

Torres-del-Rey, J. & Rodríguez-V.-de-Aldana, E. (2014). La localización de webs dinámicas: objetos, métodos, presente y futuro. *The Journal of Specialised Translation*, 21, 153-176.
<https://doi.org/10.26034/cm.jostrans.2014.388>

This article is published under a *Creative Commons Attribution 4.0 International* (CC BY):
<https://creativecommons.org/licenses/by/4.0>



© Jesús Torres-del-Rey, Emilio Rodríguez-V.-de-Aldana, 2014

La localización de webs dinámicas: objetos, métodos, presente y futuro

Jesús Torres-del-Rey and Emilio Rodríguez-V.-de-Aldana, University of Salamanca

RESUMEN

En la última década hemos asistido al crecimiento de las webs dinámicas (que presentan una muy superior complejidad estructural y dispersión física del contenido con respecto a las tradicionales webs estáticas) gracias, fundamentalmente, a la aparición de los sistemas de gestión de contenido web (WCMS), que permiten la creación y el mantenimiento de las mismas sin necesidad de conocer los lenguajes informáticos que la sustentan. Este cambio de paradigma se produce cuando las herramientas de traducción asistida por ordenador son cada vez más eficaces y usables para la gestión y localización de sitios web estáticos, de modo que parecería que estamos perdiendo las ventajas de contextualización visual y funcional de la información, la capacidad de reconstrucción y publicación inmediata de la web, y la ayuda de memorias y terminologías. En este trabajo investigamos, siguiendo una orientación didáctica, el estado de la cuestión, comparando los objetos y métodos de traducción de los sitios web estáticos y dinámicos, y analizando la posición del traductor/localizador en este proceso. Posteriormente, apuntamos hacia la investigación y los desarrollos necesarios para la recuperación del terreno perdido con los nuevos WCMS, incluido el uso de estándares como XLIFF, ITS u otras tecnologías de la web semántica.

ABSTRACT

Over the last decade, dynamic websites (which are structurally much more complex than traditional static websites, with localisable content scattered in many more files and databases) have grown exponentially, mainly due to specialised Web Content Management Systems (WCMS), which help create and maintain websites with little or no knowledge of the underlying technology and computer languages. This paradigm shift has taken place just as Computer-Aided Translation Tools have become more and more effective and usable for the translation and localisation of static websites. As a consequence, we seem to be missing out on recently acquired advantages such as visual and functional contextualisation, the ability of web re-creation for immediate publication, and translation memory and terminology aids. This research aims, from a didactic perspective, at exploring the state of the art of static and dynamic web localisation by comparing their translation objects and methods, and analysing the place of the translator/localiser in this process. We then suggest possible avenues for research and development in order to regain lost ground with the spread of new website WCMSs, including the use of standards such as XLIFF, ITS and other semantic web technologies.

Palabras Clave

Sitio web, web estática, web dinámica, sistema de gestión de contenido web (WCMS), traducción asistida por ordenador (TAO), XLIFF, base de datos, HTML, páginas activas, localización, didáctica.

KEYWORDS

Static website, dynamic website, Web Content Management System (WCMS), Computer-Aided Translation (CAT), XLIFF, database, HTML, active pages, localisation, training.

1. Introducción

El lenguaje HTML (HyperText Markup Language) debe formar parte del bagaje cognitivo y técnico de todo localizador que se precie. Sería incluso difícil de justificar que un graduado en Traducción, aunque no se especializara en el ámbito de la localización, desconociera su existencia, su estructura básica, algunos de sus elementos principales, su función y la forma de traducir el contenido codificado en dicho lenguaje. En este sentido, podríamos argumentar que las páginas en HTML se sitúan en el límite entre la especialidad de la localización y la traducción general o especializada tradicionales, debido tanto al tipo de archivos que debe manejarse para su tratamiento como a la generalización de los filtros para este lenguaje en los programas de Traducción Asistida por Ordenador (TAO) convencionales. El siguiente es un esquema que da cuenta de esta dualidad (Tabla 1): así, en unas condiciones u otras, la traducción de páginas en HTML requeriría competencias de traducción 'generales' o especializadas en localización:

	Traducción 'general'	Localización
Tipo de documento	Generalmente sólo se traduce un documento sencillo en formato (X)HTML.	En ocasiones, necesidad de localizar otros archivos asociados, que pueden contener texto traducible (imágenes, CSS, JavaScript...).
Visibilidad de códigos internos / interfaz WSIWYG	La mayoría de los programas de TAO permiten ocultar los códigos internos u ofrecer una interfaz WYSIWYG.	Mayor concentración y visibilidad de código dinámico (enlaces, objetos, <i>scripts</i> ...).
Encaje de los documentos en los programas de TAO y de localización	Encajan naturalmente y ofrecen filtros para HTML, e incluso en ocasiones JavaScript.	Encajan naturalmente y ofrecen <i>parsers</i> (analizadores) para HTML y otros lenguajes dinámicos asociados.
Facilidad de gestión de la macroestructura e hiperestructura textual (sitio web)	Conocimiento especializado y técnico, no habitual entre las competencias generales.	Competencia propia de la localización.

Tabla 1: competencias de traducción y de localización relacionadas con el lenguaje HTML y las webs

Dentro de la localización, por lo tanto, los componentes más especializados del tratamiento de las páginas web en HTML (más allá de la

superestructura o estructura superficial) podrían resumirse en los siguientes (cf. Mata Pastor 2005:200 y ss; Jiménez-Crespo 2013:92-94):

- La macroestructura (la organización de directorios y archivos).
- La hiperestructura (la intervinculación de archivos, y las rutas relativas y absolutas).
- La microestructura (a nivel avanzado, los formatos, lenguajes, las estructuras internas y la semántica de los tipos de archivos que conforman una página web: HTML, CSS, JavaScript...).

A ello podríamos añadir el manejo de los editores de páginas HTML con capacidades de gestión del sitio web sobre los tres niveles referidos anteriormente, y otros conocimientos avanzados de búsqueda, edición y sustitución de patrones o expresiones regulares, así como el tratamiento de las imágenes u otros archivos multimedia vinculados a las páginas web.

Se trata, en definitiva, de competencias que permiten al localizador diferenciarse del traductor general o especializado en géneros no electrónicos (Montalt 2003), y cuyo mayor o menor dominio determinará la posición del traductor o localizador en el escalafón del modelo de globalización o de provisión de servicios lingüísticos avanzados en el que se encuentre. Una web no excesivamente compleja, por lo tanto, estaría, completamente, al alcance del localizador, que podría ofrecer su servicio de "localización total" (cf. Pym 2009:139) o de recreación multilingüe del sitio entero con gran solvencia y, cabría esperar, buenos réditos. Es lo que ocurre, en suma, cuando el traductor se enfrenta a contenido de tipo fundamentalmente estático, con independencia de la complejidad mayor o menor de las distintas hiper, macro y micro estructuras que lo contienen.

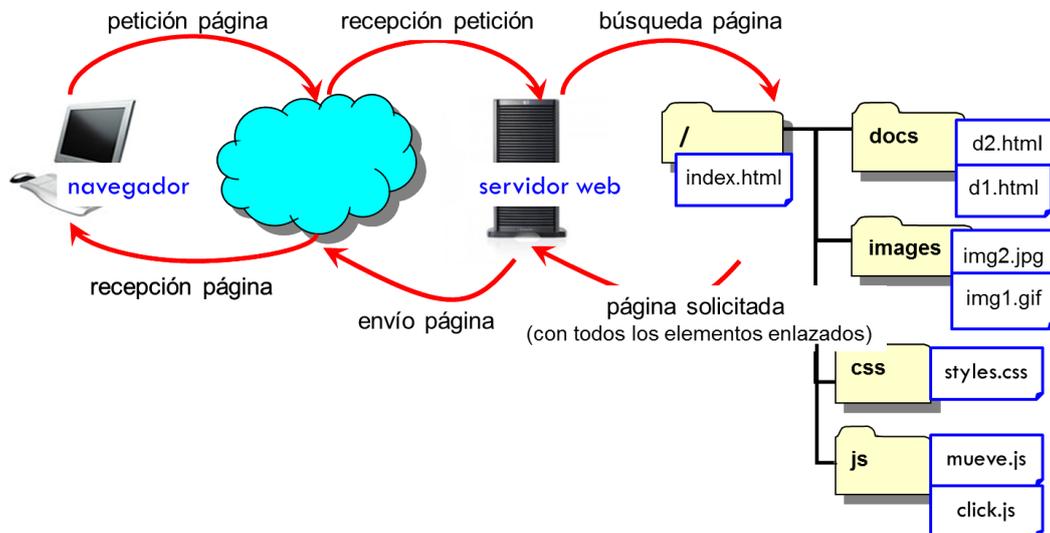
2. Webs estáticas y dinámicas

Lo que caracteriza crecientemente a los productos digitales, no obstante, es su dinamismo; y añadiríamos que, en el *continuum* que podríamos establecer entre traducción general o especializada en géneros no electrónicos, en un extremo, y localización, en el otro, es lo que identifica a esta última. La definición de dinamismo que aquí nos interesa, como veremos a continuación, tiene que ver con la generación sobre la marcha (en el momento de la petición) del producto final así como la posibilidad de una edición compartida y colaborativa del sitio web.

