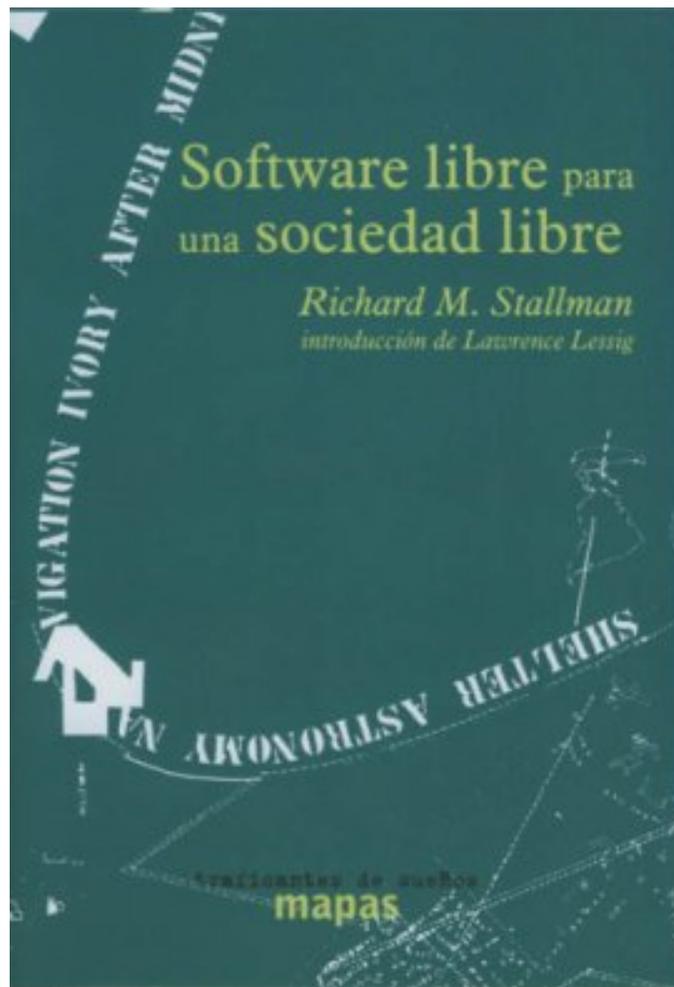


Software libre para una sociedad libre

Richard M. Stallman
Introducción de Lawrence Lessig



Diciembre 2004
Versión 1.0



Desde sus inicios **Traficantes de Sueños** ha apostado por licencias de publicación que permiten compartir, como las Creative Commons, por eso sus libros se pueden copiar, distribuir, comunicar públicamente y descargar desde su web. Entendemos que el conocimiento y las expresiones artísticas se producen a partir de elementos previos y contemporáneos, gracias a las redes difusas en las que participamos. Están hechas de retazos, de mezclas, de experiencias colectivas; cada persona las recompone de una forma original, pero no se puede atribuir su propiedad total y excluir a otros de su uso o replicación.

Sin embargo, «cultura libre» no es sinónimo de «cultura gratis». Producir un libro conlleva costes de derechos de autor, traducción, edición, corrección, maquetación, diseño e impresión. Tú puedes colaborar haciendo una donación al proyecto editorial; con ello estarás contribuyendo a la liberación de contenidos.

Puedes hacer una **donación**
(si estás fuera de España a través de **PayPal**),
suscribirte a la editorial
o escribirnos un **mail**

Software libre para una sociedad libre

Richard M. Stallman

Título original: *Free Software, Free Society: Selected Essays of Richard M. Stallman* (GNU Press, 2002)

Primera edición en castellano (en papel): Noviembre 2004

Traducción principal: Jaron Rowan, Diego Sanz Paratcha y Laura Trinidad

Edición: Traficantes de Sueños

c/ Hortaleza 19, 1º Dcha.

28004 Madrid. Tlfno: +34 1 5320928

<http://traficantes.net>

© Copyright 2004 de los artículos de este libro, Richard M. Stallman

© Copyright 2004 de la Introducción, Lawrence Lessig

© Copyright 2004 de la Edición, Traficantes de Sueños

Se permite la copia, ya sea de uno o más artículos completos de esta obra o del conjunto de la edición, en cualquier formato, mecánico o digital, siempre y cuando no se modifique el contenido de los textos, se respete su autoría y esta nota se mantenga.

