Introducción a los aspectos legales del software libre

Malcolm Bain
Manuel Gallego
Manuel Martínez Ribas
Judit Rius

P08/M2114/00342

UOC
Universitat Oberta de Catalunya
www.uoc.edu

Índice

Introducción 5

Objetivos 6

1. La libertad del software 7

2. Software libre y software abierto: lo mismo 9
2.1. La FSF y la GPL 9
2.2. La OSI y el software *open source* ('de código abierto') 10

3. Las licencias de software libre 13

4. El modelo de software libre 16
4.1. La teoría de la génesis social del conocimiento 16
4.2. Una mayor difusión 16
4.3. Otros argumentos 17

5. Terminología 19

Introducción

En este módulo vamos a introducirnos en el núcleo de estudio de esta asignatura, la identificación y la gestión de los aspectos legales del software libre. Primero, vamos a analizar lo que significa desde una perspectiva legal el término *libre*, tantas veces repetido. A continuación, comentaremos el movimiento del software libre y del software abierto y sus alegadas diferencias, y seguiremos con una breve introducción a las licencias de uso de software libre. Luego, comentaremos desde la perspectiva legal los argumentos dados para defender el movimiento libre.

Al ser ésta una unidad introductoria, nuestro único objetivo es situar al estudiante en el contenido y la terminología del curso que empezamos, por lo que en la mayoría de temas vamos a remitir a las unidades posteriores para su desarrollo.

Objetivos

Los objetivos de este módulo son los siguientes:

1. Introducir al estudiante en los aspectos legales y de explotación del software libre, ilustrarlo sobre la importancia de los aspectos legales para el desarrollo, la explotación y el uso de software libre, y situar dichos aspectos legales dentro del contexto tecnológico, comercial y político.

2. Presentar las iniciativas de software libre y software abierto, su origen histórico y los principales argumentos de desencuentro.

3. Clasificar las licencias de software libre que se van a estudiar, básicamente para introducir al estudiante en la terminología y los conceptos más relevantes.

4. Comentar desde una perspectiva legal los principales argumentos que utiliza la comunidad del software libre para defender su utilización.

5. Unificar la terminología y los conceptos más importantes de este curso.

1. La libertad del software

La palabra *libre* (*free*) en inglés tiene dos sentidos: 'libertad' y 'gratuidad'. Por ello, es necesario aclarar desde este momento que el uso del término *free* relativo al software no implica que el titular o el proveedor del software lo otorgue o lo distribuya de manera gratuita (aunque lo puede hacer).

El término *free* se refiere a que el software se distribuye bajo una licencia que permite a los usuarios aprovecharlo libremente.

Por lo que respecta a la contraprestación económica para la distribución de software libre, veremos que la mayoría de las licencias permiten al distribuidor imponer el precio que quiera.

El término *free*

Las licencias de tipo BSD autorizan a privatizar el código y, por lo tanto, a venderlo como un producto comercial. La General Public License (GPL) permite explícitamente cobrar por la distribución (cláusula 1). El precio está limitado únicamente por las reglas del mercado: como el usuario podrá publicar posteriormente el código fuente en Internet o realizar cualquier otra forma de distribución gratuita, cualquier tercero podrá obtener una copia sin pagar.

Aunque, como veremos en el apartado siguiente, el movimiento del software libre tenga diferentes ramas, lo que unifica todos los modelos de desarrollo y todos los tipos de licencias son las **libertades** que los desarrolladores de software libre desean que adquieran los usuarios. Estas libertades fueron declaradas por la Free Software Foundation (FSF) de la manera siguiente:

- **Libertad 0:** la libertad de ejecutar y usar el software para cualquier propósito.
- **Libertad 1:** la libertad de estudiar el programa y de adaptarlo a sus necesidades.
- **Libertad 2:** la libertad de distribuir copias.
- **Libertad 3:** la libertad de modificar el programa y de liberar las modificaciones al público.

Para disfrutar de estas libertades, en especial la 1 y la 3, el usuario necesita disponer del código fuente del programa. Por lo tanto, será imprescindible que las licencias de software libre contengan, efectivamente, el compromiso del titular-autor de proporcionar el código fuente a los usuarios o, al menos, de ponerlo a su disposición.

