

## Convenciones de lectura de este manual

Para distinguir con facilidad el código fuente del cuerpo literario, el primero se muestra sombreado. También he sombreado los elementos y atributos cuando se citan en las explicaciones.

Algunos fragmentos de código no están completos. Para que resulte obvio que falta un pedazo de programación, uso [...]. También, algunas veces, para no complicar la explicación con atributos que aclararé más adelante y en el mismo capítulo, he decidido obviarlos. Ténganse estas advertencias en cuenta.

Los corchetes, por otra parte, sirven para enlazar con la bibliografía. Los he preferido a los paréntesis usuales para evitar colisiones allá donde los uso para otras aclaraciones. La forma de citar las fuentes responde al autor o editor y al orden en la lista alfabética, si es necesario.

También debes estar advertido de que el uso de las cursivas, en este manual, obedece a criterios poco convencionales. Si usara letras inclinadas allá donde las buenas costumbres nos imponen, la excepción sería la redondilla. Así que el cúmulo de neologismos, extranjerismos y

argot pasan en este texto por palabras con acepción de diccionario. Sólo a veces la cursiva aparece donde toca.

Permíteme, finalmente, recordar que las marcas que aquí cito son propiedad de sus respectivos dueños. Aparecen a título informativo. Para ganar agilidad de lectura, los símbolos comerciales © y ® han sido deliberadamente omitidos en aquellos momentos en los que la ética aconseja usarlos.

# Introducción

Fíjate que el título lo dice claramente: “para periodistas”. Aunque una legión de los denominados comunicadores, redactores, copywriters, publicistas, blogueros, foreros y community managers puedan extraer un buen provecho de lo que explica este manual, no entenderán algunos de los conceptos y prácticas meramente periodísticas que se aprenden en la Facultad y en el día a día de una Redacción y que aquí se citan. Por lo tanto, el destinatario final de esta guía práctica es el periodista, aunque su trabajo se limite a comisariar textos en un blog, al refrito de notas de prensa, al cuidado de la imagen de algún cliente en una red social o a la edición de un medio de comunicación por Internet.

El periodista, de una u otra forma, siempre sacará el máximo partido del uso del lenguaje HTML5 que aquí recomiendo, logrando que sus textos y demás elementos informativos (fotografías, grafismos, vídeos, tablas, documentos anejos...) tengan la máxima eficiencia.

La presente guía tampoco está pensada para programadores o diseñadores, porque las recomendaciones se circunscriben al área de trabajo del periodista, un fragmento enorme y principal de la página, pero que, sin embargo, es sólo una fracción de un total que ha sido diseñado y programado para que el redactor pueda trabajar y verter en él conteni-

dos. Aún así, programadores y diseñadores también pueden aprender algo sobre las necesidades del periodista y contribuir a que el producto final sea más eficiente.

El objetivo de este manual es establecer y comprender unos procedimientos sencillos y estandarizados que aprovechen todas las posibilidades que brinda el lenguaje HTML5 para obtener unos contenidos estratégicos, semánticos, accesibles y eficaces con el mínimo esfuerzo posible.

Alguno pensará que ya vuelven a cargarnos, a los periodistas, con tareas que antes pertenecían a otros puestos de trabajo. Como cuando se nos obligó a cubrir actos informativos con cámaras fotográficas, ahorrando a la empresa el sueldo de un fotoperiodista, o como cuando se fueron sustituyendo los técnicos de sonido en los estudios radiofónicos e incluso en las unidades móviles. Nos dijeron que el periodista ganaba control sobre la emisión de su mensaje, pero muchos —muchos, muchos— creemos que nunca había perdido ese control y que lo que sucedía era que las empresas querían ahorrar en el capítulo de salarios.

Con el etiquetado eficiente y correcto en lenguaje HTML5 no sustituimos a nadie. Si se permite la comparación, es una circunstancia similar a la llegada de las máquinas de escribir a la mesa del redactor, sustituyendo al plumín y al tintero. Algo que forma parte de la mecánica de la escritura y que permite ganar en eficiencia. Y es un paso adelante más, personal, obligado y necesario, para mantener la capacidad profesional en un alto nivel competitivo. Lograremos que nuestros mensajes tengan más alcance en el presente y en el futuro. Afianzaremos las características de agregación y de diseminación de los núcleos de nuestras piezas informativas. Podremos presentarlas en multitud de soportes actuales y futuros. Y nos aseguraremos de que el contenido tenga vigencia más allá de su caducidad natural.

Todo esto se conseguirá etiquetando de nuevo y retocando algunas marcas del lenguaje HTML *on the fly*, es decir, mientras se escribe, o en un repaso final, antes de darlo por bueno y pasarlo al editor, redactor-jefe, supervisor, subdirector, director o el rango superior que exista en tu Redacción.