2.1. Webs estáticas

Como es conocido, toda web tradicional (estática) es un conjunto de páginas HTML entrelazadas de alguna forma entre sí y que contienen referencias a otros elementos dependientes: ficheros de imágenes, sonido, video, hojas de estilo, *scripts*.... Ante una petición de una página por parte de un usuario, el servidor que aloja la web se limitará a buscarla

y, si la encuentra, a remitirla al ordenador peticionario con el resto de elementos referenciados. Una vez recibidos, el navegador del solicitante se encargará de interpretar (presentar visualmente) el código HTML (véase ilustración 1). Por tanto, *existe una correspondencia entre los archivos o páginas HTML que se encuentran en el servidor y los que recibe e interpreta el cliente.*



**Ilustración 1. Arquitectura de las webs estáticas
(adaptada de Gutiérrez Gallardo 2010: 25)**

Como ya hemos señalado, esta estructura se encuentra al alcance del todo localizador con conocimientos y destrezas entre básicos e intermedios en la materia: los documentos se pueden traducir sin apenas dificultades mediante un programa de TAO; la adaptación de la estructura puede efectuarse mediante un editor de HTML con gestión de sitios web (como Adobe Dreamweaver o MS Expression Web), a través de la réplica de los directorios de contenido lingüístico y la redirección automática o semiautomática de los vínculos hacia los archivos comunes; la localización de imágenes se realizaría con un programa de edición de gráficos (por ejemplo, GIMP); y el resto de posibles adaptaciones estéticas, técnicas o funcionales, mediante alguno de estos programas u otros adicionales que fácilmente puedan estar a disposición del localizador.

Lo que resulta más importante, sin embargo, es que estas competencias ofrecen al traductor o localizador un grado de autonomía y de control sobre su tarea que lo sitúan en la cúspide profesional que tanto han anhelado autores como Gouadec (2002; Pym 2002), puesto que no sólo sería capaz de traducir el contenido con herramientas que le permiten centrarse en los aspectos textuales relevantes, así como contar con las ayudas contextuales, visuales y funcionales que necesite, sino que, además, sería capaz de recrear la macroestructura e hiperestructura de la web, e incluso preparar fácilmente una versión multilingüe lista para su publicación en el servidor. A ello habría que añadir, por supuesto, la ventaja de poder contar con los recursos lingüísticos (memoria,

terminología, glosarios, traducción automática...) de un programa de TAO y sus utilidades de apoyo a la traducción, verificadores, control de calidad, así como la capacidad de hacer frente a las actualizaciones de la web en el proceso de traducción gracias a dichos recursos.

2.2 Webs dinámicas

Por el contrario, podríamos definir una web dinámica como un conjunto de datos almacenados en una *base de datos* (BD) – gestionada por la capa de software conocida como sistema de gestión de base de datos (SGBD)¹– y, por otro lado, un conjunto de programas conocidos como *páginas activas* (en lenguaje PHP, ASPX, JSP...) – con su correspondiente software procesador o intérprete del lenguaje – que constituyen la interfaz para que los usuarios del sitio manipulen la BD².

Todo el contenido de una web dinámica que los usuarios editores incorporan está de alguna forma alojado en una BD, y las páginas activas se encargan, a través del entorno más o menos sencillo o ergonómico que ofrecen (menús, botones, iconos, cuadros de texto...), de acceder a la BD para reconstruir una página web solicitada por un usuario; o bien modificar aquella y, por ende, ésta, si el usuario que interactúa a través de un navegador tiene privilegios para ello. Valiéndonos de la ilustración 2, ejemplificamos el primer caso: un usuario teclea una dirección o pulsa un enlace en su navegador, con lo que se solicita una página activa. El servidor web recibe la petición y, tras cargar la página activa, el procesador del lenguaje apropiado pasa a interpretar las instrucciones que contiene. Estas últimas, de forma simplificada, se ocupan, en un primer momento, de seleccionar los datos de la BD según la petición recibida (para lo cual tendrá que comunicarse con el SGBD, en el lenguaje que éste proponga) y, a continuación, ubicar los datos recuperados para la composición o maquetación final de todo el código HTML que se enviará al usuario. Todo este proceso hace que se diga comúnmente que las páginas HTML que recibe al final el navegador solicitante se generan 'sobre la marcha', es decir, en el instante de la petición (ver ilustración 3).

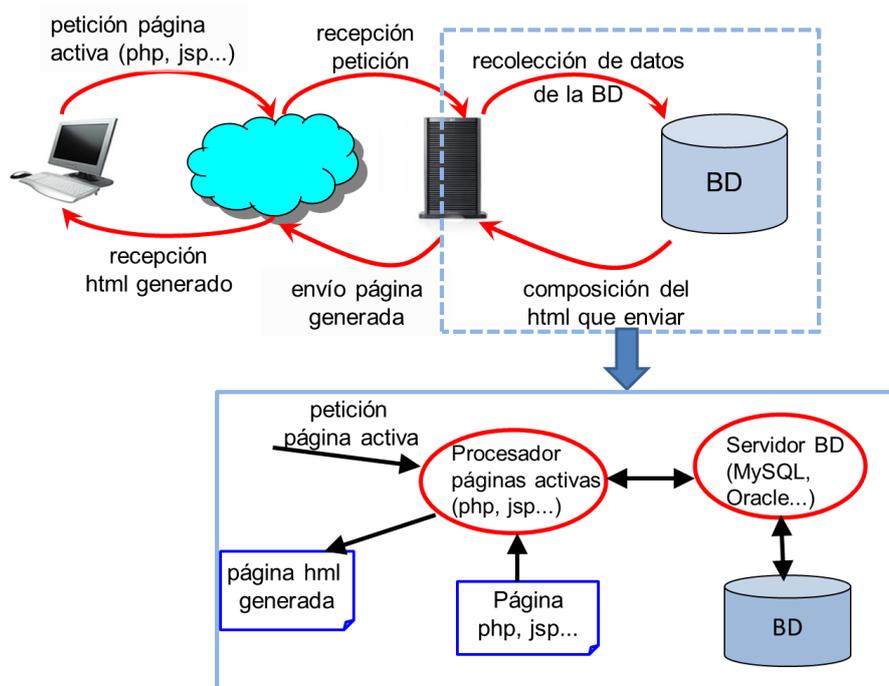


Ilustración 2. Arquitectura básica de una web dinámica

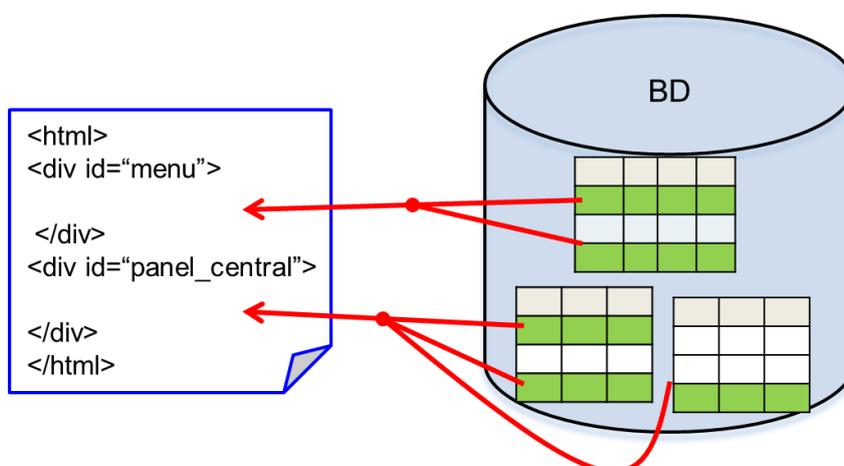


Ilustración 3. Composición 'sobre la marcha' del HTML que enviar

Creemos conveniente aclarar, asimismo, el motivo por el que las webs dinámicas necesitan valerse de un SGBD para la gestión de la web y, por tanto, la razón de esta compleja arquitectura de capas de software, que han de trabajar conjunta y coordinadamente, frente a la más sencilla infraestructura de las webs estáticas. Siguiendo a De Miguel (1993: 56), diremos que un SGBD es una compleja aplicación informática ideada para que múltiples usuarios, de forma concurrente, puedan "*describir, recuperar y manipular bases de datos, asegurando en todo momento la integridad, confidencialidad y seguridad.*" Es decir, un SGBD es, a día de hoy, la infraestructura necesaria para el sostén de la web 2.0, que se caracteriza, como todos sabemos, por la creación, actualización y mantenimiento de sus contenidos de forma flexible y cooperativa. Esta edición compartida de datos no le resulta ajena a un traductor que haya

trabajado con memorias de traducción en red, donde cualquiera de los miembros del equipo que están conectados a dicha memoria en línea, puede, a la vez que los otros, introducir o modificar en ella segmentos bilingües (unidades de traducción), así como ver 'en tiempo real' las aportaciones que están realizando sus compañeros. Como ya se sabe, una memoria de este tipo es una base de datos³ que, naturalmente, estará gestionada por un SGBD⁴.

