

Nuevas Tecnologías

Software Libre

Free Software

Resumen

Ha surgido en el mundo informático una tendencia en auge. Se trata de una apuesta por y para la libertad, abanderada por el concepto de “Software Libre”, libre para usar, libre para modificar, libre para divulgar.

El “Software Libre” es a la informática lo que la “Libertad de Expresión” es al lenguaje. Es la “expresión libre del software”. Partiendo de esta exclusiva premisa y respaldados por una vasta comunidad de usuarios y desarrolladores, podemos afirmar en plena era Web 2.0 que el software libre es el futuro.

Su particular definición (más filosófica que rigurosa), sus licencias asociadas y sus innumerables ventajas en todos los campos hacen del software libre el modelo tecnológico a seguir en todo proyecto vinculado mínimamente con las Tecnologías de la Información y la Comunicación.

Palabras clave: Software, libre, libertades, código fuente, código abierto, GNU, FSF, licencia, GPL, copyleft.

Abstract

A peaking trend has emerged in computing world. It is about a bet to and for freedom, led by the concept of “Free Software”, free to use, free to modify, free to release.

“Free Software” is to computing what “Freedom of Speech” is to language. It is the “free speech in software”. From this unique premise and supported by a vast community of users and developers, we can assert in full Web 2.0 era that free software is the future.

Its particular definition (more philosophical than rigorous), its associated licenses and its innumerable benefits in all areas make free software a technology model to be followed in any project minimally related with Information and Communication Technologies.

Keywords: Software, free, freedom, source code, open source, GNU, FSF, license, GPL, copyleft.



Todos usamos o hemos usado software libre en algún momento, quizá sin saberlo o tal vez con total conocimiento de ello.

Si en lugar de Internet Explorer, usamos Mozilla Firefox, estamos utilizando un navegador de Internet libre. Si en lugar de Microsoft Office instalamos OpenOffice, tenemos a nuestra disposición un paquete ofimático libre. La Wikipedia es una enciclopedia pública con contenido libre. Emule, Azureus, Gimp, Blender, PdfCreator, LibreOffice, Audacity, VLC, Apache, PHP y MySql son ejemplos de software libre que pueden instalarse bajo Windows, que, por cierto, es un sistema operativo no libre. Disponemos por suerte de alternativas a Windows completamente libres como Fedora,

Debian, Opensuse, Mandriva, Knoppix, Ubuntu, Guadalinex o cualquier otro sistema operativo basado en Linux.

Poco a poco el software libre se va haciendo hueco en los hogares de millones de usuarios. Uno de los sectores de mercado que ha experimentado mayor acogida es el de la telefonía móvil, con dispositivos cada vez más potentes y con un abanico cada vez más variado de aplicaciones.

¿QUÉ ES ‘SOFTWARE LIBRE’?

Para entender perfectamente a qué se refiere la terminología de ‘Software Libre’ necesitamos definir ambas palabras separadamente:



Software

No tiene traducción literal del inglés y por tanto queda admitida en nuestro idioma. El diccionario de la Real Academia Española indica que es el *“Conjunto de programas, instrucciones y reglas informáticas para ejecutar ciertas tareas en una computadora”*. Esta definición es escueta pero precisa. Hace referencia a la parte lógica de un ordenador, es decir, a los componentes no físicos de un sistema informático: aplicaciones, drivers, librerías de funciones, archivos, carpetas e incluso el propio sistema operativo. Por el contrario, los componentes físicos conforman el denominado *“Hardware”*.

Todo software está constituido en su origen por un sinnúmero de complejas instrucciones que elabora un equipo de desarrolladores informáticos. Estas instrucciones no son más que órdenes que recibe un ordenador para ejecutar una determinada acción, escritas en un lenguaje específico que *“facilita”* la comunicación entre humanos y computadoras. El texto resultante se denomina *“código fuente”* (este término es importante para entender correctamente el resto del artículo). Después, es necesario el proceso de traducción del código fuente a *lenguaje máquina* (de unos y ceros) que es el que finalmente entiende y ejecuta un ordenador. Este proceso de traducción se denomina *“compilación”*.

Libre

“Que no está sujeto ni sometido”. Esta es la acepción del término que más se ajusta a su significado cuando se asocia al concepto de *“Software”*.