Lograrás un contenido estratégico desde el primer momento. Desde el instante en el que hayas decidido que aquel término o expresión necesita un etiquetado distinto dentro de la programación HTML. Al hacerlo, obtendrás de un plumazo algunas ventajas:

- **Indexabilidad:** Lograrás que el texto, fotografía, gráfico, aplicación web, vídeo o presentación tenga una mejor indizabilidad en los buscadores.
- **Posicionamiento semántico:** Conseguirás que el contenido obtenga mejores posiciones en los buscadores semánticos, como Siri, en los iPhone.
- **Accesibilidad:** Facilitarás la comprensibilidad de un contenido de forma unívoca e inequívoca a las personas con alguna discapacidad que utilizan navegadores de Internet adaptados a sus condiciones.
- **Versatilidad:** Crearás un contenido que sea adaptable,<sup>1</sup> transmitiendo eficientemente los mensajes desde cualquier tipo de dispositivo e independientemente de su capacidad de almacenamiento y transmisión.
- **Diseminabilidad / Agregabilidad:** Manipularás con mayor facilidad los contenidos creados, en los que se podrán usar técnicas profesionales de diseminación o agregación de la información que mejor convengan en cada presentación.
- **Referenciabilidad:** En el futuro y desde nuevos contenidos será más fácil referirse al que hayas creado ahora y vincular el producto futuro al pasado.
- **Historiabilidad:** Actualizarás la información y realizarás seguimientos secuenciales de acontecimientos sucesivos con mayor facilidad.

1 Por contenido adaptable se entiende el que ha sido estructurado en piezas informativas con valor propio. Éstas pueden aglutinarse o se puede prescindir de ellas en función de las características de uso y tamaño del dispositivo de consumo, sin que pierdan su función básica. Se opone a la escuela de diseño *responsive*, que emplea todo el contenido adaptándolo a la anchura de las pantallas de los dispositivos de consulta, lo que puede resultar más caro e ineficiente.

- Recuperabilidad:<sup>2</sup> Obtendrás los textos informativos limpios —sólo texto— de forma muy sencilla, eliminando las etiquetas HTML con una simple operación informática.

Esta última capacidad extractiva resulta muy útil para la empresa que nos contrate. De hecho, el uso correcto de HTML5 en la producción informativa es un paso más en la industrialización de los procesos. Afecta tanto a los medios de comunicación como a cualquier otra actividad en la que se trabaja con contenidos tratados periodísticamente. La empresa se beneficia de reducciones de costes en el tratamiento *post mortem* de la información, en las posibles traducciones, en la exportación a otros canales y soportes, etcétera. El medio, la agencia, la empresa que paga las nóminas, consigue reducir los costes de recursos humanos y de tiempo aumentando su capacidad de reacción y manteniendo siempre un estilo unificado de tratamiento informativo, de voz, de trato y de tono, que son el distintivo de su marca.

Para poder hacer todo esto con el mínimo esfuerzo, te presento este manual.

Como verás, este volumen tiene cuatro partes. Una primera que te sitúa en lo que es la semántica web y en la operativa de trabajo que propongo. La segunda, en la que profundizaremos en los elementos clave de la redacción para Internet. Y al final, un epílogo y los anexos, en los que encontrarás información adicional útil para el trabajo diario.

La segunda parte, la mayor, no tiene un orden fijo ni deliberado. Puedes leerla de principio a fin o saltar directamente a los capítulos que más te interesen. Si lo haces desde un ebook, te será más fácil realizar búsquedas por nombres de elementos y atributos HTML5.

Mi intención ha sido que puedas encontrar siempre el fragmento de código que mejor se avenga a tus necesidades informativas, sus explicaciones correspondientes, las variaciones posibles y las fuentes en

2 La mayoría de los términos que encabezan esta lista no existen en español, aunque son de uso bastante frecuente en medios de creación de contenidos en Internet. Tómalos como argot y acéptalos escritos en redondilla.

las que puedes informarte con mayor detalle y profundidad sobre este lenguaje.

Recuerda que HTML5 es un lenguaje vivo, sometido a cambios e incorporado poco a poco por la totalidad de la industria. Esta vitalidad puede suponer que etiquetas que hoy se usan, mañana hayan sido abandonadas por la presión de los fabricantes de los navegadores, de los teléfonos móviles o de los buscadores, que son quienes cortan el bacalao. Hay que estar atento.





## Primera parte

Los tres capítulos de esta primera parte aportan las claves de lo que se desarrollará en la siguiente. Por lo tanto, sigue mi recomendación y léelas de cabo a rabo antes de adentrarte en la jungla de los ejemplos de código.

El primero de los capítulos recoge una visión global de qué es HTML5 y qué nos puede aportar como periodistas. El segundo, a riesgo de que los especialistas me corran a gorrazos, pretende ser una explicación comprensible de qué es la semántica en el lenguaje para la web, cómo funciona y cómo nos podemos y debemos beneficiar de ella.

Finalmente, el tercero es la propuesta de trabajo para los periodistas. Se trata del material con el que nos enfrentamos en el quehacer diario y de cómo solventar los problemas de edición de nuestros textos para superar las convenciones visuales y transmitir adecuadamente un valor comprensible y semántico que nos ahorre trabajo en el futuro.



# Lo esencial del lenguaje HTML5

Vale la pena que recordemos cómo se construye una página web. Sabrás, con seguridad, que la casi totalidad de los sitios web que visitas están creados con una clara separación del diseño, los datos y contenidos que muestran, y la programación que la hace posible. Son muy pocas las excepciones.

