

SOMOS 11TH Y VAMOS A PRESENTAR EL SOFTWARE LIBRE

El logo que vemos aquí corresponde a la **Open Source Initiative (OSI**, en español Iniciativa para el Código Abierto) es una organización dedicada a la promoción del [código abierto](#).

1. INTRODUCCIÓN

«Software libre» es el software que respeta la libertad de los usuarios y la comunidad. A grandes rasgos, significa que los usuarios tienen la libertad de ejecutar, copiar, distribuir, estudiar, modificar y mejorar el software.

Su definición está asociada al nacimiento del movimiento de software libre, encabezado por Richard Stallman y la consecuente fundación en 1985 de la Free Software Foundation, que coloca la libertad del usuario informático como propósito ético³ fundamental.

'*Software libre*' no significa '*no comercial*'. Un programa libre debe estar disponible para uso comercial, desarrollo comercial y distribución comercial. El desarrollo comercial del software libre ha dejado de ser inusual; el software comercial libre es muy importante.

2. HISTORIA

Entre los años 1960 y 1970, el [software](#) no era considerado un producto sino un añadido que los vendedores de las grandes [computadoras](#) de la época (las [mainframes](#)) aportaban a sus clientes para que éstos pudieran usarlas. En dicha cultura, era común que [programadores](#) y desarrolladores de software compartieran libremente sus [programas](#) unos con otros. A finales de la década de 1970, las compañías iniciaron el hábito de imponer restricciones a los usuarios, con el uso de acuerdos de [licencia](#).

Con la llegada de los años 1980 la situación empezó a cambiar. Las computadoras más modernas comenzaban a utilizar [sistemas operativos privativos](#), forzando a los usuarios a aceptar condiciones restrictivas que impedían intercambiar, compartir o realizar modificaciones a dicho software.

Todo cambia con la aparición de Richard Matthew Stallman, programador estadounidense y figura relevante del movimiento por el software libre en el mundo.

Es principalmente conocido por el establecimiento de un marco de referencia moral, político y legal para el movimiento del software libre, como una alternativa al desarrollo y distribución del software no libre o privativo, aunque también fue el inventor del concepto de Copyleft (aunque no del término), un método para licenciar software de tal forma que su uso y modificación permanezcan siempre libres y queden en la comunidad.

Se cuenta que por los años 80 en su laboratorio habían recibido una impresora de una empresa externa. El dispositivo se utilizaba en red por todos los trabajadores, pero no funcionaba correctamente, ya que el papel en ocasiones se atascaba y no se generaba ningún aviso a los trabajadores, por lo que las pérdidas de tiempo eran constantes. Decidió que el problema podría solucionarse accediendo al código fuente de los controladores de la impresora, pero la empresa no se lo consintió. A raíz de ese conflicto comenzó a trabajar en el proyecto GNU y más tarde fundó la Free Software Foundation.

3. LAS CUATRO LIBERTADES ESENCIALES.

La libertad de ejecutar el programa como se desea, con cualquier propósito (libertad 0).

La libertad de estudiar cómo funciona el programa, y cambiarlo para que haga lo que usted quiera (libertad 1). El acceso al código fuente es una condición necesaria para ello.

La libertad de redistribuir copias para ayudar a su prójimo (libertad 2).

La libertad de distribuir copias de sus versiones modificadas a terceros (libertad 3). Esto le permite ofrecer a toda la comunidad la oportunidad de beneficiarse de las modificaciones. El acceso al código fuente es una condición necesaria para ello.

La libertad de ejecutar el programa como se desee

La libertad de ejecutar el programa significa que cualquier tipo de persona u organización es libre de usarlo en cualquier tipo de sistema de computación, para cualquier tipo de trabajo y finalidad, sin que exista obligación alguna de comunicarlo al programador ni a ninguna otra entidad específica. En esta libertad, lo que importa es el propósito del *usuario*, no el del *programador*. Usted como usuario es libre de ejecutar el programa para alcanzar sus propósitos, y si lo distribuye a otra persona, también esa persona será libre de ejecutarlo para lo que necesite; usted no tiene el derecho de imponerle sus propios objetivos a la otra persona.

La libertad de ejecutar el programa como se desea significa que al usuario no se le prohíbe o no se le impide hacerlo. No tiene nada que ver con el tipo de funcionalidades que el programa posee ni con el hecho de que el programa sea o no sea útil para lo que se quiere hacer.