En inglés es un término ambiguo que puede dar lugar a cierta confusión, debido a que *“free”* significa tanto *“libre”* como *“gratis”*, y el Software Libre no tiene por qué ser gratis.

En definitiva, lo correcto es entender el término como *“que tiene libertad”*.



Software Libre

Tal y como expone el Proyecto GNU (el movimiento más importante y decisivo en el desarrollo de Software Libre) es una cuestión de libertad, no de precio: *“...you should think of «free» as in «free speech», not as in «free beer».”* - Se debería pensar en *“libre”* como en *“libertad de expresión”*, no como en *“cerveza gratis”*.

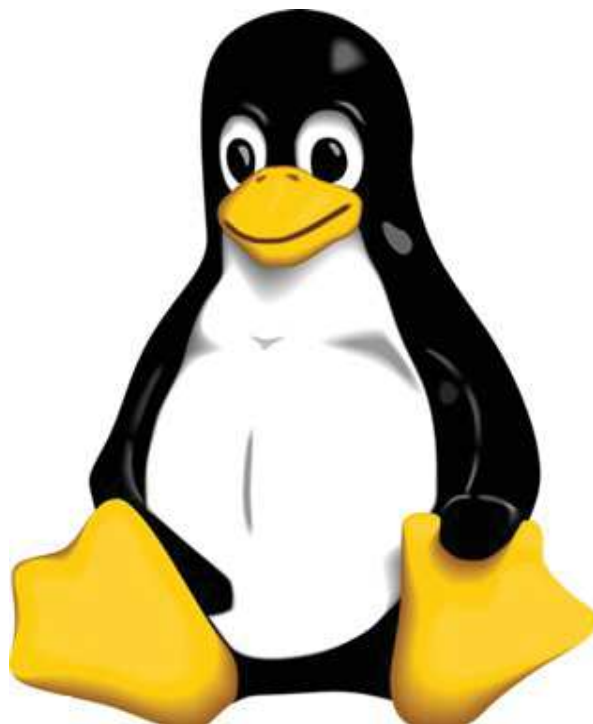
“Software Libre” es el que da al usuario la libertad de usarlo, copiarlo, distribuirlo, estudiarlo, cambiarlo y mejorarlo sin restricciones.

Normalmente, este tipo de software es gratuito, pero no tiene por qué. Un usuario tiene libertad de distribuirlo ya sea gratis o cobrando una pequeña cuantía por su distribución.

El software que no proporciona esta serie de libertades se denomina *“Privativo”*, *“Propietario”* o, directamente, *“No libre”*.

UN POCO DE HISTORIA

Hace cincuenta años, los grandes avances se realizaban a nivel de hardware. Las empresas que desarrollaban computadoras incluían el software como producto añadido para poder hacer uso de ellas (al igual que hoy en día un usuario compra una lavadora sin prestar atención al tipo de software que integra...)



TUX. La mascota oficial de Linux, núcleo de los sistemas operativos libres y principal ejemplo de software libre.

Ya en los 70, antes del boom de la informática, las compañías comenzaron a imponer restricciones al usuario mediante ciertos acuerdos de licencia. El acceso al código fuente se hacía cada vez más dificultoso o directamente imposible.

En la década de los 80, con la llegada de los primeros ordenadores personales, la informática se puso de moda y las empresas incluían en estas modernas máquinas sistemas operativos privativos, que forzaban al usuario a aceptar restricciones en cuanto a la modificación de su software.

Así pues, si un usuario encontraba un error en el sistema de su ordenador, lo máximo que podía hacer era escribir a la empresa propietaria, informar del problema y esperar a que fuese subsanado en diversas actualizaciones o en versiones posteriores del sistema. Incluso tratándose de un experto informático capaz de resolver el error por sí mismo, era imposible acceder al código fuente para corregirlo, ya que éste no se entregaba con el sistema.

Fue en esta época cuando un programador estadounidense harto de estas restricciones, **Richard Stallman**, fundó el movimiento por el Software Libre, creando a su alrededor un marco de referencia moral, político y legal para su desarrollo.

