

República Bolivariana de Venezuela.

Poder Popular para la Educación.

U. E.: "Nuestra Señora de Lourdes".

Disciplina: Informática.

9no "A" Cristóbal Mendoza.

Software Libre

Docente:
Ángel Cordero

Integrantes
Pablo Turmero #3
Javier Páez #16

26 de Junio 2009.

Introducción

En una época en que las tecnologías de la información han tenido un enorme y raudo desarrollo, existe la necesidad de que los países de Latinoamérica se integren a sus avances y beneficios. De lo contrario, no sólo habrá aislamiento en toda una serie de actividades económicas y sociales, sino también, nos quedaremos muy atrasados respecto de los países desarrollados, debido al aumento de la denominada "brecha tecnológica" que amenaza con una serie de efectos parecidos al analfabetismo: falta de oportunidades, pobreza, subdesarrollo consolidado y de difícil superación, entre otras.

Las tecnologías de la información constituyen una de las bases más importantes del desarrollo, no solo en su aspecto material, sino también con serias relaciones con programas y políticas educativos, de defensa nacional, de promoción del empleo, salud, capacitación y superación de la pobreza.

No obstante, las tecnologías de la información ofrecen un abanico espléndido y diverso de soluciones tecnológicas, cuyo nivel de eficiencia e idoneidad sobre y baja, en un movimiento constante. Tal dinamismo, parece estar inclinándose favorablemente a un tipo de tecnologías denominadas software libre, de fuente abierta, o simplemente, como optamos por denominar en este proyecto de ley, programas de libre distribución. Para este caso, creemos necesario definir correctamente estos términos.

Índice

-Introducción.....	2
-Software Libre.....	4
-Definición legal.....	4
-Historia del Software libre.....	5
-Libertades del Software Libre.....	6
-Tipos de Licencia.....	8
-Qué no es Software Libre.....	10
- Ventajas del Software Libre.....	10
- Desventajas del software libre.....	11
- Decisiones que afecta el uso del Software Libre.....	12
- Maneras de obtener software libre.....	13
- Leyes relacionadas.....	13
- Entidades relacionadas con el Software.....	14
- Importancia de afrontar el estudio del Software Libre.....	14
-Conclusión.....	16
-Anexos.....	17
-Bibliografía.....	20

Software libre

Es la denominación del software que respeta la libertad de los usuarios sobre su producto adquirido y, por tanto, una vez obtenido puede ser usado, copiado, estudiado, modificado y redistribuido libremente. Según la *Free Software Foundation*, el software libre se refiere a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software; de modo más preciso, se refiere a cuatro libertades de los usuarios del software: la libertad de usar el programa, con cualquier propósito; de estudiar el funcionamiento del programa, y adaptarlo a las necesidades; de distribuir copias, con lo cual se puede ayudar a otros, y de mejorar el programa y hacer públicas las mejoras, de modo que toda la comunidad se beneficie (para la segunda y última libertad mencionadas, el acceso al código fuente es un requisito previo).

El software libre suele estar disponible gratuitamente, o al precio de coste de la distribución a través de otros medios; sin embargo no es obligatorio que sea así, por lo tanto no hay que asociar software libre a "software gratuito" (denominado usualmente freeware), ya que, conservando su carácter de libre, puede ser distribuido comercialmente ("software comercial"). Análogamente, el "software gratis" o "gratuito" incluye en ocasiones el código fuente; no obstante, este tipo de software *no es libre* en el mismo sentido que el software libre, a menos que se garanticen los derechos de modificación y redistribución de dichas versiones modificadas del programa.

Tampoco debe confundirse software libre con "software de dominio público". Éste último es aquel software que no requiere de licencia, pues sus derechos de explotación son para toda la humanidad, porque pertenece a todos por igual. Cualquiera puede hacer uso de él, siempre con fines legales y consignando su autoría original. Este software sería aquél cuyo autor lo dona a la humanidad o cuyos derechos de autor han expirado, tras un plazo contado desde la muerte de éste, habitualmente 70 años. Si un autor condiciona su uso bajo una licencia, por muy débil que sea, ya no es del dominio público.

-Definición legal

Programa de computación cuya licencia garantiza al usuario acceso al código fuente del programa y lo autoriza a ejecutarlo con cualquier propósito, modificarlo y redistribuir tanto el programa original como sus modificaciones en las mismas condiciones de licenciamiento acordadas al programa original, sin tener que pagar regalías a los desarrolladores previos (Decreto 3390, artículo 2).