Esta licencia de distribución del software determina las libertades (o mejor dicho, los derechos) que tiene el usuario. Sin embargo, las 4 libertades pueden ser implementadas de diferentes maneras en las licencias de software libre. Tanto la BSD como la GPL ofrecen estas libertades y derechos, pero de maneras muy diferentes, como veremos a continuación y en detalle en el módulo 6.

La importancia de esta definición es doble. Por un lado, toda la comunidad está de acuerdo con ella y la respeta. Segundo, desde la perspectiva legal, es una herramienta de análisis unificador de las licencias de software libre: separa lo que es libre de lo que no lo es.

2. Software libre y software abierto: lo mismo

En 1998 hubo una cierta escisión conceptual en el movimiento de software libre que en realidad fue la concretización de una división que se venía produciendo desde principios de los años noventa. Esta división dio lugar a la creación de la *Open Source Initiative*, OSI ('Iniciativa de Software con Fuentes Abiertas'), que estableció la *open source definition*, OSD ('definición de software con fuentes abiertas'). A continuación, vamos a comentar brevemente esta evolución historico-filosófica dentro del movimiento del software libre.

2.1. La FSF y la GPL

Aunque encontramos un antecedente en la Universidad de Berkeley y en la licencia BSD que estudiaremos más adelante, para muchos, los iniciadores del movimiento del software libre, a principios de 1984, son Richard Stallman y la Free Software Foundation.

Richard Stallman, en aquella época empleado en el AI Lab del MIT, abandonó su trabajo para comenzar el proyecto GNU (acrónimo de 'GNU No es Unix') y fundó la Free Software Foundation para conseguir fondos y una estructura más formal para el desarrollo y la protección del software libre. Richard Stallman sentó los fundamentos éticos del software libre en documentos como *"The GNU Manifesto"* y *"Why Software Should Not Have Owners"*.

Web recomendada

Podéis ver:

- www.gnu.org
- www.fsf.org

Desde el principio del proyecto GNU, Richard Stallman estaba preocupado por las libertades que tendrían los usuarios del software creado. Estaba interesado en que no sólo los que recibieran los programas directamente del proyecto GNU, sino cualquiera que lo recibiera después de cualquier número de redistribuciones, pudiera seguir disfrutando de los mismos derechos (modificación, redistribución, etc.).

> Un objetivo fundamental para R. Stallman y la FSF es garantizar la libertad de uso del software (en el sentido de que el usuario goce de las cuatro libertades mencionadas) y mantener esa libertad en relación con obras derivadas de software originalmente libre.

Para conseguir este objetivo, R. Stallman escribió la General Public License (GPL) como base para asegurar la libertad de **todos** los usuarios del software libre en todo momento. La GPL va más allá de garantizar las cuatro libertades básicas, e incluye términos que obligan a utilizar la misma licencia (la GPL) a la hora de redistribuir tanto el software original como cualquier obra derivada del mismo (y potencialmente cualquier otra obra que lo integre) y a ofrecer

acceso al código fuente. Es decir, a) no se permite a los redistribuidores añadir a su licencia restricciones adicionales (a las de la licencia GPL original) y b) en general, deben acompañar cualquier binario con su correspondiente código fuente.

Esta condición particular, que se conoce como *copyleft*, establece la imposibilidad legal de "capturar" el software libre y privatizarlo –es decir, distribuirlo bajo licencia no libre. Por lo tanto, el *pool*, o la cantidad de software con *copyleft* disponible para todos, no puede hacer más que aumentar a medida que los desarrolladores crean nuevas aplicaciones sobre la base del software distribuido bajo licencia con *copyleft*. La GPL, por lo tanto, es una licencia libre con *copyleft*.

Este concepto de *copyleft* ha tenido un éxito fenomenal dentro del sector y la FSF es la institución por excelencia que defiende los valores y la ética del movimiento de software libre. En 2006-2007 la FSF volvió a considerar la GPL, ya en su versión 2, a la luz de los cambios tecnológicos y legales y presentó a la comunidad su borrador para una nueva versión de la GPL (la GPLv3). El proceso de redacción de la nueva licencia, hasta su aprobación final en junio del 2007, fue largo y complicado, e implicó a actores (*stakeholders*) tan diversos como las empresas multinacionales, el sector académico, la comunidad de desarrolladores individuales y la administración pública. Comentaremos en profundidad la GPLv3 en el módulo 6.