Stallman inició el **Proyecto GNU** con el fin de crear un sistema operativo libre que se erigiese como alternativa directa a los sistemas operativos privativos. Fundó la **Free Software Foundation** o



Richard Stallman. 1953. E.E.U.U. Fundador de la FSF y del Proyecto GNU

FSF (Fundación para el Software Libre) con el propósito de difundir este movimiento. También inventó el concepto de **Copyleft** como competencia legal a la licencia **Copyright**.



Logo del Proyecto GNU (Cabeza de ñu)

CUESTIÓN DE LIBERTAD

El Proyecto GNU establece cuatro libertades muy claras que debe cumplir todo software para recibir la condición de “Libre” (como buena enumeración informática que se precie, utiliza la secuencia numérica que se usaría en términos de programación, del cero al tres):

- **Libertad 0.-** que permite **ejecutarlo** con cualquier propósito.
- **Libertad 1.-** que permite **estudiar** cómo funciona, cambiarlo y adaptarlo a las necesidades del usuario.
- **Libertad 2.-** que permite **distribuir** copias y así ayudar al prójimo.
- **Libertad 3.-** que permite **mejorarlo y hacer públicas las mejoras**, para que toda la comunidad se beneficie.

El software deja de ser libre si falla alguna de las cuatro libertades.

Gracias a la **Libertad 0** podemos usar el software como queramos, cuando queramos y en tantos ordenadores como necesitemos, independientemente del tipo de equipo informático en el que lo instalemos, de la finalidad de su uso y sin la obligación de pedir permiso a su desarrollador o comunicárselo a alguna empresa u organización.

Según esto, podemos descargar un programa libre de Internet e instalarlo en varias máquinas sin necesidad de pagar varias licencias. En el software privativo protegido por copyright se debe pagar una licencia por cada ordenador en el que se instale. Es decir, que si disponemos de dos ordenadores en casa debemos comprar, por ejemplo, dos **Windows** para poder usarlos legalmente.

Si queremos entender cómo funciona un programa, podemos hacer uso de la **Libertad 1**, que nos permite estudiarlo y adaptarlo a nuestras necesidades. Por tanto, la posibilidad de acceder al código fuente se convierte en una condición totalmente necesaria para el software libre.

Esta libertad nos permite entender nuestro software y ser espectadores o, en su caso, actores directos del desarrollo y progreso del mismo. Ante nosotros se abre una puerta al conocimiento del código y al aprendizaje de las técnicas de programación actuales. Gracias a la disponibilidad del código fuente podemos detectar rápidamente errores, agujeros de seguridad o incluso algún tipo de código malicioso dentro del programa.

Es un sello de identidad del software libre el estar totalmente **exento de virus**.

Un programa con esta libertad se denomina “*de código abierto*” (en inglés: “*open source*”). Por tanto, todo software libre ha de ser de código abierto. Sin embargo, no todo software de código abierto tiene por qué ser libre (si no cumple con todas y cada una de las demás libertades).

Una vez tengamos en nuestro poder un programa libre, podemos hacer uso de la **Libertad 2** y copiarlo en cualquier dispositivo de almacenamiento para pasárselo a otra persona. No hay restricciones en el número ni en la finalidad de las copias. Podríamos distribuirlo gratuitamente o incluso cobrando una cantidad por los gastos de distribución. Por supuesto, el código fuente de todo lo que se distribuya debe ser accesible al público de alguna manera.

Cualquiera que compre o copie alguna de nuestras distribuciones de software tiene la libertad de distribuirlo nuevamente o usarlo en un número ilimitado de sistemas informáticos sin nuestro permiso.

Como consecuencia de las tres primeras libertades el software libre adquiere una gran versatilidad, pero es en la **Libertad 3** donde realmente radica su verdadera potencia. Existe una enorme comunidad de desarrolladores, intercomunicados vía web, que comparten experiencias en torno a este software, recogen todo tipo de observaciones de usuarios, corrigen problemas de funcionamiento e incluyen mejoras que son rápidamente publicadas en Internet a disposición nuevamente



de toda la comunidad. Como es lógico, para ello se requiere el acceso total al código fuente del software.

Los objetivos de esta práctica, tal y como promueve el proyecto GNU y la FSF, son puramente éticos, de mejora común, solidaridad y de acceso libre al conocimiento.