Los Programas Informáticos Libres son aquellos que cumplen con las siguientes cuatro libertades básicas:

Libertad Cero: "usar el programa con cualquier propósito". Es decir, el ejercicio de esta libertad implica que lo podemos utilizar con cualquier fin, ya sea educativo, cultural, comercial, político, social, etc. Esta libertad deriva de que hay ciertas licencias que restringen el uso del software a un determinado propósito, o que prohíben su uso para determinadas actividades.

Libertad Uno: "Estudiar cómo funciona el programa, y adaptarlo a sus necesidades". Significa que podemos estudiar su funcionamiento (al tener acceso al código fuente) lo que nos va a permitir, entre otras cosas: descubrir funciones ocultas, averiguar cómo realiza determinada tarea, descubrir que otras posibilidades tiene, que es lo que le falta para hacer algo, etc. El adaptar el programa a mis necesidades implica que puedo suprimirle partes que no me interesan, agregarle partes que considero importantes, copiarle una parte que realiza una tarea y añadirla a otro programa, etc.

Libertad Dos: "Distribuir copias". Quiere decir que somos libres de redistribuir el programa, ya sea gratis o con algún costo, ya sea por email, FTP o en CD, ya sea a una persona o a varias, ya sea a un vecino o a una persona que vive en otro país, etc.

Libertad Tres: "Mejorar el programa, y liberar las mejoras al público". Es la libertad de hacer mejor el programa, es decir que podemos hacer menores los requerimientos de hardware para funcionar, que tenga mayores prestaciones, que ocupe menos espacio, que tenga menos errores, entre otras modificaciones. El poder liberar las mejoras al público quiere decir que si realizamos una mejora que permita un requerimiento menor de hardware, o que haga que ocupe menos espacio, soy libre de poder redistribuir ese programa mejorado, o simplemente proponer la mejora en un lugar público (un foro de noticias, una lista de correo, un sitio Web, un FTP, un canal de Chat).

-Historia del Software libre:

Entre los años 60 y 70 del Siglo XX, 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 usarlos. En dicha cultura, era común que los programadores y desarrolladores de software compartieran libremente sus programas unos con otros. Este comportamiento era particularmente habitual en algunos de los mayores grupos de usuarios de la época, como DECUS (grupo de usuarios de computadoras DEC). A finales de los 70, las compañías iniciaron el hábito de imponer restricciones a los usuarios, con el uso de acuerdos de licencia.

Allá por el 1971, cuando la informática todavía no había sufrido su gran boom, las personas que hacían uso de ella, en ámbitos universitarios y empresariales, creaban y *compartían el software* sin ningún tipo de restricciones.

Con la llegada de los años 80 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 realizar modificaciones a dicho software.

En caso de que algún usuario o programador encontrase algún error en la aplicación, lo único que podía hacer era darlo a conocer a la empresa desarrolladora para que esta lo solucionara. Aunque el programador estuviese capacitado para solucionar el problema y lo deseara hacer sin pedir nada a cambio, el contrato le impedía que mejorase el software.

El mismo Richard Stallman cuenta que por aquellos años, en el laboratorio donde trabajaba, habían recibido una impresora donada por una empresa externa. El dispositivo, que era utilizado en red por todos los trabajadores, parecía no funcionar a la perfección, dado que cada cierto tiempo el papel se atascaba. Como agravante, no se generaba ningún aviso que se enviase por red e informase a los usuarios de la situación.

La pérdida de tiempo era constante, ya que en ocasiones, los trabajadores enviaban por red sus trabajos a imprimir y al ir a buscarlos se encontraban la impresora atascada y una cola enorme de trabajos pendientes. Richard Stallman decidió arreglar el problema, e implementar el envío de un aviso por red cuando la impresora se bloqueara. Para ello necesitaba tener acceso al código fuente de los controladores de la impresora. Pidió a la empresa propietaria de la impresora lo que necesitaba, comentando, sin pedir nada a cambio, qué era lo que pretendía realizar. La empresa se negó a entregarle el código fuente.

En ese preciso instante, Richard Stallman se vio en una encrucijada: debía elegir entre aceptar el nuevo software privativo firmando acuerdos de no revelación y acabar desarrollando más software privativo con licencias restrictivas, que a su vez deberían ser más adelante aceptadas por sus propios colegas.

