

Migración a Software Libre:  
Una Propuesta para la Universidad de Tarapacá de Arica  
Tesis para Optar al Grado de Magíster en Ingeniería de Software

Alumno: Jorge Jerónimo Benavides Escobillana <sup>1</sup>  
Profesor Guía: Héctor Beck Fernández <sup>2</sup>

8 de enero de 2009

<sup>1</sup>jbenavid@uta.cl

<sup>2</sup>hbeck@uta.cl

# Índice general

<b>1. Introducción</b>	<b>4</b>
1.1. El software libre	5
1.1.1. Definición del Software Libre	5
1.1.2. Historia del Software Libre	5
1.1.3. Consideraciones respecto al software libre	6
1.2. Contribución de la tesis	7
1.2.1. Objetivo general	7
1.2.2. Objetivos específicos	7
1.3. Organización del trabajo	7
<b>2. Ventajas y desventajas del software libre</b>	<b>9</b>
2.1. Ventajas del software libre	9
2.1.1. Principios éticos	9
2.1.2. Seguridad	9
2.1.3. Privacidad	11
2.1.4. Independencia tecnológica	13
2.1.5. Ventajas legales y económicas	14
2.2. Desventajas del software libre	15
2.2.1. Escasez de soporte técnico	15
2.2.2. Escasez de cursos de capacitación	15
2.2.3. Falta de software libre para determinadas áreas	15
2.2.4. Dificultad de uso de algunos programas	15
<b>3. Obstáculos para la Libertad Digital</b>	<b>17</b>
3.1. Gestión Digital de Restricciones	17
3.1.1. Tivoización	18
3.1.2. Computación Traicionera	19
3.1.3. AACCS	19
3.1.4. FairPlay	20
3.1.5. Restricciones de Windows Vista	21
3.2. Hardware sin especificaciones técnicas	22

3.2.1. BIOS no libre . . . . .	23
3.3. Formatos de archivo cerrados . . . . .	23
3.4. Leyes que prohíben el software libre . . . . .	24
3.4.1. Patentes de software . . . . .	24
3.5. Presiones de empresas de software privativo . . . . .	25
3.6. Inercia o costumbre . . . . .	25
3.7. Falta de conciencia sobre las libertades de los usuarios de software . . . . .	25
<b>4. Casos de migración a software libre</b>	<b>26</b>
4.1. Casos de migración en la administración pública . . . . .	26
4.2. Casos de migración en colegios y universidades . . . . .	28
4.3. Casos de migración en empresas . . . . .	29
<b>5. Plan de migración</b>	<b>31</b>
5.1. Objetivo . . . . .	31
5.2. Consideraciones . . . . .	31
5.3. Fases del proceso de migración a software libre . . . . .	32
5.3.1. Convencer sobre la necesidad de migrar a software libre . . . . .	32
5.3.2. Soporte técnico . . . . .	32
5.3.3. Identificar programas utilizados actualmente . . . . .	33
5.3.4. Buscar alternativas libres . . . . .	33
5.3.5. Identificar posibles obstáculos para la migración . . . . .	34
5.3.6. Utilizar software libre en los servidores . . . . .	35
5.3.7. Utilizar software libre sobre Windows . . . . .	36
5.3.8. Cambiar Windows por GNU/Linux con algunos componentes privativos . . . . .	37
5.3.9. Adquirir hardware compatible con software libre . . . . .	37
5.3.10. Utilizar GNU/Linux sin componentes privativos . . . . .	38
5.3.11. Desarrollo de software libre . . . . .	38
<b>6. Conclusión y trabajo futuro</b>	<b>39</b>
<b>Bibliografía</b>	<b>41</b>
<b>A. Glosario de términos</b>	<b>48</b>
A.1. Definiciones generales . . . . .	48
A.2. Definiciones legales . . . . .	49
A.3. Definiciones técnicas . . . . .	50
<b>B. Listado de programas libres y privativos</b>	<b>51</b>
B.1. Introducción . . . . .	51
B.2. Programas privativos utilizados en la Universidad . . . . .	52
B.3. Programas libres utilizados en la Universidad . . . . .	78

*ÍNDICE GENERAL*

3

**C. Programa para curso de capacitación en OpenOffice**

81

# Capítulo 1

## Introducción

La declaración de los Derechos Humanos [1] fue aprobada hace menos de un siglo, en el año 1948, con la intención de evitar que se repitan actos que atenten contra la libertad, los derechos y la dignidad de las personas. Sin embargo, aún no existe un acuerdo internacional sobre los derechos y libertades de las personas en la era digital, a pesar de que los computadores y las redes son cada vez más importantes en áreas como la comunicación, el trabajo, la educación, la investigación y la salud. Entonces, es importante que la sociedad empiece a considerar los derechos de las personas en este ámbito.

En la década de 1970, cuando la computación estaba en sus inicios, era común que los desarrolladores de software, tanto profesionales como aficionados, publicaran su trabajo para que otros lo puedan utilizar [2], tal como se realizaba en otras áreas de la ciencia, como la matemática y la física. Sin embargo, en la década de 1980, muchos desarrolladores de software comenzaron a apropiarse de su trabajo y solamente dejaban que los demás lo utilizaran bajo ciertas condiciones, como la prohibición de compartirlo con otros usuarios y la prohibición de mejorarlo. Por ejemplo, la empresa de software Microsoft envió una carta a un grupo de programadores aficionados que utilizaban copias no autorizadas de su programa BASIC [3], acusándolos de «robar» su software, argumentando que compartir el software es injusto, pues el desarrollador no recibe suficiente dinero a cambio. Esta forma de pensar atenta contra el espíritu de cooperación y solidaridad que existía en la comunidad científica y en la sociedad humana en general.