Generalmente todos los textos, menús, epígrafes, enlaces salientes, ubicaciones de fotografías, archivos de vídeo, archivos de sonido y documentos que el usuario puede descargarse, y en general la referencia de todo el contenido, se guarda en bases de datos.

Los archivos y documentos que se mostrarán en pantalla, como las fotografías, los vídeos o los documentos MP3, cuando son propios, se encuentran ubicados en un directorio del servidor.

Allí también se encuentran las hojas de estilo CSS (Cascading Style Sheets), unas instrucciones que dan forma a la página web, y los documentos con la programación. Estos archivos se conectan a la base de datos, obtienen el contenido y las instrucciones de presentación, y generan cada página HTML que después los usuarios pueden ver con un navegador en un ordenador sobremesa, una tableta, un teléfono inteligente o un televisor.

Es decir, el código HTML no existe antes de que el usuario haya llamado a una URL concreta. La programación web genera de forma dinámica la página a partir de lenguajes de ejecución en el servidor (PHP, ASP, JSP)<sup>3</sup> que combinan fragmentos de lenguaje HTML con el contenido e instrucciones obtenidos de la base de datos.

La separación de contenido, forma e instrucciones para su combinación permite una más rápida y fácil actualización de contenidos, rediseño y modernización de la programación. Pero, sobre todo, genera un documento en el único lenguaje comprensible para el navegador: HTML (HyperText Markup Language).

HTML es un lenguaje de marcación avanzado. Es decir, se compone de una serie de marcas que aportan un significado dentro del sistema de la página para que los navegadores lo interpreten de forma adecuada. Por ejemplo, un texto etiquetado con `h1` será mostrado siempre como un titular de primer nivel y cualquier máquina lo considerará más relevante que un texto etiquetado con `h2`, que corresponde con un titular de segundo nivel.

Las etiquetas, salvo alguna excepción como las imágenes (`img`), engloban los fragmentos de contenido. Antes de la frase que queremos enfatizar se escribe la etiqueta de apertura, en este caso `<em>`, y tras la frase, la de final, `</em>`. Si lo miramos de otra forma, se trata de un sistema de cajones que guardan contenido: textos, vídeos, fotografías, enlaces... A su vez, estos cajones pueden albergar otros cajones. Pero sólo determinado tipo de cajones, en determinada cantidad y en determinada posición.

Por ejemplo, el elemento `figure`, que está destinado a albergar todo aquel dato accesorio que ilustre la información principal (como foto-

3 Antes de que tengas un vahído: CSS son las hojas de estilo en cascada (Cascading Style Sheets). MP3 es un formato de compresión de audio muy común. URL es la dirección de un recurso en Internet (Uniform Resource Locator), frecuentemente utilizado para referirse a la dirección de una página web. PHP (acrónimo recursivo de Hypertext Pre-processor), ASP (Active Server Pages) y JSP (JavaServer Pages) son lenguajes de servidor. Se ejecutan en el servidor y generan las páginas HTML.

grafías, tablas o vídeos), nunca tendrá un titular. Sin embargo, sí tendrá un elemento de titulación común a todo el contenido que albergue: `figcaption`. Éste será único.

## Valores semánticos

La gramática del sistema de etiquetas aporta un valor semántico al contenido que introduzcamos en ellas. Recordemos las clases de semiótica en la Facultad y la definición más simple de la semántica como un sistema de correspondencias entre significantes y significados. Si dentro de una etiqueta destinada a un titular escribimos otra cosa, como un pie de fotografía, el lenguaje lo soportará, los navegadores lo aceptarán como un titular y las máquinas lo interpretarán como un titular.

Pero los buscadores no encontrarán nuestro titular real. En su lugar indexarán como tal el pie de fotografía. Nuestro contenido quedará enterrado entre los millones de textos inútiles de Internet y no habremos conseguido llegar a nuestra audiencia. Ergo, no habremos podido vender nuestro producto. Porque, recuerda, sólo hay dos tipos de web: las que venden productos y las que se venden a sí mismas.

Habremos fracasado.

El escrupuloso respeto sintáctico y morfológico a la calidad semántica de las etiquetas HTML, que es de lo que hablo en esta guía profesional, no es difícil. Te ahorrará mucho tiempo y dolores de cabeza y te facilitará, en el futuro, la reusabilidad y reciclabilidad de los textos que redactes ahora.

La semántica de HTML, por lo tanto, es un sistema de significados y significantes que entienden las máquinas. Pero también es un sistema jerárquico, en el que unos contenidos prevalecen sobre otros. Y esa categorización la debemos proporcionar haciendo coincidir nuestros intereses estratégicos como redactores con el funcionamiento de las máquinas que debemos comprender.

Un ordenador leerá nuestra página HTML letra a letra, de arriba abajo y de izquierda a derecha. Irá localizando los elementos principales y los dibujará (renderizará) en pantalla. Y dentro de estos, los secundarios. De las partes de la página que no son comunes al resto del sitio web,

irá tomando aquello que hayamos escrito dentro de las etiquetas principales, y lo guardará y tratará. Lo mostrará como un snippet <sup>4</sup> en una página de resultados. Será la respuesta que nos dé el buscador Siri en el teléfono móvil cuando le hagamos la consulta. Será el fragmento de la portada de nuestro Flipboard, el titular en el Paper.li <sup>5</sup> de un seguidor, el tuit corporativo o cualquier otro automatismo que podamos utilizar.