Con este antecedente, en 1984, Richard Stallman comenzó a trabajar en el proyecto GNU, y un año más tarde fundó la Free Software Foundation (FSF). Stallman introdujo la definición de *free software* y el concepto de "*copyleft*", que desarrolló para otorgar libertad a los usuarios y para restringir las posibilidades de apropiación del software.

-Libertades del Software Libre:

De acuerdo con tal definición, el software es "libre" si garantiza las siguientes libertades:¹

- Libertad 0: la libertad de usar el programa, con cualquier propósito.

- Libertad 1: la libertad de estudiar cómo funciona el programa y modificarlo, adaptándolo a tus necesidades.
- Libertad 2: la libertad de distribuir copias del programa, con lo cual puedes ayudar a tu prójimo.
- Libertad 3: la libertad de mejorar el programa y hacer públicas esas mejoras a los demás, de modo que toda la comunidad se beneficie.

Las libertades 1 y 3 requieren acceso al código fuente porque estudiar y modificar software sin su código fuente es muy poco viable.

Ciertos teóricos usan este cuarto punto (libertad 3) para justificar parcialmente las limitaciones impuestas por la licencia GNU GPL frente a otras licencias de software libre (ver Licencias GPL). Sin embargo el sentido original es más libre, abierto y menos restrictivo que el que le otorga la propia situación de incompatibilidad, que podría ser resuelta en la próxima versión 3.0 de la licencia GNU GPL, causa en estos momentos graves perjuicios a la comunidad de programadores de software libre, que muchas veces no pueden reutilizar o mezclar códigos de dos licencias distintas, pese a que las libertades teóricamente lo deberían permitir.

En el sitio web oficial de OSI está la lista completa de las licencias de software libre actualmente aprobadas y tenidas como tales.³

El término software no libre se emplea para referirse al software distribuido bajo una licencia de software más restrictiva que no garantiza estas cuatro libertades. Las leyes de la propiedad intelectual reservan la mayoría de los derechos de modificación, duplicación y redistribución para el dueño del *copyright*; el software dispuesto bajo una licencia de software libre rescinde específicamente la mayoría de estos derechos reservados.

La definición de software libre no contempla el asunto del precio; un eslogan frecuentemente usado es "*libre como en libertad, no como en cerveza gratis*" o en inglés "*Free as in freedom, not as in free beer*" (aludiendo a la ambigüedad del término inglés "*free*"), y es habitual ver a la venta CDs de software libre como distribuciones GNU/Linux. Sin embargo, en esta situación, el comprador del CD tiene el derecho de copiarlo y redistribuirlo. El software gratis puede incluir restricciones que no se adaptan a la definición de software libre por ejemplo, puede no incluir el código fuente, puede prohibir explícitamente a los distribuidores recibir una compensación a cambio, etc.

Para evitar la confusión, algunas personas utilizan los términos "libre" y "gratis" para evitar la ambigüedad de la palabra inglesa "free". Sin embargo, estos términos alternativos son usados únicamente dentro del movimiento del software libre, aunque están extendiéndose lentamente hacia el resto del mundo. Otros defienden el uso del término *open source software* (software de código abierto,

también llamado de fuentes abiertas). La principal diferencia entre los términos "open source" y "free software" es que éste último tiene en cuenta los aspectos éticos y filosóficos de la libertad, mientras que el "open source" se basa únicamente en los aspectos técnicos.

En un intento por unir los mencionados términos que se refieren a conceptos semejantes, se está extendiendo el uso de la palabra "FLOSS" con el significado de "Free - Libre - Open Source Software" e, indirectamente, también a la comunidad que lo produce y apoya.

-Tipos de 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". Pueden existir tantas licencias como acuerdos concretos se den entre el autor y el licenciataria. Desde el punto de vista del software libre, existen distintas variantes del concepto o grupos de licencias:

•Licencias GPL

Una de las más utilizadas es la *Licencia Pública General de GNU* (GNU GPL). El autor conserva los derechos de autor (copyright), y permite la redistribución y modificación bajo términos diseñados para asegurarse de que todas las versiones modificadas del software permanecen bajo los términos más restrictivos de la propia GNU GPL. Esto hace que sea imposible crear un producto con partes no licenciadas GPL: el conjunto tiene que ser GPL.