La libertad de estudiar el código fuente y modificarlo

Una manera importante de modificar el programa es agregándole subrutinas y módulos libres ya disponibles. Si la licencia del programa especifica que no se pueden añadir módulos que ya existen y que están bajo una licencia apropiada, por ejemplo si requiere que usted sea el titular del copyright del código que desea añadir, entonces se trata de una licencia demasiado restrictiva como para considerarla libre.

Si una modificación constituye o no una mejora, es un asunto subjetivo. Si su derecho a modificar un programa se limita, básicamente, a modificaciones que alguna otra persona considera una mejora, el programa no es libre.

La libertad de redistribuir copias si así lo desea: requisitos básicos

La libertad para distribuir (libertades 2 y 3) significa que usted tiene la libertad para redistribuir copias con o sin modificaciones, ya sea gratuitamente o cobrando una tarifa por la distribución, a [cualquiera en cualquier parte](#).

También debe tener la libertad de hacer modificaciones y usarlas en privado para su propio trabajo o pasatiempo, sin siquiera mencionar que existen. Si publica sus cambios, no debe estar obligado a notificarlo a nadie en particular, ni de ninguna manera en particular. La libertad 3 incluye la libertad de publicar sus versiones modificadas como software libre.

4. TIPOS DE LICENCIAS

El *software* no se vende, se licencia. Una licencia es aquella autorización formal con carácter contractual que un autor de un *software* da a un interesado para ejercer "actos de explotación legales". Es decir, el *software* no se compra, sino que se adquieren una serie de derechos sobre el uso que se le puede dar. En las licencias de *software* libre esos derechos son muy abiertos y permisivos, apenas hay restricciones al uso de los programas. De ahí que ayude al desarrollo de la cultura . Pueden existir tantas licencias como acuerdos concretos se den entre el autor y el licenciatario. Desde el punto de vista del software libre, existen distintas variantes del concepto o grupos de licencias:

- **Licencias GNU General Public License (GNU GPL):** se utilizan para el software libre, la adopción de esta licencia garantiza a los usuarios finales la libertad de usar, estudiar, compartir (copiar) y modificar el software. Su propósito es declarar que el software cubierto por esta licencia es software libre y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios. Con esta licencia se puede instalar y usar un programa GPL en un ordenador o en varios, así como modificarlo y distribuirlo, para ello lo único a lo que estás obligado es a facilitar con el programa binario el código fuente.
- **Licencia AGPL:** es íntegramente una GNU GPL con una cláusula nueva que añade la obligación de distribuir el software si éste se ejecuta para ofrecer servicios a través de una red de ordenadores.
- **Licencias BSD:** se utilizan en gran cantidad de software distribuido.. El autor, bajo tales licencias, mantiene la protección de copyright únicamente para la renuncia de garantía y para requerir la adecuada atribución de la autoría en trabajos derivados. Son muy permisivas, tanto que son fácilmente absorbidas al ser mezcladas con la licencia GNU GPL con quienes son compatibles. Puede argumentarse que esta licencia asegura "verdadero" software libre, en el sentido que el usuario tiene libertad ilimitada con respecto al software, y que puede decidir incluso redistribuirlo como no libre.
- **Licencias MPL:** Esta licencia es de Software Libre y tiene un gran valor porque fue el instrumento que empleó Netscape Communications Corp. para liberar su Netscape Communicator 4.0 y empezar ese proyecto tan importante para el mundo del Software Libre: Mozilla. Se utilizan en gran cantidad de productos de software libre de uso cotidiano en todo tipo de sistemas operativos. La MPL no es tan excesivamente permisiva como las licencias tipo BSD. Estas licencias son denominadas de copyleft débil.

5. SOFTWARE LIBRE VS CÓDIGO ABIERTO

Se puede considerar al **movimiento del software de fuentes abiertas** como una ramificación del [movimiento del software libre](#) que utiliza como estrategia de fomento de la programación libre una orientación pragmática que otorga importancia fundamental a la disponibilidad del código fuente de los programas y olvida el resto de libertades y aspectos morales y éticos. Los partidarios del **movimiento open source** tuvieron la creencia de que la calidad de los aspectos técnicos serían los que más atraerían el interés de las empresas por el software libre del que se consideraban representantes legítimos. La principal crítica que Richard Stallman realiza de esta postura es que su pragmatismo distrae a los usuarios de las cuestiones morales centrales y de las libertades que fundamentan verdaderamente al software libre.