## Lenguaje técnico-científico y no periodístico

HTML5 es la última evolución del lenguaje inventado en el CERN <sup>6</sup> para que los científicos pudieran compartir documentación electrónica usando una red de comunicaciones que heredaba protocolos de los sistemas del ejército estadounidense. Su característica esencial es el enlace entre documentos. Al principio eran textos muy esquemáticos —titular, cuerpo de texto con sus enlaces correspondientes y listas— que, poco a poco, fueron complicándose más y más. Hace unos meses se cumplieron los 20 años de la primera fotografía en el sistema: el GIF <sup>7</sup> sexy de unas científicas juerguistas que bajo el nombre de Les Horribles Cernettes amenizaban los jolgorios, guateques y festejos que los sesudos profesores se proporcionaban y proporcionan de vez en cuando.

4 Porción de contenido estructurado y reusable. Por ejemplo, los datos de una tarjeta de visita, la ubicación geográfica de un establecimiento cultural, una receta de cocina o los datos bibliográficos de un libro.

5 Flipboard es un agregador de contenidos en dispositivos móviles, capaz de organizarlos como si fueran una revista. Paper.li es una herramienta de comisariado de contenido que genera la portada de un periódico virtual sobre los intereses del usuario.

6 Siglas de la Organización Europea para la Investigación Nuclear, con sede en Ginebra (<http://cern.ch>).

La que se considera la primera página web de la historia, datada el 30 de abril de 1993, puede visitarse en: <http://info.cern.ch/hypertext/WWW/TheProject.html>. Si observas el código fuente verás que algunos elementos iniciales se mantienen en la versión actual de HTML. Más adelante se habla de Les Horribles Cernettes, aún en activo: <http://www.cernettes.com>.

7 Formato de compresión de imágenes. Las siglas corresponden a Graphics Interchange Format. Produce un alto nivel de compresión para colores planos y admite pequeñas animaciones.

La capacidad del lenguaje ha ido creciendo con el tiempo, pero vinculándose siempre a las necesidades técnico-científicas y no a las periodísticas. Ha tenido etiquetas para escribir esquemas de manual universitario, donde casi siempre las imágenes son una ilustración descriptiva y no perentoria de aquello que se explica. Pero no ha tenido elementos para escribir despieces, menús de navegación o secciones fijas. Y todo porque, simplemente, no se necesitaban para su objeto inicial: compartir información técnica entre iguales.

En su evolución histórica surgieron los lenguajes XHTML,<sup>8</sup> que añadían etiquetas especiales ampliando su capacidad expresiva original. Estas marcas adicionales se han aprovechado intensamente en el comercio online y en la transmisión de datos bancarios.

Pese a estos intentos, las carencias originales y las subversiones generadas obligaban a llevar a cabo una revisión profunda y urgente de HTML.

Fruto de esta revisión, tenemos hoy entre nosotros HTML5, que, para tu disgusto, es un lenguaje vivo. Su vitalidad proviene de que algunas de las etiquetas y procesos que aprenderás en este libro pueden cambiar a lo largo del tiempo. Incluso, ya se ha visto, volver a cambiar una vez modificado para regresar al punto de partida.

La versión que hoy usamos es la CR (Candidate Release), que se entrega a la industria de Internet para su aprobación definitiva como estándar tecnológico. La CR se publicó el 17 de diciembre de 2012 y, sorpréndete, simultáneamente se presentaba la versión 5.1.

El calendario del consorcio de Internet responsable del estándar tecnológico establece para las últimas semanas de 2014 la aprobación definitiva del lenguaje HTML5, con las correcciones que se hayan podido hacer en la versión final CR. Entretanto, la industria estará trabajando ya en la versión 5.1, cuyo borrador de trabajo se aprobará a finales de 2016.

Las previsiones de cambios en este período de dos años, antes de la

8 Siglas de eXtensible HyperText Markup Language.

aprobación definitiva, alcanzan a pocos elementos que afecten a nuestro trabajo de redactores: `hgroup`, `output`, el atributo `scoped` en `style`, el atributo `seamless` en `iframe` o el atributo `cite` en `blockquote` y `del`.

HTML5.1 desarrollará algunas tecnologías previstas para la versión 5 que no llegaron a implementarse. Tendrá dibujo vectorial, scripts que se ejecutan en segundo plano para evitar que el navegador se cuelgue (WebWorkers) y conexiones persistentes entre la página y el servidor sin que el usuario tenga que interactuar con ella (WebSockets). Además mejorará la integración de Microdata.

Pero faltan años para ello.

HTML5 aporta muchas novedades, la mayoría vinculadas a la ejecución del código en los terminales del usuario y no en los servidores: geolocalización, reproductores nativos de vídeo y audio para prescindir de plug-ins de terceras empresas, almacenamiento y consulta offline de bases de datos, etcétera. Pero, sobre todo, incluye una marcación con elementos nuevos destinados a la comunicación de algo más que simple información técnica.

