



MINISTERIO
DE EDUCACIÓN
Y CIENCIA

SECRETARÍA GENERAL
DE EDUCACIÓN
Y FORMACIÓN PROFESIONAL

DIRECCIÓN GENERAL
DE EDUCACIÓN,
FORMACIÓN PROFESIONAL
E INNOVACIÓN EDUCATIVA

CENTRO NACIONAL
DE INFORMACIÓN Y
COMUNICACIÓN EDUCATIVA

Edición HTML

Introducción al XHTML



SERVICIO DE
FORMACIÓN DEL
PROFESORADO

C/ TORRELAGUNA, 58
28027 - MADRID

Índice de contenido

XHTML.....	3
¿Qué es?.....	3
¿Cuáles son los requisitos que debe cumplir?.....	3
Crear un documento XHTML con N VU.....	4
Convertir documentos HTML a XHTML.....	4
Modificación de dos preferencias básicas de HTML-Kit.....	5
Convertir documentos de HTML a XHTML con HTM-Kit.....	6

XHTML

¿QUÉ ES?

Esta es la primera pregunta que surge cuando escuchamos este nuevo acrónimo. El lenguaje extensible de marcado de hipertexto, que sería la traducción del acrónimo XHTML y nace con la intención de sustituir al lenguaje HTML para superar las limitaciones de éste y adaptarse a la creciente cantidad de aplicaciones capaces de hacer uso del lenguaje XML. La idea base es combinar la sintaxis del lenguaje HTML, diseñado para mostrar los datos con la de XML, encargado de describir dichos datos.

El objetivo perseguido es que la información pueda ser interpretada correctamente independientemente del dispositivo desde el que se accede a ella. Para conseguir este fin, la especificación XHTML establece unas normas más estrictas que las del HTML para el etiquetado de los documentos.

¿CÚALES SON LOS REQUISITOS QUE DEBE CUMPLIR?

El esquema básico del documento, para considerarse conforme a la especificación deberá cumplir las siguientes condiciones:

1. El elemento raíz del documento debe ser `<html>`.
2. 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>
3. Debe haber una declaración DOCTYPE en el documento antes del elemento raíz. El identificador público incluido en la declaración DOCTYPE debe hacer referencia a alguna de las tres DTD definidas por el W3C usando el Identificador Formal Público correspondiente.

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

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

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

N|VU es capaz de cumplir estos requisitos y crear la base de documentos conformes con la especificación para los dos primeros tipos, ya que, igual que ocurría en HTML, no admite documentos basados en la DTD de marcos.

Además de lo anterior, y aunque las etiquetas serán las que ya conocemos puesto que proceden del HTML, habrá que cumplir algunas normas específicas para que el documento se pueda validar como "gramaticalmente correcto". Algunos de los más importantes serán:

1. Las etiquetas no podrán solaparse.
El código `<p>un fragmento <em>enfaticado</p></em>` que validaría correctamente en HTML 4 será incorrecto y tendrá que escribirse con las etiquetas correctamente anidadas: `<p>un fragmento <em>enfaticado</em></p>`
2. Los nombres de elementos y atributos deben escribirse siempre en minúsculas.
Aunque desde el principio recomendamos que se escribieran siempre en minúsculas,

- podríamos haberlo hecho en también en mayúsculas, pero XML es sensible a mayúsculas y por eso se adopta esta norma.
3. Con HTML 4.0, basado en SGML, en algunos elementos podía omitirse la etiqueta de cierre, de tal manera que la apertura de los elementos que les sucedían implicaba dicho cierre. Esta omisión no está permitida en XHTML, basado en XML. Todos los elementos que no estén declarados en la DTD como **EMPTY** deben tener una etiqueta de cierre. Por ejemplo: `<p>esto es un párrafo<p>y esto otro` sería correcto en HTML, pero tendría que convertirse en `<p>esto es un párrafo</p> <p>y esto otro</p>` para validar correctamente en XHTML.
 4. Todos los valores de atributos deben ir entrecomillados, incluso aquellos que son numéricos. Ya no valdrá `<table rows=3>` sino que habrá que escribirlo `<table rows="3">`
 5. XML no soporta la minimización de atributos. Los pares atributo-valor deben escribirse en toda su extensión, por lo que no podríamos hacer que un elemento de formulario apareciera marcado con el código `<input type="radio" name="verde" checked>`, sino que habría que escribirlo `<input type="radio" name="verde" checked="checked">`
 6. Los elementos vacíos, que no necesitaban una etiqueta de cierre en HTML, como `<br>` `<hr>` o `<img>` deben llevarla para validar correctamente en XHTML o bien terminar su etiqueta de apertura con `/>`. Para garantizar la compatibilidad es conveniente que este cierre de la etiqueta de apertura vaya precedido por un espacio con lo que las etiquetas mencionadas quedarían así: `<br />` `<hr atributos />` `<img atributos />`
 7. En HTML se podía identificar fragmentos del documento utilizando tanto el atributo `name` como el atributo `id`. En XHTML se utilizará siempre `id` incluso en aquellos casos en los que se había utilizado siempre `name`, pudiendo cada atributo `id` aplicarse a un único fragmento del documento.