ISBN: 84-933555-1-8

Depósito Legal: M-44298-2004

Edición digital a cargo de: Miquel Vidal <miquel@barrapunto.com>. Esta edición electrónica se ha realizado íntegramente con software libre, mediante el procesador \LaTeX 2 ϵ , GNU Emacs y AUCT \E X.

Traficantes de Sueños

Traficantes de Sueños no es una casa editorial, ni siquiera una editorial independiente que contempla la publicación de una colección variable de textos críticos. Es, por el contrario, un proyecto, en el sentido estricto de «apuesta», que se dirige a cartografiar las líneas constituyentes de otros órdenes de vida. La construcción teórica y práctica de la caja de herramientas que, con palabras propias, puede componer el ciclo de luchas de las próximas décadas.

Sin complacencias con la arcaica sacralidad de la cultura, sin concesiones para con los narcisismos del genio literario, sin lealtad alguna a los usurpadores del saber, TdS adopta sin ambagajes la libertad de acceso al conocimiento. Queda, por tanto, permitida y abierta la reproducción total o parcial de los textos publicados, en cualquier formato imaginable, salvo por explícita voluntad del autor o de la autora y sólo en el caso de las ediciones con ánimo de lucro.

Omnia sunt communia!

Índice

Acerca de la presente edición	9
INTRODUCCIÓN: Lawrence Lessig	11
I El proyecto GNU y el software libre	15
1. El Proyecto GNU	16
La primera comunidad que comparte software	16
EL colapso de la comunidad	17
Una elección moral radical	18
Libre en su acepción de libertad	19
El software GNU y el Sistema GNU	20
Los inicios del proyecto	20
Los primeros pasos	20
GNU Emacs	21
¿Un programa es libre para cualquier usuario?	22
El Copyleft y la GNU GPL	22
La Free Software Foundation	23
Los servicios relacionados con el software libre	24
Los objetivos técnicos	24
La donación de ordenadores	25
La lista de tareas de GNU	25
La librería GNU GPL	26
¿Un reto personal?	27
Acontecimientos inesperados	27
El GNU Hurd	28
Alix	28
Linux y GNU/Linux	28
Los retos futuros	29
Hardware secreto	29
Librerías no libres	29
Patentes de software	31

Documentación libre	31
Es necesario hablar de libertad	32
«Open Source» (código fuente abierto)	33
¡Inténtalo!	33
2. El Manifiesto GNU	35
¿Qué es GNU? ¡Gnu No es Unix!	35
Por qué debo escribir GNU	36
Por qué GNU será compatible con Unix	36
Cómo estará disponible GNU	37
Por qué quieren cooperar muchos otros programadores	37
Cómo puedes contribuir	37
Por qué se beneficiarán todos los usuarios de ordenadores	38
3. La definición de software libre	45
4. Por qué el software no debe tener propietarios	48
Insultos	49
Exageración	49
La ley	49
Derecho natural	50
Economía	51
5. ¿Qué encierra un nombre?	54
6. Por qué «software libre» es mejor que «open source»	57
Relación entre el movimiento del software libre y el movimiento «open source»	57
Comparación de los dos términos	58
Ambigüedad	58
Miedo a la libertad	59
¿Podría ayudar una marca registrada?	60
Malentendidos del «open source»	61
7. Cómo promover el software libre si trabajas en la Universidad	63
8. Vender software libre	66
¿Puede perjudicar un precio de distribución más alto a algunos usuarios?	67
¿Puede desalentar un precio de distribución más alto el uso de software libre?	67
La expresión «vender software» también puede ser confusa	67
Altos o bajos precios y la GNU GPL	68
9. El software libre necesita documentación libre	69
10. La canción del software libre	72