Es decir, la licencia GNU GPL posibilita la modificación y redistribución del software, pero únicamente bajo esa misma licencia. Y añade que si se reutiliza en un mismo programa código "A" licenciado bajo licencia GNU GPL y código "B" licenciado bajo otro tipo de licencia libre, el código final "C", independientemente de la cantidad y calidad de cada uno de los códigos "A" y "B", debe estar bajo la licencia GNU GPL.

En la práctica esto hace que las licencias de software libre se dividan en dos grandes grupos, aquellas que pueden ser mezcladas con código licenciado bajo GNU GPL (y que inevitablemente desaparecerán en el proceso, al ser el código resultante licenciado bajo GNU GPL) y las que no lo permiten al incluir mayores u otros requisitos que no contemplan ni admiten la GNU GPL y que por lo tanto no pueden ser enlazadas ni mezcladas con código gobernado por la licencia GNU GPL.

En el sitio web oficial de GNU hay una lista de licencias que cumplen las condiciones impuestas por la GNU GPL y otras que no.

Aproximadamente el 60% del software licenciado como software libre emplea una licencia GPL.

•Licencias estilo BSD

Llamadas así porque se utilizan en gran cantidad de software distribuido junto a los sistemas operativos BSD. 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, pero permite la libre redistribución y modificación, incluso si dichos trabajos tienen propietario. 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. Otras opiniones están orientadas a destacar que este tipo de licencia no contribuye al desarrollo de más software libre (normalmente utilizando la siguiente analogía: "una licencia BSD es más libre que una GPL si y sólo si se opina también que un país que permita la esclavitud es más libre que otro que no la permite").

•Licencias estilo MPL y derivadas

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 es Software Libre y promueve eficazmente la colaboración evitando el efecto "viral" de la GPL (si usas código licenciado GPL, tu desarrollo final tiene que estar licenciado GPL). Desde un punto de vista del desarrollador la GPL presenta un inconveniente en este punto, y lamentablemente mucha gente se cierra en banda ante el uso de dicho código. No obstante la MPL no es tan excesivamente permisiva como las licencias tipo BSD. Estas licencias son denominadas de copyleft débil. La NPL (luego la MPL) fue la primera licencia nueva después de muchos años, que se encargaba de algunos puntos que no fueron tenidos en cuenta por las licencias BSD y GNU. En el espectro de las licencias de software libre se la puede considerar adyacente a la licencia estilo BSD, pero perfeccionada.

•Copyleft

Hay que hacer constar que el titular de los derechos de autor (copyright) de un software bajo licencia copyleft puede también realizar una versión modificada bajo su copyright original, y venderla bajo cualquier licencia que desee, además de distribuir la versión original como software libre. Esta técnica ha sido usada como un modelo de negocio por una serie de empresas que realizan software libre (por ejemplo MySQL); esta práctica *no* restringe ninguno de los derechos otorgados a

los usuarios de la versión copyleft. También podría retirar todas las licencias de software libre anteriormente otorgadas, pero esto obligaría a una indemnización a los titulares de las licencias en uso. En España, toda obra derivada está tan protegida como una original, siempre que la obra derivada parta de una autorización contractual con el autor. En el caso genérico de que el autor retire las licencias "copyleft", no afectaría de ningún modo a los productos derivados anteriores a esa retirada, ya que no tiene efecto retroactivo. En términos legales, el autor no tiene derecho a retirar el permiso de una licencia en vigencia. Si así sucediera, el conflicto entre las partes se resolvería en un pleito convencional.

-¿Qué no es Software Libre?

- a. Software regalado: o de costo cero, pero sin el código fuente. Es el que normalmente viene en los CD's de revistas de computación o que se consigue en sitios freeware.
- b. Software con el código fuente: esto quiere expresar que el software se provee con su código fuente, pero no necesariamente brinda las libertades del Software Libre.
- c. Software de dominio público: este tipo de software no tienen licencias de uso, por lo tanto corre el peligro de dejar de serlo si alguien lo utiliza con el fin de apropiárselo.

