablas, o sea que se puede tener un solo archivo y dentro de este se puede tener varias bases de datos (proveedores, productos, clientes, etc...) separados en tablas, esta es la ventaja que actualmente hay con respecto al sistema antiguo, que obligaba a tener un archivo por cada base de datos, o sea que se tendría que tener un archivo para los proveedores, otro para los productos y así tantas bases como se tengan.

Aunque pudiera dar a pensar que se pueden alterar los datos teniendo en un mismo archivo varias tablas, en realidad están separadas, aunque estén en un mismo archivo, cada tabla esta delimitada y no existe la posibilidad que se pudieran mezclar, este es un problema por el cual no hay que preocuparse.

Actualmente hay varios tipos de bases de datos, las dbf , fox pro, oracle y otras de distinto tipo.

B.4.2 Tablas

El diseño de las bases de datos para relacionar datos, es muy importante, pero se debe tener en cuenta que quizás esta posibilidad no siempre se tenga a mano, o sea, es posible que se

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

necesite realizar un programa para comercio que ya ha tenido un programa, por lo que se va a trabajar con datos de bases ya creadas y por lo tanto se tendrá que utilizar bases de otras personas que quizás no hayan pensado en relacionar dichas bases.

Tanto si se tiene la oportunidad de crear las bases como si no se tiene, para realizar relaciones entre bases de datos (tablas, se acostumbra utilizar el nombre de tabla) se tendrá que realizar un campo que enlace las bases de datos.

Por ejemplo, para relacionar los proveedores con los productos se tendrá en la base de datos de proveedores un campo que coincida con otro campo con el mismo nombre en la base de datos de productos.

De esta forma cuando se saca un producto de la base de datos(tabla) de los productos, se busca en el campo de la base donde se guardó el código del proveedor y luego se busca en la base de datos(tabla) de proveedores dicho código, con esto se realiza el enlace entre bases de datos de distinto tipo y de distinto formato.

B.4.3 Campos clave

Anteriormente se vio la necesidad de crear la relación entre distintas bases de datos. Para dicha relación se ha creado un campo común en ambas bases de datos(tablas), dicho campo es en realidad el campo clave, o sea el campo que relaciona una base con otra.

Pero se debe tener en cuenta que se puede tener tantos campos claves como se quiera, o sea que se puede tener relación entre distintas bases(tablas).

Ahora surge la principal pregunta por qué tener distintas bases(tablas) si se puede poner muchos campos y almacenar en ellos los datos, y así se ahorra trabajo y enlace entre bases(tabla).

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

B.4.4 Creación de una base de datos

Aunque hay varias clases de tipos de datos y muchas formas de clasificarlos, a la hora de crear una tabla, la mayoría de los sistemas utilizan la misma forma.

En consecuencia el sistema de creación de una base de datos(tabla), suele ser:

- Pedir el nombre del campo que se quiere crear.
- Pedir el tipo de dato que se almacenara en dicho campo, normalmente puede ser **String**, **Numérico**, **Boolean**, **Decimal**, **Entero** y otros, que dependerán del gestor con el cual se creó la tabla, así mismo también se dará la posibilidad de definir los decimales para los campos numéricos y decidir el índice.
- **El índice** es un punto muy importante a la hora de crear una base de datos(tabla) con índices, un índice no es mas que el orden que se establece para dicho campo, con esto se consigue aumentar la velocidad de búsqueda a la hora de encontrar a un determinado registro.

En la actualidad, se podrá tener varias tablas en un mismo archivo y como consecuencia se podrá crear tantas tablas como se quiera.

B.4.5 Relaciones entre datos

La dependencia funcional en una base de datos, es aquella que determina la relación que hay entre las bases de datos que utilizan de modo conjunto una relación de datos.

Para averiguar la dependencia existente entre bases de datos, se debe analizar las propiedades de los campos, para así averiguar cual es la relación existente entre las bases y que tipo de relación hay entre ellas.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

Se debe tener en cuenta que cuando se vaya a realizar una relación entre bases de datos, aquellos campos que se han seleccionado para que sean los campos clave, que serán los que realicen el enlace entre bases, deben ser del mismo tipo, por ejemplo que se utilicen tipos numéricos, los dos campos en las bases deben ser numéricos, de éste modo no se tendrá que realizar funciones que conviertan los tipos.

B.4.6 Modelos de bases de datos

En el mercado se puede encontrar una gran variedad de bases de datos, de distintos formatos, tipos, usos, gustos, etc... pero evidentemente existe en dicho mercado una tendencia de uso hacia ciertas marcas, por así decirlo, de productos que integran totalmente o en parte a un determinado tipo de base de datos, lo que obliga de forma indirecta a conocer el funcionamiento de dichas bases de datos, o mejor dicho de la forma de utilizarla o la gestión de dichas bases.

