

Definiciones y conceptos relativos al Software Libre

Diccionario temático

[Hypatia](http://www.hipatia.info) - <http://www.hipatia.info> - info@hipatia.info - [Autores](#)

Octubre del 2003

PANEL DE NAVEGACION

Si podemos conocer, usar, crear, programar, desarrollar, comunicar y compartir sin restricciones un SOFTWARE, entonces es Software Libre.

Ejecutar, inspeccionar, modificar y redistribuir el código son derechos fundamentales de los programadores y usuarios, libertades esenciales para una Sociedad del Conocimiento ni tecnocrática ni fascista que permita convivir con la tecnología en libertad y con solidaridad.

En el comienzo de los tiempos todo el software era libre y se compartía. Los programadores expresaban y comunicaban sus ideas mediante el código fuente. Luego las compañías cerraron el código.

El PROYECTO GNU surgió en 1984, como una reacción a este intento de limitar el discurso de los programadores y con ello creó un universo de Software Libre en paralelo al producido por las corporaciones. Ver el [Manifiesto GNU \[164\]](#)

[GNU] es un acrónimo recursivo, una palabra autoreferencial que significa “gnu no es unix”.

Analizamos algunas categorías de conceptos que nos parecen importantes

Otros conceptos relacionados como: computadoras, software, hardware, formatos, información digital, etc. se pueden analizar leyendo el documento: [La Encrucijada Digital \[104\]](#), que además contiene numerosas referencias de usos de Software Libre en Estados. Se puede profundizar en el [Diccionario Hacker\[155\]](#) y en la [Wikipedia \[156\]](#).

1. Software y derechos de usuarios y programadores

Listamos a continuación distintos tipos de software, exponiendo los derechos que los usuarios y programadores tienen con respecto a los mismos. No son categorías excluyentes ni totalizadoras. Muchas veces se instrumentan mediante licenciamientos o contratos privados de sus derecho-habientes con los usuarios o desarrolladores. A veces se sostienen, incluso penalmente, mediante el mecanismo de restricción de copia de las legislaciones estatales y tratados internacionales conocidos como Copyright, que en algunos casos se refuerza con leyes tipo DCMA. (ver más adelante)

[SL] Software Libre

Un software es libre si sus usuarios y desarrolladores gozan todas estas libertades:

- 0.- (**ejecutar**) La libertad de usar el programa, con cualquier propósito. (analizar la diferencia entre usar y ejecutar: hay otras formas de usar un programa además de ejecutarlo)

- 1.- (**inspeccionar**) La libertad de estudiar cómo funciona el programa y adaptarlo a sus necesidades. El acceso al código fuente es una condición previa para esto.
- 2.- (**redistribuir**) La libertad de distribuir copias, con lo que se puede ayudar a otros.
- 3.- (**modificar** y redistribuir las modificaciones) La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. El acceso al código fuente es un requisito previo para esto.

En otros idiomas:

Francés: logiciel libre

Alemán: freie Software

Ruso: svobodny programy

Chino: zi4you2 ruan3jian4

Japones: jiyuu [na] sofuto

Esperanto: libera programaro

Sueco: fri programvara

Holandés: vrije software

Inglés: free software. Para evitar la ambigüedad de libre con gratis algunos empiezan a hablar de "Libre Software." en inglés.

Portugués: software livre

Danish: fri software

Italiano: software libero

Catalan: software lliure

Software Libre implica la libertad de distribuir copias, sea con o sin modificaciones, en forma gratuita o cobrando un monto por la distribución, a cualquiera y en cualquier lugar.

Software Libre significa, entre otras cosas, que no hay que pedir o pagar permisos.

Software Libre implica la libertad de hacer modificaciones y utilizarlas de manera privada en trabajo u ocio, sin siquiera tener que anunciar que dichas modificaciones fueron realizadas. Cuando se publican y difunden los cambios no hace falta avisar a nadie en particular ni de ninguna manera en particular.

La libertad de distribuir copias debe incluir tanto las formas binarias o ejecutables del programa (si existen) como su código fuente, sean o no versiones modificadas. Distribuir programas de modo ejecutable es necesario para que los sistemas operativos libres sean fáciles de instalar.

Para que las libertades de hacer modificaciones y de publicar versiones mejoradas tengan sentido, se debe tener acceso al código fuente del programa. Por lo tanto, la posibilidad de acceder al código fuente es una condición necesaria para el Software Libre.

Para que estas libertades sean reales, deben ser irrevocables mientras no se haga nada incorrecto; si el desarrollador del software tiene el poder de revocar la licencia aunque no haya motivos, el software no es libre.

Son aceptables, sin embargo, ciertos tipos de reglas sobre la manera de distribuir Software Libre, mientras no entren en conflicto con las libertades centrales. Por ejemplo, copyleft ["izquierdo de copia"] (expresado muy simplemente) es la regla que implica que, cuando se redistribuya el programa, no se pueden agregar restricciones para denegar a otras personas las libertades centrales. Esta regla no entra en conflicto con las libertades centrales, sino que más bien las protege.

Puede que se haya pagado por obtener copias de Software Libre, o que se haya obtenido sin ningún costo, pero independientemente de la forma de adquisición u obtención de las copias, siempre se mantiene la libertad de copiar y modificar el software e incluso vender nuevas copias.

"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 ya es muy común; el software comercial libre es muy importante.

Es aceptable que haya reglas acerca de cómo empaquetar una versión modificada, siempre que no bloqueen a consecuencia de ello la libertad de publicar versiones modificadas. Reglas como "Si haces disponible el programa de esta manera, debes hacerlo disponible también de esta otra" pueden ser igualmente aceptables, bajo la misma condición. Este tipo de reglas permiten decidir si se publica o no el programa. También es aceptable que la licencia requiera que, si se distribuye una versión modificada y el desarrollador anterior solicita una copia, sea un deber enviársela.

<http://www.gnu.org/philosophy/free-sw.es.html> [152]

<http://www.gnu.org/bulletins/bull6.html> [153]

¿Quién escribe el Software Libre?

Texto tomado en parte de: <http://www.uoc.edu/dt/20327/index.html>

La mejor forma de responder esta cuestión es clasificar los proyectos del mundo del Software Libre según su liderazgo. Esta clasificación no pretende ser exhaustiva pero podemos diferenciar claramente tres grupos de comunidades que han liderado los principales proyectos libres en los últimos años:

Proyectos vinculados a empresas, como Sun Microsystems, que mantiene OpenOffice.org; la fundación Mozilla, hasta hace poco directamente dependiente de America Online, que mantiene el proyecto Mozilla.org, o Ximian, que mantiene el sistema de correo electrónico Evolution. También hay proyectos, como Apache, que están representados por fundaciones y que reciben ayuda de empresas como IBM.