Repasemos, asimismo, la manera en que se organiza la información en una base de datos. El traductor tiene ya una idea sólida de lo que es una BD pues, como acabamos de mencionar, una memoria de traducción lo es. Como seguramente habrá inspeccionado alguna con su herramienta de TAO, tiene la noción asentada de que la información en ella almacenada se organiza en tablas, que se componen de filas y columnas⁵. Siguiendo con el ejemplo de la base de datos de una memoria de traducción, aquél sabe que cada unidad de traducción se guarda en una fila y que cada fila contiene, en su columna apropiada, información de un segmento en origen y destino, además de ciertos datos administrativos (usuario, fecha, etc.). Y, por tanto, tiene muy claro que un texto se descompone en la memoria de traducción en un conjunto de filas. Esa memoria de traducción, después de ser utilizada como depósito de varios proyectos, contiene un conjunto más o menos amplio de segmentos de múltiples documentos. Aplicando ciertos criterios a los datos, es posible filtrar (seleccionar) un subconjunto entre la totalidad de unidades de traducción existentes. Veamos, a continuación, cómo se refleja este tipo de organización en el caso de una web dinámica.

En la ilustración 4 mostramos el documento HTML que recibe un navegador al solicitar la página principal de una web dinámica que hemos creado. Resaltamos en la misma los elementos del Menú Principal ("MAIN MENU") que, como vemos, el diseñador del sitio ha decidido que se ubique en la esquina superior izquierda. Cada elemento de ese menú se aloja en una fila de una tabla concreta en la base de datos de esta web. Decimos que "en una tabla concreta" porque una BD, salvo que almacene una información muy homogénea, como es el caso de las memorias de traducción, se compone de múltiples tablas para albergar cada una datos de diferente tipo (por ejemplo, una tabla para guardar información de los elementos del menú, otra para los artículos, otra para alojar los datos de los usuarios del sitio...). En la ilustración 5 podemos ver sólo las filas de la tabla concreta que la BD destina para alojar los elementos del Menú Principal. Como puede observarse si se comparan ambas, se trata de las mismas entradas. Para obtener ese subconjunto de filas (sólo las del Menú Principal) la página activa contendrá una sentencia de selección similar a la que se recoge en la parte superior de la ilustración (SELECT ...⁶). Así, para componer toda la página que se muestra parcialmente en la ilustración 4, se necesitarán otras tantas sentencias similares que se aplicarán al resto de la página para, después, servir la petición.



Ilustración 4. Web dinámica de ejemplo, creada con Joomla!

```

SELECT id, menutype, title, level
FROM jen_menu
WHERE menutype = "mainmenu"
AND level = 1
ORDER BY lft
LIMIT 0, 30
    
```

Mostrar : Fila de inicio: Número de filas: Cabeceras cada filas

Ordenar según la clave:

	id	menutype	title	level
<input type="checkbox"/>	435	mainmenu	Home	1
<input type="checkbox"/>	294	mainmenu	Site Map	1
<input type="checkbox"/>	233	mainmenu	Login	1
<input type="checkbox"/>	238	mainmenu	Sample Sites	1
<input type="checkbox"/>	448	mainmenu	Site Administrator	1
<input type="checkbox"/>	455	mainmenu	Example Pages	1

Ilustración 5. Ejemplo de consulta SQL a la base de datos para obtener de la misma sólo las opciones de primer nivel del menú principal

La idea que acabamos de exponer nos permite entender que una base de datos es un conjunto de información almacenado en múltiples filas de diversas tablas, que el diseñador ha organizado siguiendo una cuidadosa metodología (De Miguel, Piattini y Marcos 1999; Silberschatz, Korth y Sudarshan 2006) para que los datos, aparentemente dispersos, puedan combinarse de múltiples formas en una sola tabla según el propósito de la solicitud. Así, como veíamos en las ilustraciones anteriores, la base de datos de nuestra web almacena en una tabla determinada el texto de las opciones del Menú Principal en el idioma de origen (aquí, inglés). Todas las traducciones que se efectúan al español en nuestra web se guardan en otra tabla distinta de la BD. Naturalmente, como la base de datos ha sido diseñada de alguna forma para poder emparejar los textos equivalentes de las diferentes lenguas, es posible, como antes, mediante una sentencia de selección, obtener una tabla que agrupe la información dispersa de ambas. Esto es lo que pretende expresar la ilustración 6: aunque físicamente están en dos tablas distintas de la BD, las opciones del menú principal en inglés y en español pueden reunirse, con la expresión adecuada de selección, en una nueva tabla virtual que empareja en la misma fila el texto inglés con su correspondiente traducción⁷. Es lo mismo que ocurre en realidad cuando se selecciona, en el WCMS, un elemento para traducir (véase, más adelante, ilustración 11): el gestor de contenido realiza una consulta en la BD en función de lo que se ha solicitado traducir, y reúne en una tabla el texto original y el espacio para insertar el de destino.

```

SELECT origen id, origen menutype, origen title, origen type, origen level, destino id, destino language_id,
destino reference_field, destino value
FROM jen_menu AS origen, jen_falang_content AS destino
WHERE origen id = destino reference_id
AND origen menutype = "mainmenu"
AND origen level = 1
AND destino reference_field = "title"
AND destino language_id = 3

```

id	menutype	title	type	level	id	language_id	reference_field	value
435	mainmenu	Home	component	1	1	3	title	Inicio
294	mainmenu	Site Map	component	1	6	3	title	Mapa del sitio
233	mainmenu	Login	component	1	9	3	title	Acceso
238	mainmenu	Sample Sites	component	1	12	3	title	Webs de ejemplo
448	mainmenu	Site Administrator	url	1	15	3	title	Administración de la web
455	mainmenu	Example Pages	alias	1	17	3	title	Algunos tipos de contenido

Ilustración 6. Consulta SQL para obtener de dos tablas diferentes, emparejados, los elementos equivalentes del menú inglés y español

No obstante, también existen webs compuestas por páginas activas que no alojan el texto (sencillo o con elementos HTML) de la interfaz de usuario en bases de datos, sino que lo sitúan en fragmentos de los propios archivos (PHP, ASPX, JSP...), entremezclados con el lenguaje de programación o en documentos independientes que contienen única o fundamentalmente el texto de la interfaz. En este caso, el proceso de localización, que no es el que aquí nos ocupa, es similar al de las webs estáticas puesto que consiste en ubicar y filtrar en los archivos correspondientes el texto traducible, y sustituirlo por la versión lingüística apropiada.

Igualmente, podemos encontrarnos, en webs dinámicas de contenido alojado en bases de datos (el objeto de nuestro estudio), con algunos componentes de la interfaz de usuario que proceden de páginas activas. Generalmente, se trata de términos y frases de corte administrativo (por ejemplo, el nombre de "usuario," "administrador;" o fórmulas como "Creado por," "Fecha de modificación," etc.), que forman más parte del *backend* o la interfaz del administrador del sitio, o del propio software de gestión del contenido web (WCMS), que trataremos a continuación. Aunque el localizador debe ser consciente de esta posibilidad (así como de otros elementos traducibles no basados en BD, como los recursos gráficos, o mensajes contenidos en otros archivos de texto, JavaScript, hojas de estilo, etc.), nuestro análisis se centra exclusivamente en la parte localizable según la definición de webs dinámicas que ofrecimos al comienzo de esta sección 2.2: conjunto de datos almacenados en una *base de datos*.

2.3 Sistemas de gestión de contenido

Hoy en día, los sistemas de gestión de contenido web (*Web Content Management Systems*, o WCMS) permiten a un usuario sin el conocimiento de los lenguajes e infraestructuras que subyacen a toda web dinámica crear y mantener sitios web de este tipo. Un WCMS no es otra cosa que un generador de aplicaciones web dinámicas que se superpone al procesador de páginas activas y al SGBD (la parte inferior de la ilustración 2), ocultando los detalles de esta capa al administrador del sitio. A la popularización de estos sistemas ha contribuido la aparición de WCMS de software libre, fáciles de usar, gratuitos y de gran versatilidad y extensibilidad, como Joomla!⁸, Typo3, Wordpress o Drupal, que ofrecen una interfaz de manejo y configuración. Además, como siempre en este tipo de proyectos, la comunidad de desarrolladores, empresas de servicios y aficionados en general ha ayudado a su expansión mediante la continua creación y cesión o venta de innumerables plantillas, con atractivos y variados diseños de maquetación. Mediante esta infraestructura no sólo es relativamente sencillo y rápido crear toda una web con un número de páginas elevado, sino que se facilita enormemente su modificación y actualización por parte de cualquier usuario sin conocimientos técnicos.