II	Copyright, copyleft, patentes	73
11.	El derecho a leer	74
	Nota del autor	76
	Referencias	77
12.	Malinterpretar el copyright: una sucesión de errores	78
	El copyright en la Constitución de los Estados Unidos	78
	El «contrato» del copyright	79
	El primer error: «equilibrar la balanza»	79
	¿Qué se contraequilibra?	80
	Mejor concesión que «equilibrio»	81
	El segundo error: maximizar la producción	81
	La retórica de la maximización	82
	El tercer error: maximizar el poder de los editores	83
	Resultados de los tres errores	83
	Encontrar el contrato adecuado	86
	Una nota personal	88
13.	La ciencia debe desechar el copyright	90
14.	¿Qué es el copyleft?	92
15.	Copyleft: idealismo pragmático	95
16.	El peligro de las patentes de software	98
	Evitar la patente	102
	Obtener la licencia de la patente	104
	Revocar la patente en un juicio	107
III	Libertad, sociedad y software	117
17.	¿Puedes confiar en tu ordenador?	118
	Postscriptum	121
18.	Por qué el software debe ser libre	122
	Introducción	122
	Cómo los propietarios justifican su poder	123
	El argumento en contra de la propiedad del software	123
	El perjuicio ocasionado por obstaculizar el software	125
	Obstaculizar el uso de programas	126
	La cohesión social dañada	127
	Obstruir la adaptación personalizada de programas	128
	Obstaculizar el desarrollo del software	130

No importa cómo se restringe el acto de compartir	131
El software debería ser libre	131
Por qué la gente desarrollará software	131
Programar es divertido	132
Financiar el software libre	133
¿Qué deben los usuarios a los desarrolladores?	134
¿Qué es la productividad del software?	134
¿Es inevitable la competencia?	135
«¿Por qué no nos vamos a Rusia?»	136
La cuestión de las premisas	137
Conclusión	137
19. Copyright y globalización en la era de las redes informáticas	139
La historia del copyright	140
Globalización	146
Repensar el copyright	147
Turno de preguntas	152
20. Software libre: libertad y cooperación	162
Introducción	162
Software libre: libertad y cooperación	163
Turno de preguntas	190
21. Algunas palabras y frases confusas que vale la pena evitar	198
Comercial	198
Contenido	198
Creador	199
Freeware	199
Gestión de derechos digitales (DRM)	200
Licencia de tipo BSD	200
Piratería	201
Propiedad intelectual	201
Protección	202
RAND (razonable y no discriminatoria)	202
Robo	203
Software gratuito	203
Software regalado	204
Vender software	204
IV Licencias	205
A. Licencia Pública General GNU	206
Preámbulo	206

Términos y condiciones para la copia, distribución y modificación de la Licencia Pública General de GNU	207
Apéndice. Cómo aplicar estos términos a sus nuevos programas.	212
B. Licencia Pública General Menor	214
Preámbulo	214
Términos y condiciones para la copia, distribución y modificación	216
Cómo aplicar estos términos a sus nuevas bibliotecas	223
C. Licencia de Documentación Libre GNU	225
Preámbulo	225
Aplicabilidad y definiciones	226
Copia literal	227
Copia en cantidades masivas	227
Modificaciones	228
Combinar documentos	230
Colecciones de documentos	230
Combinación con trabajos independientes	230
Traducción	231
Nulidad	231
Futuras revisiones de esta licencia	231
Addenda	232

La definición de software libre¹

Conservamos esta definición de software libre para expresar claramente el verdadero significado de los programas de software libre.

El «software libre» es una cuestión de libertad, no de precio. Para comprender este concepto, debemos pensar en la acepción de libre como en «libertad de expresión» y no como en «barra libre de cerveza».

Con software libre nos referimos a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar 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 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. De modo que deberías ser libre de redistribuir copias con o sin modificaciones, de forma gratuita o cobrando por su distribución, a cualquiera y en cualquier lugar. Gozar de esta libertad significa, entre otras cosas, no tener que pedir permiso ni pagar para ello.

Asimismo, deberías ser libre para introducir modificaciones y utilizarlas de forma privada, ya sea en tu trabajo o en tu tiempo libre, sin siquiera tener que mencionar su existencia. Si decidieras publicar estos cambios, no deberías estar obligado a notificárselo a ninguna persona ni de ninguna forma en particular.

La libertad para utilizar un programa significa que 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.

¹Escrito originalmente en 1996

La libertad para redistribuir copias supone incluir las formas binarias o ejecutables del programa y el código fuente tanto de las versiones modificadas como de las originales —la distribución de programas en formato ejecutable es necesaria para su adecuada instalación en sistemas operativos libres. No pasa nada si no se puede producir una forma ejecutable o binaria —dado que no todos los lenguajes pueden soportarlo—, pero todos debemos tener la libertad para redistribuir tales formas si se encuentra el modo de hacerlo.