Proyectos que han sido desarrollados en universidades, algunas veces con financiación de empresas o del Gobierno. Uno de los más representativos es la familia de sistemas operativos BSD, en concreto NetBSD y FreeBSD, desarrollados en la Universidad de Berkeley, en California.

Proyectos liderados por grupos de voluntarios, como por ejemplo Debian, GNU o Abiword, que son desarrollados por voluntarios de todo el mundo. Estos proyectos son, sin duda, los más altruistas desde un punto de vista ético, ya que se realizan sin apoyo financiero y desde el puro voluntariado. La motivación principal de los colaboradores de este tipo de proyectos es pensar que el Software Libre es un bien común.

[CL] Copyleft

El software protegido con copyleft es Software Libre cuyos términos de distribución no permiten a los redistribuidores agregar ninguna restricción adicional cuando éstos redistribuyen o modifican el software. Esto significa que cada copia del software, aun si ha sido modificado, debe ser Software Libre.

En el Proyecto GNU, se protege mediante copyleft casi todo el software que se escribe, porque el objetivo es dar a cada usuario las libertades que el término "Software Libre" implica.

Copyleft es un concepto general. Para proteger actualmente un programa con copyleft, se necesita usar un conjunto específico de términos de distribución. Hay muchas maneras posibles de escribir términos copyleft de distribución.

[SLSCL] Software Libre no protegido con copyleft

El Software Libre no protegido con copyleft viene desde el autor con autorización para redistribuir y modificar así como para añadirle restricciones adicionales.

Si un programa es libre pero no protegido con copyleft, entonces algunas copias o versiones modificadas pueden no ser libres completamente. Una compañía de software puede compilar el programa, con o sin

modificaciones, y distribuir el archivo ejecutable como un producto propietario de software.

El Sistema X Window ilustra esto. El Consorcio X libera X11 con términos de distribución que lo hacen Software Libre no protegido con copyleft. Se puede obtener una copia que tenga esos términos de distribución y es libre. Sin embargo, hay versiones no libres también, y hay estaciones de trabajo populares y tarjetas gráficas para PC para las cuales las versiones no libres son las únicas que funcionan. Si se usa este hardware, X11 no es Software Libre.

[SGPL] Software GPL

La GPL (General Public License/Licencia Pública General) de GNU es un conjunto específico de términos de distribución para proteger con copyleft a un programa. El Proyecto GNU la utiliza como los términos de distribución para la mayoría del software GNU

El sistema mundial de copyright se basa en acuerdos internacionales de reconocimientos, pero en cada país el alcance y restricciones impuestas por este sistema y otras normas, como las de protección al consumidor, pueden modificar la aplicabilidad real de cada término específico de la GPL, dependiendo, por ejemplo, de si el software ha sido transferido al usuario a título oneroso o no.

También puede cambiar en cada país el sentido que la legislación de copyright le da a los aportes o contribuciones de diferentes personas a un proyecto bajo la GPL. En todos los casos es conveniente asegurar que el código sumado a un proyecto GPL proviene de fuentes con derechos a hacerlo. Por ejemplo si los programadores trabajan en relación de dependencia. Esto es un requisito muy importante sobre todo si se trabaja con el modelo del bazar (Ver Bazar al final) para la creación de software.

La GPL constituye un excelente paraguas protector para el Software Libre en todo el mundo, pero algunas cláusulas pueden no ser válidas en todos los países y en todos los casos.

La GPL se distribuye exclusivamente en inglés, y los acuerdos internacionales garantizan su validez en todo el planeta, independientemente de donde se origine el programa o sus modificaciones. Las traducciones se distribuyen solo a título aclaratorio.

[SGNU]Software GNU

El Software [GNU](#) es software liberado bajo el auspicio del Proyecto GNU. La mayoría del software GNU está protegido con copyleft, pero no todo. Todo el software GNU es Software Libre.

Parte del software GNU es escrito por el personal de la Fundación para el Software Libre (Free Software Foundation), pero la mayoría del software GNU es aportada por voluntarios. La Fundación para el Software Libre es derecho habiente (titular del copyright) de parte del software aportado; otra parte es de quienes lo aportaron o escribieron.

[SISGNU] El sistema GNU

El sistema GNU es un sistema operativo libre completo estilo Unix.

Un sistema operativo libre estilo Unix consiste en muchos programas. La FSF estuvo acumulando componentes para este sistema desde 1984; la primera liberación de prueba de un "sistema GNU completo" fue en 1996.

El sistema GNU incluye todo el software GNU, así como muchos otros paquetes tales como el Sistema X Window y TeX que no son software GNU.

Debido a que el propósito de GNU es ser libre, cada componente individual en el sistema GNU tiene que ser Software Libre. No todos tienen que estar protegidos con copyleft, sin embargo; cualquier tipo de Software Libre es legalmente apto de incluirse si ayuda a alcanzar metas técnicas. Por ejemplo de hacer uso de Software Libre no protegido con copyleft como el Sistema X Window.

En el proyecto GNU, se utiliza "copyleft" para proteger de modo legal estas libertades para todos. Pero el Software Libre sin "copyleft" también existe. Existen razones importantes por las que es mejor usar copyleft, pero los programas son Software Libre sin necesariamente tener copyleft. En cualquier caso, se pueden usar.

A veces las normas de control de exportación de los gobiernos y acuerdos internacionales (Wassenaar <http://www.wassenaar.org/>) junto a las sanciones mercantiles, pueden restringir la libertad de distribuir copias de los programas a nivel internacional. Los desarrolladores de software no tienen el poder de eliminar o sobrepasar estas restricciones, pero lo que pueden y deben hacer es rehusarse a imponerlas en las condiciones de uso del programa. De esta manera, las restricciones no afectarán a actividades y gente fuera de las jurisdicciones de los gobiernos que las impongan.

Si un programa es libre, entonces puede ser potencialmente incluido en un sistema operativo libre tal como GNU, o sistemas GNU/Linux que son libres.

Estas libertades pueden construirse legalmente de diversas formas. En el marco de las leyes de copyright, se creó una licencia específica, la General Public Licence (GPL) para instrumentar el concepto de copyleft.

Algunas personas consideran que estas libertades y la comunidad de desarrolladores y usuarios formada a su alrededor utilizan la GPL como un acuerdo constitutivo, y que el código así producido requiere, merece, o se beneficiaría con un encuadre en un marco legal especial diferente de las leyes de copyright. De hecho esta comunidad se ha caracterizado por garantizar sus acuerdos por el consenso más que por el uso de los tribunales, como medio de resolución de conflictos.

En ambos casos lo importante es hacer hincapie en la libertad y no en el esquema legal para su consecución, sea este una licencia como la GPL u otro mecanismo legal.