Otra de las ventajas de estos gestores de contenido es la constante adición de extensiones y complementos al software de base, que permiten crear mayores y mejores dinamismos en las formas de interacción de la web con el usuario o el entorno, nuevas fórmulas de presentación de los elementos, etc. Para la localización son de particular relevancia extensiones de localización y gestión de una web multilingüe como FaLang, Joom!Fish o Josetta⁹, que genera estructuras paralelas de la web para los idiomas que se desean agregar, y facilita enormemente la introducción de las traducciones de todos los componentes para cualquier usuario que tenga permisos para hacerlo.

2.4 Clientes y metáforas conceptuales

De cara a los clientes que solicitan la traducción o localización de una web es importante ejercer una labor pedagógica que permita al traductor o localizador poner sobre la mesa su capacidad y sus límites de recreación del producto, así como el volumen de elementos o la cuantificación temporal que supondrá el trabajo. Es fundamental, asimismo, educar al cliente sobre la necesidad de entablar una comunicación entre el traductor, localizador o gestor del proyecto de traducción y quienes han diseñado, mantienen o alojan y publican la web, con el fin de establecer un proceso lo más fluido posible en el arco de trabajo que va desde la descomposición y traducción de los componentes de la web hasta su reintegración y publicación en formato multilingüe o en las distintas versiones lingüísticas (véase Zielinski y Beuster 2012: 4). En nuestra experiencia, éste es uno de los principales orígenes de los malentendidos que llevan a un cliente a sufrir larguísimos procesos de rediseño de la web y publicación, por no hablar del elevado coste que puede suponer.

Desde el primer momento, el traductor debe analizar el tipo de web, si es estática o dinámica, cuál es su composición, como se relacionan los distintos elementos para su publicación, las tecnologías empleadas, y cuál es el procedimiento, si existe, para generar y publicar las traducciones. Esto permitirá aconsejar al cliente sobre cuál es la mejor estrategia de trabajo y hacerle entender, tal y como comentábamos en el párrafo anterior, que debe existir una comunicación fluida entre los distintos agentes del proceso completo de traducción-publicación. Aquí, nos parece de gran utilidad mostrar al cliente, gráficamente y de una manera adecuada a la imagen que éste ya tiene formada sobre el producto digital de consumo masivo que es la web, cuál es la forma física y el mecanismo de funcionamiento de la misma, y cómo influirán estos aspectos en todo el proceso.

Las metáforas se nos antojan en este caso muy útiles para la comunicación entre el traductor o localizador y el cliente. A partir de las características definitorias de las webs estáticas y dinámicas que apuntábamos anteriormente, podría hablarse, por ejemplo, de las primeras como un conjunto de fichas unidas cada una por un brazo a un

mecanismo de rueda dentada, que permite, pulsando una pestaña en su base, elevar y presentar dicha ficha. La traducción consistiría en la sustitución de la ficha en un idioma por otra en una lengua diferente (entendiendo, además, que contaríamos con herramientas que automatizan dicha sustitución de la ficha); la recreación multilingüe supondría la adición de un segundo mecanismo, o la ampliación del número de fichas, copiando las primeras para alojar las del segundo idioma.

Sin embargo, en el caso de las webs dinámicas, podríamos pensar en un motor o en una planta de impresión, de gran complejidad interna y alimentados por energía externa, donde las distintas máquinas y componentes combinan bajo demanda los elementos textuales y no textuales. Sólo en determinadas partes del proceso, o en su producción final o impresión, se podrían ver los resultados, que, no obstante, nunca existen de forma prefijada. En este caso, el traductor podría tomar fotografías de los compuestos finales y devolver otras fotografías que contengan la traducción (también mediante herramientas que automatizarían la sustitución textual dentro de la fotografía), pero ello requeriría todo un proceso posterior de extracción de las traducciones y reintegración dentro de la maquinaria de producción, buscando los compartimentos en los que insertar cada parte textual.

Estas metáforas, u otras semejantes, muestran al cliente que una web dinámica es, internamente, un 'motor' de software compuesto por piezas de distinto tipo, en las que se entremezcla el texto traducible. El traductor o localizador puede poner todo de su parte para extraer las páginas finales en el idioma de origen y devolver un conjunto que sea lo más parecido a aquéllas, e incluso que ofrezcan cierta ilusión de funcionamiento semejante; pero sin el acceso a toda la tecnología y a los mecanismos de procesamiento y publicación, sólo se le estarán ofreciendo 'instantáneas' estáticas de la web, en lugar de un 'motor reajustado' para acomodar idiomas adicionales, o, cuando menos, las piezas adecuadas que se ajusten al software que servirá la web a los distintos idiomas.

3 Métodos de localización de las webs dinámicas

Si resumimos los apartados anteriores, y dejando de lado los elementos multimedia, podríamos decir que en las webs estáticas todo es tangible para el traductor de una manera más o menos inmediata. Puede recibir archivos HTML y, mediante programas de TAO, devolver los mismos tipos de archivos. Por otro lado, gracias a editores de lenguaje HTML capacitados para gestionar a nivel de sitio, podría recrear la estructura de archivos, carpetas y vínculos para obtener una web multilingüe. Sin embargo, en las webs dinámicas, como hemos intentado explicar, el contenido y la estructura están dispersos en múltiples filas de diversas tablas de una base de datos, y en el código de las páginas activas. En consecuencia, el ciclo de la traducción también será diferente, puesto que

se deberá extraer, de alguna forma, el contenido traducible en origen de la BD y, como paso final del proceso, alojar dicho contenido en destino dentro de la propia BD.

Puesto que este tipo de webs dinámicas se está implantando cada vez con mayor fuerza entre la comunidad de diseñadores, a continuación analizaremos cuáles son, en nuestra opinión, los métodos de localización que, en estos momentos y dado el estado de desarrollo de software, tiene a su disposición el traductor (con mayor o menor intervención de técnicos informáticos).

3.1 Método 1: uso de TAO con descarga de páginas HTML

La fórmula que permite que el trabajo del traductor se realice de manera más contextualizada, consiste, en primer término, en descargar las páginas en la versión HTML que recibe el navegador. De este modo, aquél puede manejar un programa de TAO (con su filtro apropiado para este tipo de archivos), para valerse de sus memorias de traducción, terminología y resto de utilidades. A la vez, si dicho programa dispone de módulos de vista previa, el traductor podrá obtener la experiencia gratificante de ser consciente del resultado estético y funcional de cada traducción, si bien, como ya sabemos, la publicación final requiere algún proceso adicional.

En la ilustración 7 podemos ver una representación de esta fórmula de trabajo. La mayor parte de las operaciones se realizan en el entorno de un programa de traducción asistida por ordenador (TAO): en dicha ilustración esta secuencia de pasos – elipses – queda dentro del recuadro de trazo discontinuo.

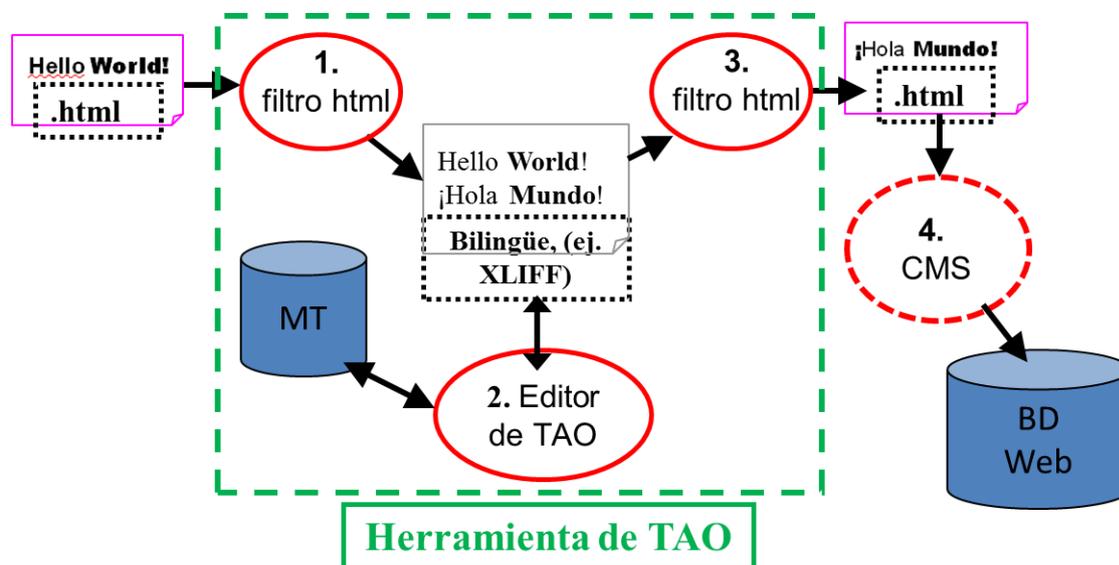


Ilustración 7. fórmula de localización de webs dinámicas mediante descarga de páginas HTML

