El misterioso mundo del encoding

Por Mauro Gullino

La desesperación que surge al ver nuestras páginas web con símbolos extraños se potencia al desconocer las causas. ¿Cómo arreglarlas?

¿Quién no ha visitado alguna vez un sitio web con signos extraños reemplazando las *eñes*, las letras con acento u otros caracteres especiales? Como desarrolladores, ¿cuántas veces hemos encontrado que al subir contenido de texto a nuestros sitios aparecen esos caracteres raros?

Si uno no los puso allí, ¿por qué aparecen?

La mayor dificultad para resolver los problemas de *encoding* es que no hay una solución universal y única. Se debe comprender el mecanismo del codificación de textos en las computadoras y el funcionamiento de varios elementos de la web para poder diagnosticar el problema y llegar a una solución. Es un tema muy complejo, pero trataremos de desglosarlo para conocerlo un poco más.

Qué es el *encoding*

El *encoding* («codificación» en inglés) es el proceso a través del cual se transforma información textual humana (caracteres alfabéticos y no alfabéticos) en un conjunto más reducido, para ser almacenado o transmitido. Podemos nombrar el Código Morse como un sencillo ejemplo que clarificará el concepto de *encoding*: cada letra tiene su correspondencia en forma de sonidos, y todas las letras se codifican con combinaciones de dos signos, el punto y la raya. El conjunto de información se transforma, se reescribe, con un código de solo dos signos, lo que hace posible una transmisión óptima en canales donde, por ejemplo, sería imposible la transmisión de la voz humana.

En el mundo de las computadoras el *encoding* asocia nuestros signos alfabéticos y no alfabéticos con ciertos números. Todos los signos que utilizamos al componer un texto en la computadora deben traducirse a estos números si queremos almacenarlos. «Almacenar» en una computadora es una operación fundamental, porque se almacena cuando algo debe mostrarse en pantalla, cuando queremos guardar un archivo y también almacenamos cuando queremos transmitir algo a través de una red. Por lo tanto, para la computadora en realidad todos nuestros signos serán números y nada más que números. Recordemos que la computadora no es más que una gran calculadora, que solo «entiende» dos signos: el uno y el cero.

Entonces el problema surge cuando una computadora tiene un conjunto de números que sabe representan a un texto y necesita mostrarlos. En ese momento acude a una tabla de *encoding* para reinterpretar de qué caracteres se trataban antes de convertirlos en números.

Dos *encodings* famosos

Veamos un primer ejemplo de tabla de *encoding*: el extendido ISO-8859-1 (más conocido como Latin1, y prácticamente coincidente con Windows-1252). Este tipo de encoding utiliza números de 8 bits para representar todos los signos. Es decir que todos los signos se transforman en un número entre el 0 y el 255 a partir de una especie de tabla predefinida. En este encoding nuestra letra eñe se transforma en el número 241 (que en lenguaje de computadora es 11110001; nosotros lo representamos en decimal 241 para hacerlo más manejable).

Otro de los *encodings* más utilizados, fuertemente recomendado y que se ha convertido en un estándar, es el UTF-8. Este *encoding* es distinto del anterior ya que no tiene una cantidad fija de bits para representar los caracteres. Utiliza un sistema de largo variable para lograr mayor

flexibilidad. UTF-8 puede representar todos los caracteres de Unicode, un estándar creado a fines de los ochenta para codificar todos los caracteres de todas las lenguas escritas del mundo: un total de más de 100 mil signos. En UTF-8 la eñe se representa con el número hexadecimal C3B1.[1]

Cuando un autor crea los contenidos de su blog está ingresando texto en algún formulario desde su computadora. Ese texto viaja hacia el servidor para ser almacenado en la base de datos. Luego, cuando alguien quiere acceder al artículo, el texto se recupera de la base de datos, se coloca en la página y la página se envía de vuelta a otra computadora.