En el mercado se encuentran 3 tendencias muy populares:

- La dbf , de DBASE(c) , que es muy conocida.
- La Mdb de Microsoft(c) , que es la que utiliza el Access(c) y otros productos de dicha compañía.
- La Oracle, es una de las más utilizadas mundialmente.

Y actualmente esta haciendo aparición la denominada Interbase, MySQL y la DB2, que aunque actualmente no sean muy utilizadas, se podrá convertir en otro estándar en la gestión de datos.

Se ha de tener en cuenta que la utilización de uno u otro tipo esta directamente ligado con las tendencias del mercado, y es el mismo mercado con su demanda quien determina cual es el que ostenta la supremacía durante un tiempo. Aunque su esquema técnico no hace falta conocer, si es necesario reconocer el tipo de base por si fuera necesaria su gestión.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

Existe otro tipo que es la db, conocida como paradox(c) pero es igual a la dbf, no presenta ningún inconveniente.

Para usar la DBF es necesario tener una de las versiones de DBASE(c), no es imprescindible pero si aconsejable tener algún gestor de datos que permita manejar este tipo de bases, en el mercado hay muchos y gratuitos.

La MDB, Microsoft(c), suele ofrecer productos que las manejan.

La más difícil es la Oracle, este tipo de base de datos, es difícil de manejar a no ser que se adquieran productos especializados, pero si se busca es posible encontrar algo.

B.4.7 El uso de las relaciones intertabla

Aunque a simple vista se pueda creer que no es necesario o funcional para una empresa, un tipo u otro de base de datos, la elección esta directamente relacionada con la productividad que vamos a tener utilizando la gestión de la base que hayamos escogido.

El motivo en que se fundamenta una elección es el siguiente:

Suponga que las bases no son relacionales, entonces se tiene una base para clientes, otros proveedores, otra factura, etc... para la gestión de cada una de las áreas de la empresa y seria de forma independiente y no relacionada, esto traería las siguientes consecuencias:

Si un cliente viene a comprar a la tienda, y se tiene que hacer la factura, conforme se vaya ingresando el código de los productos que haya comprado y las cantidades que se haya llevado, también en un papel aparte se tendría que apuntar cuales son esos productos y las cantidades, pues al no estar relacionadas las facturas con los productos disponibles en el almacén, se tendría que ir después a la base de productos y dar de baja lo que se ha vendido para poder controlar las cantidades que hay en el almacén.

Aplicación de supervisión dentro del modelo de gestión WBEM a través de terminales móviles WAP

Al estar relacionadas las facturas con los productos, cuando se realiza la factura, el programa de forma automática daría de baja las cantidades de cada producto del almacén, sin percatarse de ello.

Como este ejemplo, se puede aplicar al resto de áreas de la empresa, los pedidos, las llegadas de la mercancía, etc...

Esto conlleva a un aumento de la productividad, superior al 50%, en el tiempo y en la fiabilidad.

GLOSARIO

Autenticación Kerberos

La autenticación Kerberos solamente puede ser usada en escenarios remotos. El cliente debe estar habilitado para identificar el servidor y el servidor debe estar habilitado para identificar el cliente. Una de las mayores características de Kerberos, como aplicación de WMI, es que permite delegar la identificación del cliente a través de múltiples máquinas secundarias. En otras palabras, una petición de un cliente puede pasarse desde el servidor a otra máquina manteniendo aún la identidad del cliente.

Encriptación Hash

Hash es la transformación de cadenas de caracteres que usualmente tienen una longitud larga a un valor o llave que representa la cadena original. Hash es usada para acceder y recibir items en una base de datos puesto que es mas fácil encontrar el item usando una llave que buscando el valor original, además es usada en algoritmos de encriptación.

NTFS

Sistema de Archivos de windows NT (NTFS-NT file system) es el sistema que usan los sistemas operativos Windows NT para almacenar y recibir archivos en un disco duro. NTFS es equivalente a la tabla de localización de archivos de Windows 95 y al sistema de archivos de alto desempeño OS/2 *HPFS* (high performance file system).

TNP

Protocolo de transferencia de Noticias de la Red (NNTP-Network News Transfer Protocol) es el protocolo predominante usado por clients y servidores para gestionar las notas enviadas a un grupo de la red.

ACRONIMOS

NTFS Seguridad de Archivos de windows NT

DNS Servidor de Dominio de Nombres

IIS Servidor de Información de Internet

MMC Consola de Gestión de Microsoft

HTML Lenguaje de Marcado de HiperTexto

SMTP Protocolo Simple de Transferencia de Correo

FTP Protocolo de Transferencia de Archivos

NNTP Protocolo de Tranferencia de Noticias de la Red

MIME Extensiones Multipropósito de Correo Internet