En esta dirección se encuentran listados de licencias en cada categoría: <http://www.fsf.org/licenses/license-list.html>

Se puede profundizar estos temas en:

- <http://www.gnu.org/philosophy/free-sw.es.html>
- <http://www.gnu.org/philosophy/pragmatic.es.html>
- <http://www.gnu.org/philosophy/categories.es.html>
- <http://www.gnu.org/philosophy/words-to-avoid.es.html>

[SOL] Sistemas Operativos Libres: GNU/Linux, BSD, Hurd

Debian, SuSE, RedHat, Mandrake, etc

GNU/Linux es el sistema operativo mas conocido del mundo libre, otros son BSD y Hurd. Linux fue creado por Linus Torvalds en 1991 en Finlandia. Es un proyecto colaborativo llevado adelante por miles de desarrolladores en el Mundo. <http://www.kernel.org/>, <http://www.linux.org/>, <http://www.linux.com/>

Existen distribuciones de GNU/Linux preparadas en CDROM, algunas comunitarias como Debian y otras comerciales como Red Hat y SuSE.

[OSS] Software de Fuente|Código Abierto (Open Source Software)

El término 'Software de Fuente Abierta' es usado para dar a entender más o menos lo mismo que Software Libre. Ver <http://www.opensource.org/>

Existen al menos dos tendencias principales en el mundo del Software Libre. Ambas coinciden en las prácticas, desarrollan y distribuyen el mismo software bajo los mismos esquemas legales. Las diferencias son filosóficas. El primer grupo suele referirse a Software Libre, el segundo a Código Abierto. El primero indica que la razón fundamental para usar Software Libre es ética y su referencia es la FSF (Free Software Foundation).

El segundo indica que la razón fundamental es la conveniencia. Impulsa el Software Libre en términos pragmáticos más que filosóficos. Sus principales impulsores son Eric Raymond y O'Reilly. La definición en sus propios términos se encuentra en: <http://www.opensource.org/docs/definition.php>. Su argumento refiere a que el término "free" en inglés es confuso, si bien la propuesta de abierto es todavía más confusa, pues referencia sólo a una de las cuatro libertades. La OSI (Open Source Initiative), al igual que la FSF, mantiene una lista de licencias aprobadas que cumplen con sus requisitos: <http://www.opensource.org/licenses>. La Open Source Initiative es una estrategia de marketing para el Software Libre. La sustancia no cambia, la actitud sí. En febrero de 1998, Netscape anunció el lanzamiento de su navegador como Software Libre. Un grupo de personas con base en Palo Alto y Silicon Valley, propuso empezar tal campaña a favor del Software Libre mediante el término de Fuente Abierta. El objetivo fue lograr la aceptación de compañías y capitalistas en el boom de la nueva economía. Se eligió dejar de lado las cuestiones de largo alcance como economía, ética y sociales, pensando que estas obstaculizaban la rápida aceptación por parte de las empresas. El foco estaba en las ventajas técnicas. Muchos hoy usan el término para referirse a software con condiciones intermedias entre Libre y Propietario.

La Iniciativa OSS considera que en otros idiomas es mejor usar Software Libre, últimamente se empieza en inglés a usar "Libre Software"

[SA] Sistemas Abiertos

Como reacción a los sistemas cerrados, primero de IBM y luego de Microsoft, surgió la preocupación de permitir que las computadoras interoperen en base a estándares.

En general esta idea se materializó con servidores Unix y los estándares POSIX y han quedado completamente superadas por el movimiento de Software Libre que es la única vía real de satisfacer las necesidades expresadas en este movimiento.

[SC] Software Comercial

El software comercial es software que está siendo desarrollado por una entidad que tiene la intención de hacer dinero del uso del software. "Comercial" y "propietario" no son la misma cosa. La mayoría del software comercial es propietario, pero hay Software Libre comercial y hay software no libre no comercial.

Por ejemplo, Ada de GNU siempre se distribuye bajo los términos de la GPL de GNU y cada copia es Software Libre; pero los desarrolladores venden contratos de soporte. Cuando sus vendedores les hablan a los clientes potenciales, algunas veces el cliente dice "Nos sentiríamos más seguros con un compilador comercial." Los vendedores responden, "Ada de GNU es un compilador comercial; sólo que es Software Libre."

[SDP] Software de dominio público

Es software que no está protegido con copyright. Esto implica que terceros pueden apropiarse del mismo y realizar versiones modificadas que pueden ser o no libres.

Algunas veces la gente utiliza el término "dominio público" de una manera imprecisa para decir "libre" o "disponible gratis". Sin embargo, "dominio público" es un término legal y significa de manera precisa "sin copyright". Por claridad, se recomienda el uso de "dominio público" para ese significado solamente y el uso de otros términos para transmitir los otros significados.

[SSL] Software semilibre

El software semilibre es software que no es libre, pero viene con autorización para particulares de usar, copiar, distribuir y modificar (incluyendo la distribución de versiones modificadas) sin fines de lucro. PGP es un ejemplo de un programa semilibre. Habitualmente este tipo de software restringe el uso con fines comerciales del mismo.

El software semilibre es mucho mejor que el software propietario, pero aún plantea problemas y no se puede usar en un sistema operativo libre.

La única justificación para cualquier restricción substantiva acerca del uso de un programa es prevenir la adición de restricciones por parte de otras personas (copyleft).

[SG] Software gratuito, regalado, Freeware

Software que se distribuye sin costo. no es lo mismo ni es sinónimo de Software Libre.

Ese término específico significa que tiene precio cero. Software Libre es una cuestión de libertad y de derechos, no de precio.

Las copias de Software Libre frecuentemente están disponibles libres de cargo --por ejemplo, para ser descargadas via FTP (File Transfer Protocol) . Pero las copias de Software Libre también están disponibles por un precio en CD-ROMs. Las copias de software propietario ocasionalmente están disponibles libres de cargo en promociones y algunos paquetes propietarios están normalmente disponibles sin cargo alguno para ciertos usuarios.

El término "freeware" no tiene una definición clara aceptada, pero es usada comúnmente para paquetes que permiten la redistribución gratuita, pero no la modificación (y su código fuente no está disponible). Estos paquetes no son Software Libre, por lo tanto no es correcto usar la palabra "freeware" .

El término "freeware" fue utilizado con frecuencia durante los años 80 para hacer referencia a programas sacados al mercado sólo como ejecutables, con el código fuente no disponible. Hoy no se tiene una definición particular aceptada.

Para otros idiomas, se debe evitar palabras tomadas del Inglés como "free software" o "freeware".

Un programa distribuido como Software Libre no es un „regalo“. El uso del término „regalar“ tiene el mismo problema que el concepto de „software gratuito“: expresan un atributo basado en el precio, no en la libertad. Una forma de evitar este tipo de confusiones es hablar de software „distribuido como Software Libre“.