Se trata de una solución ideal para gestores de proyectos que quieren distribuir documentos web a traductores poco o nada expertos en localización. Lo más sencillo, en este caso, es descargar cada una de las páginas web recibidas de internet en el navegador como *sólo HTML*. Si se seleccionara la opción de "página completa" para guardar el archivo, el navegador descargaría una carpeta de archivos dependientes (imágenes, hojas de estilo, librerías JavaScript...) y trataría de modificar todo lo necesario el código del archivo HTML para que la página funcionara bien localmente, cambiando referencias de vínculos e incluso eliminando fragmentos que considere prescindibles. Por eso, la descarga en modo "sólo HTML" garantizaría dos cosas importantes: no obligar al traductor a manejar varios archivos o carpetas (lo que podría confundirle) y tener un archivo que, en el caso de webs estáticas, permitiría la rápida publicación del resultado, con el mínimo de modificaciones de código. Para la localización de webs dinámicas, el segundo beneficio no sería aplicable, aunque sí el primero. Además, al descargar (sólo) el código HTML de las páginas dinámicas, generalmente se presentarán vínculos internos con direcciones absolutas (URL) a todos los archivos dependientes en internet; o bien rutas relativas al directorio raíz, indicándose que las mismas partan de la URL de base (mediante la etiqueta, atributo y valor `<base href="url de base">` en la cabecera `<head>`). Ambos mecanismos ofrecen la ventaja de que quien maneje estos archivos descargados pueda verlos como si estuviera navegando en internet (si tiene conexión)¹⁰.

Otra opción es utilizar un programa descargador o copiador de webs que permite la navegación sin conexión a internet (*offline browser*), como HTTrack Website Copier o Scrapbook (extensión, esta última, para Mozilla Firefox). Bien configurados, y dependiendo de la complejidad de la web, estos programas podrían descargar el conjunto de archivos y la estructura de directorios que componen una web estática o los que sirven para el funcionamiento de las distintas páginas de una web dinámica, e incluso buscarían las fórmulas para que se pudiera navegar entre las distintas páginas de manera local. Evidentemente, la entrega de este conjunto al traductor requeriría cierta dosis de instrucciones si aquél no tiene experiencia en el manejo avanzado de sitios web.

Ahora bien, tal y como se recordará, cuando el traductor devuelve los archivos HTML localizados, aún es necesario reintegrar el contenido en la base de datos, para que, como hemos explicado, el 'motor' de composición HTML (procesador de páginas activas y SGBD) sirva las páginas localizadas de la web dinámica correctamente. Es decir, como fase final, aún falta descomponer el archivo HTML para que las distintas partes traducidas se ubiquen en el lugar apropiado de la base de datos. Se trataría de realizar el proceso inverso del que se mostraba en la ilustración 3¹¹. Si el localizador es el encargado de llevar a cabo esta tarea (el paso 4 de la ilustración 7), generalmente empleará un método semiautomático, consistente en copiar el contenido de las páginas HTML en cada sección,

artículo o página de la web dinámica mediante la interfaz de edición de su gestor de contenido web (WCMS); es decir, copiando y pegando (como se muestra en la ilustración 8)¹².

No obstante, hay que tener en cuenta que se trata de una tarea laboriosa (debe copiarse y pegarse, en el lugar adecuado, cada elemento de información, y, probablemente, reconstruir enlaces internos) y para la que el localizador debe adquirir, además, un conocimiento suficiente del gestor de contenido. Al final, será el propio WCMS, como ya se ha indicado, el encargado de ubicar el contenido traducido en la base de datos (véase la ilustración 8). Para este proceso de incorporación del contenido HTML de la página en su ubicación apropiada de la base de datos podría crearse un programa que lo automatizara, es decir, que lo realizara sin ninguna intervención humana¹³.

Por último, la gestión de las actualizaciones del contenido original se beneficiaría del control de cambios del propio WCMS y de la capacidad del programa de TAO para mostrar las diferencias entre el nuevo texto que traducir y su correspondencia con las frases de la versión anterior de dicho texto que se hayan almacenado en la memoria de traducción.

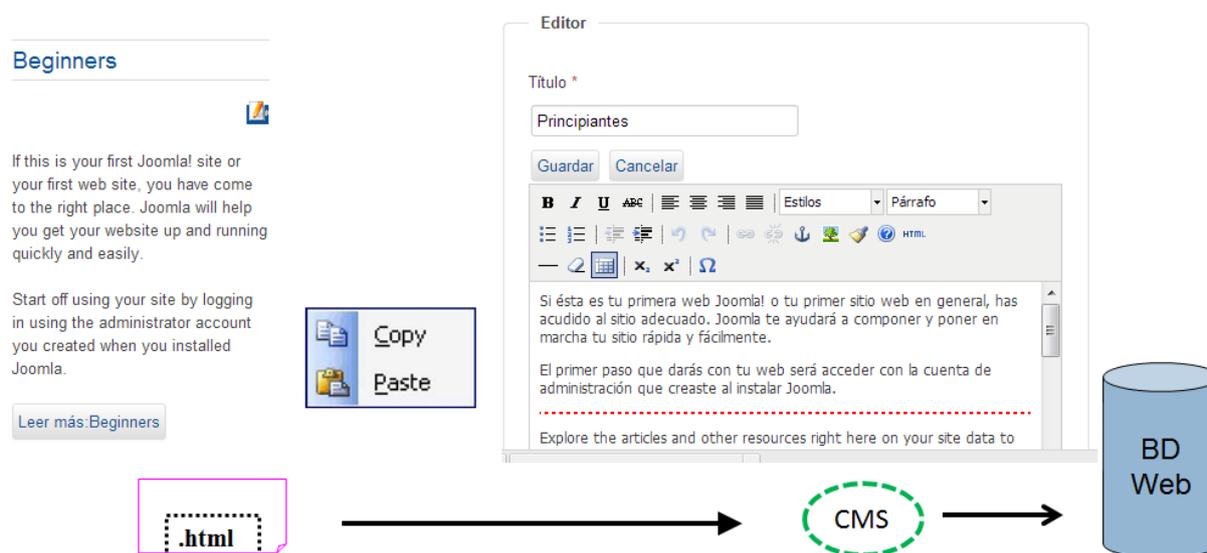


Ilustración 8. copiado y pegado del contenido HTML mediante el WCMS

3.2 Método 2: exportación e importación del contenido traducible de la base de datos

Otra fórmula posible consiste en la exportación, por parte del responsable del WCMS, de las cadenas traducibles de la base de datos a un formato de texto, que puede ser CSV (valores separados por comas), ya más en desuso, o, actualmente, alguna variedad de XML (siendo XLIFF el estándar para el intercambio de objetos e información de localización¹⁴). El

traductor recibe el contenido traducible en alguno de estos formatos, y los filtros de las herramientas de traducción o localización asistida por ordenador (predeterminados, adaptados o creados por el usuario) extraerán los campos adecuados (en el caso de CSV) o los elementos o atributos traducibles (en el caso de XML). Finalmente, el traductor devuelve al cliente el archivo en el mismo formato, quien se ocupará de realizar la importación mediante las consultas y actualizaciones que haya previsto para la base de datos.

Este proceso ya se puede realizar en Drupal mediante el módulo XLIFF Tools (Schnabel 2013); en Wordpress, se puede emplear el módulo XLIFF del complemento WPML que mencionábamos antes (véase nota nº 9); TYPO3 permite la exportación/importación de archivos XML mediante la extensión Localization Manager. En el caso de Joomla!, el componente de gestión multilingüe JDiction ha agregado recientemente una utilidad de exportación/importación de XLIFF¹⁵.

Evidentemente, aunque este método también le permite al traductor hacer uso de sus útiles, memorias y terminología, esto le supondría la pérdida de contexto visual (en qué sección, elemento u objeto aparece el texto) y funcional (como resultado de qué operaciones o acciones aparece el texto, y qué otras funciones permite realizar) en el momento de la traducción. En este caso, son menos los procesos que el traductor o localizador puede realizar por sí mismo, como apreciamos en la siguiente ilustración 9.