Para contrarrestar esta tendencia a no compartir el software, surgió el «software libre». El software libre es un movimiento ético, político y social que tiene por objetivo defender la libertad de los ciudadanos en un mundo donde los computadores afectan cada vez más nuestra forma de vivir. El software libre fomenta la solidaridad, la cooperación y la independencia de los usuarios y desarrolladores de software.

La función principal de las Universidades es la creación y la difusión del conocimiento. Por lo tanto, es deber de éstas defender la libertad de las personas de compartir el conocimiento y, en particular, el software. Es por esto que el uso de software libre debe ser una prioridad en cualquier Universidad que esté comprometida con el bienestar de la sociedad.

El presente trabajo pretende difundir la filosofía del software libre, destacando principalmente sus ventajas éticas, además de sus posibles ventajas técnicas, económicas y legales. Luego se propone un plan de migración a software libre para la Universidad de Tarapacá.

## 1.1. El software libre

En esta sección se exponen los conceptos fundamentales para comprender qué es el software libre. En el Apéndice A se definen los términos más importantes utilizados en este trabajo.

### 1.1.1. Definición del Software Libre

Richard Stallman define al software libre de la siguiente manera [4]:

«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 (libertad 0).
- La libertad de estudiar cómo funciona el programa, y adaptarlo a sus necesidades (libertad 1). El acceso al código fuente es una condición previa para esto.
- La libertad de distribuir copias, con lo que se puede ayudar a otras personas (libertad 2).
- La libertad de mejorar el programa y hacer públicas las mejoras a los demás, de modo que toda la comunidad se beneficie. (libertad 3). El acceso al código fuente es un requisito previo para esto.»

Un programa es software libre si respeta simultáneamente estas cuatro libertades.

Un programa que no respete simultáneamente estas cuatro libertades se denomina «software no libre» o «software privativo», pues priva a los usuarios de sus libertades.

El objetivo del movimiento del software libre es hacer que la sociedad rechace el software no libre, porque no respeta las libertades de las personas como usuarios de software. Para lograr este objetivo, la estrategia de este movimiento es crear software libre que reemplace al software privativo, de tal forma que los usuarios no necesiten utilizar software privativo para realizar sus tareas.

### 1.1.2. Historia del Software Libre

Según [2], en la década de 1970, era común que las empresas y los programadores compartieran el código fuente de sus programas y cooperaran unos con otros. En la década de 1980, en cambio, casi todo el software dejó de ser libre, es decir, sus autores prohibían e impedían la cooperación entre los usuarios.

Descontento con esta situación, Richard Matthew Stallman, quien trabajaba como programador en el Laboratorio de Inteligencia Artificial en el Instituto Tecnológico de Massachussets (MIT), decide publicar el manifiesto GNU [5] e iniciar el proyecto GNU [2] en 1983, lo cual da inicio al movimiento del software libre. Stallman también fundó la Free Software Foundation, creó el concepto de copyleft y es el principal autor de la Licencia Pública General de GNU (GNU GPL), la licencia de software libre más utilizada.

El objetivo del proyecto GNU (acrónimo recursivo que significa «GNU No es Unix») es crear un sistema operativo completamente libre, el sistema GNU, que permita utilizar un computador sin tener que renunciar a las cuatro libertades definidas anteriormente. En la década de 1990 ya se había escrito la mayoría del sistema GNU, solo faltaba una parte: el núcleo. De forma independiente, Linus Torvals escribió el núcleo Linux y lo publicó como software libre. Al combinar el núcleo Linux con el sistema GNU, se dio origen al sistema operativo libre llamado GNU/Linux o GNU+Linux.

### 1.1.3. Consideraciones respecto al software libre

El autor de este trabajo está de acuerdo con la mayoría de los pensamientos de Stallman ([4] y [6]) en relación a las libertades de los usuarios de software, algunos de los cuales se mencionan a continuación.

El software privativo no respeta las normas éticas de cooperación y solidaridad, presentes en la mayoría de las culturas, porque impide que las personas compartan el software. Específicamente, el software no libre limita la forma en que los usuarios utilizan su computador, no permite ser adaptado a las necesidades propias de los usuarios, impide realizar copias del software y distribuirlas o impide hacerle mejoras y publicarlas. En realidad puede haber formas de realizar estas actividades con software privativo, pero lo más probable es que sean ilegales o técnicamente muy difíciles de realizar. Por ejemplo, al no proveer el código fuente, se hace muy compleja la modificación de algún software privativo.

Una persona debe rechazar el software privativo para evitar un conflicto ético interior. Por ejemplo, cuando un amigo le pide una copia de un programa privativo, una persona debe elegir entre respetar su amistad y copiar el software, o bien respetar la licencia del programa y negarse a hacer la copia a su amigo.

Algunos programas privativos permiten realizar tareas que con el software libre resultan difíciles de llevar a cabo. Sin embargo, la humanidad ha logrado desarrollarse sin utilizar las funcionalidades o comodidades que ofrecen esos programas privativos, por lo tanto no se justifica que tengamos que sacrificar nuestras libertades para utilizar dichas funcionalidades o comodidades.

El uso de software privativo implica ciertos peligros. Al no conocer su código fuente, no podemos saber qué es lo que hace realmente un programa privativo. Por ejemplo, podría estar recolectando y enviando información privada de los usuarios a algún sitio sin la autorización de éstos. Además, el software privativo genera un monopolio, ya que solamente la empresa que crea un programa privativo puede modificarlo. De lo anterior se concluye que debemos evitar el uso del software no libre, o al menos tener conciencia del peligro que significa su uso.