Para que las cuatro libertades sean consecuentes, deben ser irrevocables. Si un desarrollador impone una restricción a alguna de ellas mediante un tipo concreto de licencia, el software se convierte inmediatamente en “no libre”.

LICENCIA PARA COPIAR

Una **licencia de software** es una autorización de carácter contractual entre el desarrollador y el usuario y establece los derechos, las obligaciones y las libertades que éste último posee.

Si un software no se protege con ningún tipo de licencia se dice que es **de dominio público**, es decir, que es un bien común para la humanidad. Ésta podría parecer la práctica lógica y habitual para conseguir la categoría de software libre, puesto que satisface sus cuatro libertades. Pero, desgraciadamente, también permitiría que un usuario de escasa moral reutilizase el código, realizase alguna modificación y lo hiciese suyo mediante una licencia restrictiva.

La mayoría de software no libre tiene una **licencia propietaria** y está amparado por derechos de autor o **copyright**, que es la forma de protección proporcionada por las leyes vigentes en la mayoría de los países para los autores de cualquier tipo de obra original.

¿Cómo puedo proteger mi software para que sea libre?

Si bien una licencia propietaria es demasiado restrictiva, la carencia de licencia lo haría vulnerable a convertirse tarde o temprano en privativo. Para ello, el proyecto GNU crea la **licencia GPL** (Licencia Pública General), que básicamente viene a defender las cuatro libertades del software libre. Esta licencia no es compatible con la protec-



• Símbolo del Copyleft.

ción que ejerce el copyright, así que Richard Stallman tiene que inventar el concepto de **copyleft**, que es un juego de palabras entre “copyright” y “left” (participio de “leave”, que a su vez puede significar “permiso”). Gracias a este nuevo término, las libertades del software que defiende la licencia GPL se hacen extensibles a todas las distribuciones y modificaciones posteriores, de modo que ningún usuario tiene la posibilidad de revocar alguna de las cuatro libertades asociadas. El copyleft, por tanto, garantiza que el usuario mantenga su libertad.

Para mucha gente la licencia GPL y el copyleft son de alguna forma restrictivos, porque *obligan* al usuario que modifica el código fuente de un programa libre a liberalizar el resultado bajo el mismo tipo de licencia.

Hay otras licencias menos *restrictivas* aún que se acercan bastante al dominio público. Este es el caso de la **BSD** (Berkeley Software Distribution), que permite el cambio de licencia de un software modificado a otra más o menos libre. Por lo tanto, un usuario puede decidir incluso redistribuir este tipo de software como no libre.

Existen intensos debates sobre qué licencia es más libre o menos restrictiva, pero no es esa la intención de este artículo, puesto que supondría entrar en planteamientos metafísicos similares a la cuestión de si un país que permite la esclavitud es más libre que otro que no la permite...

Otros tipos de software no libre

Un tipo de software gratuito pero no libre es el **freeware**, que se distribuye necesariamente sin costo, con libertad de uso y por tiempo ilimitado. Sin embargo, normalmente no incluye el código fuente.

Otra modalidad aún más extendida es el **shareware**, que distribuye el software con ciertas limitaciones, en versión de demostración, de evaluación o con límite de tiempo.

Cuando un programa es gratuito y sin limitaciones pero incluye publicidad durante su ejecución se denomina **adware**.

A veces, estas modalidades son potencialmente peligrosas, porque incluyen en su interior

(inaccesible, claro está) algún código malicioso (malware) o espía (spyware).

¿POR QUÉ USAR SOFTWARE LIBRE?

Todos hemos encontrado un poco tedioso el cambio de unas versiones de Windows a otras. Al final nos damos cuenta de que es un proceso natural que no supone más de unos cuantos días de aprendizaje hasta que nos habituemos al entorno de trabajo.

No es muy diferente el cambio a un sistema operativo libre como por ejemplo **Ubuntu o Guadalinex**, con la ventaja de que podemos optar por diferentes entornos de escritorio y elegir el que más nos guste (o el que más se parezca al de Windows).