Para la gestión de las actualizaciones, se contaría con los mismos beneficios que en el método anterior, incluida la intervención del programa de TAO para marcar las correspondencias y diferencias entre versiones, si bien no sería el traductor quien tuviera que llevar el control de los cambios en la web (y, por lo tanto, en la base de datos).

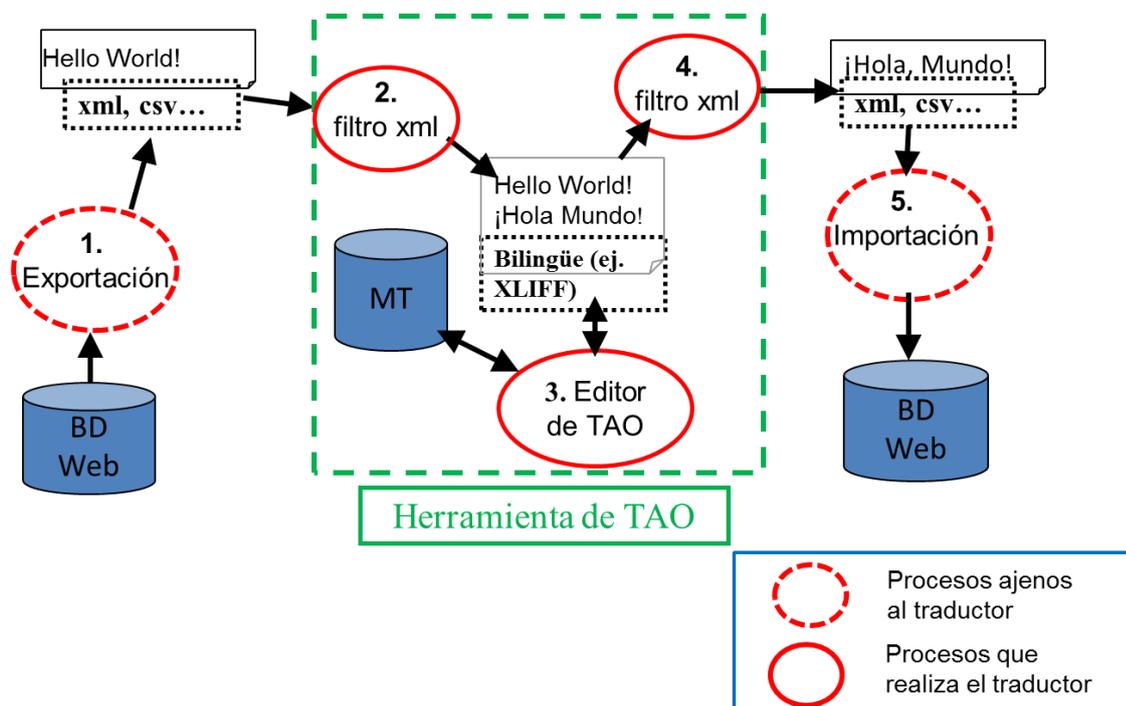


Ilustración 9. método de localización de webs dinámicas mediante exportación e importación del contenido traducible de la base de datos

3.3 Método 3: traducción desde el propio gestor de contenido

La tercera fórmula consiste en facilitar un acceso a la edición de los distintos componentes de la web a través del propio sistema de gestión de contenido web (WCMS), siempre y cuando esté preparado para la edición multilingüe (por ejemplo, en nuestra web, mediante el complemento FaLang; véase la nota nº 9). En principio, esto elimina la necesidad de todos los procesos que mediaban, en los métodos anteriores, entre la selección del contenido que traducir y la incorporación de la versión traducida a la base de datos. Por otro lado, al recibir ésta los datos directamente a través del WCMS, las actualizaciones se mostrarán inmediatamente en la web con la siguiente petición que se realice.

Además, este método simplificaría el trabajo del traductor en dos aspectos: en primer lugar, le mostraría el contexto visual del contenido que debe traducir, puesto que, en gran medida, éste realizaría su transformación 'sobre' el propio elemento de la web; en segundo lugar, no tendría que preocuparse más que de traducir una sola vez los elementos comunes del sitio (menús de navegación, pies de página, etc.) ya que dichos datos también están almacenados *una sola vez* en la BD.

En la ilustración 8 veíamos, al final del método anterior, un ejemplo visual de traducción, con un CMS como Joomla! (con la extensión FaLang), de un elemento de contenido o artículo que aparece en la portada de nuestra web. Tras ingresar en la web con el nombre de usuario que le permita realizar las modificaciones oportunas, el traductor no tiene más que

seleccionar el artículo sobre el que desea trabajar y pulsar el icono de edición que se muestra en la parte superior derecha de dicho contenido, insertar la traducción en la ventana proporcionada y guardar los cambios. Para traducir componentes más dinámicos, como los menús de navegación (por ejemplo, los de la ilustración 4), el usuario con permisos debe realizar, en estos momentos, un proceso un poco más complejo. Así, tras seleccionar la opción de traducción (“Translation”) de nuestro gestor multilingüe (accesible desde el menú “Componentes” de Joomla!), se elige el idioma de destino y el elemento que desea traducir (un menú, en nuestro caso), lo que devolverá las filas que se muestran (ilustración 10), para que se pueda marcar el texto que traducir (aquí, mediante casilla de verificación) y editarlo. En la siguiente ventana, se presenta el texto original y el espacio para insertar la traducción (ilustración 11), que tras guardarla en estado “Publicado” (Published), actualizará con estos valores las filas correspondientes de la BD y se mostrarán inmediatamente, en la siguiente solicitud de la web.