Para que las libertades 2 y 4 —la libertad para hacer cambios y para publicar las versiones mejoradas— adquieran significado, debemos disponer del código fuente del programa. Por consiguiente, la accesibilidad del código fuente es una condición necesaria para el software libre.

Para materializar estas libertades, deberán ser irrevocables siempre que no cometamos ningún error; si el desarrollador del software pudiera revocar la licencia sin motivo, ese software dejaría de ser libre.

Sin embargo, ciertas normas sobre la distribución de software libre nos parecen aceptables siempre que no planteen un conflicto con las libertades centrales. Por ejemplo, el copyleft, *grosso modo*, es la norma que establece que, al redistribuir el programa, no pueden añadirse restricciones que nieguen a los demás sus libertades centrales. Esta norma no viola dichas libertades, sino que las protege.

De modo que puedes pagar o no por obtener copias de software libre, pero independientemente de la manera en que las obtengas, siempre tendrás libertad para copiar, modificar e incluso vender estas copias.

El software libre no significa que sea «no comercial». Cualquier programa libre estará disponible para su uso, desarrollo y distribución comercial. El desarrollo comercial del software libre ha dejado de ser excepcional y de hecho ese software libre comercial es muy importante.

Las normas sobre el empaquetamiento de una versión modificada son perfectamente aceptables siempre que no restrinjan efectivamente tu libertad para publicar versiones modificadas. Por la misma razón, serán igualmente aceptables aquellas normas que establezcan que «si distribuyo el programa de esta forma, deberás distribuirlo de la misma manera» —cabe destacar que esta norma te permite decidir si publicar o no el programa. También admitimos la posibilidad de que una licencia exija enviar una copia modificada y distribuida de un programa a su desarrollador original.

En el proyecto GNU, utilizamos el «copyleft» para proteger legalmente estas libertades. Pero también existe software libre sin copyleft. Creemos que hay razones de peso para recurrir al copyleft, pero si tu programa, software libre, carece de él, todavía tendremos la opción de seguir utilizándolo.

A veces la normativa gubernamental de control de las exportaciones y las sanciones comerciales pueden constreñir tu libertad para distribuir copias a nivel internacional. Los desarrolladores de software no tienen el poder para eliminar o invalidar estas restricciones, pero lo que sí pueden y deben hacer es negarse a imponer estas condiciones de uso al programa. De este modo, las restricciones no afectarán a las actividades y a los individuos fuera de la jurisdicción de estos gobiernos.

Cuando hablamos de software libre, es preferible evitar expresiones como «regalar» o «gratis», porque entonces caeremos en el error de interpretarlo como una mera cuestión de precio y no de libertad. Términos de uso frecuente como el de «piratería» encarnan opiniones que esperamos no compartas. Véase el apartado de «Palabras que conviene evitar» para una discusión sobre estos términos. Tenemos disponible también una lista de traducciones de «software libre» en distintos idiomas.

Por último, señalaremos que los criterios descritos para definir el software libre requieren una profunda reflexión antes de interpretarlos. Para decidir si una licencia de software específica puede calificarse de licencia de software libre, nos basaremos en dichos criterios y así determinaremos si se ajusta al espíritu y a la terminología precisa. Si una licencia incluye restricciones desmedidas, la rechazamos aun cuando nunca predijimos esta cuestión al establecer nuestros criterios. En ocasiones, ciertas condiciones en una licencia pueden plantear un problema que requiera un análisis exhaustivo, lo que significa incluso debatir el tema con un abogado, antes de decidir si dichas condiciones son aceptables. Cuando llegamos a una solución sobre un problema nuevo, a menudo actualizamos nuestros criterios para hacer más fácil la consideración de que licencias están cualificadas y cuáles no.

Si estás interesado en cualificar una licencia específica como licencia de software libre, visita <http://www.gnu.org/licenses/license-list.html>. Si la licencia no aparece en la lista, envíanos un correo electrónico a licensing@gnu.org.

Capítulo 4

Por qué el software no debe tener propietarios¹

Las tecnologías digitales de la información ayudan al mundo haciendo que sea más fácil copiar y modificar información. Los ordenadores prometen hacer esto de forma más sencilla para todos.