Tampoco lancemos las campanas al vuelo, porque sigue lejos de atender satisfactoriamente todas las necesidades periodísticas, aunque se aproxima bastante. Ha sido diseñado para que los autores de blogs tengan una buena cobertura tecnológica, para que la presentación codificada del contenido sea lo más estandarizada posible y para que tenga una excelente visibilidad en cualquier dispositivo de conexión a Internet: ordenadores de sobremesa, portátiles, tabletas, teléfonos, televisores o navegadores de automóvil.

Resulta más fácil de utilizar y, en principio, más fácil de programar. Tiene más etiquetas y, por lo tanto, su nivel semántico ha aumentado. Un mismo contenido puede alcanzar mayor difusión en distintos tipos de dispositivo, de forma simultánea, y además puede ser interpretado correctamente por máquinas.

## Documentos reconocibles

El lenguaje HTML5 es fácil de reconocer. Si miras el código fuente de una página web, las primeras líneas deben ser o parecerse a:



```
<!DOCTYPE HTML>
<html lang="es">
<head>
```

La primera línea es el tipo de documento. HTML5 sólo tiene este texto, sin extras ni extensiones. Si incluye más texto, o no es HTML5 o tiene alguna extensión desconocida en el momento en el que escribo este manual.

La segunda línea es la marca del código HTML. Es el gran cajón que engloba todos los cajones y marcas de la página. Si te fijas, al final del documento debe aparecer una etiqueta de cierre `</html>`.

La etiqueta `html`, obsérvalo, tiene un atributo `lang` con un valor `es`. Significa que el idioma en el que se ha escrito el contenido de la página es español. En HTML5 el idioma de la página es fundamental. Condiciona el diccionario terminológico que utilizarán las máquinas para interpretar el contenido de cada cajón. En este caso es español, pero podría ser francés (`fr`), inglés (`en`), alemán (`de`), catalán (`ca`), gallego (`gl`) o *swahili* (`sw`).

Este atributo, además, puede incluir la versión territorial del habla, aunque en la mayoría de los casos es redundante y no aporta nada o casi nada a la difusión del contenido. Por ejemplo, para España el atributo sería `lang="es-ES"`; para México, `lang="es-MX"`.

Entre la segunda y la tercera línea, que indica el encabezamiento de la página web, no debería haber ningún otro código. Debería ser tal y como se muestra en el ejemplo.

Si el código fuente de la página en la que estás trabajando no se parece al que he reproducido —adicionalmente en la segunda línea podría haber atributos `id` o `class` generados *on the air*—, no utilices la mayoría de las etiquetas de este manual, porque no funcionarán y podrían romper el diseño, mostrando elementos de contenido fuera de lugar.

Pero podrías arriesgarte, porque una de las grandes ventajas de HTML5 es que los navegadores, cuando encuentran algún elemento irreconocible en la marcación, simplemente lo ignoran. Otras versiones

de HTML, sin embargo, pueden detener el dibujado de la página en el punto que les causa confusión o entrar en el denominado modo *quirk* e inventarse una interpretación de qué demonios quería hacer el programador cuando escribió aquella etiqueta que no conoce. En casos así, puede producirse en pantalla cualquier resultado inesperado.

Hasta aquí hemos visto, por lo tanto, unas cuantas características del lenguaje HTML5 y de la forma en la que vamos a trabajar para obtener el máximo provecho con el mínimo esfuerzo. Las resumo:

- Elementos:

El lenguaje está formado por elementos que etiquetan contenido, marcándolo antes y después, a modo de grandes cajones. Por ejemplo, un párrafo: `<p>Texto del párrafo.</p>`. Recuerda, no obstante, que en algunos casos la etiqueta no tiene marca de cierre.

- Atributos:

Cada elemento puede tener atributos dentro de la marca. Por ejemplo, un párrafo en alemán dentro de una página cuyo documento HTML se ha definido en otro idioma: `<p lang="de">Texto en alemán</p>`.

Para usar un elemento, nuestros programadores ya habrán previsto un instrumento simplificado de estilo WYSIWYG (*What You See Is What You Get*, lo que ves es lo que escribes) y un aire a los editores de texto tradicionales en ofimática. Pero algunos deberemos escribirlos manualmente y otros muchos nos llevarán a modificar o crear sus atributos. Para ello actuaremos directamente sobre el código fuente, una vez hayamos escrito el contenido que queremos mostrar y antes de guardar o enviar a la base de datos nuestro trabajo.

¿Empezamos?

## Hablando el lenguaje de las máquinas

Dicen que vivimos una nueva era de Internet, dominada por la semántica computacional. Probablemente tendrá poco de era y mucho de etapa breve hacia lo próximo.

Muchos, como siempre, pasarán de la edad de piedra al futuro más sofisticado saltándose esta fase.

El hecho es que las máquinas, las computadoras en red, comienzan a comunicarse con nosotros gracias a la aplicación de conceptos semánticos. Y para obtener el mejor provecho de nuestros mensajes presentes y futuros tenemos que entender cómo funcionan sus mecanismos de comprensión.

