

# SOFTWARE LIBRE

## y las otras categorías de software

X	A	O	P	O	I	R	A	T	E	I	P	O	R	P
P	A	S	D	F	G	H	J	K	L	Ñ	P	Z	X	C
R	Ú	H	X	E	S	O	U	R	C	E	P	D	L	Y
I	T	B	S	D	X	O	E	R	N	E	Z	C	P	U
V	G	J	L	C	H	V	F	D	N	M	E	S	R	I
A	Y	K	W	I	N	K	F	T	E	E	W	A	R	E
T	U	L	Z	R	C	L	T	T	W	W	E	Z	A	V
I	I	Ñ	A	F	N	O	Y	S	B	A	W	F	G	A
V	O	Z	Q	V	I	O	U	O	R	A	R	O	P	W
O	P	X	M	N	M	P	I	F	T	Y	C	E	L	Z
A	A	C	I	T	U	Ñ	O	T	G	H	J	R	K	C
Q	S	M	N	G	J	Q	P	W	N	M	A	A	D	A
W	O	V	B	B	M	W	A	A	Z	X	C	W	B	H
D	D	P	O	Q	W	E	R	R	T	Y	U	E	O	W
E	F	Ñ	I	L	I	B	R	E	G	H	J	E	L	Ñ
R	G	L	H	Z	X	C	V	B	N	M	A	R	D	F
Z	X	S	H	A	R	E	W	A	R	E	H	F	F	A

Francisco de Assis da Costa Silva

**FRANCISCO DE ASSIS DA COSTA SILVA**

# SOFTWARE LIBRE

## y las otras categorías de software

## Licencia de uso

Esta obra se distribuye bajo los términos de la Licencia Creative Commons Reconocimiento-CompartirIgual 4.0 Internacional (CC BY-SA 4.0).



Usted es libre de:

- Compartir – copiar y redistribuir la obra en cualquier medio o formato.
- Adaptar – remezclar, transformar y crear a partir de la obra para cualquier finalidad.

El licenciador no puede revocar estas libertades mientras cumpla con los términos de la licencia.

Bajo las condiciones siguientes:

- Reconocimiento: Debe reconocer adecuadamente la autoría de la obra.
- CompartirIgual: Debe distribuir la obra derivada bajo la misma licencia que el original.

Términos de la licencia: <https://creativecommons.org/licenses/by-sa/4.0/deed.es> ES.

Catálogo na fonte: Renata Santana CRB 15/689 - PB

S586i	Silva, Francisco de Assis da Costa
	Software Libre y las otras categorías de software [recurso eletrônico] / Francisco de Assis da Costa Silva – Patos - PB: Edição do autor, 2020.
	Dados eletrônicos (1 arquivo : 890 KB)
	Livro eletrônico Inclui bibliografia <b>ISBN:</b> 978-65-00-03175-1
	1. Software libre, 2. Software privativo, 3. TIC, 4. Software, I. Título
	CDD 004 CDU 004

Se agradece el apoyo brindado por la **Prof<sup>a</sup> Dr<sup>a</sup> Anna Escofet**, de la *Universitat de Barcelona*, y **Eduard Elias Vila**, decano fundador del *Col·legi Oficial d'Enginyeria en Informàtica de Catalunya*.

# || ÍNDICE

7	<b>1 INTRODUCCIÓN</b>
13	<b>2 LAS CATEGORÍAS DE SOFTWARE</b>
13	2.1 Introducción
15	2.2 El software libre
19	2.2.1 Las licencias de software libre
23	2.2.2 Los problemas relacionados con la expresión software libre
24	2.3 El <i>Open Source Software</i> (OSS) o software de código abierto
27	2.3.1 Los problemas relacionados con la expresión software de código abierto
28	2.3.2 Conjunción de términos software libre y software de código abierto
29	2.3.3 Los dos movimientos: el del software libre y el del software de código abierto
30	2.4 El software privativo o propietario
31	2.4.1 Las licencias en el software privativo
34	2.4.2 La <i>End User License Agreement</i> (EULA) o Contrato de Licencia de Usuario Final (CLUF)
36	2.5 El software <i>freeware</i>
37	2.6 El software <i>shareware</i>
38	2.7 El software de dominio público
40	2.8 Resumen del capítulo

<b>43</b>	<b>3 EL SOFTWARE LIBRE</b>
43	3.1 Introducción
46	3.2 La adhesión al software libre
49	3.3 Ventajas del uso del software libre
50	3.3.1 Reducción de Costes
51	3.3.2 Independencia del proveedor
57	3.3.3 Disminución de recursos de hardware
61	3.3.4 Interoperabilidad
64	3.3.5 Estabilidad y seguridad
65	3.3.6 Compartir y cooperar
66	3.3.7 Combatir la brecha digital
68	3.4 Desventajas del software libre
68	3.4.1 Cantidad de software disponible
69	3.4.2 Programas y documentación disponibles en pocos idiomas
70	3.4.3 Gastos en el proceso de migración
70	3.4.4 Desconocimiento sobre el software libre
72	3.4.5 Difícil instalación y configuración
73	3.4.6 Necesidad de conocimientos técnicos para hacer uso del código fuente
73	3.5 Resistencias a usar el software libre
74	3.5.1 Mayor aceptación del software privativo
74	3.5.2 Experiencia con el software privativo
75	3.5.3 <i>Fear, Uncertainty and Doubt</i> (FUD)
76	3.5.4 Otros intereses
77	3.6 Los falsos mitos relacionados con el software libre
78	3.6.1 El software libre es gratis
79	3.6.2 El software libre no es comercial
80	3.6.3 Sólo existe software libre para GNU/Linux
81	3.7 Algunos de los programas libres más conocidos
84	3.8 Resumen del capítulo
<b>87</b>	<b>Referencias</b>

# 1 | INTRODUCCIÓN

En la década de los 70 surgieron en los Estados Unidos de América los primeros microcomputadores, que representaron una verdadera revolución en la tecnología de los ordenadores y por consecuencia en el mundo de las Tecnologías de la Información y Comunicación (TIC).

En enero de 1983, la revista Time sustituyó su tradicional portada «*TIME's Person of the Year*» por «*TIME's Machine of the Year*»<sup>1</sup> como forma de reconocimiento al gran éxito del ordenador personal en el aquel momento. Por primera vez en su historia la portada no traía el título de «la persona del año», que es concedido a un consagrado personaje o en algunas ocasiones a un grupo de personas. Parecía ser un anuncio de que el ordenador estaría entrando en nuestro quehacer diario para siempre.

Desde entonces esas máquinas han influenciado nuestras vidas. Siempre que el intelecto humano y la tecnología se encuentran, el ordenador está presente (Norton, 1996). En la década de los 90, la aparición de Internet ha provocado otro gran impacto en el mundo de las TIC. Con eso, hubo todavía una mayor aceleración y diseminación del ordenador.

---

1 Más información en: <http://content.time.com/time/magazine/0,9263,7601830103,00.html>.

Estas máquinas están cada vez más presentes en lo cotidiano de las personas, sea en los bancos, en las oficinas, en las escuelas, en las universidades, en el comercio, en las industrias y hasta en los hogares. Su uso se ha vuelto indispensable en la sociedad contemporánea. Han influenciado significativamente el modo de vida de la ciudadanía y parece que somos cada día más dependientes.

Ante este cuadro, la revista Time eligió como la «*TIME's Person of the Year*»<sup>2</sup> de 2006 el usuario de Internet por el explosivo crecimiento e influencia de los contenidos de Internet generados por los internautas, como los *blogs* y el portal de vídeos compartidos *YouTube*. La leyenda de la portada decía: «*Yes, you. You control the Information Age. Welcome to your world*».

Prácticamente desde la década de los 90, somos testigos del surgimiento de numerosas tecnologías y hemos experimentado sus rápidos avances. Son elementos claves en una sociedad que se está informatizando día a día y ejercen un papel fundamental en la difusión de la información y del conocimiento. De hecho, en un mundo cada vez más digitalizado y donde la tecnología invade todos los ámbitos de la sociedad, las TIC ejercen un papel protagonista en la Sociedad de la Información. Vivimos en una sociedad cada vez más informatizada y globalizada. Las tecnologías forman parte de nuestro día a día, son una necesidad y resulta impensable permanecer ajeno a esa nueva coyuntura.

Es indudable que esas tecnologías, con tantas utilidades y funciones, desempeñan un papel importante para el futuro de cualquier país. Conocedores de esa situación, son justamente los países ricos los que más invierten en el desarrollo y en el consumo de esas nuevas tecnologías. Y, en consecuencia, son los que tienen más beneficios. Las TIC pasaran a ser símbolo de progreso y riqueza y, por qué no decir, de poder.

Se considera el ordenador e Internet como dos exponentes fundamentales en el proceso de consolidación de las TIC en la sociedad. Su uso ya es una realidad, es un proceso irreversible y del que no se puede prescindir. No es casualidad que se difunda cada vez más. La diseminación

---

2 Más información en: <https://time.com/4586842/person-of-the-year-2006-2016/>.



del uso de estas dos tecnologías en las más diversas áreas ha provocado profundos cambios de comportamiento, hábitos y costumbres de la ciudadanía.

Por detrás del ordenador e Internet, o quizás fuese mejor decir en paralelo con éstos, está el software. Éste es un elemento que a menudo pasa desapercibido o no se le da su valor real, pero juega un trascendental e imprescindible papel en todo el proceso de penetración y estabilización de las TIC en todas las áreas.

Con el rápido crecimiento del mercado de consumo de ordenadores y la expansión del acceso a Internet, el software se ha convertido en un producto de extremo interés en la economía globalizada, pues se ha convertido en una nueva fuente de riqueza. De acuerdo con Gartner<sup>3</sup>, una de las más importantes empresas de consultoría y de investigación de las TIC, solamente en uno de los más grandes mercados de software, el de Gestión de Relaciones con Clientes, *Customer Relationship Management* (CRM), hubo un aumento de 15,6% en 2018, moviendo la impresionante cifra de US\$ 48,2 mil millones.

Por otra parte, el mercado de software de manera general, desgraciadamente, es monopolio de grandes empresas del sector informático. Las mayores empresas de software del mundo tienen su sede en países desarrollados y son, en su mayoría, empresas estadounidenses. Eso ha creado una permanente dependencia tecnológica en algunos países, principalmente en los menos desarrollados y más pobres, con todas las consecuencias que eso conlleva.

Cuando adquirimos un ordenador, en general, suele venir con una serie de programas instalados, sobre todo el sistema operativo, cuyo importe ya está incluido en el precio. La gran mayoría de los compradores desconocen sus derechos e, incluso, que hay otras posibilidades en relación con el sistema y las aplicaciones que se pueden instalar en su computador, incluso algunos gratuitos, lo que disminuiría el coste. Los gastos de software han llegado a representar la mayor parte del dispendio en los productos tecnológicos.

---

3 Más información en: <https://www.gartner.com/en/information-technology/research>.

Si, en el pasado, era el precio del ordenador lo que dificultaba la masificación de las TIC y provocaba el aumento de los excluidos digitales, actualmente, se ha añadido también el coste del software, que suele ser uno de los grandes responsables de ese proceso de exclusión.

Frente a esa realidad, el ciudadano, la propia sociedad y los países se han visto obligados a buscar otras formas de incorporar las TIC a su mundo. Los que no pueden desarrollar programas por sí mismos y no tienen capacidad económica suficiente para adquirirlos, disponen, principalmente, de dos formas de escapar al mandato del oligopolio de las grandes empresas de software y conseguir la inclusión tecnológica. Una de las formas encontradas es usar el software libre, que es totalmente legal y ético. Otra, incluso más común, es, desafortunadamente, la utilización de la llamada piratería de software, que es ilegal y es un fenómeno mundial, que afecta principalmente los países más pobres, no excluyendo su presencia en los países de las regiones más ricas del planeta.

El término piratería es muy utilizado para referirse al uso de productos no originales o auténticos, incluido el software. Pero aceptando la sugerencia de Stallman (2004), que piensa que este término es muy fuerte y más asociado a secuestrar y asesinar, utilizaremos la expresión más neutral de «copia no autorizada» a lo largo de este libro en sustitución del término piratería, para caracterizar la acción de obtener o comprar copias ilegales de software.

La organización americana *Business Software Alliance* (BSA)<sup>4</sup> viene analizando, durante casi dos décadas, las tendencias globales del uso de estas copias de software en los ordenadores. Según el último estudio publicado por la referida organización, el 37% de todo el software instalado en los ordenadores alrededor del mundo en 2017 es de copias no autorizadas de software. Esa cifra disminuyó dos puntos en comparación con la de 2015, que fue de 39%. Mientras que en Estados Unidos la cifra es sólo del 15%, en Libia, esa cifra es del 90%. Aún según el estudio, las empresas de software tienen pérdidas estimadas en US\$ 359.000 millones al año por el uso de software ilegal (Business Software Alliance, 2018).

---

4 Más información en: <https://www.bsa.org/>.

De hecho, si ha generalizado el discurso de que el uso de copias no autorizadas de software provoca enormes perjuicios a las grandes empresas de software. Está tan extendido que se ha internalizado por la sociedad en general, pero es poco cuestionado e investigado en su origen y sus efectos (Feres, Oliveira y Gonçalves, 2017).

Además, es justamente en ese sentido que el uso de copias no autorizadas es perjudicial, por ejemplo, para el software libre. En la medida que el usuario en general opta por usar las copias ilegales está dejando de considerarlo como una alternativa al software no libre. Al ignorarlo, no se divulga, tampoco hay conciencia de haber desaprovechado una oportunidad. Eso favorece al software no libre, particularmente el privativo, que pasa a ser considerado como la única opción, aunque sea conseguido de forma ilícita.

Frente al lo expuesto, es evidente que el software es un recurso clave en el mundo de las TIC. A pesar de su discreción, es recurrente, cada vez más, en el flujo principal del debate alrededor de las tecnologías, especialmente por el uso de copias no autorizadas de software, por las facilidades de descargar y comprar programas en Internet o por las discusiones en torno a las distintas categorías de software.

En el mundo del software existen diversas categorías con características muy distintas. Durante muchos años, o incluso hasta hoy día, para una gran parte de los consumidores de tecnología es como si eso no existiera. Para éstos, dicho mundo se reduce a los programas que vienen preinstalados en sus ordenadores. Pero hay mucho más de lo que la gente conoce.

Dentro del particular y tan amplio mundo de las TIC y del software, más concretamente entre las categorías de software, una que gana cada vez más espacio y atención en las más diversas áreas es la del software libre. La comprensión de sus principios y tendencias permitirá una comparación con las otras categorías de software.

El software libre es una temática habitual en el mundo de la Informática y cada vez más ha despertado también el interés de usuarios de

otras áreas. Esto se debe, principalmente, a las diversas ventajas que ofrece cuando se compara con otras categorías. No es casualidad que sean cada vez más frecuentes discusiones sobre su uso y relevancia en diversos sectores de la sociedad, en especial en la administración pública y en el área educativa.

Sin embargo, para un gran número de usuarios, este asunto aún no es conocido, no está suficientemente claro o incluso no despierta interés. Esto es consecuencia, sobre todo, del amplio dominio de otras categorías de software en los más diversos sectores de la sociedad, en particular la de software privativo.

A grandes rasgos, el software no libre, representado en su mayoría por el privativo, es el más conocido y utilizado en las más diversas áreas, aunque otras categorías ofrezcan más ventajas, incluso económicas, y tengan una filosofía totalmente contraria.

Hay muchos intereses en juego y también muchas militancias y pasiones. Es importante aclarar algunos puntos y poner de relieve la situación real y contribución del software libre en el selecto y especial mundo de las TIC.

# 2 | LAS CATEGORÍAS DE SOFTWARE

## 2.1 Introducción

Existen varias maneras de clasificar el software. Una de las más usadas se basa en distinguirlo según la forma de su distribución o más concretamente cómo se licencia al usuario. A lo largo de este libro, trataremos de una manera general el usuario de software indistintamente de que sea una persona física o jurídica. En este último caso se referirá a corporaciones, asociaciones, sociedades y fundaciones. Cuando sea necesario hacer distinción entre el usuario persona física, del de la persona jurídica, esto estará explicitado en el texto.

La licencia es un documento legal que acompaña el software y contiene sus normas de uso. Estas normas delimitan lo que el usuario puede hacer con el programa y lo que no. O sea, cuáles son sus derechos, límites y libertades.

Una licencia de uso del software puede ser concedida por: el autor o autores originales del software, la persona física o jurídica que sea titular de los derechos de explotación o la que tenga el derecho de distribuir el software. Delante de esa diversidad, denominemos proveedor del software, el sujeto que puede conceder la licencia de software (Bain *et al.*, 2007a). Aceptando esta sugerencia, a lo largo de este libro utilizaremos la expresión «proveedor del software» para quien otorga la licencia de este producto.

La gran mayoría de usuarios de ordenadores, de una manera general, no se preocupa de la licencia de software y termina muchas veces actuando ilegalmente por desconocimiento, desatención o pereza en el cumplimiento del contrato de licencia, ya que no suele leerla cuando adquiere el software. Al instalarlo, copiarlo, prestarlo, redistribuirlo o ejecutar cualquier otra acción sobre éste es primordial conocer la licencia bajo la que ha sido distribuido.

El usuario es libre para decidir si acepta las condiciones impuestas por una determinada licencia. Una vez aceptada, aunque es normal que no haya firmado nada, está obligado a cumplir todas las cláusulas del contrato, si no, estará infringiendo la ley.

Por otro lado, también es verdad que, para el gran público, resulta muchas veces difícil o confusa la interpretación de las licencias, en especial cuando éstas no están en el idioma nativo del usuario. Otra dificultad es la terminología que se emplea, que normalmente es propia del mundo jurídico, lo que resulta pesado y no muy agradable para leer.

En este trabajo, presentamos e introducimos algunas categorías de software, que son las más utilizadas en su distribución y que frecuentemente son mencionadas en debates sobre el tema. La discusión tiene como objetivo hacer un repaso de sus aspectos conceptuales y principales características y, de este modo, aclarar diferencias y similitudes entre éstas, con la finalidad, únicamente, de delimitar el núcleo central del libro: el software libre.

Además de ésta categoría, expondremos la del *Open Source Software* (OSS) o de código abierto, la del software privativo o propietario, la

del *freeware* y la del *shareware*. También comentaremos una categoría especial: la de dominio público.