El software libre, en cambio, no posee los peligros del software privativo, y otorga ventajas prácticas, además de las ventajas éticas. Al estar disponible su código fuente, se puede hacer una auditoría de este código para saber exactamente qué es lo que hace un programa libre y, por lo tanto, los datos privados de los usuarios no corren peligro de ser enviados a terceros sin autorización. Si una persona tiene conocimientos de programación, puede realizar la auditoría del código por sí misma; en caso contrario, puede contratar a alguien para que realice la auditoría del código fuente del programa.

Cuando una persona u organización utiliza software libre, tiene independencia del fabricante, es decir, puede escoger a la persona o empresa que desee para que le adapte el software a sus necesidades. Nuevamente, la disponibilidad del código fuente permite que esto sea posible. Si una persona tiene conocimientos de programación, puede modificar el software por sí misma; en caso contrario, puede contratar a alguien para que realice las modificaciones que desea al programa. Para un país, el software libre le permite desarrollar capacidad tecnológica local, lo cual es indispensable para su desarrollo.

El software libre es una alternativa real al software privativo. Si las empresas de software privativo comienzan a abusar demasiado del control que poseen sobre sus usuarios, se puede recurrir a esta alternativa. Sin embargo, es más responsable evitar una situación en la cual los usuarios se vean obligados a soportar el control de las empresas de software privativo.

Por otra parte, existen ciertas tecnologías, como la Gestión Digital de Restricciones (DRM) y

la Computación Traicionera (Treacherous Computing), que amenazan con limitar las libertades de los usuarios de software y de los ciudadanos en general. Con el DRM, por ejemplo, se le quita a los compradores de libros ciertos derechos que siempre han tenido, tales como leer un libro cuantas veces quiera, prestarlo, regalarlo, venderlo o comprar el libro de forma anónima. La computación traicionera otorga a algunas grandes corporaciones del software y de los medios, el poder de limitar lo que los usuarios pueden hacer en sus propios computadores. Así, un computador deja de ser una máquina de cálculo multipropósito, para transformarse en una máquina para vigilar y controlar a sus usuarios.

## 1.2. Contribución de la tesis

A continuación se describen el objetivo general y los objetivos específicos de este trabajo.

### 1.2.1. Objetivo general

Proponer un plan de migración a software libre para la Universidad de Tarapacá, con el fin de reemplazar por software libre el software privativo utilizado actualmente en sus actividades oficiales, de tal forma que se puedan seguir realizando las mismas tareas y se cause el mínimo de problemas de adaptación a los usuarios.

### 1.2.2. Objetivos específicos

- Exponer los peligros del software privativo
- Exponer los beneficios del software libre
- Describir la situación actual del software libre
- Diseñar un plan de migración a software libre
- Identificar los programas que utilizan las distintas unidades académicas y administrativas de la Universidad
- Buscar programas libres que ofrezcan funcionalidades similares a los programas privativos identificados en el punto anterior

## 1.3. Organización del trabajo

Una vez vistos los conceptos y el contexto del software libre, en el Capítulo 2 se explican en mayor medida los peligros del software privativo y las ventajas del software libre, desde el punto de vista de la seguridad, la independencia tecnológica, el desarrollo tecnológico e innovación local, la privacidad de los datos y la reducción de la brecha digital.

En el Capítulo 3 se explican algunos de los principales obstáculos para la libertad digital, es decir, aquellos factores técnicos, legales y políticos que amenazan al software libre y a las libertades de los usuarios.

En el Capítulo 4 se exponen casos de organizaciones que han migrado, total o parcialmente, a software libre, considerando organizaciones empresariales, gubernamentales y educativas de distintas partes del mundo.



En el Capítulo 5 se describe el diseño de un plan de migración a software libre para la Universidad de Tarapacá. Este plan consiste en varias fases, tales como el convencimiento de las autoridades, la identificación del software que se utiliza actualmente, la búsqueda de alternativas en software libre, los distintos pasos para la implantación gradual del software libre y el posible desarrollo de software libre.

Finalmente, en el Capítulo 6 se plantean las conclusiones de este trabajo y el camino a seguir en el futuro en esta área.

## Capítulo 2

# Ventajas y desventajas del software libre

En este capítulo se exponen las principales ventajas y desventajas del software libre frente al software privativo. Se analizan solamente los beneficios cualitativos del software libre, poniendo especial énfasis en el aspecto ético. Para convencer a alguien que no valore la libertad y la solidaridad, se le puede mostrar las ventajas prácticas del software libre. Un extenso estudio cuantitativo y cualitativo sobre las ventajas prácticas del software libre se encuentra en [7].

### 2.1. Ventajas del software libre

#### 2.1.1. Principios éticos

Como ya se ha mencionado, la principal ventaja del software libre es ética, pues fomenta la creación de una comunidad de usuarios y desarrolladores que comparten el software y su código fuente. Por lo tanto, el software libre respeta los principios éticos de cooperación y solidaridad que existen en la mayoría de las culturas del mundo, si no en todas. El software privativo, en cambio, divide a la comunidad al impedir por medios técnicos o legales que los usuarios compartan el software. Entonces, el software privativo no respeta los principios éticos señalados anteriormente.

El software libre, al proveer su código fuente, no limita a quienes deseen aprender a programar viendo cómo funcionan los programas que utilizan. No se trata de que todos los usuarios de software libre se conviertan en programadores, sino que la libertad de quienes quieran aprender a programar no se vea limitada.

El software libre es un movimiento ético, político y social. Por lo tanto, la razón más importante para utilizar programas que respeten nuestras libertades, es ética. Si una organización adopta esta filosofía, podrá enfrentarse con mayor determinación a las dificultades que puedan aparecer durante la migración. En cambio, si una organización escoge al software libre por otras razones, por ejemplo, económicas y técnicas, lo más probable es que se cambie a software privativo cuando aparezca alguna dificultad con el software libre, o si el software privativo se acomoda más a sus necesidades.