-Ventajas del Software Libre

1. **Escrutinio Público**: Al ser muchos las personas que tienen acceso al código fuente, eso lleva a un proceso de corrección de errores muy dinámico, no hace falta esperar que el proveedor del software saque una nueva versión.
2. **Independencia del proveedor**:
 - a. Al disponer del código fuente, cualquier persona puede continuar ofreciendo soporte, desarrollo u otro tipo de servicios para el software.
 - b. No estamos supeditados a las condiciones del mercado de nuestro proveedor, es decir que si este se va del mercado porque no le conviene y discontinúa el soporte, nosotros podemos contratar a otra persona.
1. **Manejo de la Lengua**:
 - a. Traducción: cualquier persona capacitada puede traducir y adaptar un software libre a cualquier lengua.
 - b. Corrección ortográfica y gramatical: una vez traducido el software libre puede presentar errores de este tipo, los cuales pueden ser subsanados con mayor rapidez por una persona capacitada.

1. Mayor seguridad y privacidad:

- a. Los sistemas de almacenamiento y recuperación de la información son públicos. Cualquier persona puede ver y entender como se almacenan los datos en un determinado formato o sistema.
- b. Existe una mayor dificultad para introducir código malicioso como ser: espía (p/ej. capturador de teclas), de control remoto (p/ej. Troyano), de entrada al sistema (p/ej. puerta trasera), etc.

1. **Garantía de continuidad:** el software libre puede seguir siendo usado aun después de que haya desaparecido la persona que lo elaboro, dado que cualquier técnico informático puede continuar desarrollándolo, mejorándolo o adaptándolo.
2. **Ahorro en costos:** en cuanto a este tópico debemos distinguir cuatro grandes costos: de adquisición, de implantación (este a su vez se compone de costos de migración y de instalación), de soporte o mantenimiento, y de interoperabilidad. El software libre principalmente disminuye el costo de adquisición ya que al otorgar la libertad de distribuir copias la puedo ejercer con la compra de una sola licencia y no con tantas como computadoras posea (como sucede en la mayoría de los casos de software propietario). Cabe aclarar que también hay una disminución significativa en el costo de soporte, no ocurriendo lo mismo con los costos de implantación y de interoperatividad.

-Desventajas del software libre

Si observamos la situación actual, es decir la existencia mayoritaria de Software Propietario, tenemos:

1. **Dificultad en el intercambio de archivos:** esto se da mayormente en los documentos de texto (generalmente creados con Microsoft Word), ya que si los queremos abrir con un Software Libre (p/ ej. Open Office o LaTeX) nos da error o se pierden datos. Pero está claro que si Microsoft Word creara sus documentos con un formato abierto (o publico) esto no sucedería.
2. **Mayores costos de implantación e interoperabilidad:** dado que el software constituye "algo nuevo", ello supone afrontar un costo de aprendizaje, de instalación, de migración, de interoperabilidad, etc., cuya cuantía puede verse disminuida por: mayor facilidad en las instalaciones y/o en el uso, uso de emuladores (p/ej. Si el usuario utiliza Microsoft Windows, la solución seria instalar alguna distribución de GNU/Linux y luego un emulador de Windows, como Wine, VMWare. Terminal X, Win4Lin). Vale aclarar que el costo de migración esta referido al software, ya que en lo que hace a Hardware generalmente el Software Libre no posee mayores requerimientos que el Software Propietario.

-Decisiones que afecta el uso del Software Libre

1. Libertad de elección:

- a. Respecto al software: se obtiene la independencia del soporte de versiones. Es de público conocimiento que Microsoft ha dejado de ofrecer soporte de desarrollo a Windows 95 y Windows 98, para dedicarse de lleno a Windows Me, 2000, XP y .NET.
- b. Respecto al Hardware: al tener requisitos de funcionamiento no tan elevados, el usuario no está atado a la compra de una determinada computadora, o procesador, o disco rígido, o cantidad de memoria RAM, etc. Esa compra que haga el usuario solo depende de la velocidad con la que desee realizar sus tareas, no de si el software va a funcionar o no en esas condiciones.
- c. Respecto al soporte: al tener acceso al código fuente, cualquier persona idónea nos puede ofrecer soporte, no solo nuestro proveedor. En el software Propietario esto no sucede, ya que ninguna persona ajena al proveedor conoce el funcionamiento interno del mismo.
- d. Respecto a la Formación o Capacitación: la puede ofrecer cualquiera, no solo el proveedor.