## 2.2 El software libre

La expresión software libre ha ganado notoriedad con el americano Richard Matthew Stallman, el iniciador e ideólogo del movimiento del software libre, creador de la *Free Software Foundation* (FSF)<sup>5</sup> y reconocido divulgador de la cultura de este software. La Fundación, que es una organización sin fines lucrativos, ha sido creada precisamente con el objetivo de divulgar este movimiento alrededor del mundo.

Son diversas las definiciones de software libre, pero prácticamente todas derivan o son similares en contenido con la concepción presentada por Stallman. Su definición es una referencia sobre el tema y son diversos los autores que la han citado en sus artículos o libros. En concreto, en las palabras de Stallman (2004),

Con software libre nos referimos a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software. Nos referimos especialmente a cuatro clases de libertad para los usuarios de software:

- Libertad 0: la libertad para ejecutar el programa sea cual sea nuestro propósito.
- Libertad 1: la libertad para estudiar el funcionamiento del programa y adaptarlo a tus necesidades - el acceso al código fuente es condición indispensable para esto.
- Libertad 2: la libertad para redistribuir copias y ayudar así a tu vecino.
- Libertad 3: la libertad para mejorar el programa y luego

---

<sup>5</sup> Más información en: <http://www.fsf.org/>.

publicarlo para el bien de toda la comunidad - el acceso al código fuente es condición indispensable para esto. Software libre es cualquier programa cuyos usuarios gocen de estas libertades. (p. 59).

Tal como fue concebido en su definición, un software es libre si cumple esta suerte de mandamientos o reglas establecidas, llamadas las cuatro libertades que determinan cómo podemos utilizarlo. Para que estas libertades sean reales, deben ser irrevocables (Stallman, 2004). Estas cuatro reglas son la base para referirse al software libre y por esto presentamos, a continuación, una breve discusión acerca de cada una de ellas.

Se trata, entonces, de un tipo especial de software que se caracteriza por su forma distinta de desarrollo y distribución. Consideramos también este concepto como base para nuestro trabajo.

- **Libertad 0: la libertad para ejecutar el programa sea cual sea nuestro propósito.**

El ejercicio de esta libertad consiste en que, conseguida una copia del programa libre, “cualquier individuo u organización podrán ejecutarlo desde cualquier sistema informático, con cualquier fin y sin la obligación de comunicárselo subsiguientemente ni al desarrollador ni a ninguna entidad en concreto” (Stallman, 2004, p. 60).

Cualquier usuario, una vez que haya adquirido una copia de un programa libre, sea pagando o gratuitamente, tiene la libertad de poder instalarlo y utilizarlo ilimitadamente en cualquier ordenador de cualquier lugar, región o país, para cualquier propósito o finalidad, sea personal, comercial, educativo, de investigación, social, político e incluso militar. No hay restricciones, solamente la total libertad de usarlo.



- **Libertad 1: la libertad para estudiar el funcionamiento del programa y adaptarlo a tus necesidades - el acceso al código fuente es condición indispensable para esto.**

Para concretar esta libertad, en primer lugar, es imprescindible que el código fuente del programa que se distribuye esté disponible. Eso es un requisito esencial para ejercer ésta y la libertad 3. Si no es así, el software no es libre.

Esta libertad determina que cualquier usuario es “libre para introducir modificaciones y utilizarlas de forma privada, (...), sin siquiera tener que mencionar su existencia” (Stallman, 2004, p. 60). De este modo, independientemente de cómo haya obtenido el programa, sea pagando o gratuitamente, tiene la libertad de estudiar y analizar su funcionamiento, modificarlo, mejorarlo y adaptarlo a sus propias necesidades.

Claro que para poder ejecutar esas acciones es necesario tener algún conocimiento técnico. Si no, la solución es acudir a un técnico o profesional del área para que realice las modificaciones.

Lo trascendente de esta libertad es que se puede adaptar el programa a las necesidades del usuario, algo prácticamente imposible en otras categorías de software.

- **Libertad 2: la libertad para redistribuir copias y ayudar así a tu vecino.**

Independientemente de cómo haya conseguido la copia del programa libre, cualquier usuario tiene la libertad “de redistribuir copias con o sin modificaciones, de forma gratuita o cobrando por su distribución, a cualquiera y en cualquier lugar” (Stallman, 2004, p. 60).

Cualquier usuario puede hacer copia de un programa libre y prestarla, regalarla o venderla a quien demuestre interés en obtenerla. Esto es un proceso muy sencillo, hoy día, dado que la tecnología necesaria para efectuar esa acción es muy simple y suele estar disponible en los ordenadores o se puede usar Internet. Por eso, es cada vez más fácil obtener una copia, sea pagando o gratuitamente. En los casos en que se paga, normalmente el valor cobrado corresponde sólo a los costes de distribución, o sea, son valores muy bajos que prácticamente cualquier persona puede pagar.

Con esta libertad no se viola ninguna norma que permita emprender alguna acción legal por el acto de copiar el software e incluso redistribuirlo, independientemente de la forma como lo haya conseguido. Al contrario, se debe copiar y, sobre todo, distribuir libremente de modo que todos puedan beneficiarse. Vivimos en sociedad y nada es más natural que compartir y cooperar. No pasa lo mismo cuando se intenta hacer esto con programas de algunas otras categorías de software.

Sobre este tema, la propia Free Software Foundation (2020b) aclara que “si usted redistribuye copias de software libre, puede poner un precio y *ganar algo de dinero*. Redistribuir software libre es una actividad buena y legítima. Si lo hace, es sensato que obtenga un beneficio”.

- **Libertad 3: la libertad para mejorar el programa y luego publicarlo para el bien de toda la comunidad - el acceso al código fuente es condición indispensable para esto.**

Así como para el ejercicio de la libertad 1, también en ésta, en primer lugar, es necesario que el código fuente del programa que se distribuye esté disponible para hacer efectiva la libertad. El acceso al código fuente permite que cualquier usuario pueda hacer uso del derecho de realizar modificaciones y mejoras en el programa para atender a sus intereses y necesidades. De acuerdo con Stallman (2004, p. 60), “si decidieras publicar

estos cambios, no deberías estar obligado a notificarlo de ninguna forma ni a nadie en particular”.

De la misma manera, también es libre para distribuir la versión del programa modificado o mejorado a quien interese, sea quien sea, para beneficiarse de los cambios. No está obligado, pero si así lo desea, sólo estará ejerciendo su libertad. Para poder realizar estas acciones, es necesario, obviamente, tener algún conocimiento técnico o, entonces, pagar para que las hagan.

Así, en resumen, un programa, para ser considerado libre, debe ser distribuido de tal modo que el usuario tenga la libertad de instalarlo, utilizarlo, copiarlo, estudiarlo, modificarlo, adaptarlo, perfeccionarlo y redistribuirlo, con o sin modificaciones, cobrando o gratuitamente. También es requisito obligatorio que el código fuente del programa esté disponible.

Ser libre para hacer todo eso significa, entre otras cosas, que no tenemos que pedir permiso, ni tampoco pagar para ejecutar el programa, sea cual sea el propósito (Stallman, 2004). Y, además, siempre es legal. Es decir, es como si el programa no tuviese dueño. Sin embargo, no significa que no pertenece a nadie o a ningún proveedor, o que sea propiedad de todos. El énfasis está en la libertad y no en la propiedad o en el precio. La palabra clave para referirse al software libre es, entonces, libertad.

### 2.2.1 Las licencias de software libre

La manera usual de distribución del software libre es bajo una determinada licencia libre. Al contrario de otros tipos de licencias que son conocidas por imponer muchas restricciones a los usuarios, las libres se caracterizan por conceder derechos o libertades a éstos, tales como la copia, la modificación y/o la redistribución.

Existen diversos tipos de licencias libres. Cada una presenta algunas diferencias con respecto a sus características. Las hay que hacen exigencias más sencillas, otras presentan diferencias de matices en algunos puntos y algunas son mucho más exigentes. Pero todas garantizan las cuatro libertades anteriormente discutidas.

El uso de una u otra licencia tiene consecuencias y es por eso que siempre es fundamental conocer y entender bien la licencia bajo la que el software ha sido distribuido o que se pretende utilizar para distribuirlo. Esta observación es pertinente tanto para el proveedor del software como para el usuario, de una manera general.

Al distribuir el programa bajo una licencia libre, el autor del mismo no está renunciando a sus derechos, sino que está asegurando que sus objetivos con la distribución sean respetados. Así está defendiendo y protegiendo los derechos del usuario.

La licencia libre considerada más completa, y que también es la más popular y utilizada, es la GNU *General Public License* (Licencia Pública General) o GNU/GPL<sup>6</sup>, más popularmente conocida como GPL. Fue creada por la FSF a mediados de los 80.

La GNU/GPL ha sido formulada con el propósito de ofrecer los medios legales para imponer que toda distribución de software libre se haga preservando y respetando las cuatro libertades presentadas por Stallman. También asegurar que el software, una vez libre, siempre será libre, incluso sus versiones subsiguientes y derivaciones. Se considera una licencia muy restrictiva y particular, especialmente porque usa una condición especial para proteger la redistribución de software, conocida como *copyleft*.

Este término ha sido utilizado por primera vez por Stallman (Stallman, 2002). En realidad, es un juego de palabras utilizado en contraposición al conocido *copyright*. En inglés, *left* significa, a la vez, izquierdo y permitido. Izquierdo se opone a derecho (*right*, tanto en el sentido de lateralidad, como en el jurídico) y permitido se opone a las restricciones

---

6 Más información en: <https://www.gnu.org/licenses/gpl-3.0.html>.

que supone el *copyright reserved*. En realidad, se trata de un método utilizado para convertir un programa en software libre y exigir que todas sus versiones, modificadas o ampliadas, también lo sean (Stallman, 2004). El objetivo es preservar, sobre todo, las libertades de los usuarios y no proteger los derechos del autor. Es usar las leyes del sistema para resguardarse del propio sistema.

El *copyleft* es, entonces, un instrumento jurídico que protege los derechos de los usuarios de software libre contra el intento de que cualquier otro agregue alguna restricción adicional al software libre redistribuido, sea con o sin modificación. También garantiza que las cuatro libertades sean observadas, garantizadas y ejercidas libremente, además del acceso al código fuente del programa.

Así mismo, es utilizado para exigir que cualquier copia o programa derivado del software distribuido bajo la licencia GPL obligatoriamente deba ser redistribuido también bajo la misma licencia, garantizando así que éste también sea libre. Por esta razón, las licencias con *copyleft* son llamadas «virales», porque contagian sus características a todos los programas derivados, o, incluso, con los que se integren.

Esto impide que cualquiera pueda aprovecharse del acceso al código fuente del programa, para mejorarlo y transformarlo en un software no libre. De esta forma, evitará el riesgo de tener que competir con una versión modificada privativa de su propio trabajo (Free Software Foundation, 2020a). En cambio, el software libre no protegido con *copyleft* puede transformarse en programa no libre ya que es permitido añadirle restricciones.

Utilizaremos la expresión «software no libre» a lo largo de este trabajo para caracterizar cualquier software que no tenga las características de software libre. Es decir, que no respete las cuatro libertades fundamentales. Este término es, incluso, recomendado por la FSF, que lo usa como sinónimo de software privativo.

El recurso de *copyleft* ha sido un gran legado de la FSF al mundo del software libre y por qué no decir a toda la sociedad. “No obstante, no todo el software libre se distribuye con *copyleft*, que es lo que distingue el software libre de la FSF del resto de software libre distribuido bajo otras licencias conocido como software abierto” (Bain *et al.*, 2007a, p. 27).

La GPL es sin ninguna duda una de las licencias más importantes dentro del mundo del software libre y es por eso que goza de amplia popularidad. Tanto es así que muchas veces hay el malentendido de que software libre significa exactamente lo mismo que todo software distribuido bajo la GPL, lo que no es verdad. La última versión de la GPL, la versión 3 (GPLv3), fue lanzada en 2007, después de años de discusión con usuarios y profesionales de los más diversos sectores de la sociedad. En esta versión se ha intentado resolver algunos problemas especialmente relacionados con las patentes de software.

Otras licencias también muy utilizadas dentro del ámbito del software libre son la licencia GNU *Lesser General Public License*<sup>7</sup> (Licencia Pública General Reducida) o GNU/LGPL, también creada por la FSF, y la licencia *Berkeley Software Distribution*<sup>8</sup> (BSD) elaborada por la Universidad de Berkeley en los Estados Unidos de América. Hay otros muchos tipos de licencias, pero las citadas son sin duda las más populares. Para ampliar más la información sobre otras licencias recomendamos la página del Proyecto GNU<sup>9</sup>.

Las licencias libres son, en resumen, el mecanismo legal que se utiliza para garantizar al usuario de software, por lo menos, las cuatro libertades propuestas por Stallman, impidiendo que, de alguna forma, se puedan hacer restricciones a éstas. No se aplican solamente al proceso de distribución y uso del software libre. Hay también las licencias libres para la documentación técnica del software, que tienen por objetivo regular la utilización de su documentación. Normalmente, toda la documentación del software es considerada parte del software y, por consiguiente, también está

7 Más información en: <http://www.gnu.org/licenses/lgpl-3.0.html>.

8 Más información en: <https://opensource.org/licenses/BSD-3-Clause>.

9 Más información en: <https://www.gnu.org/licenses/license-list.es.html>.

distribuida bajo la licencia del software. En particular, la regulación europea sobre derechos de autor así lo establece. Pero no todos los programas vienen con alguna documentación. La más popular de estas licencias es la *GNU Free Documentation License*<sup>10</sup> (Licencia de Documentación Libre) o GNU/FDL, creada también por la FSF.

El acelerado avance tecnológico y los intereses cada vez más crecientes en la industria de software, han generado una proliferación de licencias, incluso libres. Eso termina haciendo difícil o complicado elegir una de ellas para el uso o la distribución del software libre. Por esta razón, en los casos de dudas es común recurrir a expertos y tener en cuenta su opinión para ayudar a decidir. De todas formas, la licencia GNU/GPL nos parece la más adecuada para distribuir software libre.

### 2.2.2 Los problemas relacionados con la expresión software libre

La expresión software libre ha provocado muchas confusiones porque deriva del inglés *free software*. La palabra inglesa *free* puede significar libre o gratis y estos significados no son equivalentes. Así, ha producido muchas interpretaciones sobre lo que realmente significa. Según el propio Stallman (2004) y la Free Software Foundation (2019a), el problema de la ambigüedad de la expresión *free software* en inglés es evidente y conlleva un significado indeseado. Este problema no ocurre en lenguas latinas como, por ejemplo, el francés que tiene términos distintos para «libre» y «gratuit». En español, se usa «libre» y «gratuito» y en portugués, «livre» y «grátis». Tanto el propio Stallman como otros han intentado conseguir un término más adecuado y suficientemente claro para evitar ese problema, pero no lo han conseguido.

A lo largo de los años hemos escuchado cada vez más la asociación de software libre con el *open source software*, también conocido como software de código abierto o de fuente abierta, para referirse al mismo software. En

---

10 Más información en: <https://www.gnu.org/licenses/fdl-1.3.html>.

realidad, se trata de dos categorías de software que presentan muchas similitudes, pero también matices y por eso confunde tanto a los usuarios.

### 2.3 El *Open Source Software* (OSS) o software de código abierto

La expresión *Open Source Software* es generalmente conocida y traducida al castellano como software de código abierto, pero también usada como software de fuente abierta o software de código fuente abierto.

Esta categoría de software muchas veces es confundida, citada o empleada indistintamente para referirse a software libre. En otros casos también se utilizan a menudo los dos términos como equivalentes, sinónimos o intercambiables, como si fuesen una sola categoría de software. Esto ocurre porque sus definiciones son muy similares y comparten prácticamente los mismos objetivos. Es verdad que hay muchos aspectos en común, pero no son idénticos, hay también matices.

El software de código abierto es “para algunos, una modalidad del software libre, para otros, el término general que engloba a todo el software libre, y para otros, una peligrosa desviación de los conceptos originales del software libre para conseguir su mayor comercialización” (Bain *et al.*, 2007a, p. 28).

El proyecto del software de código abierto surgió en febrero de 1998 en Palo Alto, en el Valle del Silicio, California, Estados Unidos de América, cuando un grupo formado por integrantes disidentes del movimiento del software libre, entre ellos Eric S. Raymond<sup>11</sup> y Bruce Perens<sup>12</sup>, decidieron crear la iniciativa Open Source Initiative (OSI)<sup>13</sup> y proponer el uso de la expresión «*Open Source Software*» como alternativa al software libre. Mientras que la FSF ha sido creada con el objetivo principal de difundir el movimiento del software libre, la OSI lo ha sido con el objetivo de

11 Líder filosófico del movimiento *Open Source*.

12 Autor de la definición del concepto de la *Open Source Definition* (OSD).

13 Más información en: <http://www.opensource.org/>.



definir el software de código abierto y de crear y difundir el movimiento *open source*.

La idea del nuevo movimiento era atraer a las empresas, sobre todo de la industria de la alta tecnología, al mundo del software libre, dejando de lado todo el idealismo defendido por sus partidarios, para centrarse en cuestiones prácticas y relevantes orientadas a este sector. En realidad, las ideas fundacionales de este movimiento, de alguna forma, les parecían una postura radical y no eran ampliamente entendidas entre los empresarios.

Para esto, el énfasis de los beneficios de la disponibilidad del código fuente del software no estaba en las libertades ofrecidas al usuario, sino en la posibilidad de desarrollar software tanto de calidad superior y más eficiente, como a una velocidad mayor que en el modelo tradicional, centrando la cuestión exclusivamente en las ventajas técnicas para las empresas.

De una manera general, el software de código abierto es el software en que, normalmente, igual como en el libre, el usuario tiene la libertad de usarlo para cualquier propósito, estudiarlo, modificarlo y redistribuirlo libremente, con o sin modificaciones. También se distribuye con el código fuente. Tanto en el caso del libre como en el del código abierto hay un principio de inclusión, es decir, todos los usuarios se benefician del software, al contrario que la mayoría del no libre que se caracteriza por la exclusión.

Un software para ser considerado libre debe respetar las cuatro libertades fundamentales del software libre, mientras que para ser considerado como software de código abierto debe cumplir un decálogo de condiciones creado por la OSI. Este decálogo es lo que sirve de base para el concepto de software de código de abierto.