No todo el mundo quiere que esto sea más fácil. El sistema de copyright permite que los programas de software tengan «propietarios», la mayor parte de los cuales pretende privar al resto del mundo del beneficio potencial del software. Los propietarios desearían ser los únicos que pueden copiar y modificar el software que usamos.

El sistema de copyright se desarrolló con la imprenta —una tecnología usada para la producción masiva de copias. El copyright se ajustaba bien a esta tecnología puesto que restringía sólo a los productores de copias en masa. No privaba de libertad a los lectores de libros. Un lector cualquiera, que no poseyera una imprenta, sólo podía copiar libros con pluma y tinta, y a pocos lectores se les ponía un pleito por ello.

Las tecnologías digitales son más flexibles que la imprenta: cuando la información adopta forma digital, puedes copiarla fácilmente para compartirla con otros. Es precisamente esta flexibilidad la que se ajusta mal a un sistema como el del copyright. Esa es la razón del incremento de medidas odiosas y draconianas ahora usadas para hacer cumplir el copyright del software. Toma, por ejemplo, estas cuatro prácticas de la Software Publishers Association, SPA [Asociación de Editores de Software]:

- Propaganda masiva afirmando que está mal desobedecer a los propietarios para ayudar a un amigo.
- Solicitar a la gente que se convierta en soplona para delatar a sus colegas y compañeros de trabajo.
- Redadas (con ayuda policial) en oficinas y escuelas, en las que se dice a la gente que debe probar que es inocente de hacer copias ilegales.

¹Escrito originalmente en 1996

- El proceso judicial —iniciado por el gobierno de los EEUU, a petición de la AES— a personas como David LaMacchia del MIT,² no por copiar software —no se le acusa de copiarlo—, sino meramente por dejar sin vigilancia equipos de copia y no controlar su uso.

Cada una de estas cuatro costumbres se asemejan a aquellas usadas en la antigua Unión Soviética, donde todas las máquinas de copiar tenían un vigilante para impedir que se hicieran copias prohibidas, y donde las personas tenían que copiar información en secreto y pasarla de mano a mano como *samizdat*. Por supuesto hay una diferencia: el motivo para el control de información en la Unión Soviética era político; en los EEUU el motivo es el beneficio económico. Pero son las acciones las que nos afectan, no el motivo. Cualquier intento de coartar el hecho de que se comparta la información, sin importar por qué, lleva a los mismos métodos y a la misma dureza.

Los propietarios hacen uso de distintos argumentos para que se les conceda el control de cómo usamos la información:

Insultos

Los propietarios usan palabras difamatorias como «piratería» y «robo», al igual que terminología experta como «propiedad intelectual» y «daño», para sugerir una cierta línea de pensamiento al público —una analogía simplona entre los programas y los objetos físicos.

Nuestras ideas e intuiciones acerca de la propiedad sobre los objetos materiales tratan acerca de si es justo *privarle a alguien de un objeto*. No se aplican directamente a *hacer copias de algo*. Pero los propietarios nos piden que apliquemos estas ideas de todas formas.

Exageración

Los propietarios dicen que sufren un «daño» o «pérdida económica» cuando los usuarios copian programas por su cuenta. Pero copiar no tiene un efecto directo sobre el propietario, y no hace daño a nadie. El propietario sólo puede perder si la persona que hizo la copia hubiese pagado por otra del propietario en su lugar.

Un poco de reflexión muestra que la mayoría de esas personas no habrían comprado copias. Aun así los propietarios calculan sus «pérdidas» como si todos y cada uno hubiesen comprado una copia. Esto es una exageración —por decirlo de forma suave.

La ley

Los propietarios a menudo describen el estado actual de la ley, así como las duras sanciones con las que nos amenazan. Implícita en este enfoque va la sugerencia de que

²El 27 de enero de 1995 el caso de David LaMacchia fue desestimado, sin que se haya apelado todavía esta decisión.

la ley actual refleja un punto de vista moral incuestionable —y aun así al mismo tiempo, se nos insta a considerar estas sanciones como hechos naturales por los que no se puede responsabilizar a nadie.

Esta línea de persuasión no está diseñada para reafirmar el pensamiento crítico; está concebida para reforzar un camino mental ya trazado.