[SW] Shareware

El shareware es software con autorización para que la gente redistribuya copias, pero bajo la condición de que aquel que continúe usando la copia deberá pagar un cargo por licencia.

El shareware no es Software Libre, ni siquiera semilibre. Existen dos razones por las que no lo es:

- En la mayoría del shareware, el código fuente no está disponible, de esta manera se limita el derecho básico de modificar el programa.
- El shareware no viene con autorización para hacer una copia e instalarlo sin pagar una cantidad por licencia, ni aún para particulares involucrados en actividades sin ánimo de lucro. (En la práctica, la gente a menudo hace caso omiso a los términos de distribución y lo hace de todas formas, pero las cláusulas no lo permiten).

[SP] Software propietario, privativo o cerrado

El software propietario es software que no es libre ni semilibre. Su código está oculto. Su uso, redistribución o modificación está prohibida o requiere autorización.

Los usuarios de software propietario no pueden acceder al código del mismo, por lo tanto se ven privados del conocimiento real de las utilidades y funciones que el software realiza. Los usuarios de software propietario están privados del derecho a compartirlo, modificarlo, adaptarlo o corregir sus errores. Los usuarios de software propietario se convierten en clientes cautivos de las empresas que los desarrollan y de sus servicios técnicos autorizados y pierden así sus derechos básicos como consumidores.

El software propietario se caracteriza por privar de derechos a sus usuarios, por lo tanto es correcto denominarlo privativo.

El software cerrado es aquel cuyo código no se puede inspeccionar.

[CND] Contratos de no divulgación

Muchas veces los programadores y otros trabajadores se sujetan a contratos de no divulgación para poder programar. Los programadores acceden al código fuente del software, lo pueden usar para sus propios sistemas, pero no pueden redistribuir el código fuente o sus aportes al mismo.

Estos contratos exigen renunciar al derecho de compartir para poder programar.

2. Formatos de almacenamiento e intercambio de datos

[FAID]

Son las convenciones o formatos que se usan para intercambiar información en el tiempo (almacenamiento), a la distancia, o entre personas usando distintas aplicaciones o programas. Frecuentemente los tipos de archivos se vinculan con distintos formatos. Así el pdf, doc, xls, sxw, rtf, y otros, representan distintas clases de formatos.

Si estas convenciones o estándares, contenedores de nuestra información, son abiertos, conocidos por todos y no tienen restricciones de uso, cualquiera podrá consultarlos y realizar programas para acceder a estos datos. Los formatos libres garantizan la libertad de los usuarios para intercambiar información con todo el mundo independientemente de la aplicación que utilicen, ya que permiten a cualquier programador desarrollar software que trabaje con estos formatos.

Lamentablemente los formatos propietarios en ocasiones pueden tener absurdas limitaciones de uso o simplemente no estar documentados. Si, por ejemplo, si utilizamos Microsoft PowerPoint para enviar una presentación, sólo tienen garantizado el acceso con todas las particularidades del documento los usuarios de este programa.

En Internet a menudo nos encontramos con creadores de sitios que sólo prueban sus páginas con Explorer, muchas veces por desconocimiento de la existencia y del grado de implantación de otros navegadores o, en ocasiones, simplemente porque no consideran la compatibilidad como un tema importante.

De hecho, Internet ha creado sus propios mecanismos para evitar este tipo de situaciones. En 1994 se creó el World Wide Consortium (W3C), que agrupa a los principales fabricantes de software de Internet, con la misión principal de definir y promover la creación de estándares para la Web. En realidad, cuando hablamos de estándares web nos referimos casi siempre a las definiciones y recomendaciones de este consorcio, que ha conseguido que prácticamente todos los navegadores modernos funcionen en un grado aceptable con los estándares más recientes.

La restricción en el acceso de la información a un determinado navegador o formato representa una

discriminación contra los usuarios de los otros navegadores o aplicaciones.

Las administraciones públicas deben velar por no discriminar ninguna plataforma del usuario y no favorecer a ningún fabricante en especial.

Texto tomado en parte de: <http://www.uoc.edu/dt/20327/index.html>

Se puede clasificar los estándares de la misma forma que el software según los derechos que sus usuarios tienen.

3. Aspectos políticos, legales, penales y comerciales

[LIC] Licencias, contratos privados, relación con el copyright

La licencia es el tipo de contrato privado que regula el uso de determinado software.

Sus partes son un editor que distribuye su material al usuario. Le concede ciertos derechos e impone restricciones. En general son contratos de adhesión. El contrato muchas veces debe ser ratificado cuando se instala un programa, apretando el botón correspondiente, o por el simple hecho de abrir el sobre que lo contiene.

Además estos acuerdos están respaldados por la legislación sobre copyright que penaliza sus violaciones y establece derechos por defecto. La legislación de protección al consumidor muchas veces limita el tipo de contratos posibles en el caso de que sean onerosos.

En el caso del Software Libre, los autores o editores indican que se distribuye bajo una licencia particular que otorga amplios derechos a los usuarios.

En el caso de dominio público sus autores especifican que su software no está sujeto a ninguna restricción y ceden a la comunidad todos sus derechos. Esto es inconveniente porque terceros pueden tomar el software, modificarlo y cerrarlo. Tienen hasta el derecho de cerrarlo. Lo mismo pasa con el Software Libre bajo licencias no copyleft.

La mayoría de las licencias de software propietario prohíben técnicas de ingeniería inversa, descompilar, desensamblar el producto e, incluso, su traducción a otras lenguas. Básicamente deniegan el derecho a cualquier modificación o mejora del mismo, lo que queda exclusivamente en manos del fabricante. La mayoría de las licencias tampoco permiten distribuirlo o usarlo en más de un PC.

En un mundo regido por el copyright (Copyrestrictions), el software, aún el binario, se considera igual que un libro o una canción.

El software propietario se licencia de tal modo que sólo puede ejecutarse en un determinado tipo de computadora, o sistema operativo. La elección de qué combinación de computadora y sistema operativo queda enteramente a discreción del propietario del programa: nadie puede obligarlo a hacer que su programa esté disponible para tal o cual plataforma, y como la licencia es meramente de uso, y no de modificación, tampoco es posible llevarlo uno mismo a la plataforma de elección. Este hecho, combinado con el monopolio inevitable al que conduce el efecto de uso por todos, hace que la elección del usuario esté dictada por las decisiones del propietario del software dominante, en vez de ser hecha en base a sus propias necesidades. Esto es claramente visible en el mercado de computadoras de escritorio, en el que la predominancia de Windows ha convertido a las computadoras basadas en procesadores de la familia Intel aparezcan como la única alternativa viable.

[DH] Derecho-habiente

Es aquel que ostenta los derechos de copia según la legislación. Autor es quien crea la obra. El derecho-habiente de una obra creada por un empleado es su empleador.