Los términos de los criterios establecidos en el decálogo en la *Open Source Definition* (OSD)<sup>14</sup> son los siguientes:

<sup>14</sup> La versión original en inglés se puede consultar directamente de la página principal de la OSI: <https://opensource.org/docs/osd>.

- 1- Redistribución libre de los productos (con o sin coste adicional).
- 2- Acceso al código fuente.
- 3- Trabajos derivados del original o modificaciones de éste pueden ser redistribuidos bajo los mismos términos.
- 4- Integridad del código fuente del autor: algunas licencias pueden restringir la difusión del código a parches (*patch files*).
- 5- No discriminación de personas o colectivos: la licencia no puede excluir a nadie.
- 6- No discriminación de áreas de iniciativa: se respeta la posibilidad de una distribución comercial del software.
- 7- Distribución de la licencia: los derechos vinculados al programa aplican a todo aquel que lo reciba.
- 8- La licencia no debe ser específica de un producto: aunque una parte del programa se distribuyera separadamente, los derechos asignados son los mismos que los del conjunto original.
- 9- La licencia no debe restringir otro software: el acceso al código no es vinculante a otros productos de software que se deriven del original.
- 10- La licencia debe ser tecnológicamente neutral: no debe requerirse la aceptación de la licencia por medio de un acceso o interficie específica. (Roca y Castells, 2006, p. 30).

Tanto Mas (2005) como Stallman (2010) consideran que estas condiciones son plenamente aplicables a cualquier software libre. En el caso de Stallman (2010), todavía comenta que la definición oficial de software de código abierto publicada por la OSI se derivó indirectamente de su criterio para el software libre. No es la misma, pero en la práctica está bastante cerca, sin embargo, es un poco más imprecisa en algunos aspectos.

No hay duda de que el software libre y el de código abierto son filosóficamente distintos, pero que presentan definiciones y principios muy semejantes. Como hemos visto, las diez reglas de la OSD son prácticamente equivalentes con las cuatro libertades que caracterizan el software libre y es

por esto que, normalmente, sus conceptos se confunden o se usan de forma indistinta.

En realidad, “La OSD ha sido diseñada para establecer una declaración abierta y comprensiva de los principios del movimiento de software abierto y una manera de clasificar y “certificar” la multitud de licencias libres que existen” (Bain *et al.*, 2007a, p. 20). Estos autores también resaltan que “es importante tener en cuenta que la OSD no es una licencia, ni un modelo de licencia, sino diez directrices para la clasificación de licencias relativas a aplicaciones y productos de software en sus diversas formas (componentes, programas, distribuciones completas)” (Bain *et al.*, 2007a, p. 29).

### 2.3.1 Los problemas relacionados con la expresión software de código abierto

La idea de establecer el uso de la expresión *open source* tenía como objetivos, “por un lado, diferenciarse del término *free software*, generalmente relacionado con el software de *copyleft* de la FSF, y, por otra parte, evitar el uso de la palabra *free*, para no tener connotaciones de gratuidad” (Bain *et al.*, 2007a, p. 28). Todo eso debido al problema de significados de la palabra *free* en el mundo anglófono, conforme discutimos anteriormente.

La motivación era, entonces, que además de crear un nombre para referirse al nuevo movimiento, éste también sirviera para intentar evitar la confusión con el término inglés *free* en *free software*. Al interpretar la expresión *open source*, entendemos que está relacionada directamente con el hecho de que podemos acceder al código fuente, lo que resulta realmente ambiguo ya que tanto el software libre como incluso algunos programas no libres también lo permiten. Esta nueva opción parece que no ha sido totalmente un éxito ya que la expresión también conlleva ambigüedades.

Los objetivos del nuevo movimiento todavía no se han cumplido pues la expresión *free software* sigue más utilizada y todavía hay mucha confusión con la libertad y la gratuidad del software libre (Bain *et al.*, 2007a). La nueva expresión al revés de aclarar el malentendido con la de «software libre», en realidad se vinculó a visiones filosóficas distintas a las del movimiento del software libre (Stallman, 2010).

No hemos encontrado datos sobre el uso de ambas expresiones, pero, basados en lecturas, Internet, charlas con usuarios y expertos en el tema, pensamos que en el mundo empresarial la opción por la expresión «software de código abierto» es predominante mientras que en el mundo de los usuarios está más extendido el uso de «software libre».

### 2.3.2 Conjunción de términos software libre y software de código abierto

Por presentar tantas similitudes, a lo largo de los años, es cada vez más común mezclar las dos expresiones y referirse a ambos tipos de software con nuevas formulaciones o acrónimos para no centrar la atención en las cuestiones filosóficas que los diferencian.

Encontramos en la literatura referencias a acrónimos como:

- FLOSS, para referirse a *Free/Libre/Open Source Software*. Éste es justamente el acrónimo más utilizado.
- FOSS - *Free and Open Source Software*.
- FS/OS - *Free Software/Open Source*.
- F/OSS - *Free/Open Source Software*.
- OSS/FS - *Open Source Software/Free Software*.
- FOSSC - *Free/Open Source Software/Code*.

- Otra variante: «libre software».

### 2.3.3 Los dos movimientos: el del software libre y el del software de código abierto

El mundo del software libre ha quedado, entonces, dividido filosóficamente, sin embargo, con los mismos propósitos. En la visión de Stallman (2004, p. 76), “el movimiento del software libre y el movimiento *open source* son como dos campos políticos dentro de la comunidad del software libre”. Dentro de esta perspectiva, los dos representan movimientos separados que se diferencian en la filosofía y en la política (Branco, 2005).

En el inicio, el movimiento *open source* ha sido duramente criticado tanto por Stallman como por miembros de la FSF. Con el pasar de los años, su consolidación, y quizás también por la actitud diplomática de sus miembros, que siempre han aclarado y comentado la importancia, influencia y herencia de los principios del software libre en el *open source*, poco a poco Stallman y la FSF han ido alterando el tono de las críticas. Efectivamente, está claro que la base del movimiento *open source* está centrada en la del software libre.

De hecho, La FSF ha ido cambiando gradualmente su postura tan agresiva contra la OSI como forma de reconocimiento de que “la incorporación de empresas grandes como IBM o Sun, bajo la perspectiva de los fundadores de la OSI, favorecerá y aún fortalecerá el movimiento del software libre frente al software propietario y acelerará su difusión y adopción” (Bain *et al.*, 2007a, p. 31). El propio Stallman (2010) ha declarado que el enemigo es el software no libre, especialmente el privativo, y no el movimiento del código abierto. Pero resalta que tampoco es partidario de este movimiento, una vez que el software libre defiende, sobre todo, la libertad.

Aunque esté claro que hay diferencias entre ambos movimientos, cabe destacar que Stallman (2010) todavía remarca que hay profesionales de los dos movimientos que trabajan juntos en los mismos proyectos. Eso

evidencia que de algún modo los dos movimientos se complementan, una vez que hay colaboración entre sus profesionales.

Después de analizar los dos movimientos, está claro que la expresión «software libre» resulta más interesante para nuestros propósitos, justamente porque aboga por el sentimiento de libertad que es tan importante en el mundo actual.

Es primordial aclarar que al utilizar la expresión «software libre», a lo largo de todo este libro, estaremos refiriéndonos tanto al software propiamente libre como al de código abierto que, independientemente de la forma en que ha sido licenciado, garantice al usuario el derecho de disfrutar de las cuatro libertades fundamentales del software libre. Por lo tanto, a partir de este punto usaremos la expresión «software libre» de forma genérica, sin centrarnos en las diferencias entre los dos movimientos.

## 2.4 El software privativo o propietario

La gran mayoría de los autores utiliza el término privativo o propietario para caracterizar el tipo de software indiscutiblemente predominante y conocido en todos los sectores de la sociedad.

Además de software privativo y propietario, algunas veces también es referenciado como software no libre, no abierto, privado, cerrado y comercial. Sin embargo, hay otras categorías de software que también son no libres, no abiertas, cerradas o comerciales.

Con respecto a privado, la Free Software Foundation (2019a) comenta que se trata de otro tipo de software, también conocido como software a medida, y que generalmente se desarrolla para una organización o compañía, que lo conserva en su poder, lo utiliza y no lo libera al público de ninguna forma.

No hay un consenso sobre cuál sería la mejor expresión para caracterizar el software privativo. Por lo visto también ocurren problemas relacionados con la ambigüedad. No deseando extenderse en la discusión acerca del término más apropiado, decidimos acoger el término «software privativo» y lo usaremos a lo largo de este libro. Entendemos que refleja mejor lo que pasa cuándo se usa esta clase de software: el proveedor del programa suele prohibir al usuario copiarlo y redistribuirlo, además de no permitir el acceso a su código fuente. Así, es prácticamente imposible realizar cualquier tipo de modificación en el software.

Además de entender que este término es el ideal para nuestro propósito, también es el utilizado por la FSF en su página web, en versión castellana, en la sección de las categorías de software libre y no libre.

La Free Software Foundation (2019a) considera que software privativo es el software en que su uso, redistribución o modificación están prohibidos de alguna forma, al punto de necesitar requerir algún tipo de autorización, o está tan restringido que de hecho no se puede hacer libremente. En contraposición al modelo del software libre, es conocido por privar a los usuarios de alguna o todas las libertades fundamentales comunes a esta categoría de software, además de negarle el acceso al código fuente del programa.

El software privativo presenta como característica principal el hecho de que, tras pagar por la licencia de uso, el usuario sólo puede utilizar el programa bajo determinadas condiciones fijadas por el proveedor. Éste mantiene plenos derechos absolutos sobre su software.

### 2.4.1 Las licencias en el software privativo

El software privativo está protegido, dependiendo del país, por la ley de derechos de autor o la de copyright. No entrando en detalles sobre el asunto, pero intentando aclarar un poco el tema, “podríamos decir que el

sistema de derechos de autor es más personalista y protege la creación como extensión de la persona, en cambio, el sistema *copyright* es más colectivista y protege la creación para estimular a los creadores como beneficio de interés general” (Bain *et al.*, 2007a, p. 50).

Los derechos de autor sirven de base para las licencias de software de todas las categorías en Europa. Las del privativo se consideran las más perversas ya que, en su amplia mayoría, simplemente permiten al usuario utilizar el programa, además de todavía especificar en qué condiciones. Es decir, de forma limitada. Para usarlo, hay que aceptar incondicionalmente las restricciones impuestas por el proveedor. El derecho de uso y de cualquier otra acción está explicitado en la licencia que acompaña el software y que establece sus condiciones de uso.

Así, por ejemplo, si el usuario desea copiar o redistribuir el programa por una necesidad cualquiera está impedido legalmente de hacerlo, por regla general. La solución en estos casos es intentar conseguir una autorización expresa del proveedor del software, que retiene todos los derechos reservados. También para modificar o adaptar el programa, incluso a sus propias necesidades, ese usuario tendrá que solicitar al proveedor para que lo haga o indique una solución, ya que no tiene acceso al código fuente del programa. Incluso pagando, no hay garantías de que esto sea posible. El código del programa suele estar muy bien protegido y guardado secretamente, ya que, en estos casos, suele ser fuente de recursos para el proveedor.

Hay que destacar que, habitualmente, acciones como copiar, redistribuir y modificar el software suelen no ser permitidas al usuario. Aunque en algunos casos, estas acciones sean permitidas, lo normal es que lo sean con determinadas restricciones. Y todo esto es así porque la normativa de propiedad intelectual reserva al proveedor del software los derechos para conceder o negar al usuario la realización de estas acciones.

De este modo, mientras las licencias de software libre aseguran derechos al usuario, las de software privativo se caracterizan por restringir sus derechos, además de impedirle el acceso al código fuente en casi todos los



casos. Así, es prácticamente imposible para el usuario estudiar, modificar y adaptar el programa a sus necesidades, aunque quisiera y tuviese conocimiento técnico para todo esto.

Sin embargo, hay algunas excepciones: en casos especiales, es permitido al usuario acceder al código fuente del programa. Por ejemplo, Microsoft accedió a las exigencias del gobierno de Alemania de poder comprobar que el paquete ofimático Office no contenía código malicioso y le permitió acceder a su código, aunque sólo para hacer esta comprobación. En general, no se puede hacer nada más con el código. En estas situaciones, cabe resaltar que el hecho de que el usuario tenga acceso al código fuente no significa que el programa deje de ser considerado software privativo. Algunos proveedores han empezado a mostrar el código de sus productos como una estrategia de mercado, pero manteniendo las restricciones características de esta modalidad de software.

De hecho, otra característica determinante del software privativo es que, por regla general, se facilita al usuario por medio del pago de una licencia de uso. El precio de esta licencia, en muchos casos, es caro para el gran público y por esa razón este tipo de software es objeto de copias ilegales. Este coste puede variar dependiendo de las características del software, de su proveedor, del tipo de usuario, de la cantidad de licencias comercializadas y de otros factores. La licencia de uso de un determinado programa, por ejemplo, puede ser más barata si el propósito de su uso es destinado al entorno educativo. También puede ser diferente si una empresa adquiere una gran cantidad de licencias.

Es necesario destacar que, al pagar por la licencia de uso, el usuario adquiere simplemente el derecho legal de utilizar el programa. Es decir, incluso pagando por la licencia, no le garantiza el acceso a su código fuente, tampoco a ejecutar acciones como copiarlo, modificarlo y adaptarlo a sus propias necesidades.

La licencia, generalmente, deja muy claro que el acto de pagar no convierte al usuario en dueño del programa. Es decir, el usuario no lo está comprando. Lo que está adquiriendo con el pago es tan sólo el permiso de

poder instalar y usar el programa, ya que éste no es objeto de venta. En realidad, parece más una especie de alquiler del programa.

#### 2.4.2 La *End User License Agreement* (EULA) o Contrato de Licencia de Usuario Final (CLUF)

En el mundo del software privativo existen muchas licencias. “Prácticamente cada fabricante de software ha creado su propia licencia adecuada al software en cuestión y el modelo de negocio del fabricante” (Mas, 2005, p. 69).

La más conocida licencia del mundo propietario es la *End User License Agreement* (EULA) o en versión traducida al castellano como Contrato de Licencia de Usuario Final (CLUF). Este tipo de licencia es la que regula la distribución de la gran mayoría de software privativo. Por ejemplo, muchos de los productos de la empresa Microsoft.

Dependiendo del software, las cláusulas de este contrato pueden variar, estableciendo ciertos permisos o restricciones al usuario al utilizar el programa, es decir, lo que se puede o no hacer con el software. La EULA suele aparecer durante la instalación del programa en el ordenador.

La lectura de una EULA de cualquier programa es relevante para entender exactamente en qué condiciones podemos utilizarlo, aunque habitualmente poquísimos usuarios la leen. En ese sentido, podemos citar un artículo publicado en el periódico español El Mundo<sup>15</sup>. Según este reportaje, una empresa americana pretendía llamar la atención sobre sí misma y destacar que su producto, a diferencia de otros de la competencia, no tenía cláusulas abusivas en su licencia. Como experimento, la empresa puso un mensaje en su EULA en cuyos párrafos finales se indicaba que a quien escribiese a una dirección de correo le esperaba un premio en metálico. Más

---

15 Más información en:

<http://navegante2.elmundo.es/navegante/2005/02/25/weblog/1109297833.html>.

de 3.000 usuarios se habían instalado el programa sin leerse entero el texto, hasta que un usuario lo leyó y escribió a la dirección electrónica. El remitente recibió 1.000 dólares. La lectura de la EULA se vio fomentada y la compañía consiguió colocar su historia en Slashdot, un sitio de noticias orientado al mundo tecnológico.

Sería muy agradable que toda EULA presentase una sorpresa de esa naturaleza. Pero, de una manera general, las sorpresas que suele deparar, desafortunadamente, no son nada interesantes para el usuario. Normalmente, viene acompañada de una serie de restricciones, que, incluso pagando por la licencia de uso, termina algunas veces actuando de forma ilegal por desconocimiento de sus cláusulas.

En el tan amplio mundo de las licencias de software privativo, podemos encontrar prohibiciones o restricciones curiosas y sorprendentes. A modo de ejemplo, en la EULA del SQL Server 2012 Standard<sup>16</sup> viene con una cláusula de que no se puede criticar públicamente el producto, por ejemplo, comparándolo con otros y publicar los resultados.

Este sería solo un ejemplo de abusos que, en la práctica, podemos encontrar en las licencias de software privativo. De todos modos, es información curiosa que nos hace reflexionar sobre la validez jurídica de algunas cláusulas que consideramos abusivas, disparates o discutibles. Incluso, algunas son difíciles de cumplir.

Frente a todo lo expuesto, creemos que ya tenemos algunos elementos para comprender la diferencia esencial entre el software libre y el software privativo. Ha quedado claro que lo que distingue el software libre del privativo no es el simple hecho de que se necesite pagar para obtener el permiso de uso. En realidad, existen muchas otras diferencias significativas.

Para entrar más en el complejo, enigmático, excéntrico y muchas veces confuso mundo de las licencias de software, hay diversos materiales sobre el asunto disponible en Internet. De entre éstos, recomendamos consultar las obras de Bain *et al.* (2007a; 2007b).

---

16 Más información en: <https://www.microsoft.com/en-us/usetterms>.

## 2.5 El software *freeware*

El término «*freeware*» es muy conocido en el mundo de la informática, principalmente, por los varios problemas de ambigüedad asociados al término. Y, quizás, es por esto que no hay un consenso general sobre su definición.

De acuerdo con Stallman (2004), este término ha sido comúnmente usado en los años 80 para programas distribuidos sólo como ejecutables, pero que no permitían acceso a su código fuente. Actualmente, no hay un consenso al respecto de su definición. Ya según la Free Software Foundation (2019a), este término no tiene una definición clara aceptada y lo utiliza para designar al software que se puede distribuir gratuitamente, pero no modificar, ya que el código fuente no está disponible. Por lo tanto, no es software libre.

De este modo, se trata de un término usado frecuentemente para referirse al conocido software gratis o gratuito. Tiene la característica de que el usuario no paga por la licencia de uso del programa y puede instalarlo y usarlo libre y legalmente y en algunos casos copiarlo y redistribuirlo con total libertad. Como ejemplo de programas de esa naturaleza podemos citar Skype, Adobe Reader, Ccleaner, Opera y tantos otros.