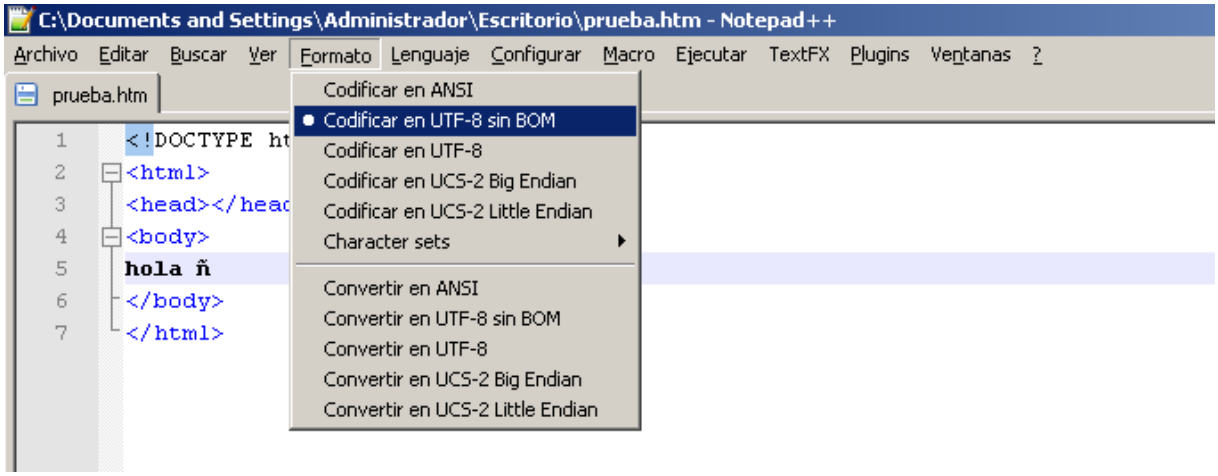
Este relato parece sencillo, pero debemos identificar el rol de los *encodings* en cada etapa:

- el navegador del autor trabaja con cierto encoding, por lo que al ingresar texto en un formulario, ese texto se convertirá en números de acuerdo a ese *encoding*.
- el lenguaje de programación (por ejemplo, PHP) que «vive» en el servidor y recibe el texto que el autor creó, también trabaja con cierto encoding y trata a los textos según ese *encoding*.
- la base de datos que almacena y recupera el texto lo hace con cierto *encoding*.
- la página web que se envía de vuelta al lector también tiene su propio *encoding*.

Los problemas ocurren cuando alguno de estos *encodings* no coincide con el resto, o cuando alguno de estos sistemas cree que está tratando con textos en cierto *encoding* cuando realmente se trata de otro. Estos errores son los que llevan a los «caracteres extraños» en nuestras páginas.

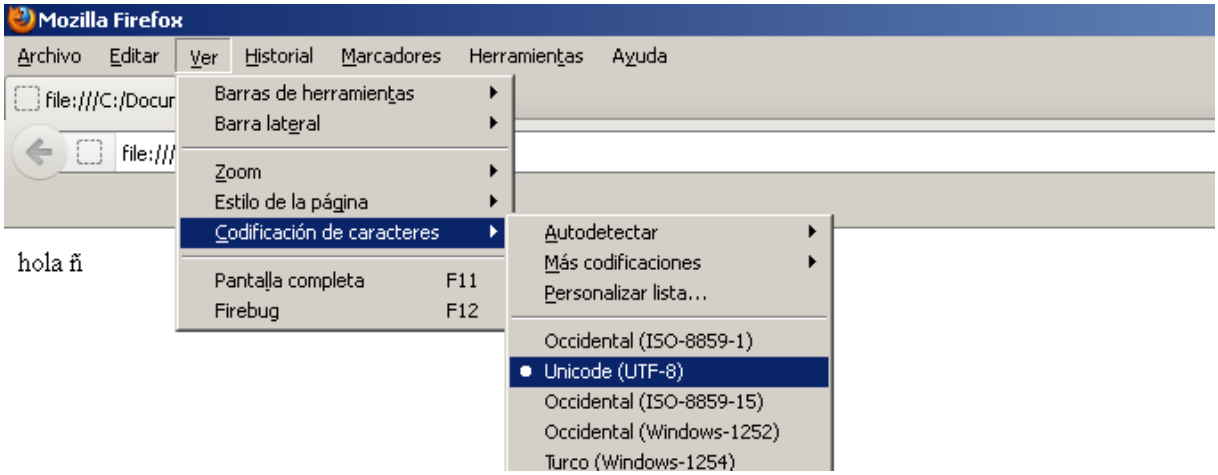
El ejemplo de la eñe

Utilizando nuestro editor favorito crearemos una simple página web que contenga un texto con eñe. El editor de texto también trabaja con un determinado encoding, por lo que indicaré que deseo trabajar en UTF-8.