[VS] Vender Software

La venta de un software es asimilable a la transferencia de los derechos de autor. Habitualmente el software no se vende, incluso, lo más común es que los programadores trabajen en relación de dependencia, por lo que su software es originariamente apropiado por sus empleadores.

Los editores generalmente **licencian** el software a sus usuarios, mediante un contrato particular que transfiere derechos de uso limitados y condicionados. Muchas veces es mejor referirse a la "distribución de copias de un programa" sea por una monto de dinero o no.

[PI] Propiedad Intelectual

Se suele englobar en esta categoría regímenes muy diferentes como las patentes, marcas, secretos industriales o derechos de autor, los cuales son más diferentes que similares. Mezclar estos sistemas legales y sacar conclusiones sin ser conciente de ello, lleva a generalizaciones incorrectas.

Existe una campaña tendiente a establecer la categoría de "propiedad intelectual" como unificador de estos sistemas. Esta concepción intenta establecer y asume que las ideas son apropiables, que lo incontable y copiable infinitamente a costo marginal tendiente a 0 es escaso y por lo tanto un bien económico. El concepto de Propiedad Intelectual es un término cargado de ideología y conlleva como presunción escondida, que la forma de pensar más natural sobre la copia está basada en una analogía con objetos físicos y nuestras ideas de ellos como propiedad.

Pero esta analogía esconde la diferencia crucial entre objetos materiales e información: la información puede copiarse y compartirse casi sin esfuerzo, mientras que los objetos materiales no pueden serlo. Usar este término es ignorar la mencionada diferencia.

Incluso el sistema legal de los Estados Unidos de América no acepta por completo esta analogía, debido a que no trata los "derechos de copia" (copyrights) sólo como derechos de propiedad de objetos físicos.

No debemos usar esta terminología sin saber o ser concientes de los conceptos que esconde, es mejor evitar el término "propiedad intelectual" en las palabras y los pensamientos.

Es mejor, utilizar los términos de copyright|copyrestrictions|derechos de autor, patentes, marcas, o el sistema legal específico del cual se esté hablando.

Es correcto hablar de derechos emergentes de la autoría de ideas, pero no de propiedad de las mismas.

[DA] Derechos de Autor, copyrights o copyrestrictions

Son un conjunto de concesiones que los Estados del mundo le dan a los editores de obras intelectuales con respecto a las mismas. Es un sistema que restringe el derecho natural de copiar las ideas que es formativo de las culturas humanas, mediante el imperio de la ley y la policía del Estado. Este sistema se justificó en algún momento para permitir la financiación de las editoriales e imprentas. Hoy, en el mundo digital es anacrónico y se constituye en un freno a la cooperación y el crecimiento, ya que cualquiera con muy pocos recursos puede editar libros y canciones y distribuirlas en segundos a todo el planeta.

La propaganda de las empresas asigna estas concesiones a derechos de copia para la gente, cuando en realidad son restricciones de copia para favorecer a los derecho-habientes o editores de las mismas. Mientras en general los autores cobran un monto pequeño por su obra, las editoriales se benefician enormemente con este sistema quedándose con la ganancia del falso "capital intelectual que poseen" .

Estas externalidades a las economías que restringen la ubicuidad de las ideas, dan un valor económico a elementos artificialmente escasos y naturalmente ilimitados.

En cuanto a la aplicación de este mecanismo de apropiación de ideas para el software es doblemente chocante cuando se usa para software binario, ya que al contrario de los libros y la música no se accede al

lenguaje humano con que fueron creados.

El hecho de aplicarse a código cerrado lo pone mas cerca de los secretos industriales que de obras intelectuales utilizables por la humanidad.

El software debería considerarse más como idea científica y regirse por sus cánones de conocimiento compartido universal. El Software Libre pone las cosas en su lugar, al proveer una alternativa que poco a poco logrará que el software sea un commodity libremente disponible sin costo de licenciamiento, creando una opción de mayor calidad y menor costo y por lo tanto, con obvias ventajas de mercado.

También se usa el término "protección" para describir el "copyright". Esta palabra expresa de cierta forma "prevención de la destrucción o el sufrimiento"; por lo tanto, impulsa a la gente a identificarse con el propietario y el publicista quienes se benefician del "copyright", en lugar de identificarse con los usuarios quienes son los restringidos por este.

Es fácil evitar el término "protección" y utilizar términos neutrales en su lugar. Por ejemplo, en lugar de "La protección del copyright está vigente por un tiempo prolongado", se puede decir, "El copyright está vigente por un tiempo prolongado".

Para criticar el copyright en lugar de apoyarlo y ajustarlo a su real funcionalidad, se puede emplear el término "copyrestrictions".

[H] Hurto, Robo

Los apologistas del copyright emplean con frecuencia palabras como "robo" y "hurto" para describir la violación del copyright. Al mismo tiempo, nos piden tratar el sistema legal como una autoridad en ética: si copiar es prohibido, debe estar equivocado.

Así es pertinente mencionar que muchos sistemas legales rechazan la idea de que la violación del copyright es "hurto". Los abogados del copyright que usan términos como "robo" están malinterpretando la autoridad que ellos mismos ejercen.

La idea de que las leyes deciden qué esta bien o mal es una equivocación en general. Las leyes son, en su mejor concepto, un esfuerzo por lograr justicia; decir que las leyes definen justicia o conducta ética es dar vuelta a todo.

[P] Piratería, o compartir las ideas

Los publicistas de las multinacionales frecuentemente se refieren a la copia prohibida como "piratería". De esta forma, ellos expresan indirectamente que hacer copias ilegales es éticamente equivalente a atacar barcos en altamar, secuestrar y asesinar a la gente que viaja en ellos. Han instalado en el sentido común y en el lenguaje la palabra piratería, con toda la carga ideológica que esta conlleva.

La copia ilegal no es como secuestrar y asesinar, entonces no se debe usar la palabra "piratería" para describirla. Términos no cargados ideológicamente como "copia prohibida, no autorizada, o ilegal" están disponibles para utilizar, en lugar de ello. Inclusive es preferible utilizar un término positivo tal como "compartir información con tu vecino".

[PS] Patentes de software

La aplicación de las patentes al software puede constituir un desafío importante al movimiento de Software Libre. Patentar es caro, y los grupos de desarrollo libre no podrían competir con las multinacionales que ven a las patentes como un mecanismo de defensa y patentan miles de ideas ridículas por año.

En una decisión histórica, la Unión Europea ratificó la no patentabilidad del software luego de manifestaciones y una acción concertada de los activistas del Software Libre, cambiando el sentido del

proyecto originariamente presentado a su parlamento.

Se reproduce una carta del Profesor Knuth a la oficina de patentes de su gobierno que resume de manera admirable lo ridículo del concepto de patentes de software.