#### 2.1.2. Seguridad

Respecto a la seguridad, la principal ventaja del software libre es que su código fuente puede ser inspeccionado o auditado, de tal forma que los programadores pueden verificar que el software realiza exactamente lo que se espera de haga.

## Problemas de seguridad con el software privativo

El software privativo, al no incluir el código fuente, no puede ser inspeccionado<sup>1</sup>. Por lo tanto, solo los desarrolladores saben lo que hace exactamente un programa privativo, lo cual da lugar a riesgos potenciales. Por ejemplo, un programa no libre podría enviar, sin permiso del usuario, información importante a algún sitio a través de la red. Además, un programa privativo podría tener una «puerta trasera» que permita a los desarrolladores o a terceros tener acceso y controlar los computadores que ejecutan ese programa. Por lo tanto, al utilizar software que no respeta nuestras libertades, existe un peligro real de que alguien pueda espiar o controlar los computadores de los ciudadanos, del gobierno, de las empresas privadas e incluso del Ejército.

En [8] se afirma que Microsoft envía el código fuente de Windows a la Agencia de Seguridad Nacional (NSA) de Estados Unidos, la cual hace algunas modificaciones y el resultado de esta modificación es el programa que finalmente es vendido. Esto hace sospechar que cada computador con alguna versión de Windows podría tener una «puerta trasera» que permita a la NSA o a Microsoft espiar el contenido del computador o bien tomar el control del computador.

El Dr. Hugo Scolnik, un experto en criptografía y seguridad, y creador del Departamento de Computación de la Universidad de Buenos Aires, en [9] afirma al respecto:

«Cuando trabajamos con Microsoft, con cada cambio teníamos que enviar el código fuente a la NSA, donde lo compilan y le agregan lo que quieren y luego vuelve como producto que nosotros distribuimos. No sé qué es lo que le pusieron.»

Si bien Windows es software privativo, Microsoft ofrece un servicio llamado Shared Source, mediante el cual, previo pago de una cantidad de dinero, se revela parte del código fuente de Windows a los gobiernos que lo soliciten. Este código no se puede modificar y un grupo de expertos requeriría años para poder auditarlo, como se menciona en [10].

Diego Saravia, un destacado defensor del software libre, en [11] explica lo siguiente:

«El software tiene que ser auditable. Porque, ¿cuál es el problema del software no libre? Es que no es auditable. Es decir, yo no sé qué cosa le pone Microsoft a sus programas. Yo no tengo forma de saber si cuando yo mando un correo, el programa indica que también se lo mande a otro. Y esto es porque no tengo acceso al código fuente. Incluso hay mucho debate en Internet acerca de si no existen “puertas secretas” para las agencias del gobierno de Estados Unidos. Es que ese país tiene una normativa que se llama CALEA<sup>2</sup>, que obliga a todos los fabricantes de equipos de comunicaciones a poner puertas para el gobierno. Es decir, cualquier router fabricado por empresas norteamericanas tiene estos mecanismos adentro y, entonces, puede ser intervenido por agencias del gobierno.»

Efectivamente, CALEA [12] es una ley de Estados Unidos cuyo propósito está escrito explícitamente y es el siguiente:

«Las empresas de telecomunicaciones tienen el deber de cooperar en la interceptación de comunicaciones con el fin de aplicar la ley, y para otros propósitos.»<sup>3</sup>

<sup>1</sup>En realidad, puede haber software privativo cuyo código fuente sea público, pero en general eso no ocurre. De todas formas, aunque el código esté disponible, las condiciones de la licencia de un software privativo pueden poner trabas legales para la modificación, copia o distribución del software.

<sup>2</sup>Siglas de “Communications Assistance for Law Enforcement Act”

<sup>3</sup>Original en inglés: “to make clear a telecommunications carrier’s duty to cooperate in the interception of communications for Law Enforcement purposes, and for other purposes.”

Si bien esta ley es específica de Estados Unidos, también afecta a personas de todo el mundo, pues gran parte del tráfico de Internet pasa por las redes de Estados Unidos. Llama especialmente la atención la frase «y para otros propósitos», lo cual deja abierta la posibilidad de interceptar comunicaciones bajo cualquier circunstancia, como sospecha por terrorismo, por ejemplo.

Jorge Hinestroza, en [13] y [14], denuncia un caso donde se utilizaron «puertas traseras» para sabotear a la petrolera PDVSA de Venezuela. Este sabotaje ocurrió durante un paro organizado por altos funcionarios de esa empresa estatal, y significó la paralización de la producción petrolera y una situación de riesgo ecológico por derrame de petróleo en el Lago de Maracaibo. Diego Saravia indica en [11]:

«Toda la informática estaba en manos de una empresa tercerizada, que estaba controlada por gente de Inteligencia de otros países. Y cuando ocurrió el paro, toda esta informática quedó en esas manos. Hicieron cosas verdaderamente terribles desde los mecanismos de software, que fueron percibidas como parte de una guerra electrónica.»

Este grave incidente, entre otros factores, significó que el gobierno de Venezuela decidiera utilizar software libre en todas las dependencias del Estado.

### **Ventajas de seguridad con el software libre**

El uso de software libre evita muchos de los problemas de seguridad que existen al utilizar software privativo. Al estar disponible el código fuente de los programas libres, cualquier gobierno, empresa u organización puede auditar el código fuente. Además, cada vez que alguien audita algún programa libre para buscar posibles fallos de seguridad y corrige esos fallos, toda la comunidad se beneficia, pues todos pueden utilizar esa nueva versión más segura.

Por otra parte, el uso de sistemas operativos libres disminuye el riesgo de sufrir ataques de virus y otros programas maliciosos.