Es importante enfatizar que muchos proveedores de software distribuyen sus productos tanto en versión *freeware* como de pago, es decir, privativo. De este modo, muchas veces hacen que sus programas ganen visibilidad. Obviamente, la versión gratis es más limitada y tiene menos recursos. De entre esos proveedores, destacamos los de programas antivirus.

Como comentamos anteriormente, es muy habitual la confusión entre este tipo de software y el libre. Incluso, algunas veces, son usados como sinónimos. Esto es así porque, igual como la mayoría de este último, el *freeware* es distribuido gratuitamente, es decir, sin coste económico, y se puede utilizar ilimitadamente. Aún así, este software no es libre y viceversa, ya que generalmente el usuario no tiene acceso al código fuente del programa, requisito esencial para que un software sea considerado libre.

Algunas veces este término también es utilizado equivocadamente como sinónimo de software de dominio público.

## 2.6 El software *shareware*

El software «*shareware*» algunas veces también es referenciado como software de evaluación. La Free Software Foundation (2019a) utiliza este término para designar al software que no es libre, pero que se permite redistribuir copias del programa. Por cada copia utilizada, el usuario debe pagar un cargo por licencia.

El usuario, habitualmente, sólo puede utilizar el software por un determinado periodo de tiempo, que puede ser de semanas o meses, para saber si el programa responde a sus necesidades y expectativas antes de decidir pagar por la licencia de uso. Normalmente, durante este período se suele permitir únicamente el uso de determinados recursos del programa.

Pasado el periodo de prueba, que significa que ha expirado el plazo de permiso de uso, el usuario ya no tiene el derecho de utilizarlo. Si desea continuar, tiene que pagar por la licencia. En el caso de no pagar e intentar seguir, pueden ocurrir situaciones en que el programa simplemente deje de funcionar. Realmente, “la exigencia de pago puede estar incentivada por funcionalidad limitada o mensajes molestos, o una simple apelación a la moral del usuario, además de que las estipulaciones legales de la licencia podrían utilizarse en contra del infractor” (González, Seoane y Robles, 2007, p. 18).

En general, este tipo de software es cerrado, ya que el usuario no tiene acceso al código fuente. Pero se le permite redistribuir copias del software, ya que es interés del proveedor divulgarlo y venderlo. Se puede obtener fácilmente en Internet.

Algunas veces el software *shareware* se confunde con el software de demostración, que también es conocido como software de prueba o más popularmente como software demo. Esto pasa porque esta última modalidad de software también se distribuye sin coste y, normalmente, sólo en versión reducida. En los casos en que se desea adquirir la versión completa del programa también hay que pagar por la licencia de uso. Es muy común en la distribución y divulgación de videojuegos.

## 2.7 El software de dominio público

A diferencia de todos los otros tipos que hemos analizado anteriormente, el software de dominio público no se distribuye al usuario bajo una licencia. Se trata de un tipo especial de software, que muchas veces se confunde o se utiliza como sinónimo de software libre. El malentendido se debe a que los conceptos, los matices y las licencias en el mundo de las categorías de software son desconocidos para la gran mayoría de usuarios de ordenadores e, incluso, también para algunos técnicos o profesionales del área de informática.

El software de dominio público, según la Free Software Foundation (2019a), es aquél que no está protegido por derechos de autor. Es un caso especial de software libre no protegido con *copyleft*, lo que significa que algunas copias o versiones modificadas pueden no ser completamente libres. Pero, la mayoría del software libre no es software de dominio público, ya que está protegido por derechos de autor, expresados en una licencia.

Este tipo de software se caracteriza por la renuncia total del autor de todos sus derechos a favor del común. Esta situación tiene que estar declarada explícitamente en el programa. Si no dice nada, el programa es privativo, es decir, no se puede hacer nada con el software. Si es de dominio público y además se proporciona el código fuente, el programa es libre (González, Seoane y Robles, 2007).

Hay que resaltar que la renuncia a los derechos de autor para ceder el código al dominio público no es válida en el derecho de la Unión Europea, puesto que los derechos morales, que son una parte de los derechos de autor, son irrenunciables. En la Comunidad Europea, un software es considerado software de dominio público sólo después de 70 años de la muerte de su autor.

En los EE. UU. y algunos otros países hay dos formas de que un software pase a ser considerado de dominio público: por decisión del autor o por cumplimiento de la ley, que determina que el derecho de autor expira normalmente “50 años después de acabada la vida del autor o de 75 desde la publicación en obras realizadas por encargo” (Bain *et al.*, 2007a, p. 95). En el caso de la expiración del derecho de autor, “dada la rápida evolución del software, en la práctica no va a existir ningún software en dominio público hasta muchos años después de haber quedado obsoleto” (Bain *et al.*, 2007a, p. 316).

En realidad, la expresión dominio público es un término legal que significa que «no está protegido por derechos de autor». Así, cualquier software distribuido bajo una licencia, independientemente de su naturaleza, no es considerado de dominio público.

El software de dominio público puede ser modificado y usado de cualquier manera, incluso sin citar al autor original. En todo caso, se considera de mal gusto actuar de esta forma (Branco, 2005). Esto no es común ya que, de una manera general, “las obras de dominio público pueden ser utilizadas por cualquiera, siempre que se respete la autoría y la integridad de la obra” (Bain *et al.*, 2007a, p. 95).

La asociación con el software libre existe porque estos dos tipos de software presentan algunas características en común. Así como el de dominio público, el libre, normalmente, también es adquirido gratuitamente. Además, ambos permiten usar, copiar y redistribuir libremente el software. En el caso en que es liberado el código fuente del software de dominio público, ambos también permiten modificar y redistribuir copias de la nueva versión sin ninguna restricción.

Para aclarar las diferencias: un software de dominio público distribuido con su código fuente puede ser considerado un software libre, pero un software libre distribuido bajo una licencia cualquiera no es de dominio público.

Si el usuario utiliza un software de dominio público y lo convierte en software libre, redistribuyéndolo con una licencia libre, nos parece que no está causando un daño a la sociedad. Si decide convertirlo en software no libre, esto sí, será un gran daño, pero es perfectamente posible. El problema que puede pasar es que el usuario oportunista puede cambiar la versión original y transformarlo en un programa privativo o de cualquier otro tipo. De esta forma, estaría prohibiendo que el usuario que reciba la nueva versión pueda disfrutar de los beneficios del software en el formato de dominio público.

Para muchos usuarios, técnicos y profesionales del área, el software verdaderamente libre es el de dominio público ya que todos los usuarios tienen total libertad de hacer uso de esa propiedad intelectual. Que un software sea de esta clase significa que el programa no tiene propietario y se vuelve un bien común que beneficia a toda la sociedad y, por consiguiente, a todos por igual.

La propia Free Software Foundation (2019b) asevera que, de hecho, si el proveedor decide poner un programa bajo dominio público, sin derecho de autor, lo está prácticamente convirtiendo en software libre. Por otro lado, también permite que lo cambian en software privativo. En este caso, al revés de las libertades que lo tendría en ser software libre, puede ser plenamente eliminadas por el intermediario, que se las ha arrebatado.

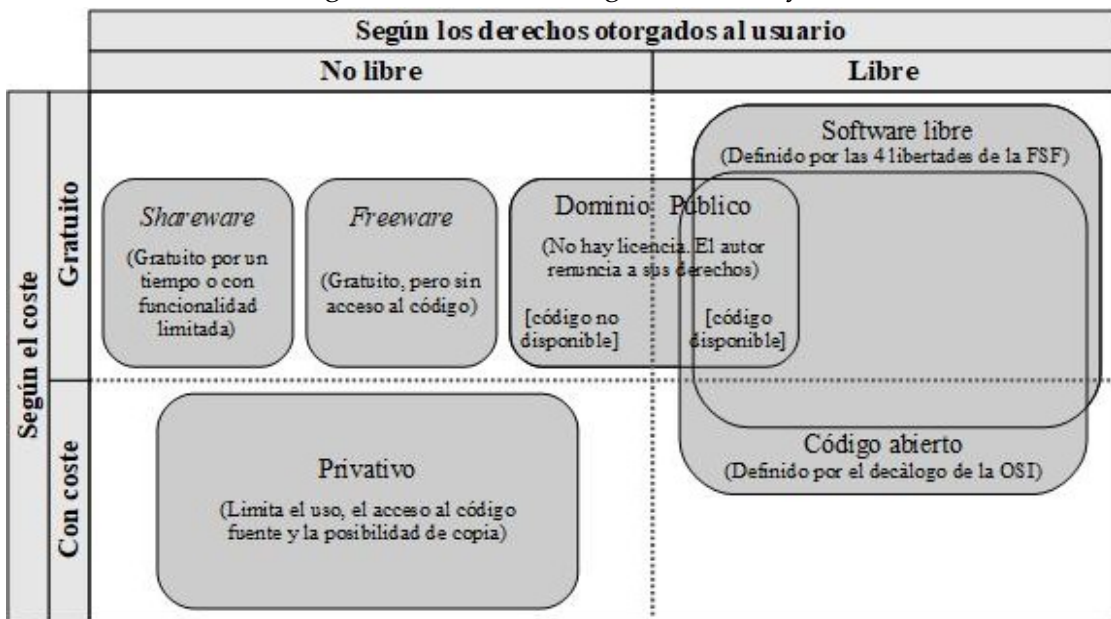
## 2.8 Resumen del capítulo

Entrando a analizar el software en sí, se puede ordenar en categorías según distintos criterios. Por ejemplo, según la posibilidad de



acceder a su código o bien de acuerdo con su finalidad, que puede ser comercial o sin ánimo de lucro. La clasificación más habitual es en función de los derechos que otorga al usuario, normalmente recogidos en una licencia de uso, y del coste de adquisición. El siguiente esquema muestra las categorías que se establecen según esta combinación de criterios.

Ilustración 1: Categorías de software, según derechos y coste



Fuente: Elaboración propia

La licencia es un documento legal que acompaña el software y contiene sus normas de uso. Estas normas delimitan lo que el usuario puede hacer con el programa y lo que no. O sea, cuáles son sus derechos, límites y libertades.

Las licencias libres son, en resumen, el mecanismo legal que se utiliza para garantizar al usuario de software, por lo menos, las cuatro libertades propuestas por Stallman, impidiendo que, de alguna forma, se puedan hacer restricciones a éstas.

Así, en resumen, un programa, para ser considerado libre, debe ser distribuido de tal modo que el usuario tenga la libertad de instalarlo, utilizarlo, copiarlo, estudiarlo, modificarlo, adaptarlo, perfeccionarlo y redistribuirlo, con o sin modificaciones, cobrando o gratuitamente. También es requisito obligatorio que el código fuente del programa esté disponible.

El acelerado avance tecnológico y los intereses cada vez más crecientes en la industria de software, han generado una proliferación de licencias, incluso libres. Eso termina haciendo difícil o complicado elegir una de ellas para el uso o la distribución del software libre. Por esta razón, en los casos de dudas es común recurrir a expertos y tener en cuenta su opinión para ayudar a decidir. De todas formas, la licencia GNU/GPL nos parece la más adecuada para distribuir software libre.

El presente trabajo se centra en el software libre, que se define por otorgar al usuario cuatro libertades, enunciadas en los 80 por Richard Stallman.

# 3 | EL SOFTWARE LIBRE

## 3.1 Introducción

A pesar de parecer un tema relativamente reciente, hace ya algunas décadas que el software libre viene despertando el interés de los usuarios de ordenadores de las más diversas áreas. Poco a poco, esta expresión está dejando de ser desconocida o un asunto trivial para muchos. A día de hoy, cada vez más, las personas han oído hablar o comentan algo relacionado con esta categoría de software, pero la gran mayoría no tiene claro qué es exactamente software libre. Incluso, llegan a usarlo sin saberlo.

El software libre se remonta al surgimiento de los primeros programas informáticos. El acto de compartir software es tan antiguo como los propios ordenadores (Stallman, 2004). Es decir, su historia se confunde con la propia historia del computador. Abundando en detalles, González, Seoane y Robles (2007) presentan un poco de la historia del origen del software libre.

Así, en el inicio, los primeros programas de ordenadores eran considerados libres. Tan libres, que nadie se molestaba en llamarles así porque nunca se había planteado una alternativa. Era común entre los programadores compartir sus códigos fuentes, o sea, sus conocimientos. En

realidad, esto es una práctica común en el mundo científico y una tradición milenaria.

A principios de los 70, algunas empresas empezaron a privatizar los códigos de sus programas y a tratarlos como secretos industriales. A partir de ahí, la evolución hasta la actualidad ha convertido el software en un lucrativo mercado, dominado por un verdadero oligopolio de empresas que controlan el negocio mundial de software. En el camino, han ido quedando prácticamente excluidos algunos sectores de la sociedad, que, en algunos países, pueden llegar a suponer una gran mayoría de la población.

Con la diseminación de los ordenadores, el software pasó a ser objeto de interés general y se volvió un mercado lucrativo, lo que llevó a un cambio de postura. El acceso al código fuente del programa, que permitía entenderlo y hasta modificarlo para atender a las necesidades del usuario, pasó a ser controlado y no disponible ya para el usuario. De este modo, surgió una nueva categoría de software. La idea inicial de compartir el código, y por consiguiente el conocimiento, para que todos pudiesen usarlo y beneficiarse parecía estar sepultada, pues los intereses económicos se estaban sobreponiendo a los intereses colectivos de la sociedad.

En la década de los 80, disconforme con esa nueva realidad, Stallman se propuso a crear un nuevo sistema operativo completamente libre, como era en el pasado. Así surgió el GNU/Linux. Para divulgar el nuevo sistema y los principios del software libre, decidió fundar la FSF en 1985 (Adell y Bernabé-Muñoz, 2007). De este modo, afloró el movimiento del software libre que, basado en la libertad y en la cooperación, se propone despertar la sociedad ante los posibles problemas derivados del monopolio y el control de los programas por los proveedores de software. La esencia es que todos puedan servirse del software libre.

A partir de este momento, Stallman viene intentando concienciar la sociedad en general de la relevancia del software libre. Y esto ha provocado muchas discusiones alrededor de este movimiento. Unos lo han calificado como uno de los más relevantes fenómenos que han surgido en el mundo de las TIC después de Internet. Algunos lo han considerado una moda, y, en

consecuencia, una tendencia pasajera. Otros lo han considerado un simple movimiento filosófico o simplemente lo han ignorado. Independientemente de como se quiera calificar, su evolución en el escenario internacional es incuestionable y ha provocado una verdadera revolución en el mundo del software y de sus usuarios y desarrolladores.

Como prueba de su repercusión, fue creado en 2004 el *Software Freedom Day* (SFD) o día de la libertad del software, o más concretamente del software libre. Es un evento mundial que se celebra cada año normalmente en el tercer sábado del mes de septiembre. La idea es concienciar, divulgar, promover e incentivar el uso de este software.

Hay también un número significativo de congresos y distintos acontecimientos alrededor del mundo sobre la temática de software libre con el propósito de intercambiar experiencias, divulgarlo o promoverlo junto a las administraciones públicas, las empresas, las organizaciones, los usuarios, los desarrolladores y la comunidad educativa, entre otros.

Podemos encontrar también artículos, informes y publicaciones que incluyen secciones informativas o, incluso, que se dedican exclusivamente al software libre. Además de éstos, han aparecido libros y noticias en los medios de comunicación relacionados con este asunto. En Internet, tenemos muchas webs de referencia, comunidades, fórums de discusión, boletines digitales, weblogs, sitios de noticias, entre otros que tratan de este tópico.

Las bases del movimiento del software libre también han servido para despertar y motivar otros movimientos. Como ejemplo de la extensión a otros ámbitos, podemos citar las licencias *Creative Commons* (CC)<sup>17</sup> y el movimiento de recursos educativos abiertos. Otro caso interesante de inspiración del movimiento que vale la pena destacar es el de la Wikipedia<sup>18</sup>. En ese sentido, quizás esas influencias del software libre podrían ser la más gran contribución del movimiento para la sociedad (González *et al.*, 2006).

---

17 Más información en: <http://creativecommons.org/>.

18 Más información en: <http://www.wikipedia.org/>.

### 3.2 La adhesión al software libre

Es innegable que un número cada vez mayor de usuarios e instituciones públicas y privadas en el mundo, incluso de renombre, se están adhiriendo al uso del software libre. Aunque no haya datos contrastables sobre su actual penetración en los más diversos ámbitos de la sociedad, es incuestionable que hay un interés creciente por este tipo de software. Parece un proceso lento, pero cada vez más miles de usuarios, empresas e instituciones de alguna u otra forma, notifican algún uso de software libre. Esto ha permitido una considerable penetración en prácticamente todas las más distintas áreas estratégicas de la sociedad: administración pública, educación, industria, comercio y otros. Es indudable que está en plena expansión. Diversos autores consultados lo destacan principalmente su presencia en la administración pública y en el entorno universitario.

En la administración pública, es cada vez más creciente el número de instituciones que han manifestado interés por esta modalidad de software. Justamente es este entorno uno de los principales impulsores y consumidores del software libre. Eso porque la administración pública tiene una serie de características y necesidades particulares que necesitan ser respetadas en un estado democrático de derecho. Y en ese sentido, el software libre es una opción interesante (CENATIC, 2008). España, Brasil, Venezuela, Chile, Alemania, Reino Unido, Rusia, Argentina, Cuba, Francia, México, Suecia, Corea del Sur y China son algunos de muchos otros países que han decidido de alguna forma invertir en el software libre en sectores de la administración pública.

Inevitablemente, es justamente el área de educación donde su adopción es cada vez más impulsada. Se considera que esta área reúne las condiciones idóneas para la expansión de este software, considerando todas sus potencialidades. Efectivamente, sus características como los principios de libertad, compartición y cooperación, son determinantes para una apuesta en la producción colectiva del saber. El fenómeno del software libre “es intrínsecamente social, no se limita al campo del software, sino que es

aplicable al conjunto de producción y distribución de conocimiento” (Roca y Castells, 2006, p. 10). Esta experiencia generada puede ser perfectamente difundida y puesta a disposición de la sociedad y, en consecuencia, todo el mundo puede usarla también de manera libre y democrática. Así, quizás, puedan sentirse realmente ciudadanos de la sociedad de la información y del conocimiento.

En este sentido, en España, podemos destacar las iniciativas pioneras de Extremadura, que ha desarrollado el proyecto gnuLinEx, y Andalucía, Guadalinex, destinadas a la consolidación del software libre en sus comunidades autónomas, especialmente en el área de educación. En mayor o menor medida, esos han sido ejemplos claros de apuestas e intentos de promover el software libre.

