

Lenguajes de marcado para Independencia Web de dispositivo

Manrique López <jsmanrique@gmail.com>

Resumen

Tradicionalmente, a la Web se ha podido acceder a través de un PC de escritorio, pero desde hace ya años es posible consultarla por medio de otros dispositivos, como los teléfonos móviles. Esto ha sido posible gracias al desarrollo de estándares de marcado para el contenido Web que se sirve a estos aparatos, sin embargo, la diversidad de dispositivos y sus capacidades ha provocado la evolución de estos estándares. Por eso, resulta interesante analizar esta evolución mostrando al final qué tendencias y desarrollos se están llevando a cabo.

Palabras clave

estándares, web, movilidad, W3C, Web Móvil, XHTML, XML, WML, XHTML Basic, XHTML Mobile Profile, DIAL

Introducción

Miles de dispositivos móviles con una amplia gama de características se usan hoy en día. La gente espera la misma calidad de servicio y riqueza de la información disponible cuando está en movimiento de la misma forma que la encuentran en la Web en sus ordenadores de sobremesa.

Esta diversidad supone retos significativos a los diseñadores Web y operadores móviles del mismo modo. ¿Cómo se puede crear contenido para más de 2.500 tipos diferentes de dispositivos móviles? ¿Cómo pueden ofrecer los operadores este contenido de forma que sea eficiente y satisfactorio para el cliente?

Una de las claves de la Web es el desarrollo del hipertexto como medio para representar los documentos, y los lenguajes de marcado como medio para generar hipertexto. Por eso, se han desarrollado lenguajes específicos para contextos de uso particulares como móviles o dispositivos especiales, donde la Web no puede ser representada de igual manera que en un PC de escritorio.

WML, el principio

En el año 1994, la empresa Openwave desarrolló un lenguaje de marcado para servir contenidos a través de Internet a dispositivos móviles, se llamaba *Handheld Device Mark-up Language* (HDML). Este lenguaje estaba inspirado en HTML introduciendo algunas características específicas enfocadas a teléfonos móviles y teniendo en cuenta el escaso ancho de banda (introduciendo el concepto de baraja (*deck*) y cartas (*card*)).

Posteriormente, Openwave se unió a otras compañías del sector (Nokia, Motorola, etc.), para crear un estándar global. Esta unión se conoció como WAP Forum (hoy conocida como *Open Mobile Alliance* (OMA), y de ella nació el *Wireless Application Protocol* (WAP) y el *Wireless Markup Language* (WML) [XHTML-MP_History].

WML es un lenguaje basado en XML utilizado para crear páginas para cualquier dispositivo basado en tecnología WAP, principalmente para teléfonos móviles, creado por la OMA en 1999 [WML].

Además de WML, el lenguaje de marcas para la creación de páginas, la tecnología WAP está formada también por el lenguaje de script WMLScript y por WTAI (*Wireless Telephony Application Interface*), un API de funciones disponible en dispositivos WAP que pueden ser invocadas desde WML y WMLScript.

WML soporta imágenes en formato WBMP, aunque ciertos dispositivos soportan más formatos

gráficos y texto al que se puede aplicar formato.

Los documentos WML se denominan barajas (*deck*) y pueden contener varias cartas (*cards*). Una carta se corresponde con la información presentada de una vez en la pantalla del dispositivo. Al acceder a un documento WML disponible en un servidor, se visualizará una de las cartas. Todas las cartas de la baraja estarán en memoria, por lo que el cambio de carta dentro del documento, utilizando la interfaz del navegador (o *micro-browser*, en la terminología WAP), no implica un intercambio entre éste y el servidor.