La Real Academia Española refiere que el concepto semántica proviene del griego *σημαντικός*, que se traduce por “significado” [RAE-1]. Por lo tanto y como era lógico suponer, la semántica de redes o computacional está relacionada con los significados. Como todo acto comunicativo.

Un significado es la atribución de un valor a un símbolo (palabra hablada o imagen icónica) y resulta esencial que el emisor y el receptor

coincidan en su misma interpretación, o la comunicación habrá fracasado.

Con las máquinas tenemos que poner en común los conjuntos de significado y significante para que nos comprendan, interpreten y puedan servirnos de algo ahora y en el futuro.

En la bibliografía anexa descubrirás todo un mundo de servicios, lenguajes y estándares tecnológicos que están al servicio de la comunicación semántica entre computadoras: XML, RDF, RIF, OWL, SPARQL... [PASTOR] Pero sólo hay uno que es realmente abierto, flexible, multicultural, de uso sencillo y que conecta al humano con la máquina: el lenguaje HTML en su versión 5, que, mira tú por dónde, es el que todos usamos en nuestra práctica profesional cotidiana.

En otras palabras, escribiendo para personas y usando el lenguaje HTML5 de forma correcta, conseguimos que las máquinas entiendan de qué estamos hablando y lleguen a responder a otros humanos con nuestra información. Es el logro para el que, desde hace más de 10 años, el consorcio que gestiona los estándares tecnológicos en Internet está trabajando desde la plataforma de la Web Semántica [W3C-3]. HTML5 es uno de sus frutos principales.

## Los antiguos buscadores y los nuevos buscadores semánticos

Se ha escrito tanto sobre las versiones de Internet —que si la web 2.0, que si la 3.0...— y sobre la web semántica, que a estas alturas o eres un experto o tienes un lío fenomenal.

Intentaré no contribuir a la confusión.

La semántica estudia la relación entre significantes y significados. El significante es el elemento del código fuente que utilizamos cuando escribimos. Un párrafo `p`, una negrita `b`, una lista ordenada `ol`, una tabla `table` son significantes. Cuando los usamos para etiquetar, marcar o envolver algún texto, imagen u otro tipo de formato de contenido, estos se transforman en sus significados.

A cada significante-elemento sólo puede corresponder un significado-tipo de contenido, al que se aplican unos atributos y que se atiene a unas reglas.

Relacionarlos correctamente implica que un buscador avanzado podrá encontrar contenido (significado) dentro de una etiqueta HTML determinada (significante), porque ateniéndose a las reglas del lenguaje sabrá que dentro del elemento de código sólo cabrá ese tipo de contenido. Un buscador semántico, además, será capaz de interpretar el contenido en su contexto y de operar con él.

Por lo tanto, nuestro contenido son datos puros y asépticos desde la perspectiva de las máquinas.

Este punto de vista robótico puede servirnos para comprender mejor su funcionamiento. En lugar de un párrafo de texto corrido, extraeremos en un ejemplo los datos fríos y los colocaremos en una tabla. Luego veremos cómo actúa cada tipo de máquinas con ellos.

Imagina un desayuno hogareño de familia de anuncio. Padre, madre, dos hijos y una hija recién duchaditos, con cara de felicidad y sonrisas relucientes y tomando tazones de cereales en la cocina mientras el jefe de la familia fuma en pipa y hojea el periódico matutino.

Esto que has imaginado es el escenario de nuestro contenido de prueba. Procedemos a desglosarlo de la siguiente forma:

Padre	Café solo	Engominado hacia atrás	Calcetines negros
Madre	Yogur con fibra y extra de bacterias	Moño alto	Panties
Hijo mayor	Bol de cereales	Raya a la derecha	Calcetines azules
Hija mediana	Bol de cereales	Coletas	Leotardos violeta
Hijo pequeño	Bol de cereales	No	Calcetines azules

Las normas lingüísticas de este ejemplo imponen que cada elemento sólo puede corresponderse con un rol familiar. Y que cada rol familiar debe tener obligatoriamente estos atributos: desayuno, peinado y calcetines.

Si empleamos un buscador habitual, como Google, intentará localizar la expresión idéntica de nuestro criterio de búsqueda en el contenido de las celdas. Si buscamos la palabra “leotardos” nos indicará que lo

ha encontrado en el registro “Hija mediana”. Si usamos parcialmente un criterio, buscará heurísticamente su coincidencia exacta, aunque no se trate de una palabra ni tenga sentido por sí mismo. Para la petición de búsqueda “acteri” nos enseñará el registro “Madre”.