Destaquemos que el movimiento *Free Software* de la FSF es, fundamentalmente, un movimiento filosófico, ético y político, no se trata de una organización tecnológica o proyecto de software libre (esto sería el proyecto GNU, el proyecto más cercano a la FSF).

Software "libre" de "propietario" o "privativo"

Distinguimos software "libre" de "propietario" o "privativo". Sin embargo, ninguno de los dos términos es realmente correcto. Todo software tiene propietario (tanto libre como no-libre) y una licencia no-libre no "priva" al usuario de algo que podría haber tenido antes. Como venimos de explicar, los derechos de uso de un software son originariamente del autor/titular y es solamente con una licencia que un tercero puede utilizar un programa. La expresión más correcta sería "no-libre" pero no tiene aceptación en el sector.

2.2. La OSI y el software *open source* ('de código abierto')

Para algunos, el término *open source* es una modalidad del software libre; para otros, es un término general que engloba todo el software libre, y finalmente, para otros es una peligrosa desviación de los conceptos originales del software libre para conseguir su mayor comercialización.

El proyecto *Open Source* surgió de una reunión estratégica mantenida en febrero de 1998 en Palo Alto, California, para reaccionar frente al plan de Netscape Inc. de liberar el código fuente de su navegador, Netscape Navigator. Entre

los presentes estaban Eric Raymond, Bruce Perens (líder del grupo Debian), John "Maddog" Hall (de la organización Linux International) y Sam Ockman (representante del grupo de usuarios de Linux de Silicon Valley).

Estas personas comprendieron que, con esta iniciativa que promulgaba una empresa comercial, se podía abandonar la actitud conflictiva que de cierta manera se había asociado con el software libre en el pasado e intentar ofrecer una perspectiva sobre el software libre más pragmática y orientada al mundo empresarial: reconciliar las libertades básicas del software libre con las necesidades comerciales de las empresas implicadas en la creación, la distribución y el uso de software libre Se consideró oportuno, por lo tanto, establecer una iniciativa para amparar licencias libres pero sin necesariamente tener *copyleft*.

Web recomendada

Podéis ver "La catedral y el bazar" en:

http://www.opensource.org/halloween

Para reforzar esta iniciativa, sus defensores también establecieron el uso de la expresión *open source software* ('fuentes abiertas' o 'código abierto', con el acrónimo OSS) para, por un lado, diferenciarse del término *free software* (general pero equivocadamente relacionado con el software de *copyleft* de la FSF) y, por otro, evitar el uso de la palabra *free*, y no tener así connotaciones de gratuidad.

free software

Es interesante señalar que actualmente estos objetivos no se cumplen del todo, en primer lugar, porque se sigue usando mayoritariamente la expresión *free software* (y en castellano, *software libre* en lugar de *abierto*), y segundo, porque hay todavía mucha confusión relativa a la libertad y la gratuidad del software libre.

La clasificación como software *open source* requiere el cumplimiento de las libertades fundamentales (copia, modificación, distribución, acceso a código fuente), pero no necesariamente las obligaciones de *copyleft*, pues considera que la distribución posterior de obras modificadas debe permanecer flexible, y que no ha de exigirse, como hace la GPL, la garantía de que la distribución posterior sea libre. Hay ahora unas 70 licencias aprobadas por la OSI, que incluyen, por supuesto, la GPL y la LGPL, y otras licencias que comentaremos en los módulos 6 y 7.

Web recomendada

Comentaremos en detalle la OSD en el módulo 6. Para más información, podéis ver:

http://www.opensource.org/docs/definition.php

OSD

La OSD surge de las directrices Debian de software libre (*"Debian Free Software Guidelines"*, DFSG), adaptadas en 1998 básicamente por la eliminación de las referencias a Debian. En efecto, la definición de software abierto en las DFSG era suficientemente amplia como para incluir las licencias de tipo BSD, la GPL y su hermana la LGPL, y otras como la del MIT/X y la Apache. Por lo tanto, sus requerimientos fueron adoptados por la OSI como las pautas generales que toda licencia abierta debía cumplir.

Además, la OSI ha elaborado una marca de certificación, la OSI Certified, que es una manera clara de indicar que una licencia cumple con la OSD. La marca sirve también para diferenciar el término general *open source*, que no tiene un uso suficientemente definido para garantizar esta conformidad.