En su artículo [Por qué el código abierto pierde el punto de vista del software libre](#), Richard Matthew Stallman establece su preferencia por el término <<software libre>> por el sentido de libertad que

implica, mientras que el término <<código abierto>> no tiene porque implicar libertad, de hecho mucho de este software no da a sus usuarios la libertad de distribuir sus modificaciones, restringe el uso comercial, o en general restringe los derechos de los usuarios. Solo ofrece disponibilidad de acceso al código fuente. Mientras que las libertades de un producto libre están establecidas de base por la propia [filosofía del movimiento](#) (sin las cuales no tendría sentido ni existiría), los programas considerados <<open source>> tan sólo garantizan el acceso al código fuente como único requisito para el cumplimiento de sus fines que se resumen en lograr que el mundo empresarial se interese por los programas libres, en lugar de las soluciones privativas, en base a poner un mayor énfasis en la accesibilidad del código fuente y evitando otros elementos menos pragmáticos de tipo filosófico y moral.

6. CONCLUSIONES. ¿POR QUÉ USAR SOFTWARE LIBRE? EJEMPLOS

Razones para usar Software Libre

- Libre: Entre otras cosas eres libre para usar, modificar, regalar o vender los programas de software libre
- La copia es legal: es legal repartir software libre a otras personas. Usar un sistema libre evita en gran medida los problemas de la piratería. Si lo natural es compartir tus programas con otras personas, con software libre es legal
- Abierto: Se puede usar el código de los programas y modificarlo.
- Colaborativo: El modelo de desarrollo de software libre es colaborativo y participativo. Todo lo puedes modificar o criticar.
- Ayuda: Existen innumerables grupos de usuarios que se ayudan entre sí a través de Internet. Es decir si te surge un problema es muy probable que a otras personas les ha ocurrido y su ayuda será la mas valiosa que puedas encontrar.
- Auditable: El software libre se puede inspeccionar al disponer de su código fuente.
- Robusto frente a los virus: Existen muy pocos virus para GNU/Linux ya que el problema de los virus se debe al diseño del sistema operativo.
- Personalizable: Puedes personalizar toda la interfaz que el sistema le presenta al usuario.
- Bajo costo: De estudios realizados para empresas se han encontrado reducciones de de hasta un 30% en TCO (Costo total de propiedad).
- Reutilización de equipos: GNU/Linux puede ejecutarse perfectamente en equipos que han sido desechados por las nuevas versiones de sistemas operativos propietarios.

En el software libre hay de todo.

Uno de los principales problemas para el arraigo del software libre en sus primeros años era la falta de aplicaciones. En los malos viejos tiempos se solía decir aquello de “Linux para jugar, Windows para trabajar”, por un lado porque faltaban aplicaciones de productividad (como un buen procesador de textos, hojas de cálculo fáciles de utilizar o un programa de contabilidad en condiciones donde llevar las cuentas domésticas o de una PYME), por otro porque era altamente especializado, coto privado de ingenieros y geeks. Sin embargo, las cosas avanzan que es una barbaridad y nada de lo anterior es cierto hoy en día. **Firefox**, por ejemplo, es un navegador web sencillo de utilizar, con muchas más prestaciones que Explorer, que además es seguro, fiable y con un aspecto visual muy agradable. **OpenOffice** es un suite ofimático que comprende procesador de textos, hoja de cálculo, base de datos, una utilidad para crear presentaciones y un programa para el diseño vectorial que satisface las necesidades de cualquier usuario doméstico y profesional. The **GIMP** es un programa de diseño de bitmaps y retoque fotográfico que, sin llegar a los límites de Photoshop, es tremendamente útil para corregir las instantáneas de las vacaciones, diseñar vistosas imágenes o incluso crear divertidos cortos de animación.

7. BIBLIOGRAFÍA.

<http://www.monografias.com>

<https://es.wikipedia.org>

<https://www.gnu.org>

<http://hispalinux.es/SoftwareLibre>

http://biblioteca.uclm.es/Archivos/Investigacion/Software_libre.pdf

<http://www.mundodigital.net/>