Es común que se piense que si no se conoce el código de un programa, entonces el programa es más seguro. A esta técnica de seguridad se le llama «seguridad por oscuridad», pues se oculta lo que realiza el algoritmo o el programa. Sin embargo, muchos expertos en seguridad señalan [15] que esta técnica es poco efectiva. La seguridad de un programa debería estar dada en un buen diseño, y los defensores del software libre (y del «open source») aseguran que una forma de lograr un buen diseño es que muchos programadores tengan la libertad de inspeccionar y corregir el programa.

Lo anterior no significa que el software libre no tenga fallos de seguridad, pero estos fallos pueden ser detectados por la comunidad y generalmente son solucionados en muy poco tiempo.

#### **2.1.3. Privacidad**

Un programa privativo puede incluir código que recoja los datos del usuario y los envíe a través de la red sin su consentimiento.

Un caso real es el del programa llamado «Notificador de Windows Genuine Advantage», el cual se instala a través del sistema de actualizaciones de Windows. Este programa verifica si se ha pagado la licencia de la copia de Windows que se está ejecutando. Según [18] y [19], este programa se conecta con un servidor de la empresa Microsoft y envía información que podría utilizarse para identificar el computador. Dependiendo de la configuración del sistema de actualizaciones, este programa se podría instalar de forma automática o bien pedir confirmación al usuario, aunque en este último caso, no se informa claramente al usuario sobre lo que hace el programa. Si el usuario decide rechazar

la instalación de este programa, de todas maneras se envía información. Microsoft ha reconocido este hecho, argumentando que se utiliza para mejorar la calidad de su servicio Windows Genuine Advantage, que intenta frenar el uso de copias de Windows sin licencia. Sin embargo, esto no justifica que un programa viole la privacidad de los usuarios.

Un usuario de un sistema operativo libre no tiene estos problemas de privacidad, pues los desarrolladores de software libre tienen muy pocos incentivos para colocar en el sistema un programa que recoja datos de los usuarios. En el caso de que un desarrollador, por algún motivo, inserte en el sistema un programa que viole la privacidad de los usuarios, otros desarrolladores pueden detectar ese programa, pues el código está siempre disponible para todos, y eliminarlo del sistema.

En resumen, con el software privativo, siempre existirá el riesgo de que se viole la privacidad de las personas. Con el software libre, este riesgo disminuye de forma importante.

Además, existen programas libres que permiten garantizar la privacidad de los usuarios en Internet. Estos programas permiten principalmente cifrar, autenticar datos, realizar conexiones seguras y mantener el anonimato en la red. A continuación se definen algunos de los protocolos, estándares y software que se utilizan para brindar privacidad a los usuarios:

**Pretty Good Privacy o PGP (privacidad bastante buena):** es un programa no libre que permite cifrar y firmar archivos. El cifrado consiste en hacer ilegible un archivo excepto para quien tenga autorización de leerlo. El firmado de archivos permite autenticar archivos, es decir, determinar quién es el autor del archivo.

**OpenPGP:** es un estándar de Internet basado en el diseño de PGP y definido en [20].

**Secure Sockets Layer o SSL:** es un protocolo criptográfico que proporciona comunicaciones seguras en Internet. Mientras OpenPGP está diseñado para proteger archivos almacenados en discos, SSL protege datos transmitidos por la red.

**The GNU Privacy Guard, GnuPG o GPG:** es un software libre que permite cifrar y firmar datos, de tal forma que solamente las personas deseadas pueden leerlos. GnuPG es una implementación libre y completa del estándar OpenPGP. GnuPG es una herramienta con interfaz de línea de comandos, pero puede ser integrado con otras aplicaciones. Existen muchas interfaces gráficas de usuario para utilizar fácilmente GnuPG, y bibliotecas para facilitar la incorporación de GPG en otras aplicaciones.

**Seahorse PGP Key Manager:** es una interfaz gráfica que facilita el uso de GnuPG. Permite administrar claves públicas con distintos grados de confianza, además de administrar su propia identidad PGP.

**Privoxy:** es un *web proxy* con capacidades avanzadas de filtrado para proteger la privacidad, modificar datos de páginas web, administrar *cookies*, controlar acceso, y remover propagandas, *banners*, ventanas emergentes, y otras molestias de la web.

**Tor Internet Anonymizer:** usado en combinación con Privoxy, permite enmascarar la dirección IP de un usuario cuando accede a recursos de Internet, con lo cual se hace prácticamente imposible para un tercero espiar sus actividades en Internet.

El uso de estas herramientas libres, y posiblemente otras, en conjunto con buenas prácticas, permiten proteger la privacidad de los usuarios en el ciberespacio. En cambio, usar herramientas no libres para mantener la privacidad es inseguro, pues, al no estar accesible su código fuente, no se puede saber qué hacen exactamente esas herramientas con los datos del usuario.

### 2.1.4. Independencia tecnológica

La independencia tecnológica se puede definir desde tres puntos de vista. Primero, la capacidad de los usuarios de software de no depender de una única empresa desarrolladora de software para obtener versiones mejoradas o modificadas. Segundo, la capacidad de los usuarios de acceder a sus datos sin requerir el uso de un programa específico de un desarrollador específico. Tercero, que un país pueda contar con profesionales capacitados técnicamente para modificar programas y prestar servicios relacionados con el software.

A continuación se discuten estos tres aspectos de la independencia tecnológica.

#### Independencia del desarrollador

Según la libertad 1 definida en [4], todo usuario tiene el derecho de modificar el programa que utiliza, ya sea para agregar una nueva funcionalidad, para corregir un fallo o por cualquier otra razón. No es necesario que el usuario sea programador para modificar un programa, ya que puede contratar a alguien para que lo haga.