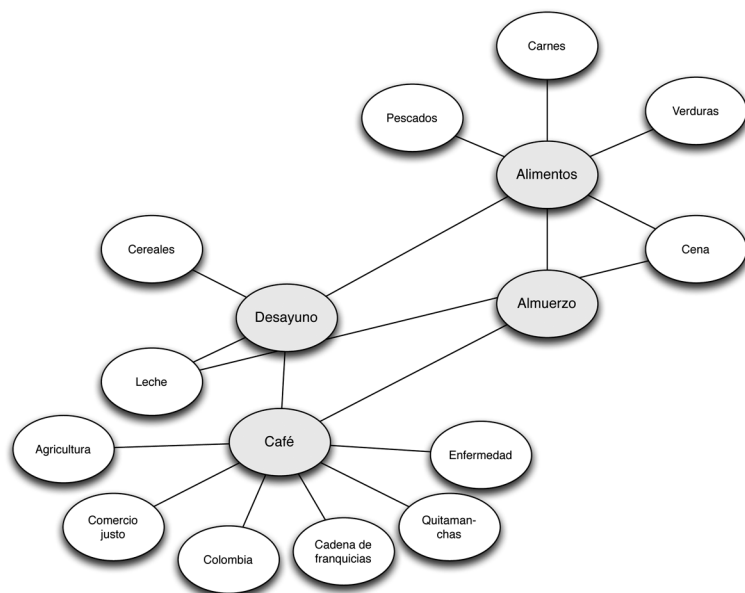
En cambio, si utilizamos un buscador más avanzado y establecemos como criterio de búsqueda “desayuno del padre”, pondrá en relación el atributo “desayuno” (segunda columna) para el elemento “Padre” (primera línea) y mostrará el resultado: “Café solo”. La única comprobación que realiza el buscador es que el tipo de contenido se corresponde con las normas. Es decir, el término “Padre” es realmente un rol familiar que puede localizar en su diccionario interno y “café solo” es, en todo o en parte (o “café solo” o “café” a secas), un término posible y de frecuencia recurrente en el ámbito de los alimentos y la esfera de los desayunos.

Para que comprendas esto último mejor, escribe “alimentos” en un círculo (Figura 1). El término aparece escrito con frecuencia elevada junto a otra terminología fácilmente clasificable. Por tipo (“verduras”, “carnes”, “pescados”) o por consumo (“desayuno”, “almuerzo”, “cena”). Escríbelos alrededor, envuélvelos cada uno en un círculo y traza líneas entre estos y “alimentos”. Ahora, escribe “café”, “leche” “cereales” alrededor de “desayuno” y nuevamente envuelve cada término en un círculo y vincúlalo con una línea a “alimentos”. Ahora tenemos “café” dentro de la esfera “desayuno” de la esfera “alimentos”. Pero también podría estar vinculado a la esfera “almuerzo”, porque se puede tomar en la sobremesa. Así que traza una línea que los vincule. Y lo mismo sucede con “leche”, que puede estar vinculado también a la esfera “cena”.

Pero “café”, además, puede aparecer en otras esferas distintas: “agricultura”, “comercio justo”, “Colombia”, “cadena de franquicias”, “quitamanchas”, “enfermedades del sueño”... Así que iríamos trazando líneas de vinculación con las distintas esferas.

Tú y yo resolvemos estas vinculaciones porque como humanos conocemos el significado ontológico de los términos y somos capaces de establecer rápidas asociaciones naturales. Una máquina desconoce el significado, más allá de su coincidencia, letra por letra, en su diccionario interno y de las variaciones lingüísticas a las que pueda someter el

término. La máquina crea un gráfico conceptual por frecuencia de apariciones en distintos entornos. Si una mayoría de sus usuarios escribe “café” en relación al “desayuno”, asociará la palabra “café” en la esfera “desayuno”.



**Figura 1**

Mapa de las relaciones entre los términos “café” y “desayuno”, que permiten validar la respuesta

Cuando se le exige una búsqueda, la máquina simplemente comprueba que el tipo de significante posible, “desayuno” para el “padre”, alberga un significado que está en la esfera “desayuno” de su propio mapa conceptual (Figura 2). Si lo encuentra, lo dará por válido y lo ofrecerá como resultado.





al corpus que van adquiriendo. Si el resultado de la aplicación de las normas es positivo, la terminología detectada pasa a su corpus lingüístico y se integra en su mapa conceptual.

El proceso no descansa, aunque es complejo y requiere continuas correcciones humanas de los algoritmos que emplean las máquinas. Cada idioma, además, presenta sus particularidades, lo que añade complejidad al proceso. Cuanto más extendido y rico es el lenguaje, mayor dificultad de implantación tienen los buscadores semánticos.

Pero ya lo estamos usando, aún sin darnos cuenta. E irá a más.

Como periodistas y comunicadores estamos en la vorágine semántica y no podemos obviarla. Si queremos que nuestro trabajo tenga valor, debe ser influyente. Para que sea influyente tiene que tener alcance. Para que tenga alcance tiene que ser localizable con facilidad. Para que sea localizable, tiene que ser semántico. Para que sea semántico, tiene que estar escrito correctamente en HTML5.

De lo contrario, fracasaremos.

Lo que tenemos a la vuelta de la esquina aportará todavía más valor a nuestros mensajes. Ya tenemos ejemplos de lo que puede ser el resultado del lenguaje de las máquinas en Siri, el buscador del sistema iOS de Apple, Assistant (Majel) en los Android, o la sofisticación de las capacidades del buscador que desarrolla Wolfram Alpha [Wolfram]. Es cuestión de tiempo que todos tengamos acceso a ellas.