En efecto, éstos también han sido casos de los más destacados en que la introducción del software libre en el sector público ha empezado por el área educativa. A partir del análisis y observación del pleno éxito de la experiencia, ha surgido la idea de expandirla a otros o todos los ámbitos de la administración.

En el ámbito universitario, destacamos la comparativa de Universidades en Software Libre (RuSL)<sup>19</sup>, que es una clasificación de universidades de países hispanohablantes en función de diversos indicadores que miden el compromiso de las instituciones en el uso, difusión y creación del software libre. En la última comparativa, ha destacado la Universidad Nacional del Callao, en Perú.

Además de la administración pública y el área de educación, otro entorno que ha apostado decididamente por el software libre ha sido el empresarial. La estrategia de algunas grandes empresas, sobre todo del área tecnológica, a favor de esta categoría de software ha abierto las puertas del entorno económico a los programas libres. A partir de entonces, un incontable número de empresas alrededor del mundo han seguido la misma dirección. Es indudable también su creciente popularidad en este sector.

---

<sup>19</sup> Más información en: <https://www.portalprogramas.com/software-libre/ranking-universidades/acerca-de>.

Algunas de las grandes empresas del mundo que financian proyectos de software libre son: Google, IBM, HP, Intel, Samsung, AMD, Nokia, Oracle, Dell y Red Hat entre tantas otras.

La diseminación mundial del software libre en las más diversas áreas ha provocado cambios en algunos sectores de la sociedad. Eso se debe, sobre todo, a las ventajas que presenta en relación a las otras categorías de software. Éstas son de carácter económico, técnico, social y ético. La ventaja económica es una de las más citadas en general en la literatura. De hecho, el ahorro en costes de licencias es una de sus principales características. Aunque la expresión tiene sentido, ya que la gran mayoría de software libre es distribuida gratuitamente, hay que resaltar que no siempre es gratis. Esta es una gran equivocación, conforme ya hemos comentado.

Para ilustrar la relevancia de esta característica, podemos citar el caso de los países en vías de desarrollo. En estos casos, el uso del software libre puede contribuir a una reducción drástica de costes, ya que no se suele pagar por las licencias de uso de los programas. Además, la reducción de costes puede, por ejemplo, posibilitar el desarrollo de políticas públicas que tengan el objetivo de promover la inclusión digital.

El software libre surge, o mejor, reaparece y es, actualmente, una importante alternativa a tener en cuenta al software no libre, especialmente, el privativo. Sea en la administración pública, en las grandes corporaciones, en el mundo empresarial de las grandes y pequeñas empresas, en la educación en general, en las organizaciones y en la salud entre otras áreas, el software libre se hace presente. Es una realidad, y cada vez más consolidada. Entre los usuarios de ordenadores, también gana espacio y parece ser una tendencia.

La relevancia del software libre ha ganado tanta repercusión que incluso ha pasado a ser citado en el marco del programa electoral de algunos políticos. Todo esto, indudablemente, manifiesta su consolidación y el avance que ha adquirido en los últimos años.



En un área dinámica como la tecnológica, es difícil predecir el futuro. Y pasa lo mismo con respecto al software libre. Pero es innegable que éste gana espacio en las más diversas áreas y se consolida como una tendencia, por lo menos en el segmento de ordenadores.

Por otro lado, una preocupante situación despunta con la reciente revolución tecnológica e invasión de los *smartphones* y *tablets* en el área tecnológica. El uso de software libre en dispositivos móviles se ve complicado por tres factores: los fabricantes de teléfonos inteligentes producen hardware y software que son extremadamente dependientes entre sí, los teléfonos inteligentes se conectan a la red telefónica a través de una capa adicional de software restringido y la gran cantidad de dispositivos y sensores en teléfonos inteligentes operados por controladores restringidos. Así, hoy día utilizamos servicios en Internet sin saber exactamente lo que hacen, desde dispositivos que ejecutan software que no podemos verificar y que acceden a nuestros datos en circunstancias oscuras (Cardoso, 2015). Es decir, nada muy distinto de lo que ya pasa cuando usamos software privativo. Ese parece ser el más grande desafío para el mundo del software libre en ese momento.

### 3.3 Ventajas del uso del software libre

El avance de la adopción del software libre en el contexto general se debe, sobre todo, al hecho de que éste conlleva una serie de ventajas sobre el software no libre. Las principales están directamente relacionadas con las cuatro libertades fundamentales del software libre, sobre todo por los derechos que otorga a los usuarios. La posibilidad de instalar, usar, copiar, modificar, adaptar y redistribuir libremente, además de poder usarlo para cualquier propósito, es una parte de los beneficios que ofrece.

La adhesión al software libre no solamente viene justificada por los beneficios económicos. Las motivaciones son muy variadas. Algunos otros de los principales factores son: independencia del proveedor, la disminución de recursos de hardware, el fomento a la interoperabilidad, la estabilidad y

seguridad, el compartir y cooperar y el combatir la brecha digital.

Presentamos a continuación una serie de ventajas, entre otras, que son determinantes y representativas para impulsarlo en los más diversos entornos. Vale resaltar que no se trata de una secuencia en orden de importancia o de prioridad, ya que todas, a su modo, gozan de estas características.

### 3.3.1 Reducción de Costes

Un ordenador necesita programas para que se pueda utilizar. Si optamos por el software privativo, dependiendo de los programas, el coste final para usar la máquina puede aumentar significativamente, considerando la necesidad de pago por las licencias de uso de software. Eso sucede porque, a lo largo de los años, hemos tenido un considerable descenso en el precio del ordenador, mientras que el coste de los productos de software no ha seguido en la misma dirección.

Rompiendo con todas esas tendencias, existe la opción de utilizar el software libre. Éste surge, entonces, como una interesante solución ya que es, normalmente, distribuido sin ningún coste adicional para el usuario, igual que sus versiones actualizadas o nuevas.

Eso es sólo una de las razones que lo hace significativamente más atractivo en el aspecto económico que el software privativo. Parece que realmente esta es la puerta de entrada de los programas libres en los más diversos ámbitos. Y para la administración pública eso todavía parece esencialmente adecuado, considerando la economía con el dinero público. Con la reducción de costes, y en algunos casos de forma drástica, los recursos liberados por el ahorro de las licencias se pueden usar para otras finalidades.

Esta modalidad de software, entonces, al contrario que el privativo, aporta la ventaja, generalmente, de no pagar por ningún tipo de licencia de

uso de software. Por eso, uno de los argumentos más convincentes y utilizados inicialmente por diversos autores consultados para justificar su uso está relacionado con la reducción de costes. Por otro lado, cabe añadir que, “a medida que pasan los años y la sociedad de la información avanza, el gasto en tecnologías de la información será mayor y mayores serán aún los ahorros en coste de licencia que el software libre representa” (Mas, 2005, p. 154).

Otro punto a considerar, es que las versiones actualizadas del software libre, o las nuevas, al igual que su obtención, también se suelen distribuir sin ningún coste adicional para el usuario. Además de estar permitido, también, redistribuirlas libremente. Cuando hay costes de actualización o por nueva versión, éstos son, muchas veces, irrisorios. Las nuevas versiones de programas privativos, al contrario, por regla general, son pagadas, aunque los costes de migración a la nueva versión acostumbran a ser menores que el coste inicial.

De modo general, se cita el ahorro del coste de adquisición de licencias como una de las principales ventajas derivadas del uso del software libre. Pero eso no es lo primordial, ni tampoco es el único beneficio que aporta al usuario. El precio es muy atractivo, pero no es determinante, ya que otras categorías de software, como la *freeware*, también tienen sus productos distribuidos sin ningún coste adicional al usuario, o sea, gratis.

### 3.3.2 Independencia del proveedor

Uno de los requisitos que debe cumplir el software para ser libre es tener acceso al código fuente del programa. Ese hecho es el gran diferencial en relación con las otras categorías. Es una de las principales y más destacadas ventajas de usarlo. Por el contrario, con el software no libre, normalmente, no se tiene acceso al código fuente de los programas. Eso, en general, presenta una serie de inconvenientes al usuario, en especial, la dependencia frente al proveedor del software.

Esa dependencia significa que el proveedor también tiene exclusividad en la prestación de determinados servicios al usuario, ya que es el único que tiene acceso al código fuente de los programas. Dichos servicios incluyen desde tareas de mantenimiento, como corregir un error, mejorar, actualizar o adaptar el programa a las necesidades específicas del usuario, hasta el soporte técnico.

Con el software libre se reduce la dependencia del proveedor de software y, por consiguiente, también puede conllevar una reducción de precios en algunos servicios derivados. Esta independencia del proveedor es importante, por ejemplo, en las tareas de mantenimiento del software y en el soporte técnico. Se trata de aspectos a tomar en cuenta a la hora de elegir el software.

## # Soporte técnico

Otra ventaja de usar software libre está relacionada con la flexibilidad de opción del servicio de soporte adquirida por el usuario. Un problema frecuente en el mundo del software está relacionado con los fallos existentes en los programas, más conocidos como *bugs*, algunos capaces de causar perjuicios incalculables al usuario. Además de éstos también hay los derivados de las traducciones, como, por ejemplo, los gramaticales y ortográficos.

El proceso de corrección de errores en el software privativo no es sencillo, pues, como el usuario no tiene acceso al código fuente del programa, hay que contactar y esperar una solución del proveedor, que, en muchos casos, puede tardar mucho o incluso decidir que no le interesa, por ejemplo, porque no le resulta rentable.

Cabe resaltar que, en la gran mayoría de casos, aunque un fallo en el código fuente del programa, por ejemplo de seguridad, pueda causar un gran perjuicio al usuario, el proveedor del software privativo se exime de

cualquier responsabilidad. Todo eso suele estar especificado en la propia licencia de distribución del software.

En el caso del software libre, aunque tampoco el proveedor se hace responsable, la situación en cuanto a solución es notablemente distinta. Los fallos detectados pueden ser más fácilmente resueltos porque el usuario tiene acceso al código fuente del programa. De este modo, como antedicho, hay varias formas de buscar una solución o, incluso, esperar una actualización del programa con el error corregido. También en este caso, suele ser distribuida gratuitamente por Internet.

En los casos en que el proveedor decide no dar más soporte a las versiones más antiguas, usando software libre, ese problema es minimizado, una vez que se tiene acceso al código fuente del programa. También en esos casos, lo que prevalecerá es la relación precio/calidad/eficiencia del soporte, además de la atención dispensada al usuario por el prestador del servicio. Se acaba definitivamente el monopolio del proveedor de software, lo que caracteriza aún más las facilidades que el usuario obtiene al optar por el uso del software libre.

Esta independencia también trae beneficios para el desarrollo local de la región y el país en el sector económico, científico y tecnológico. De este modo, además de fomentar la competencia entre las empresas y profesionales en un sector marcado, habitualmente, por la presencia de multinacionales extranjeras, se está estimulando e impulsando la industria local de software y de servicios. Por otra parte, también es relevante para potenciar la capacitación y formación tecnológica de profesionales cada vez más cualificados en un área que siempre estará en pleno desarrollo y necesitará mucha mano de obra especializada.

Es aclarador comentar que las megaempresas innovadoras, como Yahoo, Twitter, Google y Facebook, construyeron su infraestructura y soluciones basadas en software libre y gratuito. Debemos incentivar y apoyar con recursos públicos la invención de nuevos proyectos en áreas poco exploradas por la industria del software privativo. Con esto, podremos crear grupos de inventiva, que no serán posibles en un mundo dominado por

patentes y licencias de propiedad (Silveira, 2015).

Frente a lo expuesto, es evidente que las instituciones quedan independientes del monopolio del proveedor del software al optar por el uso del software libre.

## # Adaptación de programas

En general, los programas son desarrollados para propósitos generales y así bastantes de sus funciones son inútiles, mientras que otras, más necesarias, no están disponibles para determinadas situaciones o usuarios. Además, habitualmente, no toman en consideración las diversidades culturales de las regiones.

En estas situaciones sería muy interesante que los programas pudiesen ser adaptados a las necesidades de sus usuarios. Se podrían modificar para suprimir lo que no es importante o lo que no es preciso y añadir las características propias y específicas de cada uno. En el caso del software no libre, poco o casi nada se puede hacer, ya que no se suele tener acceso al código fuente. Todos quedan en la dependencia del proveedor del software para solucionar esos problemas, ya que es el único que puede acceder al código y cambiarlo. Además, por regla general, se tiene que pagar para hacer las debidas adaptaciones, lo que se puede volver inviable, ya que ese tipo de ajustes para atender necesidades específicas, generalmente, son caros.

Eso sin contar que, muchas veces, puede ser difícil conseguir soporte del proveedor, ya que puede resultar no lucrativo o, incluso, no ser de su interés. Además, no está obligado a implementarlo, o sea, los usuarios son totalmente dependientes de él.

Por otro lado, nada de eso pasa cuando se opta por el software libre. Con el acceso al código es posible modificar, adaptar e, incluso, mejorar cualquier programa para atender las necesidades de los usuarios. De forma

totalmente legal, estarían simplemente haciendo uso de una de las libertades del software libre, la libertad 1.

Esa asistencia la puede prestar cualquier empresa, independientemente del tamaño y de la proyección en el mercado, o cualquier profesional con los conocimientos técnicos necesarios. Lo más atractivo es que todos los proveedores de servicios se encuentran en régimen de igualdad para competir en el mercado, ya que lo fundamental es la calidad ofrecida.

De este modo, el usuario puede negociar y elegir libremente el proveedor para determinadas tareas, tomando en consideración, por ejemplo, lo que ofrezca la propuesta que se adapte a sus necesidades y, sobre todo, la que ofrezca la mejor opción de precio, calidad, eficiencia y facilidades para efectuar las tareas. Consecuentemente, suele ahorrar dinero y, en muchos casos, puede obtener una mejor calidad en los servicios de mantenimiento, e incluso de soporte, ya que en esos casos lo que prevalece es la calidad y eficiencia de los servicios prestados.

Por otra parte, también tiene la libertad de cambiar de proveedor en los casos en que los servicios no presenten la calidad garantizada o no atiendan con precisión a sus expectativas. Lo que es normal en un mercado libre y muy distinto en un mercado caracterizado por el monopolio. En general, hay una relación cercana entre la competencia y la calidad. Cuanto mayor es la competencia, normalmente, mejor es la calidad de los servicios prestados y menor su coste.

Como podemos observar, también en esos casos, las ventajas de usar el software libre frente al no libre son evidentes.

## # Las lenguas minoritarias o minorizadas

Un ejemplo que ilustra muy bien la importancia de poder adaptar el

software a necesidades específicas es el caso de las lenguas minoritarias o minorizadas, que se han beneficiado mucho con el software libre. Hace más fácil traducir un programa a cualquier idioma y así respetar la identidad lingüística y cultural de regiones, pueblos y países. Es una forma de impulsar el uso y la divulgación de una lengua, además de contribuir a su normalización lingüística.

Con el software privativo, “el proceso de localización<sup>20</sup> de este tipo de programas sólo puede llevarse a cabo por el fabricante del programa, que es quien determina, bajo criterios comerciales e, incluso, alguna vez políticos, que estos tipos de traducciones no son viables” (Mas, 2003, p. 37). Eso pasa cuando las dimensiones del mercado son demasiado reducidas y no hay interés para la adaptación de los productos privativos a ese mercado específico (Tong, 2004). De este modo, el usuario es obligado a utilizar el software en otra lengua que no es la nativa. En España, por ejemplo, los nativos de lengua catalana son obligados, muchas veces, a usar algunos programas en castellano. Peor todavía, cuando el software sólo está disponible en inglés.

Estas situaciones ejemplifican muy bien el grave problema de la dependencia que se crea cuando se hace la opción por el software no libre, especialmente el privativo. Si con el catalán pasa eso, imaginemos con tantas otras lenguas aún más minoritarias o con menos recursos económicos, como las lenguas indígenas de América del Sur u otras regiones del mundo. Para las lenguas minoritarias, las ventajas del software libre son trascendentales. No se trata solamente de traducir el programa. También es una forma de preservar el multilingüismo y la multiculturalidad de pueblos que a lo largo del tiempo han quedado marginados. Es una oportunidad para que no se queden para siempre en el olvido o, en la peor de las hipótesis, se pierda parte de ese patrimonio cultural de la humanidad. Según la revista *Ethnologue*<sup>21</sup>, existen actualmente 7.117 lenguas habladas en el mundo. El sistema operativo más usado en los ordenadores alrededor del mundo, el

20 Localización es la “adaptación de un producto o programa al idioma y a las preferencias de los consumidores de un país. Va más allá de la pura traducción, ya que por ejemplo, una enciclopedia en CD-ROM debe destacar ciertos conceptos y dar la versión de los hechos históricos comúnmente aceptados en ese país” (Plágaro y Palacios, 2007, p. 317).

21 Más información en: <https://www.ethnologue.com/guides/how-many-languages>.



Windows de Microsoft, de acuerdo con el sitio de la empresa, en su versión 10, solamente está disponible en 208 lenguas<sup>22</sup>.

Por lo tanto, el incentivo a la producción, traducción y adaptación de programas libres es fundamental para consolidar las TIC, sobre todo, en la educación.

### 3.3.3 Disminución de recursos de hardware

Con el rápido avance que ocurre en el hardware, normalmente, los ordenadores quedan obsoletos y los programas desactualizados. Así, es común la necesidad de adquirir la nueva versión del software y renovar el hardware. En poco tiempo, este ciclo se repite y se necesitan más inversiones en esas tecnologías.

Todo eso se debe a que los proveedores de software no libre, especialmente privativo, para aumentar sus beneficios y consolidarse en el mercado, deciden sacar nuevas versiones de sus productos. Para eso añaden una serie de funciones y recursos tecnológicos, como algunos efectos visuales. En la gran mayoría de los casos son innecesarios, pero los usan para crearle al usuario la sensación de necesidad de actualizarse para no quedar al margen de un supuesto avance tecnológico.

Programas como el sistema operativo, cuando lanzan al mercado sus nuevas versiones, es habitual que exijan muchos recursos de hardware. De este modo, obligan a que el usuario compre un nuevo ordenador con más recursos, aunque el suyo esté en pleno rendimiento, o invierta en la adquisición de determinados componentes más sofisticados para que sean instalados en su máquina si desea trabajar con la nueva versión. Es frecuente en esos casos necesitar, por lo menos, más espacio libre en el disco duro, más memoria RAM (*Random Access Memory* - Memoria de Acceso Aleatorio) y un microprocesador con más velocidad para que el usuario pueda usar el

---

<sup>22</sup> Más información en: <https://support.microsoft.com/pt-br/help/14236/windows-language-packs>.

software.