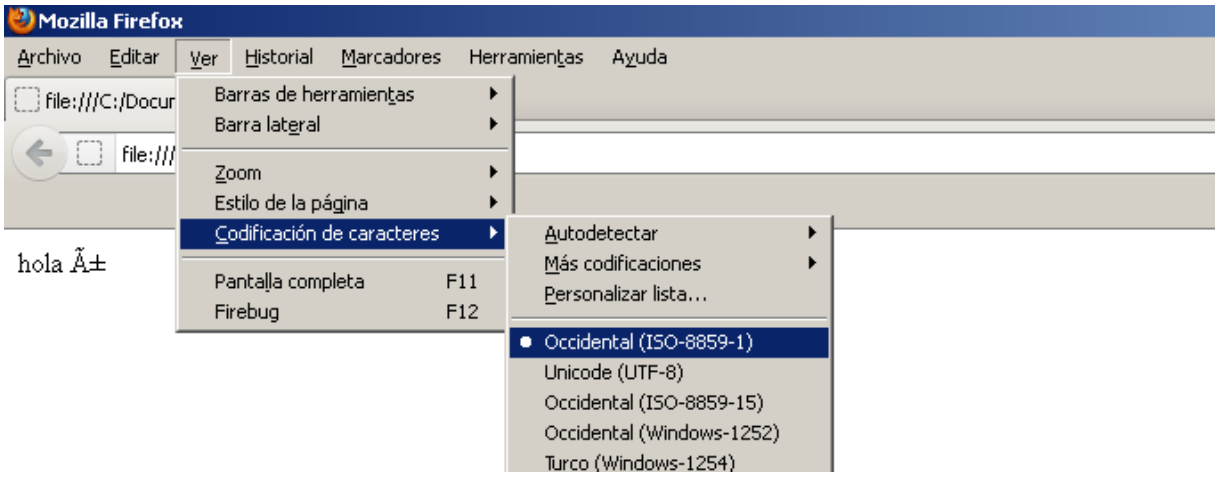
El editor de texto Notepad++ y su menú de selección de encoding. «ANSI» se corresponde con ISO.

Almacenaremos esta página web de prueba y la abriremos en un navegador. El navegador reconocerá que el archivo está en UTF-8 y mostrará correctamente la eñe.



Encoding UTF-8 correctamente identificado por Mozilla Firefox.

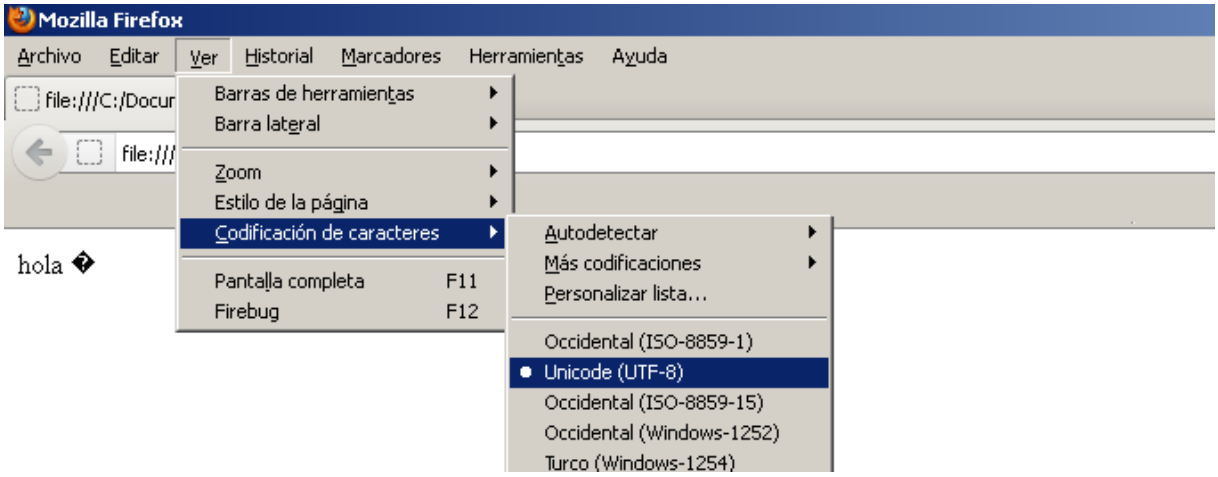
Si fuerzo al navegador para que interprete este archivo según otro encoding, entonces el resultado es el de nuestras pesadillas:



Un texto UTF siendo interpretado con el encoding erroneo, ISO.

Si observamos detenidamente este caso, veremos que este comportamiento tiene absoluto sentido para la computadora. En ISO-8859-1, el byte «C3» se corresponde con el signo «Ã» y el byte «B1» con el signo «±». Recordemos que «C3B1» era la representación de eñe en UTF-8. Lo que ocurre es que se están decodificando los números con una tabla distinta a la que se utilizó para codificarlos. Facil, ¿verdad?

También se puede probar el caso inverso: crear una página con una eñe utilizando el editor en modo ISO-8859-1 y abrirla en un navegador web indicando, erróneamente, que se trata de un archivo codificado en UTF-8.



Caso inverso: un archivo ISO decodificado con UTF. Aparece el signo de «caracter desconocido» definido en UTF porque la secuencia de números encontrados no es válida en este encoding.

Además podemos confirmar que el editor crea documentos en distintos encodings porque el archivo en UTF-8 ocupa 67 bytes en disco, mientras que los mismos caracteres en ISO-8859-1 ocupan 66 bytes. En este caso de prueba la diferencia la hace la eñe. Esto es así porque los restantes caracteres presentes en el documento se representan con un solo byte tanto en ISO como en UTF.

El navegador y los encodings

Es importante notar que el navegador web no pregunta al usuario qué encoding desea utilizar. Deberá deducir el encoding a partir de la información que la página provea y, en el peor de los casos, deberá adivinar de qué encoding se trata. Las formas de indicar cuál es el encoding de un documento son las siguientes:

- utilizar una etiqueta meta con los atributos «http-equiv» o «charset» en la sección «head» del documento HTML.
- configurar nuestro servidor HTTP (por ejemplo, Apache) para que sirva los documentos con la cabecera Content-Type adecuada (esta es una configuración del hosting, y puede no ser accesible para los desarrolladores).
- en documentos XHTML se puede utilizar el atributo «encoding» de la etiqueta xml.

Si tenemos en cuenta estas posibilidades y los distintos sistemas que listamos más arriba, vemos claramente que son muchas cosas las que pueden salir mal. La respuesta rápida ante un problema de visualización de una página web es que alguna etapa está tratando los textos con un *encoding* erroneo. Resolverlo es más dificil que enunciarlo, porque hay que investigar dónde está el problema. Para resumir podemos decir que la mayor parte de las veces lo que ocurre es:

- El navegador interpreta erróneamente el encoding del documento. Lo más probable es que alguna indicación (etiqueta meta, cabeceras HTTP, etiqueta xml) sea incorrecta. Solución: corregir las indicaciones.

- El navegador no cuenta con la información de qué encoding se trata y adivina, haciéndolo incorrectamente. Solución: agregar las indicaciones.
- El contenido se está almacenando en una base de datos con un encoding que no coincide con el de la página. Cuando el documento llega al navegador es interpretado con un encoding que, en la parte donde ese contenido aparezca, no será el correcto. Solución: corregir el almacenamiento en la base de datos.

Conclusión

Los encodings son una característica fundamental de las computadoras. Desde el inicio de la informática los ingenieros debieron representar con números nuestros signos de escritura. Los desarrolladores y diseñadores web serán interpelados por estos conceptos, ya que el futuro de Internet es decididamente multilenguaje, multicultural y multiplataforma.

Publicado el 15/08/2012

1. El modo de cálculo sería muy extenso de explicar, y probablemente poco útil a nuestros fines. Puede encontrarse detallado aquí.

FOROALFA
ISSN 1851-5606
https://foroalfa.org/articulos/el-misterioso-mundo-del-encoding