Aunque en un principio pueda haberte parecido complicado, el objetivo de este epígrafe es sencillamente presentarte algunas normas específicas para que las conozcas, pero no deberías preocuparte por su aplicación, ya que N|VU se ajusta a las especificaciones y se encarga de crear el código acorde con las normas para que sea un documento bien formado y gramaticalmente correcto.

CREAR UN DOCUMENTO XHTML CON N|VU

Podemos decidarnos a escribir crear documentos XHTML como opción por defecto, para lo cual bastará con que lo indiquemos en el menú herramientas ➡ preferencias ➡ avanzado donde marcaremos como lenguaje preferido XHTML 1 y optaremos por utilizar un DTD transicional o strict. A partir de este momento, cada vez que pulsemos el icono  sin indicar ningún cambio el documento se creará de acuerdo a lo especificado.

Por otra parte, si desplegamos las opciones, elegimos archivo ➡ nuevo o pulsamos  +  dispondremos de dos casillas de verificación para indicar para indicar si queremos que el nuevo documento sea en lenguaje XHTML y con DTD strict. Ten en cuenta que existe un error en el programa por el que, cuando al crear un documento nuevo especifiquemos un lenguaje y una DTD esta elección sobreescribirá las que hubiéramos marcado en las preferencias generales.

CONVERTIR DOCUMENTOS HTML A XHTML

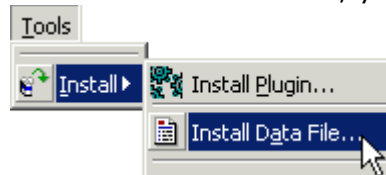
Podríamos convertir manualmente documentos creados originalmente en lenguaje HTML aplicando las normas que hemos comentado, pero puede ser una tarea muy tediosa.

El W3C desarrolló inicialmente una herramienta entre cuyas funcionalidades, además de la de analizar y validar documentos estaba la de corregir y convertir el código de HTML 4 a XHTML 1, siempre que el documento original no contuviera un número excesivo de errores. Dicha herramienta se denomina TidyHTML pero tiene el problema de tratarse de una aplicación en línea de comandos. Para facilitar el uso se ha desarrollado un interfaz grafico de usuario (GUI) que recibe el nombre de HTML-Kit que podrás descargar desde <http://www.chami.com/html-kit/download/>

Una vez instalado el programa original en inglés tendrás que descargar el parche de traducción al castellano desde