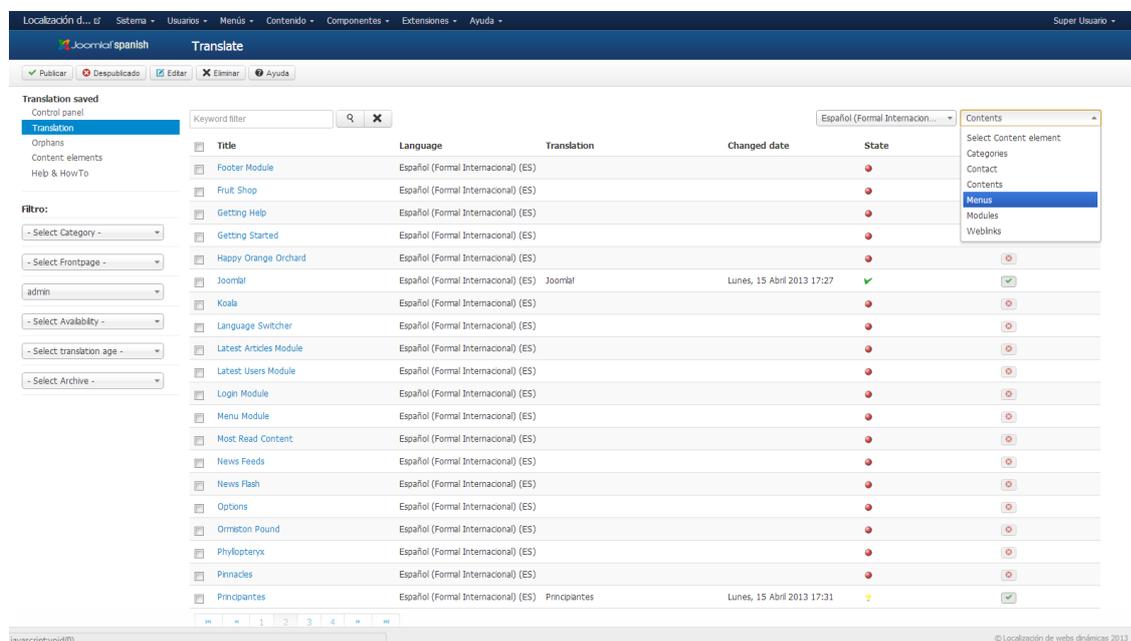


Ilustración 10. selección de idioma y elemento que traducir mediante FaLang

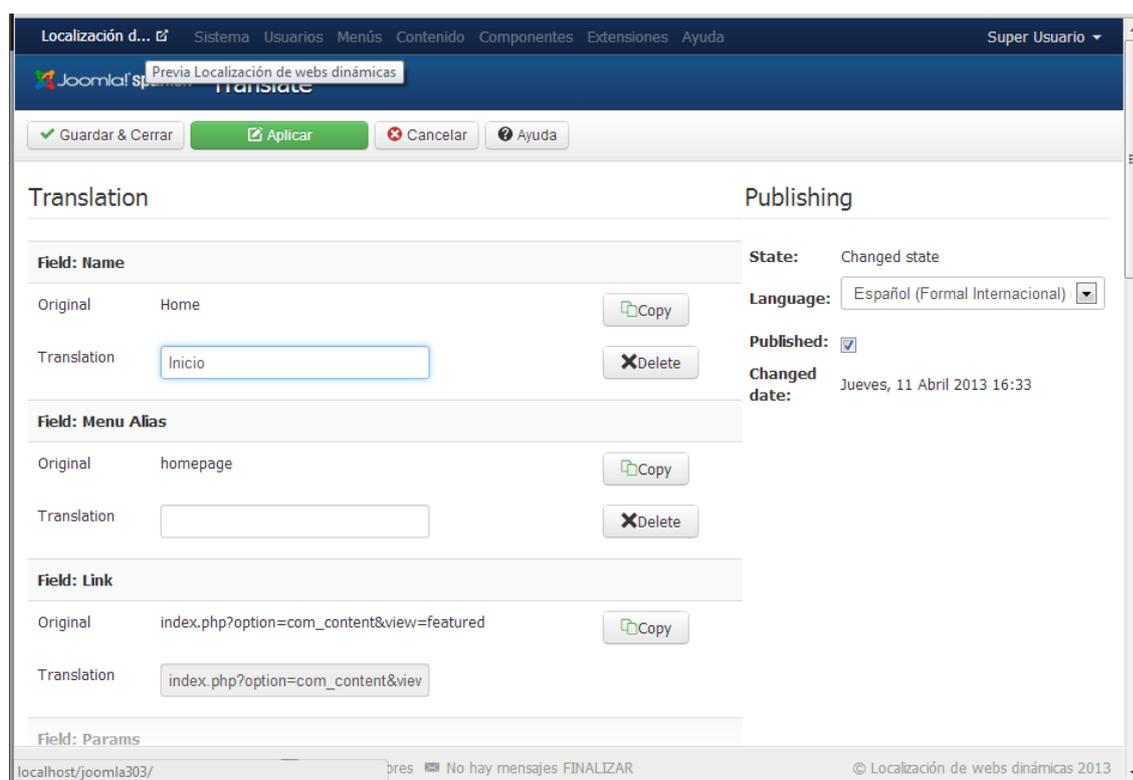


Ilustración 11. Inserción de la traducción en la ventana de edición del elemento de menú "Home"

Varios son los inconvenientes de este método (cf. Schnabel 2013: 12): en primer lugar, el traductor ha de recibir derechos administrativos del sitio; por otro lado, no estaría beneficiándose de todas las herramientas que le proporciona un programa de TAO, incluidas las correspondencias de los textos actualizados con los fragmentos almacenados en la memoria de traducción; por último, debe conocer el funcionamiento del gestor de contenido, incluida la manera de localizar (adaptar) los vínculos internos, y recibir indicaciones muy precisas sobre los elementos concretos que debe traducir, así como si puede, o debe, tocar algunos de los campos asociados al texto (alias, etc.).

Propuestas de futuro

El 'perfecto' engarce entre la localización y los sistemas de desarrollo y gestión de contenido de una web dinámica multilingüe comenzará a producirse solamente cuando haya un compromiso mutuo entre las formas de interpretar, estructurar y dar significado a la información y la tarea de cada uno de estos ámbitos, haciendo uso, eso sí, de la evolución de los lenguajes y tecnologías que buscan la máxima interoperabilidad técnica y semántica entre herramientas y agentes implicados, en sintonía con los esfuerzos e iniciativas que se despliegan en eventos periódicos como FEISGILTT (*Federated Event for Interoperability Standardization in Globalisation, Internationalisation, Localisation and Translation Technologies*), que en su edición de 2013 dedicó un amplio espacio

monográfico de exposición y debate a la localización mediante CMS, en el que participamos los autores de este artículo.

En línea con el análisis de métodos de trabajo presentado en las anteriores secciones, adivinamos una convergencia de los mismos hacia distintas fórmulas de integración que privilegien, en un primer estadio, alguno de los siguientes objetivos: 1. los archivos HTML finales (de cliente), generados por el WCMS y procesados visualmente por el sistema de TAO; 2. el entorno de producción (el gestor de contenidos, en sintonía con un sistema de traducción asistida por ordenador); 3. el estándar de intercambio de localización (XLIFF), alimentado tanto por el WCMS como por la herramienta de TAO. Cada uno de ellos presentan diversos grados de complejidad técnica, así como ventajas e inconvenientes para el localizador, que le aportan una mayor o menor capacitación o 'empoderamiento' en los términos que indicábamos al principio de nuestro artículo.

Uno de los elementos clave de las propuestas 1 y 2 es el contexto visual y funcional que ofrece al traductor trabajar sobre los objetos web en sí, ya sea en su versión resultante (HTML) o dentro del mecanismo de producción (WCMS). La primera parece promulgarse en algún artículo relacionado con el CMS Typo3 (Zielinski y Beuster 2012: 15), donde éste generaría archivos HTML marcados convenientemente para su posterior reimportación en el CMS, tras ser traducidos mediante la herramienta adecuada. La segunda, a día de hoy, posiblemente sea la de mayor complejidad técnica, al tener que integrar y sincronizar un entorno y unos mecanismos de gestión de contenido, vinculados a una base de datos, con el entorno y los mecanismos de traducción asistida por ordenador, vinculados con otras bases de datos. Mientras se trabaja en dichas vías (de manera aún marginal), cabe destacar que ambas estrategias presentan potenciales puntos débiles contrapuestos: las versiones navegables en HTML no contienen, superficialmente, todas las posibles formas de interacción dinámica con la misma, y las herramientas de TAO no siempre son capaces de procesar contextualmente las dependencias de estos archivos con otros; por el contrario, el sistema de gestión de contenidos podría dar acceso al localizador a los mecanismos significativos de producción, agrupación y definición de contenido, estructuración del mismo y dependencias existentes, a costa de cierto alejamiento, durante la traducción, de las condiciones finales de uso.

El formato de intercambio de contenidos de webs dinámicas mediante XLIFF es una de las soluciones de futuro a las que se están adaptando los WCMS poco a poco, como ya comentábamos anteriormente. Su principal ventaja consiste en que el archivo podría exportarse con el texto traducible y con una serie de metadatos que ayuden a contextualizar la información, e, idealmente, conseguir cierta visualización y ubicación de lo que se traduce. Como el propio acrónimo indica, sus puntos fuertes se encuentran en el formato estándar (XML) de intercambio (exportación –

importación) de (elementos de) localización, esto es, que podría proporcionar tanto la información textual como las características y los procesos que intervienen en la tarea y, dicho sea de paso, la especializan con respecto a la traducción. Esto permite, asimismo, que intervengan distintos agentes (traductores, revisores, terminólogos, etc.) sobre un corpus extenso, idealmente anotado para el proceso, y que se puedan emplear herramientas de diverso tipo para extracción de terminología y control de calidad, por ejemplo.

El camino de la estandarización y la interoperabilidad parece ser uno de los más prometedores, entre otras cosas porque el contenido de una web puede transformarse y emplearse en otros soportes y tipos de información por la misma organización, como promueven estándares de redacción y publicación (generalmente técnica) como DITA; o el más reciente CMIS, relacionado con el intercambio de información entre CMS, si bien, a día de hoy, está más orientado a los repositorios documentales que a formatos web¹⁶. El primero, junto con las categorías de internacionalización de información del estándar ITS¹⁷, podría facilitar la creación de contenido internacionalizado, anotado en función del tipo de recurso y otros elementos relevantes para la traducción y la localización (posición relativa, tamaño, vínculos de los objetivos localizables, anotaciones lingüísticas o terminológicas, etc.). Entre los servicios que promuevan la interoperabilidad de los WCMS, por otro lado, sería interesante contar con una capa de abstracción que defina, de manera estandarizada, dónde (en qué base de datos y campos) se encuentra la información extraíble para la localización, lo que se comunicaría de manera transparente a las herramientas de procesamiento circular o de ida y vuelta (*roundtrip*), que exportarían y reimportarían los datos traducibles.

Por último, el formato XLIFF se beneficiaría de la estandarización e interoperabilidad que pueden aportar los estándares que acabamos de mencionar, pero también debería aprovechar las novedades de su próxima versión 2.0 para aportar una mayor contextualización para el localizador, mediante la adquisición o el enriquecimiento de metadatos relevantes, la vinculación con hojas de estilo o con los propios recursos en línea, todo lo cual debe realizarse en sintonía con los desarrolladores de herramientas de traducción asistida por ordenador, de modo que se tiendan todos los puentes posibles entre los conceptos y mecanismos de los sistemas de gestión de contenido y los de traducción y localización.