Pudo ser también incómodo el paso de Word 2003 a 2007 por cuestiones de incompatibilidad de sus archivos. Es algo fácil de solucionar, puesto que hay programas especialmente diseñados para hacerlos compatibles, aunque es lógico pensar que habrá ciertas características de los documentos de Word 2007 que se pierdan al producirse la conversión. Aún así, todos sabemos que para un usuario poco experimentado todo esto puede causar problemas.

Existen alternativas libres al paquete ofimático Office, como por ejemplo **OpenOffice o LibreOffice**. Tienen la ventaja de que son compatibles con los documentos que genera Microsoft Office, además de que tienen un diseño muy sencillo que facilita su aprendizaje.

Es comprensible que un usuario normal de software privativo tenga ciertas reticencias al decidirse a apostar únicamente por el software libre, ya que está muy acostumbrado a realizar una serie de operaciones con su ordenador de una forma concreta y con unos programas determinados que ya domina (en mayor o menor medida) y con los que se siente cómodo.

El cambio a software únicamente libre implica un periodo de adaptación y aprendizaje que no mucha gente está dispuesta a llevar a cabo. Aun así, son muchas las ventajas que supone este cambio.

Ventajas Económicas

Una vez que comienza a circular un producto de software libre, rápidamente está disponible gra-

tuitamente o a un costo muy bajo. Esto, unido a la libertad de usarlo y distribuirlo, permite que a menudo encuentre un hogar entre usuarios para los cuales el precio del software privativo es prohibitivo, convirtiéndose así en una alternativa a la piratería.

Si nos disponemos a instalar un programa privativo en diferentes ordenadores, tenemos que comprar diferentes licencias. Por el contrario, si conseguimos un programa libre (ya sea gratis o pagando una pequeña cantidad) podemos instalarlo en cuantas máquinas queramos sin problemas legales.

Así, una empresa podría disminuir sensiblemente los costes de una solución informática completa, ahorrando en licencias y asignando más presupuesto al soporte técnico o al propio equipo de hardware.

Además, los requisitos de hardware para el funcionamiento del software libre suelen ser menores. Por ello, el usuario no está atado a la compra de un determinado procesador o cantidad de memoria RAM. Esa compra dependerá únicamente de la velocidad con la que se deseen realizar ciertas tareas.

Ventajas Educativas

La gran adaptabilidad del software libre resulta de gran utilidad en sectores específicos como el de la educación. Así, además de economizar gastos, este software permite satisfacer necesidades específicas de enseñanza-aprendizaje, mediante la creación de programas especialmente diseñados para cubrir una materia y un nivel concretos.

Se favorece de esta forma la libertad y la independencia de la comunidad educativa en la elección tecnológica, potenciando la capacidad crítica y la autonomía en este campo. También se facilita que el alumnado disponga en casa de las mismas herramientas educativas que se utilizan en su centro, de manera totalmente legal.

Existen distribuciones de algunos sistemas operativos libres destinadas a la educación como **Guadalinex Edu** o **Edubuntu**, que contienen infinidad de aplicaciones con fines pedagógicos.

Ventajas Empresariales

Pese a las reticencias que algunos sectores plantean ante la renuncia a las ganancias que obtie-



nen con la venta de licencias, hay diversas maneras de hacer negocio con el software libre.

Algunas empresas comienzan a estructurarse alrededor de este software para ofertar servicios adicionales como soporte técnico, asesoría, patrocinio, mantenimiento, instalación, personalización, etc. Se abre, por tanto, un futuro prometedor para el desarrollo de industria local y para aquellas personas con nivel informático avanzado.

Gracias a la libertad de modificar sin restricciones un programa libre se acorta el ciclo de desarrollo de aplicaciones. Una empresa de diseño de software que parte de un programa base para construir otro adaptado a ciertas necesidades ahorra mucho más tiempo (y dinero) que si parte desde cero.

Todo esto queda bien resumido en el decálogo publicado por el CENATIC (**C**entro **N**acional de Referencia de Aplicación de las **T**ecnologías de la **I**nformación y la **C**omunicación basadas en fuentes abiertas) titulado "10 razones para que los autónomos usen Software Libre":

1. Permite ahorrar en la adquisición, mantenimiento y renovación de tecnologías.
2. Las aplicaciones libres tienen mayor calidad y son más completas.
3. Garantiza la seguridad.
4. El uso de software libre favorece la independencia tecnológica del autónomo.
5. El software libre es una tecnología de fácil acceso y se adapta mejor a la realidad del autónomo.
6. El software libre es una tecnología 100% legal.
7. Las tecnologías libres tienen un soporte técnico más accesible.
8. Fomenta la creación de un modelo productivo más colaborativo basado en la cooperación.
9. Seguir la tendencia de los clientes en el uso de software libre.
10. Las aplicaciones en software libre son más fáciles de aprender.