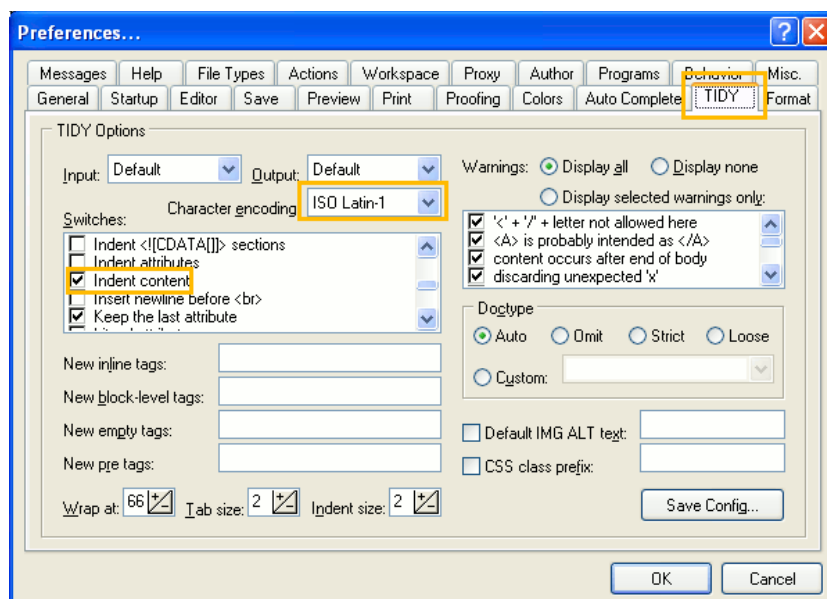
<http://www.chami.com/html-kit/plugins/info/irtranslationspanish/>

Se trata de un archivo comprimido pero no será necesario que lo instalemos manualmente, ya que el propio HTMLKit se encargará de instalarlo. Para ello arrancamos el programa y vamos al menú tools ➡ install ➡ install data file, elegiremos el archivo que acabamos de bajar y pulsaremos ok, contestando afirmativamente si se nos hace alguna pregunta más durante el proceso de instalación. Aunque comprobaremos que el programa cuenta con multitud de posibilidades vamos a concentrarnos en la que nos ha llevado hasta él. Para ello vamos primero a modificar un par de opciones en las preferencias y luego veremos el procedimiento para convertir un archivo a XHTML.



MODIFICACIÓN DE DOS PREFERENCIAS BÁSICAS DE HTML-KIT

Si no se aplica inmediatamente la traducción reiniciamos el programa y, ya con él en castellano, vamos al menú editar ➡ preferencias. Veremos aparecer un panel con multitud de pestañas y opciones (que siguen en inglés porque esta parte no está traducida).



Como ves en la imagen hemos seleccionado la pestaña **TIDY** y hemos modificado dos características:

- En **CHARACTER ENCODING** hemos utilizado el desplegable para seleccionar **ISO LATIN-1** (que es otra forma de denominar al conjunto de caracteres ISO-8859-1 que estamos usando habitualmente). Si no lo hacemos así el programa tomará por defecto el conjunto US ASCII con lo que nos encontraremos con problemas con los caracteres acentuados que no están incluidos en este conjunto.

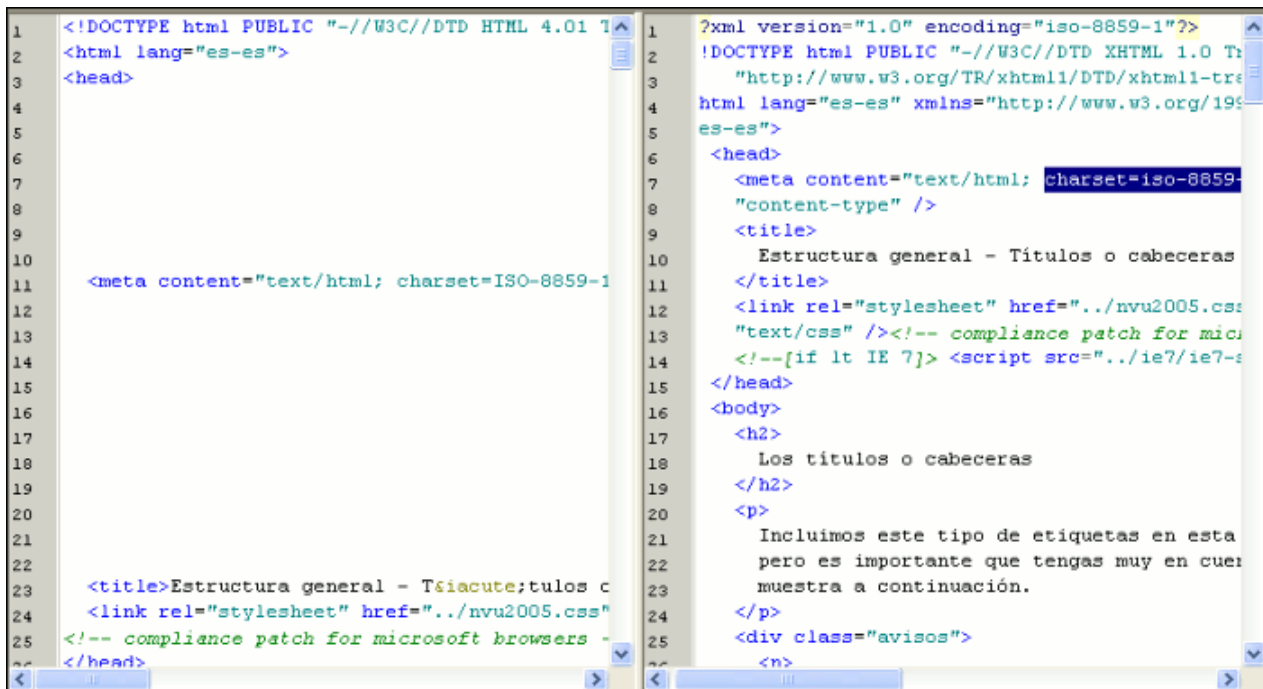
- La segunda opción que viene desmarcada por defecto pero que aquí hemos activado es **INDENT CODE** que hace que el programa tabule el contenido para facilitar su lectura cuando vemos el código. Si la activamos aumentará ligeramente el tamaño de los archivos pero, a cambio resultará mucho más fácil consultarlos cuando veamos el código. Queda pues a tu elección activarla o no.