Ejemplo 1. Ejemplo de 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="carta1" title="CARTA 1">
    <p>
      Esto es una carta.
    </p>
    <p><a href="#carta2">Pulsa aqui para acceder a otra carta</a></p>
  </card>
  <card id="carta2" title="CARTA 2">
    <p>
      Esto es otra carta.
    </p>
  </card>
</wml>
```

Existen varias versiones de WML (1.1, 1.2, 1.3, 2.0,...) y de WMLScript (1.0, 1.1, 1.2, 1.3) en el parque actual de dispositivos móviles, lo cual puede complicar la adaptación de contenidos en cierta medida.

WML ha sido diseñado pensando en cómo proporcionar servicios a los teléfonos móviles básicos, y no pensando en distribuir información a través del canal Web. Por eso no se recomienda su uso para servir información, puesto que no aprovecha todo el potencial disponible a través de los perfiles de dispositivo o de las preferencias del usuario de los distintos dispositivos.

Una de las desventajas de WML es que obliga a los creadores de contenidos Web a aprender un lenguaje diferente del tradicional HTML, y por eso se desarrolló hacia una convergencia con este último. De esta manera, nació a través del W3C la propuesta del estándar XHTML *Basic* y a través de la OMA del XHTML *Mobile Profile* que se comentan a continuación.

XHTML *Basic* 1.0

Es un subconjunto de XHTML, que incluye un conjunto de módulos considerado mínimo, diseñado para clientes Web que no puedan soportar el conjunto completo de características de XHTML (por ejemplo, teléfonos móviles, PDAs, set-top boxes y paginadores). El DTD de XHTML *Basic* está implementado utilizando el concepto de módulo XHTML, tal y como se define en el documento "*Modularization of XHTML*" de W3C [XHTMLMOD]. Este tipo de documento es suficientemente rico para la autoría de contenidos e incluye imágenes, formularios, soporte básico de tablas y soporte de objetos.

Estructura	<body>, <head>, <html>, <title>,
Texto	<abbr>, <acronym>, <address>, <blockquote>, , <cite>, <code>, <dfn>, , <h1>, <h2>, <h3>, <h4>, <h5>, <h6>, <kbd>, <p>, <pre>, <q>, <samp>, , , <var>
Hipertexto	<a>

Listas	<code><dl></code> , <code><dt></code> , <code><dd></code> , <code></code> , <code></code> , <code></code>
Formularios básicos	<code><form></code> , <code><input></code> , <code><label></code> , <code><select></code> , <code><option></code> , <code><textarea></code>
Tablas básicas	<code><caption></code> , <code><table></code> , <code><th></code> , <code><td></code> , <code><tr></code>
Imágenes	<code></code>
Objetos	<code><object></code> , <code><param></code>
Metainformación	<code><meta></code>
Enlace	<code><link></code>
Base	<code><base></code>

XHTML *Basic* está diseñado como una base común que puede ser extendida con módulos totalmente nuevos o con módulos no incorporados inicialmente al lenguaje (por ejemplo, el módulo de *Scripting*, que no forma parte de XHTML *Basic* a día de hoy).

Las características de XHTML que no están presentes o están presentes parcialmente en el subconjunto XHTML *Basic* son:

- Hojas de estilo: No existe el elemento `<style>`. Se recomiendan hojas de estilo externas, que pueden ser incluidas mediante el elemento `<link>`. Los elementos `<div>` y `` y el atributo `class` son soportados para asociar información de estilo con la estructura. Se soporta también el atributo `media` para seleccionar las hojas de estilo apropiadas.
- Scripts y eventos: No están soportados los elementos `<script>` y `<noscript>`. Se ha considerado que, al estar XHTML *Basic* orientado a pequeños dispositivos con poca memoria y capacidad de procesamiento, no tiene sentido permitir la ejecución de programas mediante *scripting*. Es importante tener en cuenta que los contenidos deben estar diseñados para que sean legibles sin la ejecución de scripts. Tampoco hay soporte en el lenguaje para atributos asociados a manejadores de eventos (que son utilizados en definitiva para invocar código de scripts).
- Presentación: Se considera que muchos de los clientes Web no pueden mostrar fuentes que no sean monoespaciadas, por lo que el texto bidireccional, en negrita y otros elementos para extensión de texto no están soportados.
- Formularios: Se soportan formularios XHTML básicos. Puesto que solamente aquellos dispositivos con un sistema de ficheros local pueden utilizar elementos `<input>` de los tipos `file` e `image`, y los dispositivos móviles más simples no disponen de sistema de ficheros local, éstos no están incluidos en los formularios básicos. Los desarrolladores de contenidos deberían tener en cuenta también que el usuario no podrá introducir ciertos tipos de caracteres desde ciertos dispositivos con mecanismos de entrada de datos limitados.
- Tablas: Existe soporte de tablas XHTML básicas, pero se considera que las tablas pueden ser complicadas de mostrar en pequeños dispositivos. En concreto, hay que reseñar que en este lenguaje está prohibido el uso de tablas anidadas. La recomendación de W3C sobre XHTML *Basic* sugiere, a su vez, seguir las pautas WCAG 1.0 para crear tablas accesibles ([WCAG]).
- Frames: No hay soporte de frames, al considerar que dependen de una interfaz de pantalla que puede no ser aplicable a pequeños dispositivos.

Un documento conforme a la recomendación XHTML *Basic* es aquel que incluye las características consideradas obligatorias en la especificación. Estos documentos deben seguir todos los criterios siguientes:

- El documento debe ser creado conforme a las restricciones expresadas en el Apéndice B de la recomendación XHTML *Basic* [XHTML-Basic], que muestra los distintos DTDs útiles para la especificación de documentos (el DTD SGML, el DTD XML y el DTD basado en modularización).

- El elemento raíz del documento debe ser `<html>`.
- El elemento raíz del documento debe indicar el espacio nominal XHTML usando el atributo `xmlns`. El espacio nominal para XHTML es `http://www.w3.org/1999/xhtml`.
- Debe haber una declaración `DOCTYPE` en el documento antes del elemento raíz. El identificador público incluido en la declaración `DOCTYPE` debe hacer referencia a la DTD comentada usando el Identificador Formal Público correspondiente. El identificador del sistema puede ser modificado apropiadamente para reflejar convenciones de rango local.

Ejemplo 2. Declaración `DOCTYPE` para un fichero XHTML *Basic*