Con el software libre, cualquier empresa puede dar soporte a cualquier programa libre. Es por esta razón que Stallman señala en [16] que el software libre genera un mercado libre de servicios relacionados con el software. El hecho de que el código fuente esté disponible de manera libre hace que esto sea posible. El software privativo, en cambio, solamente puede ser adaptado por los autores de ese software, lo cual genera un monopolio. Por lo tanto, una empresa u organización que utilice programas libres, no está obligada a pagar por soporte y actualizaciones a los autores de esos programas, sino que tiene la libertad de escoger a la persona o empresa que le brindará esos servicios.

#### Formatos abiertos

Es muy común que empresas de software privativo realicen acciones para mantener cautivos a sus usuarios (en inglés esto se conoce como *vendor lock-in*). Entre estas acciones la más notable es el uso de formatos cerrados o secretos, lo cual impide que los documentos realizados con un programa puedan ser leídos por otros programas. Los programas privativos generalmente almacenan la información y documentos de los usuarios en formatos cerrados. Esto significa que para recuperar esa información o para abrir esos documentos, se tiene que utilizar obligatoriamente el programa que los ha creado, pues otros programas no pueden decodificarlos.

Los formatos abiertos, por el contrario, permiten que la información y los documentos de los usuarios puedan ser recuperados y manipulados por cualquier programa que soporte esos formatos. Por lo tanto, utilizar formatos abiertos da la posibilidad de elegir el programa que más se acomode a las necesidades del usuario. Esto implica además, que el usuario puede estar seguro de que siempre existirá, o se podrá crear, un programa libre con el cual pueda leer su información y sus documentos.

El software libre generalmente utiliza formatos abiertos. Incluso, si el formato de archivo que utiliza no está definido de manera pública, éste se puede deducir fácilmente, pues al estar disponible el código fuente del programa, es posible ver cómo éste manipula los datos.

El software libre también puede leer algunos formatos cerrados. La comunidad de programadores de software libre ha conseguido descifrar el secreto de algunos formatos de archivo cerrados, por ejemplo, los documentos de Microsoft Office. Sin embargo, no siempre se logra descifrar completamente estos formatos cerrados, por lo tanto un documento escrito en formato cerrado no es leído de forma totalmente correcta.

### Desarrollo tecnológico local

Debido a que el código fuente de los programas libres está a disposición de cualquier persona, los programadores e ingenieros locales pueden estudiarlo y así capacitarse para brindar servicios de soporte o de adaptación de software a organizaciones que decidan utilizar software libre. Esto no sería posible con software privativo, a menos que se pague para conocer el código fuente y se firme un acuerdo para no divulgar esa información.

El Estado, para garantizar su independencia tecnológica, debe utilizar software libre en todas sus oficinas y debe promover la formación de profesionales capaces de utilizar, desarrollar y dar soporte al software libre.

En Chile ya existen empresas que dan soporte a software libre libre, como se puede ver en [17]:

- Blue Company (<http://www.bluecompany.biz>)
- IBM (<http://www.ibm.cl>)
- LinuxCenter (<http://www.linuxcenter.cl>)
- NetLinux (<http://www.netlinux.cl>)
- Microserv (<http://www.microserv.cl>)
- Mundo Linux (<http://www.mundolinux.cl>)
- Sonda (<http://www.sonda.com>)
- Igloo Sistemas (<http://www.igloo.cl>)
- Pegasus (<http://www.pegasus.cl>)
- 0Day (<http://www.0day.cl>)

La Fundación GNUCHILE (<http://www.gnu.cl/>) también ofrece soporte para software libre.

#### 2.1.5. Ventajas legales y económicas

El software libre puede ser copiado legalmente sin pagar regalías por concepto de licencias. Por lo tanto, las organizaciones que utilizan software libre se evitan el pago de licencias y no deben preocuparse de los problemas legales derivado del no pago de licencias.

El uso de software libre puede significar una disminución del costo de un computador, pues se evita el pago de la licencia de un sistema operativo no libre que generalmente viene preinstalado en los computadores, y que el usuario está obligado a pagar. Además de evitar el pago de la licencia del sistema operativo, el software libre también evita el pago de otros programas no libres, como el procesador de texto, planilla electrónica, antivirus, programa de retoque de imágenes, etc. Al disminuir el costo de los computadores, se hacen más accesibles para los ciudadanos. Además, las escuelas y universidades pueden utilizar el dinero con que pagan las licencias para otros fines, como financiar cursos de capacitación para utilizar software libre, pagar por soporte técnico local o para otro fin que se estime conveniente.

Se debe recordar que el software libre no necesariamente es gratuito. Si bien puede ser descargado de Internet sin pagar por ello más que la conexión, también puede ser vendido en algún soporte físico, como un CD o DVD. El modelo de negocio de las empresas de software libre se basa en los servicios y en el soporte técnico, no se cobra por el derecho a usar un software.

## 2.2. Desventajas del software libre

Es importante tener en cuenta las desventajas que tiene el software libre, con el fin de poder enfrentarlas de forma oportuna.

### 2.2.1. Escasez de soporte técnico

Se define el término «soporte técnico» como la capacidad de resolver problemas en el uso del software, y de realizar cambios en el software para corregir errores o para adaptarlo a las necesidades de los usuarios.

En la mayoría de los casos, los propios desarrolladores y usuarios de un software libre ofrecen soporte técnico para éste, a través de Internet, pero lo hacen durante su tiempo libre y no reciben remuneración por ello, por lo cual no pueden garantizar siempre un buen funcionamiento del software.

Sin embargo, a las empresas les interesa que alguien les garantice que su software funcione correctamente. Lamentablemente, existen pocas empresas locales que entreguen soporte técnico para software libre (ver listado en la sección 2.1.4) y que garanticen su buen funcionamiento, y las que existen son pequeñas en comparación con las grandes empresas desarrolladoras de software privativo.