Cuando pulsemos **OK** el programa quedará preparado para realizar el trabajo que le vamos a pedir.

CONVERTIR DOCUMENTOS DE HTML A XHTML CON HTM-KIT

El procedimiento a seguir es:

1. Abrimos el documento original con la opción habitual **ARCHIVO ➡ ABRIR ARCHIVO**
2. Acudimos al menú **ACCIONES ➡ TOOLS ➡ HTML TIDY ➡ CONVERT TO XHTML**
3. Veremos dividirse la pantalla en dos zonas: en la izquierda el documento original y en la derecha la conversión propuesta. Es conveniente que comprobemos que aparece el conjunto de caracteres ISO-8859-1 que habíamos establecido en las preferencias. Por cierto, puedes comprobar en la imagen que una de las correcciones que ha realizado el programa ha sido la supresión de los espacios en blanco que erróneamente introduce N|VU en el código.



4. Si estamos de acuerdo con las modificaciones propuestas pulsaremos sobre la zona del resultado con el botón derecho y elegiremos la opción  **Copiar Resultado al Editor** que trasladará el contenido de la zona derecha a la izquierda.
5. Por último salvaremos el archivo, bien con el mismo nombre o bien con otro diferente con las opciones habituales **ARCHIVO ➡ SALVAR O ARCHIVO ➡ SALVAR COMO**

Puedes encontrar un manual en castellano bastante amplio en <http://www.ignside.net/man/html-kit/index.php>

75

- Como es posible que te inquiete someter tus archivos a un programa que modificará automáticamente el código vamos a empezar a hacer "experimentos con gaseosa". Utilizaremos archivos ajenos, como puede ser éste que estás leyendo en este momento.
- Abre este archivo en N|VU y guárdalo en una carpeta en tu disco duro. La razón para hacer esto es evitar problemas con las rutas hacia las imágenes cuando modifiquemos el archivo original: al encontrarse en el CD-ROM no podríamos escribir en la ruta original y podría ocurrir que perdiéramos alguna referencia al guardarlo en el disco duro, mientras que haciéndolo así todos los archivos necesarios deberían guardarse en la misma carpeta que el código de la página.
- Abre el archivo que acabas de guardar en tu disco duro con HTML-Kit.
- Conviértelo a XHTML tal como se ha indicado y guárdalo con un nombre diferente. Puedes utilizar el propio HTML-Kit para abrir ambos archivos y comparar las diferencias de código.
- Abre ambos archivos con el navegador y compáralos para ver si encuentras alguna diferencia en la presentación. (No debería ocurrir y, gracias a la conversión tendrás un archivo escrito de acuerdo a las recomendaciones más actualizadas del W3C)
- Sigue practicando con otros archivos hasta que tengas suficiente seguridad para atreverte a convertir tu propio trabajo.