Muchas veces los usuarios no tienen condiciones de hacer esas inversiones, sobre todo continuamente, y terminan por continuar usando ordenadores con pocos recursos y como consecuencia, programas desactualizados. Poco a poco, esa dependencia está cambiando. Para ilustrar, presentamos en la Tabla 1, a continuación, los requisitos básicos de hardware recomendados por los proveedores de software para instalar dos de los más conocidos sistemas operativos, el Windows 10, sistema no libre, y el Ubuntu 18.04 LTS, libre, en sus últimas versiones.

Tabla 1. Descripción general de los requisitos de hardware recomendados para instalar los sistemas operativos.

REQUISITOS DE HARDWARE RECOMENDADOS				
TIPO	SISTEMA OPERATIVO	PROCESADOR (64 bits)	MEMORIA (RAM)	DISCO DURO (ESPACIO MÍNIMO)
LIBRE	Ubuntu 18.04 LTS <sup>23</sup>	700 MHz	1 GB	10 GB
NO LIBRE	Windows 10 <sup>24</sup>	1 GHz	2 GB	32 GB

MHz: *Megahertz*; GHz: *Gigahertz*; GB: *Gigabytes*; 1 GB = 1.024 MB

Fuente: Elaboración propia

En el caso del sistema operativo no libre, algunos recursos del sistema pueden no funcionar bien o no estar disponibles con los requisitos mínimos recomendados. Además, suele tener un rendimiento crítico con esos valores porque son muy dependientes de los recursos de hardware para funcionar bien. No hay duda que Microsoft ha refrenado su tradicional voracidad de recursos por la presión en contra que ha supuesto la existencia de una

23 Fuente: [https://ubunlog.com/guia-de-instalacion-de-ubuntu-18-04-lts-bionic-beaver/#Requisitos\\_para\\_instalar\\_Ubuntu\\_1804\\_LTS](https://ubunlog.com/guia-de-instalacion-de-ubuntu-18-04-lts-bionic-beaver/#Requisitos_para_instalar_Ubuntu_1804_LTS).

24 Fuente: <https://www.microsoft.com/es-es/windows/windows-10-specifications>.

alternativa más respetuosa, como es el software libre.

Observamos que, por lo general, el sistema libre necesita menos recursos de hardware que el no libre. De hecho, las distribuciones GNU/Linux, normalmente, ofrecen opciones de instalación en ordenadores con pocos recursos de hardware. Es decir, son menos dependientes de la tecnología. Eso porque tienen como preocupación principal la funcionalidad y eficiencia del sistema.

La gran mayoría de los usuarios de ordenadores no necesita grandes recursos informáticos para sus actividades. Excepto para casos específicos, lo usan para navegar y tareas de ofimática. Estas aplicaciones cargan poco los ordenadores. Un sistema operativo libre, en general, tampoco exige grandes requerimientos de hardware. De esta forma, el usuario no necesita estar substituyendo o renovando su ordenador con la misma frecuencia con que sería necesario si estuviese utilizando programas no libres. Esto es importante pues se prolonga la vida útil del ordenador y disminuye la preocupante basura electrónica alrededor del mundo.

Cada año se venden más ordenadores. Las adquisiciones se destinan, principalmente, a la sustitución de ordenadores considerados obsoletos. Entonces, la pregunta clave es: ¿qué hacer con esos ordenadores? Ese es un problema que preocupa el entorno tecnológico, ambientalista y la sociedad en general. Las máquinas son descartadas y en lugar de recicladas, son abandonadas, contribuyendo masivamente a aumentar la basura electrónica. Pero eso es un fenómeno mundial y creciente, sobre todo, en los países emergentes.

Una buena parte de los ordenadores descartados van a parar a los basureros, especialmente en el continente africano, en vez de ser reciclados en el país donde han sido rechazados. No por casualidad, está en el barrio de Agbogboshie de la ciudad de Acra, capital de Gana, el mayor cementerio electrónico del mundo. Durante años ha recibido toneladas de basura electrónica de países ricos, especialmente de Europa y Estados Unidos. Y las consecuencias para las personas que trabajan allí son devastadoras. Pero, hace algunos años que la Unión Europea ha decidido prohibir la exportación de

basura electrónica a países en desarrollo.

El problema consiste, por ejemplo, en que parte de los componentes de algunos tipos de ordenadores usan metales tóxicos como mercurio, plomo y cadmio, que pueden contaminar el medio ambiente si no son bien reciclados. En el ser humano, la exposición a ese material puede provocar desde serios problemas cardíacos y/o respiratorios hasta algún tipo de cáncer. Lo que pasa es que, por lo general, los ordenadores, y muchos otros aparatos electrónicos, son destruidos de forma irregular y acaban agrediendo al medio ambiente.

Los datos sobre esa situación son preocupantes. Solamente en 2016, según el informe elaborado por *United Nations University* (UNU), *International Telecommunication Union* (ITU) y *International Solid Waste Association* (ISWA), el mundo produjo 44,7 millones de toneladas de basura electrónica. De ese total, solamente 8,9 millones de toneladas, es decir, el 20%, fue reciclado. Del 80% restante, el 4% de la basura en los países ricos se arroja a los desechos residuales. Se desconoce el destino del otro 76%. Es probable que sea arrojado, comercializado o reciclado en condiciones no adecuadas. El mismo informe estima el valor total de todo el material reutilizable presente en la basura electrónica en aproximadamente 55 mil millones de Euros (Baldé *et al.*, 2017).

Una de las posibles soluciones para minimizar el problema con los ordenadores descartados puede ser, perfectamente, el software libre. Con este tipo de software también es posible reciclar ordenadores que se consideran obsoletos, porque ya no disponen de recursos de hardware suficientes para ejecutar determinados programas. Esa posibilidad abre una serie de alternativas a la adquisición y renovación de ordenadores, liberándose del imperativo del mercado tecnológico.

### 3.3.4 Interoperabilidad

En el área informática, un asunto cada vez más relevante es la interoperabilidad, que es la “capacidad de comunicación entre diferentes programas y máquinas de distintos fabricantes” (Plágaro y Palacios, 2007, p. 295). En el caso del software, hay estándares informáticos creados con esa finalidad. Los datos son almacenados usando determinadas reglas que facilitan su lectura y recuperación de manera correcta por el programa utilizado para mostrar la información. Hay transparencia en la forma como esos datos son manejados. Esas reglas son conocidas como formatos y pueden ser clasificados, por ejemplo, en abiertos y cerrados.

Los formatos abiertos “disponen de una documentación pública que define con exactitud todos los aspectos del estándar, de forma que cualquier experto podría diseñar un programa para gestionar dichos datos” (CENATIC, 2008, p. 13). Los estándares cerrados “sólo están disponibles para el propietario, de forma que sólo 'sus' programas pueden gestionar los datos así almacenados. La diferencia no está en la titularidad del formato, sino en la inexistente disponibilidad del código para terceros” (CENATIC, 2008, p. 13). Se trata de una estrategia utilizada por los proveedores de software no libre para forzar al usuario a actualizar o cambiar de versión del programa, que, muchas veces, es innecesaria. Esa estrategia de mercado hace que el coste final de la migración sea un arma poderosa en las manos del proveedor. Es una forma de fidelizar el usuario, ya que cambiar hacia otro software significa entre otras cosas, perder todo el dinero ya invertido.

Además, en muchos casos sólo es posible recuperar parte de los datos. En otros, siquiera es posible recuperar nada. Es decir, a pesar de que el usuario pague por una licencia para utilizar el software privativo, tiene toda su información y datos de trabajo almacenados y condicionados a un determinado formato que, por lo general, sólo el proveedor sabe cómo manejar correctamente. Resumiendo, el usuario es totalmente rehén del proveedor del software y de sus productos.

Cuanto más usuarios se sirven de programas que utilizan formatos cerrados, más estable y fuerte pasa a ser el formato. Como consecuencia, éste invade el mercado informático y pasa a ser considerado como un patrón, comúnmente llamado estándar *de facto*, para cualquier intercambio de información. Y el usuario, normalmente, no se da cuenta que está frente, y es víctima, de un gran monopolio. Todo eso es preocupante, considerando que cada vez más la sociedad utiliza los medios informáticos como forma de comunicación.

El software libre, al contrario del privativo, está normalmente basado en formatos abiertos. Algunos de los formatos usados en los programas libres están en conformidad con las normas creadas y reconocidas por organismos internacionales. Eso garantiza y favorece la interoperabilidad entre programas. De este modo, garantiza el respeto a los usuarios para que puedan acceder sin restricciones a la información y cualquiera tiene acceso a la información almacenada por ellos. Y, así, los beneficios son para todos. Eso tiene una importancia incuestionable.

La interoperabilidad entre las aplicaciones es otro punto básico y favorable en pro del software libre. Como consecuencia, el usuario tiene mayor flexibilidad y libertad en su trabajo, ya que para intercambiar información puede elegir cualquier software que también utiliza un determinado formato para almacenar los datos. Esa independencia también asegura la perennidad de la información. Lo que significa que se puede preservar y estar disponible permanentemente durante toda su vida útil. Eso es así porque el usuario no es cautivo de ningún formato en especial y tampoco está atrapado por ningún proveedor de software. Aunque uno de ellos, por ejemplo, decida no trabajar más con un determinado formato, hay formas de continuar usándolo o, incluso, convertirlo en otro. De este modo, el usuario jamás correrá el riesgo de quedarse imposibilitado de acceder a su propia información.

Eso puede pasar desapercibido para el público general, sobre todo para los usuarios, pero es fundamental tratándose de instituciones, especialmente las públicas, que lidian con información de los ciudadanos.

De hecho, no es raro que los organismos públicos nos obliguen a utilizar determinados programas que usan formatos cerrados para acceder a su información o comunicarnos con ellos. Como consecuencia, es común para el usuario, o incluso para otros sectores de la propia administración, encontrar dificultades en el acceso a la información. Resulta realmente extraño que eso pase, sabiendo que, en muchos casos, hay estándares abiertos y libres que facilitan el acceso a la información almacenada, ya que cualquier programa que los utilice puede interpretarla. El usuario quedaría libre y podría usar el software que mejor le pareciese para contactar con esos organismos. Eso es lo que se considera, por lo menos, un buen funcionamiento de los organismos públicos: respeto a los derechos de los ciudadanos y libre acceso a la información pública.

Claro que hay excepciones, pero, en la gran mayoría, o incluso nos arriesgaríamos a decir casi en la totalidad, la comunicación entre los organismos públicos y sus ciudadanos se podría realizar utilizando software libre, que utiliza estándares abiertos. Por otro lado, también es verdad que aún hay pocos formatos estandarizados de software. Eso también es una clara demostración del dominio del software privativo y de los intentos de monopolizar un área esencial para el pleno ejercicio de la ciudadanía en la Sociedad de la Información y Comunicación.

Por detrás de toda esa discusión está el interés de llamar la atención hacia el problema de los formatos cerrados, enseñar que hay otra opción, los formatos abiertos, y, además, aprovechar para divulgar la cultura del software libre.

Ante ese panorama, es evidente que es preferible el uso del software libre. Y, en especial, por las razones expuestas, aquellos programas que usan formatos abiertos que sean estándares aprobados por organismos internacionales.

### 3.3.5 Estabilidad y seguridad

El software libre, por lo general, también es considerado estable y seguro. Eso es consecuencia del hecho de que el usuario tiene acceso al código fuente de los programas. Cualquier persona con conocimientos técnicos puede revisarlos y así, es más probable obtener programas más estables y seguros.

La posibilidad de escrutinio público del código fuente por una gran cantidad de personas que, normalmente están envueltas en los proyectos de software libre, favorece la estabilidad y fiabilidad de los programas, ya que los problemas y los errores son tratados de forma más rápida.

Otro considerable aspecto que se beneficia del acceso al código fuente de los programas es el de la seguridad, palabra clave en cualquier sistema informático. Eso permite, en general, mayores garantías para el usuario.

Por otra parte, una preocupación cada vez más creciente es con la inseguridad que existe cuando se trata del software no libre, es decir, la falta de transparencia. No se trata aquí, sencillamente, de estar seguro contra posibles invasores externos. Como el usuario, normalmente, no tiene acceso al código fuente del programa, eso ha provocado muchas discusiones sobre la seguridad, en particular, por el hecho de no saber exactamente cuál es el contenido del código. Eso deja dudas sobre qué tareas exactamente se están realizando, cómo los programas procesan la información, o, peor todavía: la posibilidad de que datos estén siendo enviados sin consentimiento del usuario.

Es evidente que, en situaciones de esa naturaleza, las instituciones en general alrededor del mundo son dependientes y vulnerables a los proveedores de software no libre, considerando que, en su gran mayoría, son multinacionales extranjeras. No es de extrañar que situaciones de espionaje puedan pasar. En este mundo conectado globalmente y de acceso masivo a Internet, se pueden dar muchos casos sospechosos. Además, las prácticas para conseguir satisfacer los intereses comerciales, políticos y económicos



suelen sobrepasar los límites de la moralidad y decencia de la sociedad.

Con el software libre ese tipo de preocupación ya no existe. “Por su carácter abierto, dificulta la introducción de código malicioso, espía o de control remoto, debido a que el código lo revisan muchos usuarios y desarrolladores que pueden detectar posibles puertas traseras” (Mas, 2005, p. 52). Éstas son también comúnmente conocidas como *backdoors*. Cualquier usuario con conocimiento técnico puede revisar el código fuente de los programas y comprobar si efectivamente no hay ninguna puerta trasera. Con el privativo, “la solución de los errores no llega hasta que el proveedor del software puede asignar los recursos necesarios para solventar el problema y publicar la solución” (Mas, 2005, p. 48).

El acceso al código fuente de los programas también hace una gran diferencia a la hora de hacer auditorías informáticas. El software libre puede ser libremente auditado e inspeccionado por cualquier persona que tenga los conocimientos necesarios para realizar dichos procedimientos, ya que el código fuente está disponible para el usuario.

Auditar el software es la única forma de comprobar si el programa hace exactamente sólo lo que se propone hacer. Todo ese proceso es mucho más sencillo con el software libre. Hay transparencia sobre lo que hace el programa. Así, el usuario tiene la certidumbre de que no está siendo manipulado o engañado. Por otro lado, en relación con el software no libre, hay que creer en lo que dice o garantiza el proveedor del software, ya que otro modo de certificar la seguridad es más difícil, e, incluso, mucho más caro. En esos casos, es evidente la duda, considerando, especialmente, todo el historial de algunos de estos proveedores.

### 3.3.6 Compartir y cooperar

El software libre presenta una característica trascendente que es el hecho de poner en práctica valores éticos como compartir y cooperar. El

software libre nos aporta mucho más que el simple hecho de poder usar el software.

Otro punto a considerar es que el software libre, al ofrecer la libertad de copiar el software y poder redistribuirlo, o sea, compartirlo, está también compartiendo, directa o indirectamente, el conocimiento y contribuyendo a su difusión. Indirectamente, por la difusión del conocimiento a través de la distribución de las copias de software. Directamente, por el acceso al código fuente del programa.

El trabajo cooperativo, sobre todo en proyectos, se aplica especialmente en las empresas y en muchos otros sectores de la sociedad. Ha demostrado que es productivo y eficiente, porque todos los participantes pueden aportar ideas y conocimientos a las actividades desarrolladas. Y así es con el software libre. Su desarrollo es un claro ejemplo de la importancia y eficiencia de la cooperación entre las personas.

Las personas están distribuidas alrededor del mundo y componen una comunidad virtual que, cada día o cuando pueden, se dedican a colaborar en el proyecto, en su casi totalidad de forma voluntaria. En su gran mayoría, nunca se han encontrado físicamente.

Esos colaboradores, que normalmente son muchos y algunas veces miles, dependiendo de la envergadura del proyecto, presentan los más diversos perfiles. Tienen culturas, lenguas, costumbres, ideologías, ideas, experiencias, conocimientos y formación distintos. Pero un ideal único y común a todos: producir software libre.

### **3.3.7 Combatir la brecha digital**

Las características fundamentales del software libre, asociado al hecho de que normalmente puede ser usado en ordenadores con menos recursos de hardware, incluso en los que ya son considerados obsoletos, contribuyen a

que tenga una gran dimensión social.

El alto coste de la tecnología, tanto del hardware como del software, es uno de los factores que ha creado una barrera inmensa para el acceso de los ciudadanos a las TIC en varias regiones del planeta. Son dos componentes que han contribuido mucho a profundizar la brecha digital y así, aumentar las desigualdades, cada vez mayores, entre los que pueden tener acceso y hacer uso de las nuevas tecnologías, sea entre personas o naciones.

En algunos países, es casi imposible comprar un ordenador, por lo menos, para la gran mayoría de la población. Además, es, prácticamente, inalcanzable poder adquirir todas las licencias de uso de los programas necesarios. Esta situación, muchas veces, incita a que se haga un uso irracional de copias no autorizadas de software. No es casualidad que las tasas más altas de éstas copias estén justamente concentradas en las regiones más pobres.

Las limitaciones económicas, y por consiguiente las tecnológicas, dificultan en gran medida la penetración y uso de las TIC por parte de la población de esos países, contribuyendo a aumentar las diferencias entre los países ricos y los pobres o en vías de desarrollo.

Pero, los países y sus ciudadanos tienen derecho a ser personajes activos en la Sociedad de la Información. Y una verdadera inclusión digital en esa sociedad digital solo es coherente si se lleva a cabo con el software libre. El principal capital de toda esa revolución tecnológica es el conocimiento digital. Es decir, los excluidos digitales son aquellos que no tienen acceso a ese conocimiento (Branco, 2005).

El software libre contribuye a la construcción de una sociedad más inclusiva, ya que potencia el acceso a las TIC y ofrece la posibilidad de incluir digitalmente un mayor número de personas, promoviendo así la igualdad social y de oportunidades. Es evidente que el software libre se ha convertido en un arma valiosa y una alternativa interesante en la ardua tarea de disminuir, o incluso eliminar, la llamada brecha digital alrededor del mundo.