Original en inglés: <http://www.progfree.org/Patents/knuth-to-pto.txt>

Traducción por: Colibrí - <http://bachue.com/colibri/patentes/knuth-pto/knuth-pto.es.txt>

Estimado Comisionado:

En nombre mío y de muchos otros científicos informáticos, quisiera solicitarle que reconsidere la política actual de otorgar patentes de procesos computacionales. Noto una ansiedad considerable en la comunidad informática debido a que las decisiones de las cortes de patentes y de la Oficina de Patentes y Marcas Registradas están haciendo que la vida de los programadores sea mucho más difícil.

En el periodo comprendido entre 1945-1980 se aceptaba que la ley de patentes no aplicaba al software. Ahora algunas personas han recibido patentes por algoritmos de importancia práctica --por ejemplo, compresión Lempel-Ziv y la encriptación de llave pública RSA--, y están utilizando las leyes para evitar que otros programadores los usen.

Este es un cambio enorme con respecto a la política anterior, que permitió la revolución computacional, y temo que perjudicará a la sociedad. Para mí es clarísimo que este cambio habría tenido un efecto profundamente negativo en mi trabajo: Por ejemplo, desarrollé un software llamado TeX que actualmente es usado para producir más del 90% de todos los libros y revistas matemáticos y físicos, y para producir cientos de miles de reportes técnicos en todas las disciplinas científicas. Si las patentes de software hubieran sido frecuentes en 1980, no habría podido crear tal sistema; probablemente ni siquiera habría considerado hacerlo y tampoco puedo imaginar a nadie que se hubiera atrevido a hacerlo.

Me han dicho que las cortes están tratando de hacer una distinción entre los algoritmos que son matemáticos y los que no lo son. Para un científico de la computación ésto no tiene sentido: nada es más matemático que un algoritmo. Un algoritmo es un concepto abstracto, que no tiene ninguna relación con las leyes físicas del universo.

Tampoco es posible distinguir entre algoritmos "numéricos" y "no numéricos", como si los números tuvieran alguna característica que los distinguiera de las otras formas de información precisa. Todos los datos son números y todos los números son datos. Los matemáticos trabajan mucho más con entidades simbólicas que con números.

Por este motivo, la idea de aprobar leyes que afirmen que algunas clases de algoritmos pertenecen a las matemáticas y otras no me parece tan absurda como los intentos de la legislatura de Indiana en el siglo XIX de aprobar una ley diciendo que la relación entre la circunferencia y el diámetro de un círculo es exactamente 3, y no aproximadamente 3.1416. Es como la iglesia medieval decretando que el sol da vueltas sobre la tierra. Las leyes hechas por el hombre pueden ser muy útiles pero no cuando contradicen verdades fundamentales.

El congreso decidió sabiamente hace mucho tiempo que las entidades matemáticas no pueden ser patentadas. Claramente, nadie podría aplicar las matemáticas si fuese necesario pagar un valor por una licencia cada vez que se use el teorema de Pitágoras. Las ideas algorítmicas básicas, que hoy en día muchos están patentando, son tan fundamentales que las consecuencias amenazan con ser equiparables a las que se tendría permitir a los autores patentaran individualmente palabras y conceptos. Los novelistas y columnistas no podrían escribir historias excepto en los casos en que las editoriales fueran autorizadas por los propietarios de las palabras. Los algoritmos son tan básicos para el software como lo son las palabras para los escritores: son las piezas fundamentales que se necesitan para armar productos interesantes. ¿Que pasaría si los abogados pudiesen patentar sus métodos de defensa o si la justicias de la Corte Suprema pudiesen patentar sus antecedentes?

Soy consciente de que las cortes de patentes hacen sus mejores esfuerzos para servir a la sociedad cuando formulan las leyes de patentes. La oficina de patentes ha cumplido su misión de manera admirable con respecto a aspectos tecnológicos que involucran leyes físicas concretas en vez de leyes abstractas del pensamiento. Yo mismo tengo patentes en algunos dispositivos de hardware. Pero creo firmemente que la tendencia reciente de patentar algoritmos solamente beneficia a un reducido número de abogados e inventores mientras que perjudica seriamente a la mayoría de personas que quieren hacer cosas útiles con computadores.

Cuando pienso en los programas para computadoras que necesito a diario para trabajar me doy cuenta de que ninguno existiría si las patentes de software hubiesen sido comunes en los 1960s y 1970s. Cambiar las reglas ahora tendrá el efecto de congelar el progreso en su estado actual. Si las tendencias actuales continúan, el único recurso para la mayoría de los brillantes desarrolladores de software de Estados Unidos será dedicarse a otras labores o emigrar. Los Estados Unidos perderán pronto su posición dominante.

Por favor hagan lo que puedan para retroceder esta alarmante tendencia. Hay formas mucho mejores para proteger los derechos de propiedad intelectual de los desarrolladores de software que quitarles el derecho a usar las piezas fundamentales con las que se construye el software.

Sinceramente,

Donald E. Knuth

Professor Emeritus

[IT] Informática traidora o

TCPA, TCG, Palladium, NGSCB, DRM (explicaciones mas adelante)

Longhorn (nuevo Windows), Centrino (Tecnología Intel)

1.- ¿Qué es esta historia de la "Informática Fiable"?

El Grupo para la Informática Fiable ([Trusted Computing Group](#) – TCG) es una alianza de Microsoft, Intel, IBM, HP, y AMD que promueve un estándar para un ordenador "más seguro". Su definición de "seguridad" es controvertida; las máquinas construidas según sus especificaciones serán mas fiables desde el punto de vista de los proveedores de software y la industria del contenido, pero menos fiable desde el punto de vista de los dueños. De hecho, las especificaciones TCG transfieren el control último del ordenador a quienquiera que escribiera el software que este ejecuta. (Sí, incluso más que ahora)

Al proyecto TCG se le conoce por una variedad de nombres. "Informática Fiable" (Trusted Computing) fue el original, que todavía usa IBM, mientras que Microsoft lo denomina Trustworthy Computing (NT: en castellano tendría un sentido similar a informática fiable) y la Free Software Foundation (FSF) lo llama [Informática Traidora](#) – treacherous computing. De aquí en adelante, lo llamaré simplemente TC. Entre otros nombres también puedes encontrar TCPA (el nombre original del TCG), [Palladium](#) (el antiguo nombre de Microsoft para la [versión](#) que se publicaría en el 2004) y [NGSCB](#) (el nuevo nombre de Microsoft). Intel en este momento lo empieza a denominar "Informática más segura" (Safer computing). Muchos observadores creen que esta confusión es deliberada – los promotores quieren distraer la atención sobre lo que TC hace realmente.

2.- ¿Qué hace TC, en castizo?