A lo mejor cuando estás leyendo estas líneas ya puedes hacer estas otras preguntas en un ejemplo como el anterior y obtener estas respuestas:

- ¿Quién usa calcetines azules?: El hijo mayor y el pequeño.
- ¿Quién usa calcetines?: El padre, el hijo mayor y el pequeño.
- ¿Cuál es el desayuno preferido del hogar?: El bol de cereales
- ¿De los que usan moño, qué tipo es el más frecuente?: El moño alto.
- ¿Cuántos hombres hay en la cocina?: Tres, porque sus roles corresponden en el diccionario del sistema con el género masculino.
- ¿Cuántos componentes tiene la familia?: Cinco.

El buscador semántico no sólo cruzará los datos de significados y significantes, validándolos. También los pondrá en relación con el con-

texto, lo que permite obtener resultados por superposición de valores. Observando detenidamente las respuestas, son una mezcla de la coincidencia exacta del criterio de búsqueda y de la aplicación de fórmulas muy básicas de estadísticas: frecuencias, medias, sumas, desviaciones... todo lo que una máquina hace más rápido.

Los inventores que están detrás de estos buscadores de nueva generación nos prometen que el buscador será capaz de aprender de las búsquedas del usuario. Ya lo han experimentado.

Google puso en marcha un buscador semántico sobre cuestiones de salud, Google Health, que estuvo operativo durante unos meses, a caballo entre 2011 y 2012, y al que se podía acceder desde Estados Unidos [EFRATI]. Su principal característica era que proporcionaba patologías coincidentes —nunca diagnósticos, por cuestiones éticas y legales— con síntomas. Pero, además, aprendía del usuario. Era capaz de guardar y relacionar las búsquedas previas para mejorar sus respuestas.

Si alguien buscaba el síntoma “dolor de cabeza”, el buscador ofrecía una lista extensa de las patologías posibles. Si el mismo usuario, en una segunda consulta, establecía como criterio “estornudo”, la lista de patologías combinaba “dolor de cabeza”, no planteado en ese momento, y “estornudo”, y producía una lista más ajustada a estos síntomas: “resfriado”, “gripe” o “rinitis”.

## Herramientas y procedimientos de trabajo

Echemos un vistazo a nuestro instrumental.

Para crear contenidos HTML5 semánticos partiremos de tres factores: conocer qué etiquetas de código necesitaremos en cada caso, saber qué posibilidades de edición nos ofrece el sistema y disponer de un buen editor WYSIWYG que acepte elementos en lenguaje HTML5 sin sabotear nuestro trabajo. O por lo menos que sea un editor lo suficientemente flexible para que podamos manipular el código a nuestro gusto.

De entrada, una mala noticia: los editores que existen cuando escribo estas líneas no soportan HTML5. Y, de los que anuncian lo contrario, resulta que sólo utilizan la capacidad `contentEditable` que aporta el lenguaje. Desde un punto de vista práctico, estos últimos pueden trabajar en modo *air*, directamente sobre el texto, tal y como lo verá el usuario final.

Pero todo tiene remedio. Vayamos por partes.

Un editor WYSIWYG es una emulación dentro de un formulario web de los controles de un editor de textos típico de un paquete de ofimática. Es decir, tiene una apariencia muy similar a Word de Microsoft, a Pages de Apple, etcétera. WYSIWYG significa *What You See Is What*

*You Get*, es decir, lo que ves es lo que obtienes. De modo que el editor ofrece un control visual de la producción. Las cursivas marcadas se ven como cursivas; las negritas, como negritas. Un clic en un botón sirve para escribir muchos caracteres de código sin tener que conocer cómo se programa HTML.

Cada editor transforma un área de texto de un formulario web —una especie de gran casilla— en un espacio de trabajo en el que se muestra el efecto del código sobre el texto que se va escribiendo. En estos formularios se crean y se modifican los contenidos de una página.

Sobre el espacio de escritura, se muestran los controles del editor, una barra de herramientas a modo de botones y con una apariencia que intencionadamente imita a los editores de texto de ofimática.

No se necesita gran explicación sobre su uso habitual: se selecciona la palabra que se quiere transformar en una negrita, se pulsa el botón correspondiente en la parte superior y en pantalla se ve cómo ha cambiado a negrilla. El usuario evita tener que escribir la etiqueta anterior y posterior a la palabra afectada: `<b>ejemplo</b>`. Si se desea escribir un enlace, se selecciona la expresión que actuará como gancho, se activa un botón en el editor y en la ventanilla emergente se escribe la dirección de destino y los demás atributos del elemento. Todo a la vista.

No olvides, sin embargo, que generalmente estarás utilizando este editor en un back-end,<sup>9</sup> una parte privada para los creadores de contenido que los usuarios finales no ven, y que aquello que aparece con una letra del cuerpo 16 y color negro, puede ser al final una letra del cuerpo 11, porcentualmente proporcional al diseño, y con otros colores o transformaciones: mayúsculas, minúsculas, texto añadido antes o después, colores de fondo y de la tipografía, subrayados, tachaduras, etcétera.

Algunos gestores de contenido utilizan editores de tipo rich text —texto enriquecido—, que limitan las posibilidades de uso. Un editor

9 Back-end es la parte privada de administración de un gestor de contenidos.

Generalmente son páginas-aplicación con formularios en los que se introduce el contenido que luego se muestra en el front-end, la interfaz pública del sitio web.