1. Protección de la inversión:

- a. En Software: Los desarrollos en software siempre son aprovechables para otros desarrollos, si se licencian de manera "libre". Además, el fomento de la comunidad de usuarios supone un potencial extraordinario en cuanto a la generación de nuevo y mejor software.
- b. En Hardware: no se necesita reemplazar el hardware constantemente, porque no se necesita cambiar el software si este ya no funciona como se quiere.
- c. En Soporte: la experiencia ganada por los técnicos propios puede extenderse a otro tipo de software.
- d. En Formación: como no existen artificios para ocultar información, puede elegirse a otra persona que brinde capacitación con un precio más accesible.

1. Relación rendimiento/ Precio:

- a. Costo del Software: el software libre tiene la ventaja de no obligar a pagar una licencia por cada computadora en que se lo instale.
- b. Costo del Hardware: hay cierto Software Propietario que tiene requerimientos excesivos de hardware para funcionar. Es decir, no hay relación razonable entre la tarea que realiza y el hardware que requiere.
- c. Costo de Soporte: su costo es menor porque cualquiera lo puede ofrecer, si bien es cierto que no tenemos una calidad garantizada.
- d. Costo de Formación: su costo es menor, al no estar monopolizada. Aquí cabe el mismo comentario que el realizado con respecto al costo de soporte.

1. **Comunicación e interoperabilidad de sistemas:** el Software Libre garantiza el respeto a los estándares en los formatos, protocolos e interfaces. En cambio el Software Propietario generalmente los cambia, para obligar al usuario a cambiar de versión. Pero un costo importante en esta parte es el de adaptar los estándares cerrados a estándares abiertos (por ejemplo, XLS a XML, DOC a Tex, etc.)

-Maneras de obtener software libre

- a. **A través de copias en CD:** los que a su vez se pueden conseguir en revistas especializadas, o comprándolos en una casa de computación, o pidiéndoselos a un amigo, pariente, etc.
- b. **A través de Internet:** a su vez, por medio de FTP, sitios Web, canales de chat, foros de noticias, programas de intercambio de archivos, etc.
- c. **A través de una computadora:** en este caso, comprando una que venga con Software Libre preinstalado, ya sea de fabrica o por su vendedor.

-Leyes relacionadas

En el ámbito nacional tenemos:

- a. **Ley 11723:** es una ley compuesta por 89 artículos, sancionada en 1933 (y todavía vigente), conocida como "Ley de Propiedad Intelectual" o también como "Ley de Propiedad Científica, Literaria y Artística". Esta ley regula todo lo referente a derecho de propiedad de una obra artística, científica o literaria, derechos de coautor, enajenación o cesión de una obra, licencias, etc. Además, establece sanciones tanto pecuniarias (multa) como privativas de la libertad (prisión) a quienes violen sus normas. Su ultima reforma data de Noviembre de 1998, cuando por Ley 25036 se le introdujeron modificaciones referidas al software, para darle fin a las discusiones doctrinarias y jurisprudenciales sobre la cuestión de si el software estaba o no bajo el amparo de esta ley. Ahora establece expresamente en su Art. 1 que "... las obras científicas, literarias y artísticas comprenden los escritos de toda naturaleza y extensión, entre ellos los programas de computación fuente y objeto; las compilaciones de datos o de otros materiales, ..." y en su art. 55 bis que "La explotación de la propiedad intelectual sobre los programas de computación incluirá entre otras formas los contratos de licencia para su uso o reproducción".
- b. **Proyecto de ley sobre Software Libre:** es un proyecto presentado en Marzo de 2001 por Marcelo Luis Dragan, Diputado Nacional por la provincia de Tierra del Fuego, del Partido Acción por la República. Originalmente lleva el nombre de "Utilización de Software Libre por el Estado Nacional", y establece la obligación de usar prioritariamente Software Libre en todas las dependencias de la Administración Pública Nacional, salvo excepciones. Entre los aspectos que motivaron el proyecto, se destacan el económico (por el costo de las licencias y por la libertad de copiar que otorga el Software Libre), el moral (es conocido que

en todos los ámbitos de la Administración Pública se utiliza Software Ilegal, ya sea por cuestiones de costos, negligencia, etc., lo cual coloca al Estado como uno de los principales infractores a la Ley 11723), el cultural, el educativo, el de seguridad nacional, etc. Actualmente este proyecto se encuentra en estudio en la comisión de Legislación General y Comunicación.