TC aporta una plataforma informática donde las aplicaciones no pueden ser alteradas o modificadas, y donde éstas se pueden comunicar de forma segura con su fabricante y entre sí. Su aplicación original era el [Control de Derechos Digitales](#) (digital rights management - DRM): Disney podrá venderte DVDs que sólo se descodifiquen y ejecuten en plataformas TC, pero que no podrás copiar. La industria discográfica podrá venderte descargas de música que no podrás intercambiar. Podrán venderte CDs que sólo podrás oír 3 veces, o solamente el día de tu cumpleaños. Toda una nueva gama de posibilidades en marketing a su alcance.

TC hará también muy difícil ejecutar Software sin licencia. En la primera versión de TC, el software pirateado se podía detectar y borrar de forma remota. Desde aquello, Microsoft ha denegado en algunas ocasiones que quisiera que TC hiciera esto, pero en el [WEIS 2003](#) un director senior se negó a desmentir que luchar contra la piratería es una meta: "Ayudar a que la gente ejecute software robado simplemente no es nuestra meta en la vida". Los mecanismos ahora propuestos son mas sutiles, sin embargo. TC protegerá los [procesos de registro](#) de las aplicaciones, de tal forma que el software sin licencia esté bloqueado. Además, las aplicaciones TC funcionarán mejor con otras aplicaciones TC, de tal forma que ya no merecerá la pena usar aplicaciones no-TC (incluyendo las piratas). Del mismo modo, algunas aplicaciones TC podrán rehusar abrir ficheros de viejas aplicaciones cuyos números de serie hayan sido prohibidos. Si Microsoft opina que tu copia de Office es pirata, y tu gobierno local se cambia a TC, entonces los documentos que intercambies con ellos se pueden volver ilegibles. TC también hace más fácil alquilar software que comprarlo; si dejas de pagar el alquiler, no solamente el software dejará de funcionar, si no probablemente también los ficheros creados con él. Así que si dejas de pagar actualizaciones para el Media Player, puede que pierdas acceso a todas las canciones que compraste con él.

Durante años, Bill Gates ha soñado con una forma de [hacer pagar a los chinos por el software que usan](#), esta puede ser la respuesta a sus plegarias.

Hay muchas otras posibilidades. Los Gobiernos serán capaces de ajustar las cosas de tal forma que todos los documentos de Microsoft Word creados en PCs de funcionarios civiles sean 'clasificados' y no se puedan filtrar electrónicamente a la Prensa. Los sitios Web de subastas pueden hacer obligatorio el uso de TCPA/Palladium para

pujar. Hacer trampas en juegos en línea también puede ser más difícil.

También hay desventajas. Por ejemplo, puede que haya censura remota. En su forma más simple, se pueden diseñar aplicaciones para borrar música pirateada bajo petición. Si una se extrae canción protegida de una plataforma TC comprometida (crackeada) y es puesta en el Web como un fichero MP3, el reproductor de ficheros compatible con TC puede que lo detecte usando una marca de agua, informe de ello y sea informado de que debe borrarlo (del mismo modo que cualquier otro dato que viniera de esa fuente comprometida). Este modelo de negocio, denominado traitor tracing (búsqueda del traidor) ha sido investigado extensivamente por Microsoft (y otros). En general, los objetos digitales creados usando sistemas TC, permanecen bajo control de sus creadores, en lugar de bajo control de los dueños de las máquinas que los albergan (como sucede en la actualidad). Entonces, el autor de un escrito designado como difamatorio por un tribunal, puede ser requerido a censurarlo – y la empresa dueña del procesador de textos usado, ordenada a borrar el fichero si el autor se niega. Dadas estas posibilidades, debemos esperar que TC sea usado para suprimir cualquier cosa desde pornografía a textos que critican a líderes políticos.

La desventajas para las empresas es que los proveedores de software pueden hacer muy difícil cambiarse a un producto de un competidor. Por ejemplo, Word podría cifrar todos sus documentos con claves a las que solo tendrían acceso otros productos de Microsoft; de tal forma que solo tendrías acceso usando programas de esa compañía, y no con ningún otro procesador de textos de la competencia. Un bloqueo tan clamoroso probablemente sea prohibido por las autoridades antimonopolio, pero hay estrategias de bloqueo más sutiles que son mucho más difíciles de regular (Explicaré algunas de ellas más adelante).

Para ver el texto completo de preguntas frecuentes sobre informática fiable ver: <http://linuca.org/body.phtml?nIdNoticia=207>

Economía de la seguridad informática: <http://www.cl.cam.ac.uk/users/rja14/econsec.html>

[DMCA] DMCA, EUCD

La European Copyright Directive (EUCD) y la Digital Millennium Copyright Act (DMCA) <http://anti-dmca.org/>, implementan el tratado sobre Copyright de diciembre de 1996 de la Organización Mundial de la Propiedad Intelectual. (OMPI o WIPO)

Existe enorme inquietud por los efectos de restricción de libertades que su aplicación conlleva.

El software propietario pone un área que antes estaba regulada por representantes electos democráticamente en manos de las corporaciones, estableciendo una tecnocracia multinacional.

4. Economía y ecología

[LNP] Libertad, no precio

El "Software Libre" es un asunto de libertad, no de precio. Para entender el concepto, es necesario pensar en "libre" como en "libertad de expresión", no como en "consumo libre".

En inglés una misma palabra (free) significa tanto libre como gratis, lo que ha dado lugar a cierta confusión.

Cuando se habla de Software Libre, es mejor evitar términos como: "regalar" o "gratis", porque esos términos implican que lo importante es el precio, y no la libertad. En cada idioma es fundamental usar las palabras que refieran directa y concretamente al concepto de libertad, y no simplemente usar y repetir de forma mecánica alguna traducción errónea de un concepto misterioso de mercados extranjeros.

[MESL] Modelo económico del Software Libre, Economía solidaria,

El Software Libre permite construir una economía solidaria, que opere bajo el concepto de la prestación de servicios y donde los ingresos se relacionan directamente con las horas de trabajo efectivo y no con los derechos, potencialmente infinitos, de la explotación comercial de una licencia de software.

Dicha economía es de escala múltiple: en ella tienen lugar las grandes corporaciones internacionales para

proyectos de gran magnitud y las empresas unipersonales que prestan servicios a individuos, todos los puntos intermedios son válidos. Esto contrasta de manera notable con el esquema de las pocas multinacionales actuales del soft que se constituyen en recaudadoras de impuestos en todo el mundo por el uso de sus programas en cada PC bajo el poder de policía del Estado y la protección legal de las restricciones de copia (Copy-restrictions).

El acceso al código fuente, libremente visible y modificable permite construir una cultura de la cooperación donde todos participan y contribuyen a la medida de sus posibilidades y necesidades a la construcción de un acervo de conocimiento universal que va creciendo, como un edificio a partir de sus cimientos.