Se argumentó que al establecer estándares de esta manera, la definición permitiría a desarrolladores, usuarios, organizaciones comerciales y a la administración pública entender mejor el movimiento de software libre, respetar más sus principios y, por qué no, encontrar nuevos modelos de negocio que aseguraran su futuro.

Ved también

Todos estos temas se desarrollarán con más profundidad en el módulo 6.

3. Las licencias de software libre

Tanto la FSF como la OSI implementan su filosofía y/o estrategia por medio de una herramienta de trabajo que es de naturaleza legal: la **licencia de software libre**. La FSF, por la publicación de las licencias GPL y LGPL, y ahora la licencia Affero GPL (AGPL, específicamente diseñada para software distribuido como servicio –SaaS– o la prestación de servicios remotos de tipo ASP). La OSI, por la catalogación y la clasificación de las diferentes licencias libres de mayor o menor uso por parte de la comunidad, publicadas en su sitio web: www.opensource.org.

¿Por qué? Porque la licencia es el único elemento que realmente distingue un software libre de un software propietario o "privativo".

Veremos en el módulo 2 que la legislación sobre los derechos de autor concede en exclusiva al titular del software el derecho a realizar o autorizar a terceros la reproducción (instalar y ejecutar), la copia, la modificación y distribución del software. Por tanto, para que un usuario pueda beneficiarse de las libertades del software, necesita el permiso explícito del titular de estos derechos: de lo contrario, se entendería que utiliza el software de forma ilegítima.

Luego veremos, en los módulos 4 y 5, que la licencia "de uso" de software es, entonces, el instrumento legal que autoriza a los usuarios del software a realizar estos actos reservados por ley: el instrumento que "libera" al usuario de la posición por defecto que es la prohibición de explotación del software. Por otro lado, la licencia permite al autor reservarse los derechos que no se ceden e imponer y otorgar al usuario otras obligaciones y derechos no necesariamente vinculados con el derecho de autor (confidencialidad, limitaciones de garantías, etc.).

Los **derechos** (o "libertades") de los que deben gozar los usuarios de software se plasman, entonces, en la licencia de distribución del software. La diferencia entre las licencias de software libre y las de software no-libre reside en los derechos y las obligaciones especificados en dicha licencia. Aquéllos otorgados por las licencias de software libre ("licencias libres") suelen ser directamente opuestos a los otorgados y reservados por las licencias de software no-libre (licencia no-libre, "propietaria" o "privativa""), sobre todo en cuanto al uso, la distribución y la modificación del software.

Una vez diferenciadas las licencias de software no-libre de las licencias de software libre, es importante destacar que **no todas las licencias de software libre son iguales**. El abanico de posibilidades va desde unas exigencias mínimas (en las licencias de tipo BSD), que obligan únicamente a mantener el aviso de autoría y la negación de garantías y de responsabilidad (*disclaimer*), hasta el "máximo" (en cierto sentido) de la cláusula *copyleft* de la GPL, que obliga al usuario a distribuir cualquier modificación y obra derivada bajo la misma licencia GPL. En los módulos 6 y 7 comentaremos con más detalle las diferentes maneras de conceder estos derechos y de imponer algunas restricciones.

Reflexión

A causa de la obligación de distribuir de manera libre cualquier obra modificada o derivada, se ha dicho que "la GPL es menos libre" que otras licencias abiertas. La FSF rechaza esta calificación, pues argumenta que, por el contrario, la licencia GPL es más libre porque garantiza más libertad para el usuario final. Hay que matizar:

- La BSD, por ejemplo, otorga más libertad a los desarrolladores, porque éstos pueden incorporar y distribuir implementaciones de "código BSD" bajo ambas licencias, libres y propietarias.
- La GPL transmite más libertad a los usuarios finales, porque éstos siempre recibirán aplicaciones con código fuente abierto y una licencia libre.