Ventajas Sociales

Al disponer del código fuente es fácil modificarlo

localmente, lo cual posibilita traducciones a idiomas no del todo rentables comercialmente.

También existen proyectos de voluntariado en el tercer mundo que utilizan software libre con el fin de expandir el conocimiento a lugares que no poseen los medios económicos necesarios para usar software privativo común.

La Free Software Foundation argumenta que el software es conocimiento y, como tal, debe poderse difundir sin trabas. El conocimiento es un bien común que debe estar al alcance de todos.

Otras Ventajas

El software libre es siempre *legal*. Por lo tanto su uso nos exime de toda posible sanción con respecto a licencias y permisos.

Siempre está sometido al *escrutinio público*, con lo que el proceso de corrección de errores es muy dinámico y no hay que esperar a que el proveedor de software saque a la luz una nueva versión.

Al ser de código abierto, existe mayor dificultad para introducir en su interior *código malicioso* tipo espía, troyano, puerta trasera o virus. Las distribuciones más comunes de sistemas operativos libres están exentas de ello.

El software libre garantiza el *respeto a los estándares en los formatos*, mientras que el software propietario los cambia a menudo para obligar al usuario a cambiar de versión.

EN DEFINITIVA

Gracias a todas estas características, libertades y ventajas y a la idiosincrasia de la comunidad que las promueve, el software libre se está convirtiendo poco a poco en algo cada vez más cotidiano y está dejando de ser asunto de técnicos o usuarios avanzados.

Será, sin duda, un concepto que dentro de unos años formará parte de nuestro vocabulario común.

DOCUMENTOS BIBLIOGRÁFICOS

- FSFE (Free Software Foundation Europe). "What is Free Software?". En <http://fsfe.org/about/basics/freesoftware.en.html>



- GNU Operating System. "The Free Software Definition". En <http://www.gnu.org/philosophy/free-sw.html>
- HISPALINUX. Hacia la Sociedad del Conocimiento Libre. "Qué es el Software Libre?". En <http://www.hispalinux.es/SoftwareLibre>
- Red Costarricense de Software Libre. "¿Qué es Software Libre?". En http://www.softwarelibrecr.org/faq/software_libre
- Escuela Virtual para América Latina y el Caribe del Programa de las Naciones Unidas para el Desarrollo. "Software Libre". En http://www.escuelapnud.org/files/pub_pages/about_us/attachments/software_libre_v2.pdf
- Del Río, J.A.; Cedano, K. (2007). "Software Libre" Academia de Ciencias de Morelos, A.C. Contribución del 10-12-2007. En http://www.acmor.org.mx/descargas/dic10_software.pdf
- GrULiC (Grupo de Usuarios de Software Libre de Córdoba). "¿Qué es GNU / Linux?". En <http://www.grulic.org.ar/node/10>
- Monografias.com. El Centro de tesis, documentos, publicaciones y recursos educativos más amplio de la Red. En <http://www.monografias.com/trabajos12/elsoflib/elsoflib.shtml>
- Macuarium. Actualidad Macintosh en Español. "Qué es el Software Libre?". Publicado el 7-12-2002. En http://www.macuarium.com/macuarium/actual/especiales/2002_12_07_opensource.shtml
- Daccach, J.C. DELTA. (Delta Asesores). "¿Para qué Software Libre?". En <http://www.deltaasesores.com/articulos/tecnologia/5108-ipara-que-el-software-libre>
- CENATIC (Centro Nacional de Referencia de Aplicación de las TIC basadas en fuentes abiertas). "10 razones para que los autónomos usen Software Libre". En <http://www.cenatic.es/documentos/10-razones-para-que-los-autonomos-usen-software-libre>

JUAN JOSÉ MUDARRA GÁMIZ