Es evidente que las leyes no distinguen lo que está bien de lo que está mal. Todo americano debería saber que, hace cuarenta años, en muchos estados iba contra la ley que una persona de raza negra se sentase en la parte frontal del autobús; pero solamente los racistas dirían que sentarse ahí no estaba bien.

Derecho natural

Los autores a menudo apelan a una conexión especial con los programas que han escrito y añaden que, en consecuencia, sus deseos e intereses respecto al programa simplemente prevalecen sobre los de cualquier otra persona —o incluso sobre los del resto del mundo. (Normalmente son las empresas, no los autores, los que detentan el copyright sobre el software, pero se espera de nosotros que ignoremos esta diferencia.)

Para los que presentan esto como un axioma ético —el autor es más importante que tú— sólo les puedo decir que yo mismo, un notable autor de software, lo considero una tontería.

Sin embargo, la gente, por lo general, sólo suele sentir alguna afinidad hacia las pretensiones basadas en el derecho natural debido a dos razones.

Una razón viene de una analogía forzada entre el software y los objetos materiales. Cuando yo cocino espaguetis, me quejo si otra persona se los come, porque entonces yo ya no me los puedo comer. Su acción me perjudica exactamente tanto como lo que le beneficia a él; sólo uno de nosotros se puede comer los espaguetis, así que la pregunta sería, ¿quién? La más mínima distinción entre nosotros es suficiente para inclinar la balanza ética.

Pero el hecho de que tú ejecutes o modifiques un programa que yo he escrito te afecta a ti directamente y a mí sólo indirectamente. Si tú le das una copia a tu amigo te afecta a ti y a tu amigo mucho más que lo que me afecta a mí. Yo no debería tener el poder de decirte que no hagas estas cosas. Nadie debería.

La segunda razón es que a la gente se le ha dicho que el derecho natural de los autores sen una tradición indiscutida y aceptada en nuestra sociedad.

Desde un punto de vista histórico, lo cierto es lo contrario. La idea del derecho natural de los autores fue propuesta y decididamente rechazada cuando se redactó la Constitución de los EEUU. Ésa es la razón por la que la Constitución sólo *permite* un sistema de copyright y no *obliga a que exista otro*; por esa razón dice que el copyright debe ser temporal. Establece asimismo que el propósito del copyright es promocionar el progreso —no recompensar a los autores. El copyright recompensa a los autores en cierta medida, y a los editores más, pero está concebido como un medio para modificar su comportamiento.

La tradición realmente establecida de nuestra sociedad es que el copyright recorta los derechos naturales del público —y que esto sólo se puede justificar por el bien del público.

Economía

El último argumento que se emplea para justificar la existencia de propietarios de software es que esto conduce a la producción de más software.

A diferencia de los demás, éste argumento por lo menos adopta un enfoque legítimo sobre el tema. Se basa en un objetivo válido —satisfacer a los usuarios de software. Y empíricamente está claro que la gente producirá más de algo si se les paga bien por ello.

Pero el argumento económico tiene un defecto: se basa en la presunción de que la diferencia es sólo cuestión de cuánto dinero debemos pagar. Asume que la «producción de software» es lo que queremos, tenga el software propietarios o no.

La gente acepta gustosamente esta presunción por que está de acuerdo con nuestra experiencia acerca de los objetos materiales. Consideremos por ejemplo un bocadillo. Es posible que puedas conseguir un bocadillo equivalente ya sea gratis, ya por un precio. Si es así, la cantidad que pagas es la única diferencia. Tanto si lo tienes que comprar como si no, el bocadillo tiene el mismo sabor, el mismo valor nutricional, y en cualquier caso te lo puedes comer sólo una vez. El hecho de si el bocadillo lo obtienes de un propietario o no, no puede afectar directamente a nada más que la cantidad de dinero que tienes después.

Esto es cierto para cualquier objeto material —el hecho de que tenga o no tenga propietario no afecta directamente a lo que *es*, o a lo que puedes hacer con ello si lo adquieres.

Pero si un programa tiene un propietario, esto afecta en gran medida a lo que es, y a lo que puedes hacer con una copia si la compras. La diferencia no es sólo una cuestión de dinero. El sistema de propietarios de software incentiva a los propietarios de software a producir algo —pero no lo que la sociedad realmente necesita. Y causa una contaminación ética intangible que nos afecta a todos.