Brevemente (porque volveremos a ver estos aspectos en detalle en el módulo 6), podemos clasificar las licencias libres en tres categorías: las **licencias permisivas**, las licencias libres con ***copyleft robusto*** y las licencias libres con ***copyleft suave*** o "híbridas".

a) Las licencias libres permisivas (o "académicas")

La licencia *Berkeley Software Distribution* (BSD) es quizás la versión más simple de todas las licencias libres, y es también la primera licencia libre creada. Otorga todos los derechos de explotación y obliga solamente a mantener los avisos de autoría y los *disclaimers* de garantías y responsabilidades. Surge de las distribuciones de versiones de Unix de la Universidad de California, Berkeley, en las décadas de 1970 y 1980. La filosofía que late tras esta licencia se encuentra en que el código es fruto de las investigaciones y los trabajos universitarios financiados por el Gobierno de los Estados Unidos (y los impuestos del pueblo americano). Por lo tanto, debe ser de acceso libre y debe proteger lo que llamaríamos aquí los "derechos morales" de los autores por la simple obligación de mantener los avisos de autoría (*copyright notice*).

La BSD ha sido modelo de muchas licencias parecidas, entre las cuales citaremos la licencia Apache y las licencias de la familia X (X, XFree86, XOpen, X11).

b) Las licencias libres con *copyleft* robusto

La General Public License (GPL) es el principal exponente del tipo de licencias *copyleft*. Su objetivo es asegurar a todos los usuarios las cuatro libertades fundamentales del software libre, y también que las modificaciones sean distri-

buidos en las mismas condiciones. Otras incluyen la Common Public License de IBM o la licencia Sleepycat. Las llamamos de *copyleft robusto* porque no permiten su integración en aplicaciones mayores con otro tipo de licencia.

c) Las licencias libres con *copyleft* suave

Son licencias que mantienen las obligaciones de *copyleft* para el núcleo del programa distribuido, pero que, por otro lado, permiten su integración en obras con otras licencias. Pensemos en la licencia *Mozilla Public Lisence* (MPL), la *Lesser GPL* (LGPL), la *Open Source License* (OSL) o la *Common Development and Distribution License* (CDDL).

4. El modelo de software libre

El movimiento del software libre utiliza varios argumentos para defender su postura, y es importante considerar que estos argumentos tienen raíces fundamentales o impactos legales. Para entender un poco mejor la iniciativa del software libre, consideramos adecuado comentar brevemente estos argumentos.

4.1. La teoría de la génesis social del conocimiento

El primer argumento utilizado por los defensores del software libre es filosoficoideológico. El principio básico es que el conocimiento como tal no pertenece a nadie, pues cualquier conocimiento proviene de unos conocimientos anteriores y es una copia en mayor o menor medida de otras ideas. Pensamos en los "hombros de gigante" de Newton.

En otras palabras, nadie tiene ideas que no hayan estado directa o indirectamente influenciadas por las relaciones sociales que mantiene en las comunidades de las que forma parte, y si la génesis es social, el uso debe permanecer social a su vez. La función principal de la generación de conocimiento es mejorar la sociedad y, por lo tanto, llegar al máximo número de personas. Si consideramos el software como conocimiento, la argumentación de organizaciones como la FSF parece sencilla.

La consecuencia más directa de esta filosofía de la génesis social del conocimiento desde una perspectiva jurídica es que la limitación de la copia, uso o reproducción del software no tiene sentido, pues dificulta la generación de nuevo conocimiento y no permite que cumpla su principal objetivo: revertir en la comunidad.

4.2. Una mayor difusión

Existe otra razón, que podríamos considerar la razón "pragmática". Los defensores del software libre sostienen que distribuir una obra mediante una licencia libre (en el sentido de asegurar al usuario las cuatro libertades básicas) es la mejor manera de obtener beneficios de la publicación de contenidos cuando no se es un autor consagrado que pueda obtener beneficios de un *copyright* cerrado.

El argumento es que los únicos que realmente ganan con el sistema de derechos de autor o *copyright* son los autores famosos y los intermediarios, es decir, las editoriales. El resto de autores viven del prestigio que les dan sus obras, que les permiten prestar "servicios adicionales", como proporcionar mantenimiento o consultoría, o dar conferencias, cursos, charlas, escribir en periódicos, etc.

Siguiendo este argumento, para la mayoría de autores, y con el objetivo de aumentar su rendimiento económico, lo realmente importante es que la obra llegue al máximo número de personas. Así, su conclusión es que, para un autor desconocido, el sistema de derechos de autor supone una barrera para llegar al público y obtener beneficios de sus creaciones. Para contrarrestar el impacto del marco legal, hay que distribuir las obras bajo licencias libres, de manera que se asegure una plena libertad de redifusión.

