



TEMA 2.

DIFERENCIAS ENTRE XHTML Y HTML

Autor: Pascual Ligeró

Revisado por:

Fco. Javier Valero – franciscojavier.valero@ceedcv.es

2019/2020

Licencia



**CC BY-NC-SA 3.0 ES Reconocimiento - NoComercial -
CompartirIgual (by-nc-sa):** No se permite un uso comercial de la obra

original ni de las posibles obras derivadas, la distribución de las cuales se debe hacer con una licencia igual a la que regula la obra original.

NOTA: Esta es una obra derivada de la original realizada por Pascual Ligeró.

1. HTML Y DIFERENCIAS CON XHTML

HTML es un lenguaje que se utiliza para la creación de páginas en la WWW. Por página entenderemos el documento que aparece en el visualizador o navegador.

HTML se compone de una serie de comandos, que son interpretados por el visualizador, o programa que utilizamos para navegar por el WWW. En última instancia es el visualizador el que ejecuta todas las órdenes contenidas en el código HTML, de forma que un visualizador puede estar capacitado para unas prestaciones, pero no para otras. Así, podremos especificar que una página tenga una imagen de fondo, o un texto parpadeando, pero si nuestro visualizador no está capacitado para esas funciones, no podremos verlas.

El lenguaje **XHTML** es muy similar al lenguaje HTML. De hecho, no es más que una adaptación de HTML al lenguaje XML, el estándar XHTML 1.0 sólo añade pequeñas mejoras y modificaciones menores al estándar HTML 4.01, por lo que este último está prácticamente incluido en el primero, lo que hace que pasar del HTML 4.01 Strict a XHTML no requiere casi ningún cambio.

El lenguaje HTML tiene una sintaxis muy permisiva, por lo que es posible escribir sus etiquetas y atributos de muchas formas diferentes. Las etiquetas, por ejemplo, podían escribirse en mayúsculas, en minúsculas e incluso combinando mayúsculas y minúsculas. El valor de los atributos de las etiquetas se pueden indicar con o sin comillas. Además, el orden en el que se abrían y cerraban las etiquetas no era importante.

La flexibilidad de HTML da lugar a páginas con un código desordenado, difícil de mantener y muy poco profesional. XHTML soluciona estos problemas añadiendo ciertas normas en la forma de escribir las etiquetas y atributos.

1.1 Normalización. Versiones de HTML

Los programas capaces de traducir el código HTML y producir una salida en pantalla de los mismos son los navegadores (browsers en inglés). Se trata de un software gráfico que se inició con la creación de Mosaic a principios de los 90 y que poco a poco produjo más productos hasta llegar a una guerra de navegadores a finales de los 90 entre Internet Explorer de Microsoft y Navigator de Netscape que ganó Microsoft pero que ahora continúa con otros navegadores como el propio Internet Explorer, Mozilla Firefox (sucesor de código abierto de Netscape), Google Chrome, Opera o Apple Safari entre otros.

El problema surgió en cuanto unos navegadores incorporaron elementos HTML que el resto no traducía, con lo que aparecieron diferentes dialectos HTML. Así una página se podía mostrar de forma totalmente diferente según el navegador.

La solución pasó por intentar estandarizar el lenguaje. Por ello el propio Tim Bernes Lee fundó la **World Wide Web Consortium** (abreviado **W3C**) como organismo de estandarización del lenguaje HTML ante la industria.

En la actualidad las directrices de W3C son seguidas por la mayoría de navegadores aunque no al 100%, lo que sigue generando problemas a los creadores de páginas web.

La situación, sin embargo se ha complicado en estos últimos años con la aparición de diferentes estándares, en concreto actualmente se consideran estándares a HTML 4.01, XHTML 1.0 y 1.1 y ya se considera de la misma forma a HTML 5. Las diferencias entre ellos son:

- HTML 4.01. Se trata de la versión estándar del HTML tradicional hecha en el año 1999 y que sigue teniendo mucha vigencia actualmente. Hay tres versiones: la transicional (que permite seguir usando algunas etiquetas que se consideran obsoletas), la estricta (que elimina numerosas etiquetas y atributos para forzar a crear un HTML con menos formato y más significado) y la frameset orientada a usar los ya muy poco utilizados marcos. La versión más popular es la transicional al ser más libre.

Para avisar de que nuestro documento sigue las normas de HTML 4.01, se coloca una etiqueta DOCTYPE que permite llegar al DTD estándar de esta versión. Para HTML 4.01 transicional es:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"  
"http://www.w3.org/TR/html4/loose.dtd">
```

- XHTML. XHTML era planteado como el sustituto de HTML, la primera versión, la 1.0 sigue siendo la más usada y en la fecha de escritura de este texto es la versión HTML más utilizada. Hay también versión estricta, transicional y frameset; la más utilizada, otra vez es la transicional porque permite el uso de numerosos atributos y elementos que se consideran obsoletos, pero que muchos diseñadores utilizan.

Las páginas XHTML obligan a que la escritura de HTML siga las reglas del lenguaje XML bien formado.

Para indicar que utilizamos XHTML 1.0 transicional se debe poner este DOCTYPE:

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

1.2 XHTML: diferencias sintácticas y estructurales con HTML.

La diferencia fundamental entre HTML y XHTML está en el cumplimiento que exigen de XML. Así XHTML es, en definitiva, un dialecto de XML que, por lo tanto, cumple sus reglas de forma absoluta. De tal manera que cumple las reglas del XML bien formado:

- Los elementos se escriben en minúsculas obligatoriamente
- Todo elemento se debe cerrar y además se cierra primero el último que se abrió. Los elementos vacíos también hay que cerrarles (por ejemplo se escribe `
` para el salto de línea)
- Todos los valores de los atributos van entrecomillados
- Todo atributo debe de tener un valor
- Los símbolos `<`, `>`, `&` y comillas deben utilizar entidades y no escribirse tal cual

Sin embargo HTML no es tan estricto y permite que:

- Los valores de los atributos no tienen que ir entrecomillados (a no ser que tengan espacios en blanco)
- No todos los atributos tienen valores. Es decir no siempre se usa **atributo=valor** hay atributos que no tienen valor
- Las etiquetas HTML se pueden escribir como queramos, en mayúsculas o minúscula.
- Las etiquetas vacías no es obligatorio cerrarlas (se puede escribir `
`)