### 3.4 Desventajas del software libre

Independientemente de las ventajas y virtudes que presenta cuando se compara con el software no libre, el uso del software libre, de una manera general, no está exento de algunas desventajas.

En este apartado discutimos sobre algunas desventajas asociadas al software libre. Presentamos las más citadas, por lo general, en la literatura y que, a nuestro modo de ver, son algunos argumentos y justificaciones utilizados contra su adopción. Son los obstáculos a superar para consolidarse aún más en el mercado.

Cabe resaltar, aquí también, que la secuencia en que serán presentadas las desventajas no obedece a ningún orden de importancia. Entre éstas destacaremos: la cantidad de software disponible, programas y documentación disponibles en un pequeño número de idiomas, gastos en el proceso de migración, desconocimiento sobre el software libre, difícil instalación y configuración y necesidad de conocimientos técnicos para hacer uso del código fuente.

#### 3.4.1 Cantidad de software disponible

En determinadas áreas, sobre todo en entornos muy específicos, todavía existe poca cantidad de software libre cuando se compara con la cantidad disponible en otras categorías de software. No existen estudios o números precisos que demuestren ese hecho, pero, de algún modo, no descartamos esa posibilidad, ya que no existe el interés puramente comercial como en el caso del software privativo.

Una de las áreas más citadas en que el software libre aún no ha alcanzado una considerada penetración es la de los juegos electrónicos. En ese caso, quizás, esa sea una razón decisiva por la que sistemas operativos libres

aún no tengan una mayor penetración en el mercado doméstico, considerando que este tipo de programas de entretenimiento atrae un gran número de usuarios, sobre todo jóvenes.

Curiosamente, por otro lado, algunas veces ocurre lo contrario, es decir, para una determinada área hay diversas opciones. Entendemos que es una gran ventaja que el usuario tenga muchas posibilidades a su disposición, pero lo que también puede pasar es que quede confuso ante tantas alternativas. La gran cantidad de distribuciones GNU/Linux, por ejemplo, ilustra muy bien ese problema. Quizás tantas opciones causen alguna confusión a los usuarios a la hora de elegir una distribución.

Para muchos críticos, las discusiones en el mundo del software libre alrededor de cuál es la mejor distribución acaban perjudicando y debilitando las acciones en pro de la consolidación del GNU/Linux como una alternativa real, eficaz y poderosa a los sistemas operativos no libres, en especial, el Windows.

### **3.4.2 Programas y documentación disponibles en pocos idiomas**

Otra desventaja relacionada con el software libre es que una cierta cantidad de programas libres sólo está disponible exclusivamente en un determinado idioma, sobre todo en inglés, o en un pequeño número de ellos. También pasa que hay programas que están traducidos a más lenguas, pero sus manuales y documentación no. O, en algunos casos, es excesivamente técnica.

Aunque para situaciones de esa naturaleza, existe la posibilidad de traducir los programas y toda la documentación a cualquier idioma que se desee, ya que sus códigos fuentes son disponibles, eso continúa siendo un problema.

### 3.4.3 Gastos en el proceso de migración

Cambiar a software libre también requiere gastos en adaptaciones de sistemas y entrenamiento del personal y, dependiendo de las circunstancias, esos gastos pueden ser altos, sobre todo, en el periodo inicial de la migración.

El proceso de migración se complica, todavía más, cuando se acaba optando por la solución libre sin hacer una sólida planificación. El resultado final puede no ser el esperado y afectar significativamente todo ese proceso. Todo cambio, sobre todo, cuando implica dinero, y especialmente si éste es público, necesita ser analizado rigurosamente para orientar bien la inversión y lograr lo esperado. Como no hay una solución universal, es importante examinar cada caso.

El software libre, normalmente, se distribuye gratuitamente o por precios considerablemente menores al privativo. Eso hace que muchos usuarios piensen que el proceso de migración hacia esta categoría de software es sencillo y muy económico. Pero, esa no es la realidad. Todo proceso de migración conlleva un gran esfuerzo, tiempo y, principalmente, inversiones que pueden parecer inviables.

Además, todo cambio requiere adaptaciones que, muchas veces, no son bien aceptadas por los usuarios, ya que estaban acostumbrados a usar determinadas aplicaciones. Ahora han de aprender a trabajar, descubrir y adaptar sus actividades al nuevo software. Existen también las incertidumbres y las dudas sobre el software libre. Así, algunos prefieren continuar utilizando los programas no libres, especialmente los privativos, porque ya los están usando.

### 3.4.4 Desconocimiento sobre el software libre

Existe todavía un desconocimiento por parte de la mayoría de usuarios

de ordenadores de qué es el software libre, su filosofía y sus ventajas. Por consiguiente, tampoco saben qué programas pertenecen a esta clase de software. Es evidente que todavía hay una gran ignorancia sobre el tema, que dificulta la comprensión de las ventajas que aporta el software libre.

Los usuarios aún no son conscientes de que existen otras alternativas a los programas ya consolidados de otras categorías de software. Normalmente, desconocen, no buscan o no saben dónde encontrar una solución libre que atienda a sus intereses.

Todo eso es una de las razones que aún contribuye a que muchos usuarios sepan poco o casi nada sobre el software libre y sus atractivos. Aunque reconocemos que ha habido mejoras en ese aspecto, especialmente, en la última década.

Esa desinformación es fruto, sobre todo, de la falta de propaganda o de divulgación del software libre, ya que no tiene los mismos recursos e intereses que las empresas de desarrollo de software no libre. Habitualmente no hay un trabajo de *marketing* o divulgación de los programas libres como lo hay con los no libres, especialmente los privativos, que, normalmente, pertenecen a grandes empresas que invierten muchos recursos económicos para su divulgación. El secreto del éxito de un producto está, muchas veces, relacionado directamente con su publicidad y, en esos casos, la competencia entre el software libre y el no libre es desigual.

Internet es la gran responsable del avance del uso del software libre en las más diversas áreas. A medida que crece a diario el número de usuarios conectados a esta red, la tendencia es que cada vez más usuarios tendrán acceso a la información relacionada con esta modalidad de software. Por lo menos es lo que se espera.

Además, la adhesión de varias instituciones, públicas y privadas, de grandes empresas del área de informática y de varios gobiernos alrededor del mundo impulsan su divulgación. Pero, normalmente, son los propios usuarios los que son grandes divulgadores. Esa tendencia se está consolidando, aunque parezca un proceso lento.

### 3.4.5 Difícil instalación y configuración

Un punto muy criticado relacionado con el software libre, por lo general, es la dificultad de instalar y configurar determinados programas, sobre todo, algunas distribuciones del sistema operativo GNU/Linux.

En realidad, en algunas distribuciones GNU/Linux, el proceso de instalación es prácticamente todo automático y controlado por el propio sistema, no requiriendo mucha intervención del usuario. Mientras que, en otros casos, ese proceso suele exigir un poco más la participación y, por consiguiente, la experiencia del usuario. Sin embargo, hay que remarcar que, por lo general, independientemente del sistema operativo, libre o no libre, la gran mayoría de los usuarios prefiere no arriesgarse y acaba por solicitar ayuda a amigos o pagar para que un técnico la realice.

También es común, en algunos casos, que el sistema libre no detecte automáticamente algunos dispositivos de hardware. Así, necesitan ser configurados manualmente. Esos casos suelen exigir un poco de experiencia por parte del usuario. Es verdad que, cada vez más, tiende a simplificarse. En otros casos, algunos componentes de hardware o, incluso, periféricos no funcionan correctamente.

De igual modo, hay situaciones en que además de no proporcionar los controladores, o *drivers*, de sus dispositivos, los fabricantes tampoco facilitan el acceso a las especificaciones del hardware para que los desarrolladores puedan crearlos. En los casos en que los proporcionan, también es común que los controladores basados en software libre lleguen al mercado con algo de retraso.

No es difícil imaginar las estrategias de presión utilizadas por determinadas empresas para consolidar sus productos en el mercado de software. Tanto es así que, incluso, en algunas ocasiones, algunos proveedores de software han sido condenados por prácticas comerciales abusivas, principalmente, en Europa. Como ejemplo, la propia Microsoft.



### 3.4.6 Necesidad de conocimientos técnicos para hacer uso del código fuente

Uno de los problemas, bastante común, relacionado con el software libre es que, a pesar de tener acceso al código fuente, en la práctica, poquísimos usuarios tienen los conocimientos técnicos para hacer las debidas adaptaciones en los programas para atender a sus necesidades. De este modo, esa ventaja de poder acceder al código termina sin efectividad.

Eso sucede, prácticamente, en todas las áreas. Por lo general, el usuario de software libre acaba no sacando provecho de esa gran ventaja. Sin embargo, como discutimos anteriormente, también es verdad que hay otras formas de solucionar este problema. De este modo, es posible sacar un gran provecho de la ventaja de acceso al código fuente del programa.

Hay que remarcar que, para algunas de esas desventajas, podríamos argumentar o, incluso, citar algunas soluciones o tendencias que, quizás, podrían anularlas o compensarlas. En todo caso, esos aspectos, de alguna forma, son reales y no pueden ser ignorados, porque en mayor o menor medida, contribuyen a dificultar la adhesión al software libre.

## 3.5 Resistencias a usar el software libre

A pesar del gran número de ventajas que el software libre aporta, aun así, existe resistencia a su uso. Igual que hay militantes, también hay resistentes. De hecho, no es fácil vencer inercias. En mayor o menor medida, se trata de desafíos y obstáculos que tiene que superar para consolidarse, todavía más, en el mercado de software.

Los prejuicios o resistencias a adherirse al software libre se deben a factores como: el software privativo es más aceptado y utilizado; el usuario en general ya tiene experiencia con el software privativo; la conocida técnica FUD, del inglés *fear, uncertainty, doubt*, que en español significa miedo,

desconocimiento, duda; y otros intereses.

En este apartado discutimos sobre las resistencias al software libre que dificultan su penetración en los más diversos sectores. Presentamos algunas de las principales que, por lo general, se encuentran en la literatura.

### **3.5.1 Mayor aceptación del software privativo**

Es innegable que el pleno dominio del software no libre, especialmente del privativo, prácticamente en todos los sectores de nuestra sociedad favorece y refuerza la idea de que no vale la pena adherirse al software libre. Aún hoy día, hay quien defiende que se debe continuar utilizando los programas privativos porque son los más aceptados y utilizados, por regla general, y eso es valioso para su futuro, especialmente en el mercado de trabajo.

En ese sentido, es interesante destacar que las principales aplicaciones necesarias para el usuario en general ya disponen de una alternativa libre al software no libre. Y, por lo tanto, todo es una cuestión de necesidad e interés en cambiar esa realidad.

### **3.5.2 Experiencia con el software privativo**

Una de las resistencias naturales al cambio del software utilizado en cualquier sector de la sociedad está relacionada con el hecho de que los usuarios de ese entorno ya tienen experiencia en determinados programas no libres, sobre todo en los privativos. De este modo, rechazan la migración porque, en principio, no lo ven como una necesidad o algo realmente esencial. Imaginemos cuántos años de uso de aplicaciones no libres tiene el usuario de los ordenadores en general.

El cambio exigirá tiempo para estudiar, practicar, dominar y alcanzar el mismo nivel con los programas libres. Además, las dudas naturales: ¿qué hago o que pasará con mis archivos? y ¿lo perderé todo? Más preocupantes, todavía, son los que ven en esas iniciativas una imposición que les obligan a utilizar determinadas herramientas que ellos no están seguros de que sean las más adecuadas para sus actividades.

### 3.5.3 *Fear, Uncertainty and Doubt (FUD)*

Traduciendo al español, FUD significa miedo, desconocimiento y duda, que consiste de una técnica bastante utilizada en el área tecnológica para generar incertidumbre sobre un tema. Es innegable que todavía hay muchas dudas e incertidumbres asociadas con el software libre. Muchas de éstas son frutos de desinformación. Mientras que otras han sido manejadas astutamente y provocadas deliberadamente con la intención de desacreditarlo y así frenar el evidente avance de su propagación.

El objetivo de la estrategia FUD “consiste en el confundir al público dando información incompleta, una visión distorsionada de la realidad y dejando el resto a la imaginación del receptor del mensaje con el objetivo de crear miedo, incertidumbre y duda en el momento de tener que evaluar soluciones de la competencia” (Mas, 2005, p. 56).

Uno de los casos más emblemáticos y conocidos en el uso de esa técnica es el de los famosos documentos de *Halloween*, que tiene la empresa Microsoft como personaje central. Aunque ya hace más de dos décadas del episodio, el caso es una prueba de hasta que punto el software libre ya empezaba a traer preocupaciones para los proveedores de software, especialmente, los privativos.

En octubre de 1998 se filtraron, con pocos días de diferencia, varios documentos internos de Microsoft que describían y

analizaban la amenaza que representaba el software libre para la compañía y recomendaba algunas estrategias en la línea FUD. Microsoft a los pocos días confirmó la autenticidad de los documentos y reconoció que habían sido escritos por algunos de sus empleados, pero dijo que no representaban la posición oficial. Estos documentos se conocen con el nombre de documentos de *Halloween* ya que coincidió con esta festividad el día que el primer documento se filtró. Los documentos de *Halloween* reconocen ya en el año 1998, que el software libre había alcanzado niveles de credibilidad importantes, que tiene una calidad similar o superior al software propietario y que representa una amenaza para las ventas de Microsoft. Los documentos además constatan que, según la opinión de Microsoft, las técnicas FUD no han tenido gran efecto sobre el software libre y que la mejor estrategia es extender los estándares actuales de forma que sólo Microsoft pueda proporcionar el software que los soporte correctamente. Popularmente se conoce esta técnica como contaminación de estándares. (Mas, 2005, p. 57).

Con un arma tan poderosa como ésa en ese mundo conectado virtualmente, por supuesto que las iniciativas de desgaste del software libre siempre destacarán de manera especial. Los mensajes son muy influyentes y tienen un gran alcance. Y en época de las famosas *fake news*, ¿qué esperar?

Es evidente de que “para combatir el FUD y sus efectos negativos, la mejor estrategia es dar la máxima información sobre el software libre” (Mas, 2005, p. 58). El miedo, la incertidumbre y la duda es una constante en el mundo tecnológico y no hay otra forma sino afrontarlos.

### 3.5.4 Otros intereses

Quizás existan muchos «otros intereses», especialmente comerciales, por detrás de todo para que el software libre no avance y consiga su merecido

y justo espacio en los más diversos sectores de la sociedad, especialmente, en la administración pública. Un ámbito trascendental y atractivo para los más diversos intereses de los proveedores de software.

Es normal escuchar o leer que determinados organismos, normalmente de administraciones públicas, han cambiado de idea o han repensado sus políticas hacia la adhesión al software libre después de recibir las visitas de los representantes comerciales o, incluso, de los propios números uno de la sucursal o empresa de los proveedores de software privativo. No queremos aquí generalizar, lo que sería irresponsable y peligroso, además de indelicado, pero es evidente que, en algunos casos, pasan cosas extrañas o no aclaradas completamente a los ojos de los ciudadanos.

Hace casi dos décadas que Abella *et al.* (2003, p. 26), ya alertaba de que en esos casos “generalmente el acceso a los ámbitos de decisión se realiza a través de mandos intermedios que pueden filtrar o tergiversar la información por algún tipo de beneficio "personal". (Había pensado ser suave en este punto, pero he cambiado de idea; a esto se le llama Corrupción)”. ¿Alguien será capaz de imaginar que algo haya cambiado a día de hoy?

### 3.6 Los falsos mitos relacionados con el software libre

A lo largo de los años, mientras el software libre ganaba espacio en las más diversas áreas, ha ido apareciendo una serie de falsos mitos sobre el mismo, que se desvanecen ante el más mínimo análisis racional. Sin embargo, éstos han sido asimilados rápidamente, persisten y se perpetúan hasta hoy, aprovechándose del desconocimiento de los usuarios sobre este asunto.

Algunos de estos mitos han sido creados intencionadamente y son información falsa propagada, sobre todo, por proveedores de software, especialmente privativo, para quitarle credibilidad al software libre. Quizás se sientan amenazados de alguna forma por su avance. Por supuesto, hay muchos intereses por detrás de todo esto.

Otros mitos son fruto de experiencias del pasado y de la falta de conocimiento sobre la materia. Todavía hoy es común encontrar algunas interpretaciones equivocadas e imprecisas acerca del software libre. Hay muchas confusiones a su alrededor. No es difícil encontrar referencias a los falsos mitos en la literatura consultada sobre software libre

### 3.6.1 El software libre es gratis

Éste, quizás, sea el mito asociado al software libre más conocido y popular. Por lo menos en la literatura consultada, ha sido el más citado.

Como comentamos anteriormente, eso es provocado por la confusión con la expresión inglesa *free software*. Libre y gratis, las dos traducciones de la palabra polisémica *free*, no son sinónimas. Esto significa que software libre no es lo mismo que software gratuito. Sin embargo, la gran mayoría de programas de esta modalidad se suele distribuir gratuitamente, ya que se puede conseguir en Internet sin coste económico.

En realidad, la palabra libre de la expresión está relacionada con las libertades otorgadas al usuario y no tiene ninguna relación con el precio del software. Es más, como contraejemplo irrefutable para desmentir definitivamente el mito de la gratuidad del software libre, basta citar que hay diversas aplicaciones y sistemas operativos libres que se paga por su obtención.

Como hemos visto, la adquisición de un software libre puede ser o no gratis. Pensar que todo programa libre se obtiene gratuitamente es una equivocación, ya que, conforme antedicho, podemos encontrar situaciones en que es necesario pagar para obtenerlo. Pero eso no cambia su característica de libre. Según las libertades que determinan su esencia, ninguna hace referencia a la prohibición de poder cobrar por la distribución de software.

El propio Stallman (2004) también aclara esa equivocación comentando que independientemente de la forma como haya obtenido el software, pagando o gratuitamente, siempre tendrás garantizado la libertad de hacer lo que quiera con el software libre. Lo importante es la libertad y no el precio. Es decir, el énfasis y los principios básicos del software libre están en las libertades y no en el precio.

### 3.6.2 El software libre no es comercial

Otro de los mitos relacionados con el software libre más citados es que no es comercial, que no hay software libre comercial o que software comercial es sinónimo de software privativo, por lo tanto, excluyendo el libre comercial. El propio Stallman (2004) pide para que no se use el término comercial como sinónimo de no libre. De este modo, evitamos asociación y confusión del tipo de que no hay software libre comercial.