Web recomendada

Hay un artículo muy clarificador sobre el tema, escrito por el colectivo Wu Ming, que puede leerse aquí:
www.wumingfoundation.com/italiano/Giap/giap2_IV.html#copyright1

4.3. Otros argumentos

Podemos citar otros argumentos usados por el movimiento de software libre para defender su postura, los cuales tienen todos un impacto o consecuencia legal:

- La mayor difusión de la obra no solamente permite la obtención de beneficios, sino que también **mejora su calidad** (corrección de errores, enriquecimiento por comentarios, el hecho de compartir diferentes perspectivas, etc.). Para ello, se requiere dar al usuario el derecho de modificación y acceso al código fuente (si de programas se trata). Ello es la filosofía detrás de la Wikipedia, cuyo *ranking* en términos de calidad es similar al de la *Enciclopedia británica*, a pesar de los orígenes a veces humildes de sus diferentes contribuidores.

- El modelo de software libre se basa en la participación del usuario no solamente en la identificación de errores, sino también en el momento del diseño y el desarrollo. Para ello, es necesario distribuir las obras (por ejemplo, el software) como versiones beta, con la libertad de instalar, usar, probar y aportar una contribución al proyecto u ofrecerle "retroalimentación" (aquí intervienen la licencia y la ausencia de garantías, puesto que se trata de una Beta).

- El modelo de desarrollo del software libre es más eficiente: no hace falta reinventar ninguna rueda, ya que las ruedas –los componentes de software, los textos, los gráficos, los iconos o las fotos– están disponibles para ser reutilizadas. Ello es posible en un mundo digital, ya que el "consumo" (el uso) del objeto digital no implica o no requiere la "propiedad" exclusiva de la cosa. De nuevo, esto sólo se puede hacer con licencias que permitan el libre uso (reproducción, distribución) en forma no exclusiva por los usuarios.

Vemos pues que tras los principales argumentos a favor del software libre (y sus contenidos) subyacen diferentes elementos o aspectos que caracterizamos como legales. Estos aspectos son los que desarrollaremos a lo largo de los capítulos siguientes.

Web recomendada

Podéis ver:

http://www.unctad.org/Templates/webflyer.asp?docid=4255&intItemID=1634&lang=1

5. Terminología

Cualquier aproximación al tema del software libre implica encontrarse con multitud de definiciones y términos diferentes. Para unificar criterios, en este apartado vamos a enumerar algunos de los principales conceptos del curso y a proponer la definición que los autores utilizaremos cuando nos refiramos a ellos.

Por **propiedad intelectual**, **derechos de autor** o ***copyright***, nos referiremos al sistema de protección de las creaciones humanas originales, literarias, artísticas o científicas, entre las que se incluye el software, y que reserva a sus titulares los derechos de reproducción, transformación, distribución y comunicación pública (subir a Internet) (ver el módulo 2 para más detalles).

Por **propiedad industrial** nos referimos al sistema legal que protege el uso o la explotación de signos distintivos que identifiquen productos o empresas (marcas), de las invenciones (patentes) y de la información confidencial con valor económico (secretos industriales) (ver el módulo 3 para más detalles).

La **licencia de uso** es el instrumento legal (un contrato) que emplea el titular de un software para otorgar permisos de uso de dicho software a terceros, a cambio, o no, de una remuneración económica (ver los módulos 4 y 5 para más detalles).

En cuanto a la nomenclatura relativa a las licencias de software libre en sentido amplio, usaremos la terminología siguiente:

- **Software libre y licencia libre:** cualquier licencia que respete las cuatro libertades, permitiendo así su reproducción, distribución y modificación, y el acceso al código fuente del mismo.

- **Software abierto y licencia abierta:** aquel software que cumpla las directrices de la definición de *software de código abierto* (OSD), en gran parte "sinónima" de *libre*, pero con otra perspectiva (más comercial, más orientada al acceso al código fuente).

- **Software *copyleft*** y **licencia con *copyleft*:** aplicaciones y licencias que se distribuyan con una cláusula de *copyleft*, que puede ser robusto (como la GPL) o suave (como la LGPL o la MPL).

- **Software ylicencia no-libre**, **"propietaria" o "privativa":** aquellas aplicaciones distribuidas bajo licencias que no son libres.