No hay arquitectos centrales, mas allá de lo aceptado por todos, ya que es simple y está bien visto hacerlo, crear desarrollos nuevos a partir de proyectos con cuyas decisiones no se acuerda. Esta meritocracia del consenso y con disenso aceptado es el motor del cambio. Esta es la más grande aventura colectiva de la historia mundial de la humanidad en la construcción de otro mundo mas real que posible.

Si las ideas fuesen bienes económicos el valor total del Software Libre producido en el planeta sería monstruoso.

[ET] Etica del Trabajo

El movimiento hacker posee una cultura del trabajo diferente de la cultura protestante definida por Weber [099].

El trabajo se considera más un placer que una obligación. No se siguen horarios ni rutinas. El trabajo consiste en la creación de nuevas herramientas que permiten definir nuevos espacios y mecanismos de comunicación. <http://www.hackerethic.org/>

[ESL] Ecología

El siguiente material en edición discute la vinculación entre el Software Libre y la ecología.

http://ourproject.org/cgi-bin/moin.cgi/Versi_f3n_20para_20hackers

[PROS] Prosumidores

Productor, consumidor, término tomado de la economía solidaria y los movimientos de trueque. No es utilizable para el software, ya que el software no se produce, pero sí para los servicios relacionados. Refleja la idea de comunidad que crea, mejora, comparte y usa su conocimiento con un modelo económico sustentable.

[BD] Brecha Digital

Expresa la preocupación mundial relacionada con la ubicuidad creciente de las tecnologías de la información y las comunicaciones, y las consecuencias negativas aparejadas por un acceso diferencial a las mismas. Los sectores marginados por motivos estructurales, económicos, geográficos, culturales y sociales entre otros empiezan a ser considerados analfabetos digitales.

Los ciudadanos de países en vías de desarrollo donde la renta per cápita es muy baja, deben trabajar meses y dedicar su salario completo para pagar una licencia de Microsoft Office y Windows XP. No es de extrañar que los índices de uso ilegal de software sean tan altos. Lo que está mal es la legislación no las costumbres. Esta clase de legislación impuesta a posteriori, crea de la noche a la mañana miles de criminales y no satisface los mínimos criterios para ser considerada una legislación legítima.

5. Modelos de desarrollo de software

[BC] Bazar y Catedral

Se refiere a dos modelos de desarrollo de software, uno típico del Software Libre como el Bazar y otro usado tanto en Software Libre como en el propietario denominado la Catedral. Existen muchas vinculaciones con la metodología de desarrollo “Programación Extrema” (XP).

El bazar es un método dinámico, no estructurado y comunitario, muy apto para formar comunidades en la red.

Para mas detalles ver <http://www.sindominio.net/biblioweb/telematica/catedral.html>

El modelo de cooperación utilizado por el Software Libre no es nuevo. Sin ir más lejos, el principal diccionario de referencia en lengua inglesa, el Oxford English Dictionary (OED), fue desarrollado de forma cooperativa siguiendo un modelo muy similar al usado por el Software Libre. A finales del siglo XIX, James Murray, el primer editor del OED, pidió ayuda públicamente para completar el diccionario. Casi 400 personas le enviaron información sobre palabras de la lengua inglesa y ejemplos ilustrativos de su uso de forma totalmente altruista, un material que fue recogido en dicha obra. Hoy en día, el OED sigue admitiendo colaboraciones. Texto tomado en parte de: <http://www.uoc.edu/dt/20327/index.html>

6. Organizaciones, grupos, listas y comunidades

[FSF] FSF

La Free Software Foundation (FSF) es el principal patrocinador organizativo del Proyecto GNU. Creada a tal efecto en los orígenes del movimiento, es la principal responsable del desarrollo de Software Libre en el mundo y sostenedor con coherencia del movimiento. <http://www.fsf.org/> Su referente es Richard Stallman.

[H] Hypatia

Es una organización mundial, popular y democrática para promover la adopción de políticas públicas, junto a conductas humanas y sociales, que favorezcan la libre disponibilidad, sustentabilidad y socialización de la tecnología y el conocimiento, su uso solidario y la viabilidad del modelo económico y social que lo construya en términos de igualdad e inclusión de todos los seres humanos y los pueblos del mundo.

Considera que el Software Libre es un camino hacia un conocimiento socialmente justo, tecnológicamente sustentable y económicamente viable.

Participa activamente en los Foros Sociales Mundiales considerando que **OTRO MUNDO ES POSIBLE**.

Ver <http://www.hipatia.info>

[PSLB] Proyecto Software Libre Brasil

Movimiento que impulsa el desarrollo del Software Libre, primero en Porto Alegre, Rio Grande Do Sul y ahora en todo Brasil. <http://www.softwarelivre.org>

[PROP] Proposición

Lista originaria de Argentina donde se discute la legislación para el uso de Software Libre en el Estado. Una buena parte del material aquí presentado y usado en muchos lugares ha sido generado al calor de las discusiones de esta lista. <http://proposicion.org.ar/>

[LUG] LUGs

Linux Users Groups son las asociaciones de usuarios del sistema GNU/Linux. Fueron los primeros núcleos duros de hackers que trabajaron y trabajan con Software Libre.

La asociación Linux Internacional: <http://www.li.org/> suele agruparlos y contiene listados de ellos.

[HL] Hacklabs

Son comunas organizadas alrededor de laboratorios participativos, donde se difunden las ideas libertarias de Internet. Movimiento principalmente vivo en Europa. Muy activas en Software Libre y enlaces inalámbricos de Internet (WIFI).

[HC] Hacker y Cracker

Los enemigos del movimiento intentan establecer que los hackers son peligrosos y están al margen de la ley. Nada mas falso, el movimiento hacker se caracteriza por tener personas inquietas y curiosas que aportan conocimiento en forma solidaria a su comunidad. La palabra correcta para describir conductas inapropiadas de violación de información personal es Cracker.

6. Varios

[TCO] Total cost of ownership, Costo total de posesión.

Mide los costos de IT, la complejidad y las mejores practicas en el transcurso del tiempo. Incluye los costos directos (costo de adquisición, hardware, licencias, tasas, manejo, operacion) e indirectos (operación de usuarios finales, tiempo fuera de servicio, etc.) [193]

Panel de Navegación

HTML: [Inicio](#) [Definiciones](#) [Requisitos](#) [Fases](#)
[Usos](#) [Acciones](#) [Preconceptos](#) [Referencias](#)
[Autores](#)



PDF: [Global](#) [Inicio](#) [Definiciones](#) [Requisitos](#)
[Fases](#) [Usos](#) [Acciones](#) [Preconceptos](#)
[Referencias](#) [Autores](#)

Documento: <http://www.hipatia.info/docs/dsl>



[Hypatia](#) [Noticias](#)