Bibliografía

- **De Miguel, Adoración y Mario Piattini** (1993). *Concepción y diseño de bases de datos. Del Modelo E/R al Modelo Relacional*. Madrid: Ra-ma.
- **De Miguel, Adoración, Piattini, Mario y Esperanza Marcos** (1999). *Diseño de bases de datos relacionales*. Madrid: Ra-ma.
- **Elmasri, Ramez and Shamkant B. Navathe** (2007). *Fundamentos de sistemas de bases de datos*. Madrid: Addison Wesley.
- **Gouadec, Daniel** (2002). "Training Translators: Certainties, Uncertainties, Dilemmas." Belinda Maia, Johann Haller and Margherita Ullrych (eds) (2002). *Training the Language Services Provider for the New Millennium*. Porto: Universidade do Porto, 31-41.
- **Gutiérrez Gallardo, Juan Diego** (2010). *Desarrollo Web con PHP 6 y MySQL 5.1*. Madrid: Anaya Multimedia.
- **Jiménez-Crespo, Miguel A.** (2013). *Translation and Web Localization*. London and New York: Routledge.
- **Montalt, Vicent** (2003). "La traducción de géneros electrónicos: el caso de la localización." Miguel Á. García Peinado and Emilio Ortega Arjonilla, Emilio (dirs) (2003). *Panorama actual de la investigación en traducción e interpretación*, vol. II, parte II, Sección III, 313-328.
- **Pym, Anthony** (2002). "Training Language Services Providers: Local Knowledge in Institutional Contexts." Belinda Maia, Johann Haller and Margherita Ullrych (eds) (2002). *Training the Language Services Provider for the New Millennium*. Porto: Universidade do Porto, 21-30.
- **Pym, Anthony** (2009). *Exploring Translation Theories*. Oxford: Routledge.
- **Rodríguez-V.-de-Aldana, Emilio y Jesús Torres-del-Rey** (2014). "Localización del texto de una web multilingüe creada con un gestor de contenidos: el ejemplo de Joomla!". Belén Santana López and Crispulo Travieso Rodríguez (eds) (2014). *Puntos de encuentro: los primeros 20 años de la Facultad de Traducción y Documentación de la Universidad de Salamanca*. Salamanca: Universidad de Salamanca.
- **Schnabel, Bryan** (2013). "Making the Multilingual Web Work: When Open Standards Meet CMS". *W3C Workshop Program: Making the Multilingual Web Work*. Roma, 12-13 de marzo de 2013. http://www.w3.org/International/multilingualweb/rome/slides/11_schnabel.pdf (consultado 13.09.2013).
- **Silberschatz, Abraham, Korth, Henry F. and S. Sudarshan** (2006). *Fundamentos de bases de datos*. Madrid: McGraw-Hill.
- **Zerfass, Angelika** (2002). "Evaluating Translation Memory Systems." *LREC 2002: Language Resources in Translation Work and Research*. (Las Palmas de Gran Canaria, 29-13 May 2002), 49-52.
- **Zielinski, Daniel and Martin Beuster** (2012). "Localizing Dynamic Websites Created from Open Source Content Management Systems." *memoQfest 2012* (Budapest, 10 de mayo de 2012).

http://www.loctimize.com/fileadmin/user_upload/termine/20120510_memoQfest/20120510_WebL10N_memoQfest_Budapest.pdf (consultado 13.09.2013).

Notas biográficas

Jesús Torres del Rey es profesor del Departamento de Traducción e Interpretación de la Universidad de Salamanca, donde imparte asignaturas de grado y posgrado sobre tecnologías de traducción, localización y gestión de proyectos, y donde se doctoró en 2003 con una tesis sobre las implicaciones filosóficas y prácticas de las nuevas tecnologías en el campo de la traducción y en su enseñanza. Es miembro de The Institute of Localisation Professionals (TILP).

jtorres@usal.es



Emilio Rodríguez Vázquez de Aldana es profesor del Departamento de Informática y Automática, adscrito a la Facultad de Traducción y Documentación de la Universidad de Salamanca, donde ha impartido docencia en diferentes asignaturas tanto de Documentación como de Traducción e Interpretación. Su labor docente e investigadora se ha desarrollado en el ámbito de la recuperación de información y el procesamiento del lenguaje natural, y en la actualidad se centra en las tecnologías de traducción y la localización.

aldana@usal.es

¹ El acrónimo inglés es Database Management System –DBMS–. Productos de este tipo son, entre otros muchos, MySQL y Oracle.

² Con la expresión “manipular una base de datos” nos referimos tanto a consultar como a actualizar (insertar, borrar o modificar) información.

³ Nos referimos a los sistemas de memorias de traducción organizados en bases de datos, según la clasificación de Zerfass (2002).

⁴ Por ejemplo, las memorias de traducción de SDL Trados, Kilgray memoQ y Atril Déjà Vu X2 en red se gestionan mediante el SGBD Microsoft SQL Server.

⁵ Las tablas no son la única forma de organizar datos en las bases de datos. Existen otras, pero ésta es la dominante (Elmasri y Navathe 2007).

⁶ Dicha sentencia pertenece al lenguaje SQL (Structured Query Language), un lenguaje estándar (ANSI e ISO) para la manipulación de BD basadas en lo que se conoce como modelo relacional. Muchos de los SGBD actuales incorporan dicho lenguaje. Por la misma razón, muchos de los productos incorporan dichas siglas en su nombre: MySQL, PostgreSQL, SQLite, SQL Server... Naturalmente hay productos, por otro lado muy destacados, que también adoptan dicho lenguaje sin incorporar tales siglas: DB2, Oracle... La expresión SQL que se recoge en la ilustración 5 está indicando al SGBD que se quieren obtener los datos de las columnas "id", "menutype" ... (línea 1: SELECT...) de la tabla en la base de datos llamada "jen_menu" (línea 2: FROM jen_menu), pero sólo de aquellas filas que satisfagan la condición que se expresa en las líneas 3 y 4: WHERE... Además, se mostrarán, en la consulta, en el mismo orden determinado para la web (campo de anidamiento "lft").

⁷ Se las denomina tablas "virtuales" porque, aunque físicamente no existan en la base de datos, se obtienen, en tiempo de ejecución de la consulta, a partir de los datos de las tablas realmente almacenadas. Por otro lado, la sentencia SQL de la ilustración 6, en la cláusula SELECT..., está indicando (como en la expresión que comentamos anteriormente) que recupere información de unas determinadas columnas. Ahora, si nos fijamos, se señala en la cláusula FROM... que esas columnas se encuentran en dos tablas ("jen_menu" –origen– y "jen_falang_content" –destino–), cuya forma de combinarse adecuadamente se expresa en la cláusula WHERE...

⁸ Todas las ilustraciones que hemos mostrado en este artículo se han creado mediante el WCMS Joomla! 3.0.3.

⁹ Nuestros ejemplos se han realizado con el complemento para Joomla! FaLang 1.2.0 (<http://www.faboba.com/composants/falang/presentation.html>). Existe para Drupal una serie de módulos que, tras instalarse y activarse, facilitan la gestión multilingüe e internacionalización (<http://drupal.org/project/i18n>). En el caso de Wordpress también existen complementos (*plugins*) para la gestión multilingüe, como WPML (<http://wpml.org>) o el gratuito .qTranslate (<http://www.giangin.de/qtranslate>).

¹⁰ Si no se diera ninguna de estas dos alternativas, el gestor del proyecto podría insertar la URL de la página descargada como <base> de las rutas, en la cabecera (<head>) del archivo HTML.

¹¹ Es decir, en vez de componer HTML a partir de los datos dispersos en la BD, ahora se trata de descomponer HTML para reubicarlo adecuadamente en la BD.

¹² Además de las referencias visuales y funcionales que le ofrece al traductor manejar una versión HTML de la página, otra de las ventajas de trabajar con este formato es que presenta poco o ningún riesgo de 'ensuciar' el contenido de la página web con códigos extraños.

¹³ Seguramente esta última opción se realice en muy pocos casos, puesto que probablemente nuestro cliente desconozca los entresijos del gestor de contenidos. Sería posible llevarlo a cabo si contara con personal informático cualificado.

¹⁴ El estándar *XML Localization Interchange File Format* está gestionado por el consorcio sin ánimo de lucro OASIS: <https://www.oasis-open.org/committees/xliff>.

¹⁵ Véase el proyecto XLIFF Tools de Drupal en <http://drupal.org/project/xliff>, que está basado en la hoja XSL de la aplicación XLIFF Roundtrip de Bryan Schnabel (<http://sourceforge.net/projects/xliffroundtrip>). La extensión Localization Manager se puede encontrar en: <http://typo3.org/extension-manuals/l10nmgr/3.4.0/view>. JDiction es accesible desde <http://jdiction.org/en/jdiction/about-jdiction>.

¹⁶ Ambos son estándares mantenidos por OASIS: *Darwin Information Typing Architecture* (<https://www.oasis-open.org/committees/dita>) y *Content Management Interoperability Services* (<https://www.oasis-open.org/committees/cmisis>).

¹⁷ Se trata del *Internationalization Tag Set*, del consorcio W3C (<http://www.w3.org/TR/its20/>), que acaba de aprobar su nueva versión 2.0, con importantes novedades para el ámbito de la localización y la traducción.