

Claves para entender LA LOCALIZACIÓN

El investigador **Juan José Arevalillo** nos brinda elementos precisos para comprender los procesos de localización. Para ello es necesario combinar traducción y manejo de herramientas informáticas.

El futuro ya llegó... Y las tecnologías de la información han revolucionado la modalidad de trabajo de los traductores, específicamente en lo que concierne al uso de las herramientas informáticas. En consecuencia, los traductores han desarrollado nuevas especializaciones, entre ellas la localización, que es **"el proceso de adaptación y fabricación de un producto para que presente el aspecto y el funcionamiento de un artículo fabricado en un país en concreto"**. (*Guía de Introducción a la localización, Localization Industry Standards Association - LISA* (2003)).

Ahora bien, para hablar de localización y tratar de explicar en qué consiste su aplicación nos basaremos en un extracto del texto *"Introducción a la localización, su presencia en el mercado y su formación específica"*, de Juan José Arevalillo, publicado en *"La linterna del traductor"*, número 8 de marzo de 2004.

En lo que se refiere a localización son cuatro los componentes que nos interesan desde el punto de vista del traductor: interfaz de usuario, ayuda en línea, documentación impresa y material complementario.

La interfaz de usuario es la carta de presentación visual del programa de localización. Este componente debería ser el primero en traducirse, ya que la terminología empleada en él marcará de modo significativo el resto de los componentes, dadas las continuas referencias que se hacen en la ayuda y la documentación a los menús, cuadros de diálogos y mensajes que se presentan en pantalla. El traductor debe poseer ciertos conocimientos de programación, aunque existen aplicaciones especializadas que permiten una traducción visual. Antes de abordar las distintas formas de realizar la traducción, es conveniente disponer de una visión general del proceso que recorre un programa desde su escritura en el denominado **código fuente** hasta que se llega al archivo ejecutable. Esto es de suma importancia, porque según el método de traducción se utilizarán unos programas u otros, y el proceso cambiará radicalmente, con sus ventajas e inconvenientes. Dentro de la interfaz gráfica de usuario, tres son los componentes principales: menús, cuadros de diálogo y mensajes.

La traducción de archivos de código fuente es el método de traducción más tradicional, en el que

se trabaja con los archivos originales escritos en el código correspondiente (por lo general, el lenguaje de programación C o alguna de sus variantes). Para ello, el traductor debe saber muy bien las características principales del código y, sobre todo, las del texto traducible.

El problema con este método de traducción reside en el código que circunda al texto. Ese código no debe tocarse bajo ningún concepto, ya que puede dar problemas en la compilación e incluso evitar que ésta se produzca. Un borrado accidental de un solo carácter puede dar lugar a un problema con una función del programa. De ahí que se recomiende el uso de un programa de traducción asistida que proteja el código y que sólo presente para su traducción el texto debido. En la actualidad, la mayoría de estos programas dispone de filtros específicos para abordar estas traducciones, siempre con la garantía añadida de respetar la integridad del código próximo.

Dentro del texto entrecorrido que sí debe traducirse pueden encontrarse determinados elementos que precisan de un tratamiento especial, como son los aceleradores,

los caracteres de control, las variables, los códigos de control, etcétera.

En el planteamiento de localización de archivos binarios, todas las actividades se centran en los módulos compilados de la aplicación, en lugar de los archivos de código fuente. Esto también supone una reducción importante en el número de archivos y de los trabajos iniciales de preparación. Después de la traducción, tampoco se precisa ninguna compilación, ya que la traducción y la modificación visual del formato se producen directamente en el archivo binario ya traducido, por lo que pueden pasar de inmediato al proceso de control de calidad.

Este planteamiento se vio favorecido por el desarrollo informático en general. De hecho, la manera en que se programa en la actualidad ha cambiado diametralmente en comparación con el método más antiguo, en el que el programador escribía el código, texto, coordenadas, funciones, etcétera, sin ningún tipo de ayuda que no fuera su propio ingenio y capacidad, y algunas plantillas predefinidas del lenguaje en cuestión.

Las exigencias del mercado dieron lugar a la programación orientada a objetos (*Object-Oriented Programming*, OOP), que indirectamente facilitó la labor del traductor hasta límites insospechados, ya que éste disponía de una ayuda visual para la creación de los archivos de código y de recursos: basta con la selección de los objetos y las características y funciones deseadas (por ejemplo, los botones de un cuadro de diálogo) que se incluyen en un archivo de apariencia gráfica. Al guardar ese archivo, toda esa representación gráfica se convierte internamente en código, que el programador se evita escribir, compatible con la sintaxis del lenguaje de programación en sí; es decir, el programador puede crear un archivo de recursos prácticamente sin saber programación: es suficiente con su intuición para elegir los objetos o elementos visuales correspondientes, puesto que el programa en uso se encarga de la codificación interna, con lo que el programador puede centrarse realmen-

te en el trabajo de diseño y olvidarse de la tediosa labor de picar el código.

Este tipo de programación es la que dio origen a los programas de traducción para interfaces de usuario, como **Passolo** o **Catalyst**, que permiten ver el objeto y el contexto que lo rodea con una representación visual, y no su código. A esta representación se la denomina **WYSIWYG** (del inglés *What You See Is What You Get*), y permite traducir las cadenas de texto directamente sobre el menú o cuadro de diálogo y ver el resultado al instante, sin necesidad de otros procedimientos de programación.

En lo
que se refiere
a localización
son cuatro
los componentes
que nos
interesan
desde el
punto de vista
del traductor:
interfaz de usuario,
ayuda en línea,
documentación
impresa
y material
complementario.



Juan José Arevalillo es Licenciado en Filología Inglesa por la Universidad Complutense de Madrid y Traductor Superior Especializado por el IULMyT.

Lleva 21 años en el campo de la traducción y es el director gerente de Hermes Traducciones y Servicios Lingüísticos.

Participa en numerosos seminarios universitarios y es miembro de pleno derecho de la LISA, la ATA, la ACT y miembro benefactor de la Fundación Pro Real Academia de la Lengua Española. A su actividad laboral, se añade su dedicación como profesor de Traducción en la Universidad Alfonso X el Sabio de Madrid.