### 2.2.2. Escasez de cursos de capacitación

Existe una escasez de cursos de capacitación para software libre. La mayoría de las instituciones educacionales enseñan a utilizar software privativo, pero casi ninguna enseña a utilizar software libre. Esta falta de capacitación puede generar un rechazo en los usuarios cuando se instale software libre en los computadores de su lugar de trabajo o de estudio.

### 2.2.3. Falta de software libre para determinadas áreas

Si bien hay una gran cantidad de software libre orientado a tareas específicas, existen áreas donde no existe software libre de buena calidad que pueda reemplazar al software privativo. Por ejemplo, hasta el momento no existe software libre para diseño asistido por computador (CAD) en tres dimensiones que ofrezca funcionalidades similares a las que tienen los programas privativos más usados en esta área.

En este caso, para una organización que desee migrar a software libre, existen dos posibilidades: seguir utilizando software privativo temporalmente, o bien, como sugiere Stallman, dejar de utilizar el programa privativo, incluso si no tiene alternativas libres adecuadas, y comenzar a buscar otra forma de realizar la tarea, o bien desarrollar un software libre en conjunto con la comunidad que permita realizar esa tarea.

### 2.2.4. Dificultad de uso de algunos programas

El software libre puede ser más difícil de aprender que el software privativo. Existen varias razones que pueden explicar este fenómeno. Una de ellas, es que algunos programadores de software libre programan como una forma de desafío personal, y no piensan demasiado en el usuario final. Otra razón es que en la mayoría de los proyectos de software libre no existe un financiamiento adecuado para garantizar estándares de usabilidad.

Sin embargo existen excepciones, pues hay proyectos de software libre que toman muy en serio la facilidad de uso y que incluso reciben financiamiento de grandes empresas, que contratan a



programadores para que se dediquen a mejorar el software libre que les interesa.

## Capítulo 3

# Obstáculos para la Libertad Digital

La Free Software Foundation creó la licencia GPL con el fin de resguardar las cuatro libertades de los usuarios de computadores, definidas por esta misma organización. Sin embargo, existen algunos factores externos que afectan a las libertades de los usuarios y al software libre. Entre estos factores están los de tipo técnico, como la Gestión Digital de Restricciones (DRM), la venta de hardware sin especificaciones técnicas y los formatos de archivo cerrados. Entre los factores de tipo legal se encuentran las leyes que prohíben algunos programas libres y las patentes de software. Otros factores externos que amenazan al software libre son las presiones de las empresas de software privativo, la inercia o costumbre de las personas que utilizan software privativo y la falta de conciencia sobre las libertades. A continuación se explica cada uno de estos factores.

### 3.1. Gestión Digital de Restricciones

La Gestión Digital de Restricciones (DRM, Digital Restrictions Management) es un concepto que engloba a distintas tecnologías que permiten a las grandes empresas de medios audiovisuales, software y hardware, restringir la forma en que los usuarios acceden y utilizan ciertos datos digitales, tales como software, música y películas. Cada vez más dispositivos electrónicos están incluyendo DRM, tales como computadores, reproductores de audio y video, teléfonos móviles y consolas de videojuegos.

A pesar de que el DRM cambia muchas reglas de la sociedad, por ejemplo, la forma en que se copia la música, no se está realizando una discusión política y social acerca de estos cambios. Esto se debe en parte a que algunos creen erróneamente que no se verán afectados, pero quienes piensan eso es porque no se han informado acerca del DRM o bien toda la información que han recibido ha sido proporcionada por las empresas que pretenden beneficiarse de la implantación del DRM. Estas empresas prefieren referirse al DRM como «Gestión Digital de *Derechos*» («Digital *Rights* Management») o «Medidas de *Protección* Tecnológica» («Technological *Protection* Measures», TPM), pero los derechos que se protegen con el DRM son los de estas empresas y no los de los usuarios.[21]

La Free Software Foundation se opone al DRM porque las restricciones que esta tecnología impone pueden violar las libertades de los usuarios de software. Por esta razón inició una campaña en contra del DRM llamada «Defective by Design»<sup>1</sup> o Defectuoso por diseño, cuyo nombre hace alusión a las limitaciones artificiales e intencionales que impone esta tecnología a los usuarios.

---

<sup>1</sup><http://www.defectivebydesign.org/>

Algunas de las tecnologías más importantes que son englobadas por el concepto de DRM son la Tivoización, la Computación Traicionera, AACS y FairPlay. A continuación se describen cada una de estas tecnologías.

### 3.1.1. Tivoización

La «tivoización» es la creación de un sistema de hardware y software, que incorpora software libre, pero utiliza una técnica para evitar que los usuarios ejecuten versiones modificadas del software en ese hardware. Richard Stallman creó el término tivoización a partir de un dispositivo de hardware y software llamado TiVo, el cual permite grabar programas de televisión. TiVo es en realidad un computador que ejecuta un sistema operativo GNU/Linux, que es software libre. Sin embargo, como se explica en [22], si el usuario intenta ejecutar una versión modificada de este sistema operativo, el TiVo se apaga automáticamente, lo cual priva al usuario de la libertad 1, que es la libertad de estudiar y modificar el software según sus necesidades.

La versión 2 de la GPL (GPLv2) obliga a los distribuidores de software a publicar el código fuente de esos programas. El objetivo de este requerimiento es permitir a los usuarios de software con GPL la modificación de éste para adaptarlo a sus necesidades. Sin embargo, TiVo evitó este objetivo al hacer que sus productos no permitan ejecutar software modificado por el usuario. Para esto utiliza un sistema de firmas digitales del software, que compara la firma del software original autorizado por TiVo con la firma del software que se ejecuta en el aparato. Si éstas no coinciden, el dispositivo deliberadamente no funciona. Si bien TiVo cumple con el requerimiento que establece la GPLv2 de distribuir el código fuente, a través de otros medios está vulnerando la libertad de los usuarios de modificar el software.