```
<!DOCTYPE html  
PUBLIC "-//W3C//DTD XHTML Basic 1.0//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml-basic10.dtd">
```

- El subconjunto del DTD no debe ser utilizado para sobrescribir ninguna de las entidades parámetro en el DTD.

Actualmente se está desarrollando la propuesta para la versión 1.1 de éste estándar, que incluye características de XHTML *Mobile Profile*, que se explica a continuación.

XHTML *Mobile Profile*

El tipo de documento XHTML *Mobile Profile* o XHTML-MP es un superconjunto de XHTML *Basic* creado por la *Open Mobile Alliance*, con algunas extensiones añadidas. Es una alternativa de facto a XHTML *Basic* y forma parte de la especificación WAP 2.0 (aunque se sigue manteniendo también el soporte de WML 1.x).

Los módulos adicionales que añade XHTML-MP son:

Formularios	<code><fieldset></code> , <code><optgroup></code>
<i>Legacy</i>	<code>start</code> , <code>value</code>
Presentación	<code></code> , <code><big></code> , <code><hr></code> , <code><i></code> , <code><small></code>
Hojas de estilo	<code><style></code>
Atributos de estilo	<code>style</code>

Como se puede ver, XHTML-MP permite romper con la separación de contenido y presentación. Esto se debe al problema asociado al retardo que supone para la visualización completa de la página la descarga e interpretación de una hoja de estilos externa y las limitaciones en la implementación de los navegadores Web móviles en cuanto a la carencia de gestión de cache de hojas de estilo. En la actualidad, se utiliza la versión 1.0 de XHTML-MP y está en preparación la versión 1.2.

El encabezado para los documentos XHTML-MP es el que se muestra a continuación:

Ejemplo 3. Encabezado de documentos XHTML-MP.