-Entidades relacionadas con el Software

Hay una innumerable cantidad de organizaciones relacionadas con esta temática:

Por el lado del Software Libre: la FSF, la Organización Open Source, la Fundación Vía Libre, etc.

Por el lado del "Software Legal", tenemos dos:

1. **La BSA:** "Business Software Alliance" es una organización mundial sin fines de lucro, creada en 1988 por un grupo de empresas fabricantes de software. Tiene tres funciones fundamentales: a) Inculcar a los gobiernos y a los consumidores que pagar por usar en Software redundante en beneficio de la economía, incrementando la productividad de los trabajadores y el número de puestos en empleo. b) Emprender acciones legales contra aquellos que no cumplan con las leyes de propiedad intelectual y/o que hagan mal uso de las licencias de software. c) Promover legislaciones que protejan los derechos de propiedad intelectual, y conseguir que los gobiernos las ejecuten.

2. **La organización "Software Legal":** es una asociación civil sin fines de lucro, creada en 1992 por un grupo de empresas fabricantes de software. Tiene las mismas funciones y características de la BSA.

-Importancia de afrontar el estudio del Software Libre

1. **Económica:** el costo de las licencias de Software Propietario es bastante importante, y por la situación económica actual, imposible de afrontar de la manera que los fabricantes de Software lo piden.

2. **Legal:** el Software Libre es siempre legal, salvo contadas excepciones (p/ej., que compilemos el código fuente y lo vendamos como propietario). Por lo tanto, al utilizar este tipo de software estaremos siempre "por derecha", por lo que no seremos pasibles de multas y/o prisión.

3. **Técnica:** es sabido que Microsoft ha dejado de ofrecer soporte de desarrollo para Windows 95 y Windows 98, por lo que si hoy o mañana se descubre un error en ellos, Microsoft no está obligado a repararlo. Para solucionar esto, tendríamos dos caminos: a) Migrar a otras versiones de Sistema Operativos de Microsoft: esto lleva aparejado una serie de costos, principalmente en licencias, luego costos de implantación, soporte e interoperabilidad, y además implica volver a hacer lo mismo dentro de dos o tres años. b) Utilizar Software Libre.

4. **Laboral:** la implementación de Software Libre plantea un futuro muy prometedor para aquellas personas que sepan programar, traducir, utilizar un programa, enseñar, etc. Si tenemos que elegir entre pagar una licencia de software a un coloso informático o darle trabajo directamente a una persona, es de esperar que nos volquemos a la segunda alternativa.

Conclusión

-Es importante que como estudiantes de Ciencias Económicas vayamos cambiando nuestras costumbres con respecto al Software que utilizamos.

-Los que tienen computadora en su casa: instalando alguna distribución de GNU/Linux (tener en cuenta que pueden "convivir" con un Sistema Operativo propietario como Microsoft Windows) o utilizando Software Libre para Windows (como Open Office, LaTeX, Mozilla, FreeAmp, The GIMP, etc.)

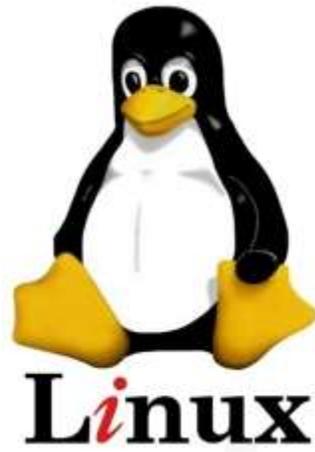
-Los que trabajan en la Administración Pública: hablando con el Jefe de Sistemas sobre esta "Filosofía del Software Libre" y sobre las posibilidades de implementación, o instalando Software Libre por nuestra cuenta pero con la debida autorización.

-Los que trabajan en el estudio de un/a Contador/a: charlando sobre este tema con el/la y sugiriéndole que le pida a su proveedor de software (Tango, Bejerman, Oracle, etc.) que desarrolle sistemas para GNU/Linux u otra plataforma.

-Los que trabajan en otra parte o los que no trabajan: hablar con compañeros de trabajo o amigos sobre este tema, compartir experiencias, sugerir trucos, etc.

-No debemos olvidarnos de que podemos ser pasibles de sanciones, y que es preferible siempre estar del lado de la ley que fuera de ella.

Anexos



Bibliografía

Páginas Web:

-www.monografías.com

-www.wikipedia.com