¿Qué es lo que la sociedad necesita? Necesita información que esté verdaderamente a disposición de sus ciudadanos —por ejemplo, programas que la gente pueda leer, arreglar, adaptar, y mejorar, no solamente ejecutar. Pero lo que los propietarios de software ofrecen de forma característica es una caja negra que no podemos ni estudiar ni modificar.

La sociedad también necesita libertad. Cuando un programa tiene un propietario, los usuarios pierden la libertad de controlar una parte de sus propias vidas.

Y sobre todo una sociedad necesita incentivar el espíritu de cooperación entre sus ciudadanos. Cuando los propietarios de software nos dicen que ayudar a nuestro vecino de una manera natural es «piratería», están contaminando el espíritu cívico de nuestra sociedad.

Por eso decimos que el software libre es una cuestión de libertad, no de precio.

El argumento económico para justificar la propiedad es erróneo, pero la cuestión económica es real. Algunas personas escriben software útil por el placer de escribirlo o por admiración y amor; pero si queremos más software del que esas personas escriben, necesitamos conseguir fondos.

Desde hace ya diez años, los desarrolladores de software libre han probado varios métodos para encontrar fondos, con algo de éxito. No hay necesidad de hacer rico a nadie; los ingresos medios de una familia norteamericana, alrededor de 35.000 dólares anuales, demuestran ser incentivo suficiente para muchos trabajos que son menos satisfactorios que programar.

Durante años, hasta que una beca lo hizo innecesario, yo me ganaba la vida realizando mejoras a medida sobre software libre que yo había escrito. Cada mejora se añadía a la versión estándar lanzada y así, eventualmente, se ponían a disposición del público en general. Los clientes me pagaban para que trabajase en las mejoras que ellos querían, en lugar de en las características que yo habría considerado como la máxima prioridad.

La Free Software Foundation (FSF), una entidad sin ánimo de lucro exenta de impuestos para el desarrollo de software libre, consigue fondos mediante la venta de CD-ROMs de GNU, camisetas, manuales y distribuciones «deluxe», (que los usuarios son siempre libres de copiar y modificar), así como mediante donaciones. Ahora cuenta con un equipo de cinco programadores y tres empleados que se encargan de los pedidos por correo.

Algunos desarrolladores de software libre ganan dinero mediante la venta de servicios de soporte. Cygnus Support,³ que cuenta con alrededor de 50 empleados [en 1994, cuando se escribió este artículo], estima que en torno al 15 por ciento de la actividad de su equipo es desarrollo de software libre —un porcentaje respetable para una compañía de software.

Algunas compañías, incluyendo Intel, Motorola, Texas Instruments y Analog Devices, han unido esfuerzos para financiar el desarrollo continuado del compilador GNU para el lenguaje C. Mientras, el compilador GNU para el lenguaje Ada está siendo financiado por la Fuerza Aérea de los EEUU, que cree que ésta es la manera más efectiva de conseguir un compilador de alta calidad. (La financiación de la Fuerza Aérea se acabó hace algún tiempo; el Compilador GNU de Ada está ahora en servicio, y su mantenimiento se financia comercialmente)

Todos estos ejemplos son pequeños; el movimiento de software libre es aún pequeño y aún joven. Pero el ejemplo de las radios mantenidas-por-la-audiencia en los EE.UU muestra que es posible mantener una actividad grande sin forzar a cada usuario a pagar.

Como un usuario de informática de hoy en día, te puedes encontrar usando un programa propietario. Si un amigo te pide hacer una copia, estaría mal negarse a ello. La cooperación es más importante que el copyright. Pero una cooperación clandestina,

³Cygnus Support siguió teniendo éxito, pero luego aceptó inversiones foráneas, se volvió ambiciosa y empezó a desarrollar software no libre. Entonces fue absorbida por Red Hat, que ha lanzado la mayor parte de esos programas como software libre.

oculta no contribuye a mejorar la sociedad. Una persona debería aspirar a vivir una vida honrada abiertamente con orgullo, y esto significa decir «no» al software propietario.

Tienes derecho a poder cooperar abierta y libremente con otras personas que usan software. Tienes derecho a poder aprender cómo funciona el software, y a enseñar a tus estudiantes con él. Tienes derecho a poder contratar a tu programador favorito para arreglarlo cuando se rompa.

Tienes derecho al software libre.