```
<?xml version="1.0" encoding="UTF-8"?>  
<!DOCTYPE html PUBLIC "-//WAPFORUM//DTD XHTML Mobile 1.0//EN"  
"http://www.wapforum.org/DTD/xhtml-mobile10.dtd">
```

En el uso cotidiano, el encabezado con la declaración XML y la declaración `DOCTYPE` que indica el DTD contra el que validar el documento podrían ser omitidos. Sin embargo, algunas pasarelas de red y navegadores la utilizan para rechazar contenido que no está bien formado, aunque este tipo de validación es más bien la excepción que la regla.

Es importante también tener en cuenta que los distintos fabricantes de navegadores Web para móviles han añadido en ocasiones atributos propietarios a algunos elementos del lenguaje XHTML-MP (por ejemplo, para añadir al lenguaje características de WML no presentes en XHTML). A pesar de no seguir la recomendación, en determinados contextos puede ser útil aplicar técnicas de adaptación de contenidos para aumentar la experiencia del usuario de estos dispositivos. En estos casos, los documentos XHTML-MP no podrán ser validados contra el DTD sugerido anteriormente, aunque los fabricantes ofrecen un DTD propio en el que se soportan sus extensiones propietarias al lenguaje.

El soporte de formatos gráficos en los navegadores WAP 2.0 también ha aumentado: si en WAP 1.x sólo se permitía el uso de WBMP, en esta versión más reciente se añade el soporte de GIF, JPEG y algunos sabores de PNG. Al igual que en WAP 1.x, los distintos fabricantes podrían soportar más formatos gráficos.

La fusión: XHTML *Basic* 1.1

Todos los estándares están en continua evolución, y este no va a ser una excepción. Ya se está trabajando en nuevas versiones de la especificación, en concreto la versión 1.1 del estándar XHTML *Basic* dirigido a unir los dos estándares existentes en uno solo [XHTML-Basic_1.1].

Las nuevas versiones trabajan para añadir otros módulos y extender las posibilidades con opciones de otras recomendaciones, como es el caso del atributo `inputmode` que se cuela desde la recomendación XForms 1.0 [XFORMS]. Este atributo, utilizado en controles de formulario vacíos `<input>` y `<textarea>` sirve para ofrecer al agente de usuario las pistas necesarias para proporcionar asistencia a la hora de completar los controles de formulario en base a la información que se espera sea introducida.

Ejemplo 4. Ejemplo de uso del atributo `inputmode` en controles de formulario

```
<input type="text" name="fax" inputmode="digits" />
```

En el ejemplo anterior el valor del atributo `inputmode` tiene como valor `digits`, lo que servirá para indicar al navegador que debería activar el teclado numérico para completar dicho campo (esto resultaría muy útil en teléfonos móviles, en los que para llegar a un carácter numérico hay que recorrer todas las opciones de un botón o cambiar el modo de edición).

Otras características añadidas a esta versión son:

- Formularios XHTML
- Eventos intrínsecos
- Atributo `value` para el elemento ``
- Elemento `<style>`
- El atributo `target`

Todas ellas vienen definidas en el documento sobre modularización de XHTML [XHTMLMOD]

DIAL, ¿el futuro?

Aparte de las nuevas versiones de XHTML *Basic* encaminadas a unificar los dos estándares existentes actualmente (*Basic* y *Mobile Profile*), se están desarrollando nuevas propuestas.

En este sentido, destaca el *Device Independent Authoring Language* [DIAL]. El *Lenguaje de Autoría Independiente de Dispositivo* es un perfil de lenguaje XML basado en los vocabularios XHTML2 y XForms además de hacer uso del vocabulario de metadatos para Selección de Contenido con Independencia de Dispositivo (DSelect) creado por el grupo de trabajo de Independencia de Dispositivo del W3C.

El objetivo de DIAL es facilitar la creación de páginas Web para ser presentada en dispositivos con diferentes capacidades, y consumido por usuarios con diferentes preferencias.

Cuando se solicita un documento DIAL este pasa por al menos un procesador de DIAL, que filtra los contenidos marcados con las expresiones DSelect. Para evaluar de estas expresiones, el procesador consulta el Contexto de Envío [DCO] y finalmente, se genera el contenido que es enviado al usuario que ha realizado la petición. Un ejemplo de código fuente de un documento DIAL podría ser:

Ejemplo 5. Selección de contenidos en DIAL