Como hemos visto, aunque en un primer momento pueda sonar paradójico, el software libre también es comercial. Siempre se puede comercializar, pero obviamente respetando las libertades otorgadas a los usuarios. No existe ninguna contradicción entre que un programa sea libre y que haya sido desarrollado con finalidad comercial.

Cabe remarcar aquí también que, según las libertades que determinan la esencia del software libre, ninguna hace referencia a la prohibición de comercializarlo. Al revés, si así lo hace, no se trata de un software libre.

Otra vez más, como contraejemplo irrefutable para desmentir definitivamente este mito, basta citar las diversas distribuciones del sistema operativo GNU/Linux que ofrecen la versión comercial de sus productos, que se trata de una versión más completa, pues ofrece una serie de servicios que no son contemplados en la versión destinada al usuario más sencillo.

Esta realidad parece ser una tendencia en el mundo del software libre, pues es cada vez más común encontrar empresas, incluso renombradas, que ofrecen sus productos basados en un modelo de software libre comercial.

### 3.6.3 Sólo existe software libre para GNU/Linux

Otro popular mito relacionado con el software libre es que sólo existe para GNU/Linux. Es decir, no hay programas libres disponibles para otras plataformas. Esto es un absoluto error.

Confrontando ese falso mito basta comentar que hay diversos programas libres que se encuentran disponibles para otras plataformas no libres, incluyendo aquí el famoso y popular sistema operativo de la Microsoft, el Windows. Como ejemplo, podemos citar uno de los más populares programas libres disponible en el mercado, el paquete ofimático LibreOffice, que tiene versiones para Windows, Mac y otras plataformas. Además de ése, tenemos el navegador de Internet Mozilla Firefox, el software educativo GCompris y tantos otros programas libres.

En otras palabras, esto significa que un programa libre utilizado en cualquier otra plataforma, incluso Windows, no pierde su característica y esencia de libre. Por esto, muchas veces, los usuarios utilizan una determinada aplicación libre en la plataforma Windows y no son conscientes de que están usando software libre. De hecho, un gran número de aplicaciones libres también están disponibles en versiones para otras plataformas, además de la GNU/Linux.



### 3.7 Algunos de los programas libres más conocidos

Actualmente existe una gran cantidad de programas libres disponibles como alternativa a los no libres, que cubren prácticamente todas nuestras necesidades en el quehacer cotidiano informático. Sin embargo, muchos usuarios lo usan, pero no son conscientes de eso. Son programas que ya están consolidados en el mercado de software, tienen buena calidad y en su gran mayoría son distribuidos gratuitamente.

Hay también para áreas más técnicas como redes, bases de datos, servidores web, lenguajes de programación, etc. De hecho, hoy en día existe un amplio espectro de programas que cubren las más diversas áreas del conocimiento humano y una amplia gama de necesidades de los usuarios. Aunque no se hayan encontrado datos precisos sobre la cantidad y utilización de programas libres.

Actualmente, muchos de los programas libres se consideran de calidad igual o hasta superior a sus correspondientes no libres. Hay muchos casos en que los programas libres son considerados de mejor calidad que los no libres y han tenido una amplia aceptación, como es, entre otros, el del navegador web Mozilla Firefox. Incluso, en algunos nichos, el software libre ya es líder absoluto de uso, así lo demuestran algunas estadísticas. Como ejemplo, podemos citar el segmento de los servidores de Internet, donde se destaca el Apache, considerado uno de los más fiables.

Para algunas funcionalidades, la calidad alcanzada por los programas libres es tan alta que han llegado a impedir la consolidación de ninguna alternativa privativa. Por ejemplo, gestores de contenidos para webs o, precisamente, plataformas de formación, como Moodle o Sakai.

De acuerdo con el Proyecto TOP500<sup>25</sup>, que se dedica a publicar estadísticas semestrales sobre las 500 supercomputadoras con mayor rendimiento del mundo, en noviembre de 2019, todas las 500 más potentes utilizaban sistemas derivados del Linux.

---

25 Más información en: <https://www.top500.org/statistics/sublist/>.

Hoy en día, innegablemente el abanderado más ilustre de los programas libres es sin duda alguna el sistema operativo GNU<sup>26</sup>/Linux<sup>27</sup>. Además de éste, otros exitosos iconos son la suite ofimática LibreOffice<sup>28</sup>, el navegador de Internet Mozilla Firefox<sup>29</sup>, el programa de edición de imágenes digitales GIMP<sup>30</sup>, el reproductor y framework multimedia VLC<sup>31</sup> y la aplicación multiplataforma Audacity<sup>32</sup>. La importancia de estos programas reside, sobre todo, en romper el monopolio del software no libre, especialmente el privativo, que había en sus respectivas áreas, consideradas estratégicas para la diseminación de las TIC.

Cabe resaltar, una vez más, que una mayoría de los programas libres están también disponibles en versiones para otras plataformas, incluso para sistemas privativos como los de la familia Windows de Microsoft. Por ejemplo, la suite ofimática LibreOffice.

En el caso del GNU/Linux, el sistema operativo libre más conocido y usado, es destacable la gran cantidad de distribuciones existentes. Unas destinadas a usuarios con poca experiencia en el mundo informático, otras para los que tienen más. También las hay generales y específicas para determinadas áreas, como la educativa. Toda esta diversidad es consecuencia directa de la libertad que el software libre otorga al usuario. Por otro lado, para éste, seleccionar la distribución que atienda a sus necesidades, muchas veces, no es una tarea sencilla.

En el sitio web de The LWN.net<sup>33</sup>, una conocida referencia sobre las distintas distribuciones GNU/Linux existentes, encontramos un listado de 577, incluyendo la descripción del proyecto y enlace para cada una. Sin embargo, es difícil tener un número exacto de todas las distribuciones. Por ejemplo, en ese sitio no hay referencias a las distro Linkat, catalana, y Linux Educacional, brasileña. De este modo, ratifica que el número de distros debe

---

26 Más información en: <http://www.gnu.org>.

27 Más información en: <http://www.linux.org>.

28 Más información en: <https://es.libreoffice.org/>.

29 Más información en: <https://www.mozilla.org/es-ES/>.

30 Más información en: <https://www.gimp.org/downloads/>.

31 Más información en: <https://www.videolan.org/vlc/index.es.html>.

32 Más información en: <https://www.audacityteam.org/>.

33 Más información en: <http://lwn.net/Distributions/>.

ser mucho mayor. Esto ocurre justamente por las características del software libre que permite crear una distribución para atender a las necesidades específicas del usuario.

Hay también diversos otros sitios web que presentan información sobre las distintas distribuciones, como comparativa de visitas, las más populares y precios que se cobran por éstas. De entre estos sitios, podemos mencionar el de DistroWatch.com<sup>34</sup>.

Cabe remarcar también que se puede encontrar en Internet: páginas, sitios, artículos, libros, revistas especializadas, catálogos o guías con listados de programas libres o de no libres con sus correspondientes libres existentes en las más diversas áreas, que sirven de base para aclaración y consulta.

Internet es sin duda alguna la mejor fuente donde se pueden encontrar programas libres a disposición del usuario. De hecho, son los repositorios de software libre existentes en la red una de las mejores referencias sobre programas libres. Éstos sirven de canal de distribución y, por consiguiente, de difusión de una gran cantidad de software de esta categoría. Uno de los más conocidos de estos repositorios es el SourceForge<sup>35</sup>. Según datos obtenidos en el propio sitio web, hay, actualmente, 4 millones de descargas por día y su directorio de software comercial enumera más de 50.000 títulos de software. Obviamente no es el único, pero es de los más destacados y visitados. Esto ejemplifica y ratifica la gran cantidad de software libre que existe en la actualidad.

Si realmente uno está interesado o necesita encontrar alguna información sobre determinado programa o alguno que atienda a sus necesidades, la forma más práctica hoy día es, sin ninguna duda, buscar en Internet. No es casualidad que ésta ha sido a lo largo de los años una de las grandes responsables de la difusión y propagación del software libre, además de contribuir, y mucho, a su desarrollo. Ha sido una pieza fundamental en todo este proceso. Por otro lado, también es verdad que esta relación beneficiosa ha sido en los dos sentidos, una vez que el software libre tiene

---

34 Más información en: <https://distrowatch.com/?language=ES>.

35 Más información en: <https://sourceforge.net/>.

una importancia esencial en el avance de Internet.

### 3.8 Resumen del capítulo

El software nació al mismo tiempo que los primeros ordenadores, siendo todo libre. Se compartía y se trataba del mismo modo que el conocimiento científico. De hecho, no se hablaba de «software libre» porque nadie había planteado que pudiera ser de otro modo.

Es innegable que los programas no libres, principalmente, los privativos, predominan en los más diversos entornos, pero el avance del software libre es incuestionable. No hay estadísticas de uso contrastadas, pero sí indicadores significativos y la tendencia es clara. Especialmente el software privativo aún es mayoritario porque hace más tiempo que se mantiene en el mundo de la Informática y, además, sus esforzados intentos por mantenerse se ven envueltos por un complejo entramado de intereses. Sus proveedores son conscientes de que cualquier esfuerzo para mantener su posición dominante es una sabia inversión para el futuro.

El software libre, por otro lado, todavía es un desconocido para la mayoría de usuarios, hasta el punto que hay personas que lo usan sin saberlo. No son conscientes de los beneficios que les puede aportar. Mejorar el conocimiento sobre esta materia es la mejor forma de darle impulso.

Alrededor de los ideales que definen el software libre, ha surgido un movimiento que está alterando todo el sector, aunque los intereses económicos reinantes se oponen a él con estrategias como: sembrar miedo, incertidumbre y dudas, conocido por las iniciales inglesas FUD, pervertir los principios del movimiento con iniciativas parecidas pero no coincidentes, aprovecharse del desconocimiento de la mayoría o intentar hacer que el software sea patentable. Como éxitos de estas estrategias, se han extendido algunos falsos mitos sobre este software. Entre los más conocidos figuran: que se trata de software gratuito, que no es comercial y que sólo funciona sobre

GNU/Linux.

Sin embargo, actualmente, cada vez hay más adhesiones al software libre, tanto en las administraciones públicas, como en el ámbito empresarial y en la educación. Su avance se puede predecir sin miedo a equivocarse.

El software libre es muy provechoso porque aporta una serie de ventajas en relación a las otras categorías de software. De entre éstas, destacamos: reducir costes al dejar de pagar licencias de uso de software privativo; ser independientes del proveedor, lo que facilita el soporte técnico y la adaptación de programas, especialmente a las lenguas minoritarias o minorizadas; extender la vida útil de los ordenadores, ya que los programas libres no necesitan tantos recursos de hardware; beneficiarse de la interoperabilidad; usar aplicaciones de tan buena calidad, o incluso mejor, que las no libres; tener estabilidad y seguridad en el uso de programas; poner en práctica valores éticos como compartir y cooperar; evitar o disminuir el uso de copias no autorizadas de software, y combatir la brecha digital.

Por otro lado, al software libre se le han atribuidos desventajas que, con el paso del tiempo, se han desvanecido, bien porque respondían a problemas de maduración del modelo, bien porque eran falsas. Pero hay limitaciones que conviene recordar, para no generar falsas expectativas, cuya frustración dejaría un rastro de usuarios insatisfechos difíciles de recuperar. Algunas de las más conocidas son: la cantidad de software disponible, que en determinadas áreas es un tanto limitada, mientras que en otras, el problema es el contrario: el exceso de alternativas provoca confusión; los programas libres que sólo tienen versión en inglés o en muy pocos idiomas, o su documentación no está traducida; los costes del proceso de migración, que pueden ser inalcanzables para algunos usuarios; la dificultad de instalación y configuración de algunos programas libres; la falta de soporte; y la necesidad de conocimientos técnicos para hacer uso del código fuente.

A pesar de las ventajas que el software libre aporta, aún perviven resistencias. De entre éstas, resaltamos: la mayor aceptación del software privativo; la experiencia de los usuarios con este software; los FUD, y otros intereses, como casos en que algunos cargos de la administración han tenido

sospechosos cambios de postura después de la visita de altos directivos de grandes corporaciones de software privativo.

Pese a la inercia, las naturales resistencias al cambio y desconfianzas respecto aquello que es menos conocido, los argumentos a favor de su uso van mucho más allá del ahorro económico y la tendencia en este sentido es clara. Por otro lado, aún falta mucho por hacer en cuanto a la promoción y difusión de programas libres. Esto es reflejo, también, de la inexistencia de políticas concretas por parte de los organismos gubernamentales.

# REFERENCIAS

- Abella, A., Sánchez, J., Santos, R. y Segovia, M. A. (2003). *Libro blanco del software libre en España 2003*. <http://docplayer.es/5545365-Libro-blanco-del-software-libre-en-espana-2003.html>.
- Adell, J. y Bernabé-Muñoz, I. (2007). Software libre en educación. En J. Cabero (Coord.), *Tecnología educativa* (pp. 173-194). Madrid: McGraw-Hill.
- Bain, M., Gallego, M., Martínez, M. y Rius, J. (2007a). *Aspectos legales y de explotación del software libre (Parte I)*. Barcelona: Fundació per a la Universitat Oberta de Catalunya. <https://openlibra.com/es/book/aspectos-legales-y-de-explotacion-del-software-libre-parte-i>.
- Bain, M., Gallego, M., Martínez, M. y Rius, J. (2007b). *Aspectos legales y de explotación del software libre (Parte II)*. Barcelona: Fundació per a la Universitat Oberta de Catalunya. <https://openlibra.com/es/book/download/aspectos-legales-y-de-explotacion-del-software-libre-parte-ii>.
- Baldé, C. P., Forti, V., Gray, V., Kuehr, R. y Stegmann, P. (2017). *The Global E-waste Monitor 2017: Quantities, Flows and Resources*. United Nations. [https://www.researchgate.net/publication/21797215\\_The\\_Global\\_E-waste\\_Monitor\\_2017\\_Quantities\\_Flows\\_and\\_Resources](https://www.researchgate.net/publication/21797215_The_Global_E-waste_Monitor_2017_Quantities_Flows_and_Resources).

- Branco, M. d'E. (2005). Software livre. En A. Ambrosi, V. Peugeot y D. Pimienta (Coords.), *Enjeux de mots. Regards multiculturels sur les sociétés de l'information* (pp. 573–596). Caen, France: C&F Éditions.
- Business Software Alliance. (2018). *BSA Global Software Survey. Software Management: Security Imperative, Business Opportunity*. Washington: BSA. [https://gss.bsa.org/wp-content/uploads/2018/05/2018\\_BSA\\_GSS\\_Report\\_en.pdf](https://gss.bsa.org/wp-content/uploads/2018/05/2018_BSA_GSS_Report_en.pdf).
- Cardoso, R. S. (2015). Privacidade, Software Livre e a era pós-PC. *Revista da Sociedade Brasileira de Computação*, (27), 44-49. [https://www.sbc.org.br/images/flippingbook/computacaobrasil/computa\\_27/02-2015\\_18.06.pdf](https://www.sbc.org.br/images/flippingbook/computacaobrasil/computa_27/02-2015_18.06.pdf).
- CENATIC. (2008). Centro Nacional de Referencia de Aplicación de las TIC basadas en fuentes abiertas. *Guía básica del software de fuentes abiertas*. Almendralejo, Badajoz, España: CENATIC.
- Feres, M. V. C., Oliveira, J. V. de y Gonçalves, D. D. (2017). Robin Hood às avessas: software, pirataria e direito autoral. *Revista Direito GV*, 13(1), 69-94. <http://www.scielo.br/pdf/rdgv/v13n1/1808-2432-rdgv-13-01-0069.pdf>.
- Free Software Foundation. (2019a). *Categorías de software libre y software que no es libre*. <https://www.gnu.org/philosophy/categories.es.html>.
- Free Software Foundation. (2019b). *¿Qué es el copyleft?* <https://www.gnu.org/licenses/copyleft.es.html>.
- Free Software Foundation. (2020a). *Preguntas frecuentes acerca de las licencias de GNU*. <https://www.gnu.org/licenses/gpl-faq.es.html#WhyUseGPL>.
- Free Software Foundation. (2020b). *Vender software libre*. <http://www.gnu.org/philosophy/selling.es.html>.
- González, J. M., Rodríguez, E., Rodríguez, N., Cagide, M. C., Fillat, N., Cueva, J. de la, Malina, T. y Moglen, E. (2006). *Copyleft. Manual de uso*. Madrid: Traficantes de sueños.



- González, J. M., Seoane, J. y Robles, G. (2007). *Introducción al software libre*. <http://softlibre.unizar.es/manuales/softwarelibre/sobre.pdf>.
- Mas, J. (2003). El software libre y las lenguas minoritarias - una oportunidad impagable. *Novática*, (163), 36-39. <https://dialnet.unirioja.es/servlet/articulo?codigo=2063750>.
- Mas, J. (2005). *Software libre: técnicamente viable, económicamente sostenible y socialmente justo*. Barcelona: Infonomía.
- Norton, P. (1996). *Introdução à Informática*. São Paulo: Makron Books.
- Plágaro, J. y Palacios, J. (Eds.). (2007). *Diccionario LID de Tecnologías de información y comunicación*. Madrid: LID Editorial.
- Roca, M. y Castells, M. (2006). *El software libre en Catalunya y en España. Informe de investigación*. <https://programarilliure.files.wordpress.com/2007/11/uoc-meritxell-roca-sl-en-cat-y-espana.pdf>.
- Silveira, S. A. da. (2015). Dez teses sobre software livre no Brasil Pós-Snowden. *Revista da Sociedade Brasileira de Computação*, (27), 56-63. [https://www.sbc.org.br/images/flippingbook/computacaobrasil/computa\\_27/02-2015\\_18.06.pdf](https://www.sbc.org.br/images/flippingbook/computacaobrasil/computa_27/02-2015_18.06.pdf).
- Stallman, R. M. (2002). *Free software, free society: selected essays of Richard M. Stallman*. Boston, MA, USA: GNU Press.
- Stallman, R. M. (2004). *Software libre para una sociedad libre*. Madrid: Traficantes de sueños.
- Stallman, R. M. (2010). *Por qué el «código abierto» pierde de vista lo esencial del software libre*. <https://www.gnu.org/philosophy/open-source-misses-the-point.es.html>.
- Tong, T. W. (2004). *Softwares livre/de fonte aberta. Educação*. Nova Delhi: Reed Elsevier India Private Limited.