```
...
<sel:select>
<!-- check if the requesting user is a subscriber-->
<sel:when expr="dcn:search('userSubscriptionStatus') = premium">
  <!-- if so select this movie -->
  <object sel:selid="HelloWorldMovie" src="HW_movie_portrait.mov"/>
</sel:when>
<sel:otherwise>
  <!-- if not send an image instead -->
  <object sel:selid="HelloWorldMovie" src="HW_image.jpg"/>
</sel:otherwise>
</sel:select>
...
```

Los procesadores de DIAL al evaluar las expresiones DSelect generan una representación XHTML2 para el contenido en función del contexto de envío en el que se aplican. El documento *Delivery Context Interfaces* describe cómo representar el contexto de envío y cómo puede ser consultado por los procesadores de DIAL [DCI].

DIAL debería ser considerado para aquellos casos en los que el contenido Web debe adaptarse a contextos específicos de envío para los usuarios. Por ejemplo:

- para un gran rango de dispositivos (ordenadores, PDA, WebTV, etc.)
- cuando se debe acceder a la Web con diferentes conexiones y modalidades de interacción
- cuando las preferencias y políticas deben determinar qué contenidos representar

Conclusiones y desarrollos futuros

Actualmente, la adaptación de contenidos Web a dispositivos móviles se basa en la combinación de diferentes tecnologías para producir resultados acordes con los lenguajes de marcado WML o XHTML. Por tanto, existen alternativas al uso de DIAL a día de hoy, pero este estándar permite mejorarlas por usar características de negociación de contenidos unidas a una sintaxis declarativa en el propio lenguaje de marcado. Analizando las alternativas generales y cómo DIAL las conjunta, se tiene, en resumen:

- **Negociación de contenidos:** DIAL complementa esta alternativa al permitir al autor describir en la

propia página Web qué contenido y estructuras son relevantes para diferentes contextos de envío.

- **Sintaxis declarativa:** frente al uso de XSLT o plantillas en el servidor, DIAL permite al autor determinar qué partes del contenido deben tratarse en cada contexto de envío.

Además, como la lógica de selección se indica en el marcado de la página, independiza la adaptación del lenguaje de programación utilizado para seleccionar los contenidos.

- **Múltiples versiones del contenido:** El uso de un único contenido que se adapta a cada contexto de envío que se le presenta representa una mayor flexibilidad que tener todos los potenciales contenidos generados de antemano.

Pero además, DIAL presenta un par de ventajas adicionales:

- **Proceso distribuido:** los procesadores de DIAL pueden actuar en diferentes partes del envío (servidor, usuario, capas intermedias, etc.), con lo que se flexibiliza el proceso y se mejora la calidad de la adaptación.
- **Validez gramática:** Dado que DIAL es una gramática XML puede ser validada, cosa que no ocurre con otras soluciones enfocadas a la adaptación de contenidos como el uso de plantillas en el servidor que mezclan marcado y código (JSP, PHP, etc.).

Sin embargo, el mayor reto a día de hoy es conseguir una implementación de este estándar, y poder evaluar con detalle el rendimiento del sistema propuesto. Así que se está ante un reto de cara a los próximos años.

Bibliografía

- [XHTML-MP_History] Luca Passani XHTML-MP Style Guide Chapter 1 - History of XHTML Mobile Profile
http://developer.openwave.com/dvl/support/documentation/guides_and_references/xhtml-mp_style_guide/
- [WML] OPEN MOBILE ALLIANCE Wireless Markup Language Specification
<http://www.openmobilealliance.org/tech/affiliates/wap/wapindex.html>
- [XHTMLMOD] WORLD WIDE WEB CONSORTIUM. Modularization of XHTML™.
<http://www.w3.org/TR/2001/REC-xhtml-modularization-20010410/>
- [WCAG] WORLD WIDE WEB CONSORTIUM Web Content Accesibility Guidelines 1.0
<http://www.w3.org/TR/1999/WAI-WEBCONTENT-19990505/>
- [XHTML-Basic] WORLD WIDE WEB CONSORTIUM. XHTML Basic.
<http://www.w3.org/TR/2000/REC-xhtml-basic-20001219/>
- [XHTML-Basic_1.1] WORLD WIDE WEB CONSORTIUM. XHTML Basic 1.1.
<http://www.w3.org/TR/xhtml-basic/>
- [XFORMS] WORLD WIDE WEB CONSORTIUM XForms 1.0
<http://www.w3.org/TR/2006/REC-xforms-20060314/>
- [DIAL] WORLD WIDE WEB CONSORTIUM. Device Independent Authoring Language
<http://www.w3.org/TR/dial/>
- [DCO] WORLD WIDE WEB CONSORTIUM. Delivery Context Overview for Device Independence
<http://www.w3.org/TR/di-dco/>
- [DCI] WORLD WIDE WEB CONSORTIUM. Delivery Context: Interfaces (DCI) Accessing Static and Dynamic Properties
<http://www.w3.org/TR/DPF/>