Como la GPLv2 no puede evitar la tivoización, la Free Software Foundation creó la versión 3 de la GPL (GPLv3), la cual evita este y otros problemas que amenazan a las libertades de los usuarios. Eben Moglen, el abogado que asesora a la Free Software Foundation en los temas legales, afirma lo siguiente en [26]:

«La licencia [GPLv3] debería prohibir los medios técnicos para evitar la evasión de sus reglas, con la misma claridad con que prohíbe la evasión legal de sus reglas.»

Por otra parte, Linus Torvalds, el autor del núcleo Linux, no cree que exista un problema ético en utilizar firmas digitales para restringir al software que se puede ejecutar en ciertos dispositivos, siempre y cuando ese software pueda ser utilizado en otro hardware [24]. Linus cree además que el uso de DRM puede ser beneficioso desde un punto de vista de seguridad, aunque esto vaya en desmedro de la libertad de los usuarios. Sin embargo, otros desarrolladores del núcleo Linux, incluyendo a Alan Cox [25], han expresado opiniones contrarias. En respuesta al tema de la seguridad, la GPLv3 permite el uso de las firmas digitales, siempre y cuando el usuario tenga la opción de desactivar esta restricción.

Además, la cláusula anti-tivoización de la GPLv3 fue limitada para que no se aplique cuando el software es distribuido en una empresa, sino solo cuando se distribuye a usuarios finales[27]. Esta medida se tomó porque muchas empresas externalizan su infraestructura de TIC y desean que solo la empresa que les presta servicios pueda acceder a sus sistemas, con lo cual ceden algunas de sus libertades a cambio de mayor seguridad. Esta medida es solo una transigencia y no un cambio en los ideales de la Free Software Foundation, la cual es clara al señalar que lo más importante es la libertad de los usuarios.

### 3.1.2. Computación Traicionera

La Computación Traicionera es una tecnología que permite que las empresas que fabrican el hardware y el software controlen los computadores de los usuarios. El usuario solamente puede ejecutar los programas que estas empresas le permitan, es decir, el usuario no puede modificar el software ni puede escribir su propio software. Esto viola la libertad 1, que es la libertad de estudiar y modificar el software.

Quienes están a favor de esta tecnología y, por lo tanto, a favor de restringir a los usuarios, la llaman «Trusting» Computing o Computación «Confiable». El término «confiable» se refiere a que el sistema es consistente y seguro, es decir, se comportará de la forma en que «alguien» haya definido. Si ese «alguien» no es el usuario, entonces él ha perdido la libertad de hacer lo que desee con su computador. Una opción puede ser utilizar computadores que no tengan componentes de Computación Traicionera, pero los grandes fabricantes de hardware y software quieren que todos los computadores los incorporen.

Microsoft ha desarrollado un sistema, llamado originalmente Palladium, que permite que una aplicación «selle» los datos de tal forma que ningún otro programa pueda acceder a éstos. Ni siquiera una versión modificada del mismo programa podría leer esos datos. Esto nuevamente viola la libertad 1, la libertad de estudiar y modificar el software, pues al modificar el software, el usuario ya no podrá acceder a sus datos. Además, Microsoft podría integrar este sistema en alguna versión futura de Windows [22], con lo cual se lograría «amarrar» aún más a los usuarios con el software de esa empresa, pues los usuarios solamente podrían acceder a sus datos con versiones específicas de sus programas.

Por otra parte, la Computación Traicionera permitirá restringir el acceso a documentos y la copia de éstos. Esto puede ser atractivo para algunas empresas o usuarios, pero los críticos afirman esto facilitará la censura indebida. Por ejemplo, una persona podría no tener acceso a un documento que ella misma escribió, si su jefe decide que ya no tiene autorización para leerlo.

Algunos destacados expertos en seguridad [28] [29] se oponen a la Computación Traicionera, pues creen que otorgará a los fabricantes de hardware y autores de software un mayor control para imponer restricciones a los usuarios. A pesar del debate que se ha generado, las empresas más influyentes en el área de la computación, tales como los fabricantes de chips Intel y AMD, los fabricantes de computadores como Apple y Dell, y los desarrolladores de software como Microsoft, tienen planeado incluir la Computación Traicionera en nuevas generaciones de sus productos [30].

### 3.1.3. AACS

AACS o Advanced Access Content System es una tecnología desarrollada por Disney, IBM, Microsoft, Intel, Sony y otras empresas, cuya finalidad es restringir el uso de grabaciones de televisión de alta definición (High Definition Television, HDTV) [22]

AACS es un estándar que fue creado para restringir el acceso a y la copia de discos ópticos de nueva generación y DVD. Ya ha sido adoptado como el esquema de restricción de acceso para los discos HD DVD (High Definition DVD) y Blue-ray.

Desde su aparición en dispositivos en 2006, varias claves de descifrado han sido extraídas de programas reproductores débilmente protegidos y se han publicado en Internet. Con esta clave es técnicamente posible reproducir discos HD DVD y Blue-ray con software libre (y también con software privativo), pero legalmente esto puede estar prohibido por leyes como la DMCA (Digital Millenium Copyright Act) de Estados Unidos.

Algunas de las características de AACS, según [23], son:

- El contenido del disco está cifrado y (supuestamente) solo los reproductores autorizados tienen una clave para descifrarlo
- Cada disco con contenido posee un identificador o número de serie que solamente puede ser grabado por hardware especial, por lo tanto si un usuario copia el disco, no podrá copiar el identificador. Como este número es necesario para descifrar el contenido, el usuario no podrá leer el disco copiado. Sin embargo, algunos crackers han logrado modificar el firmware de algunos reproductores HD DVD para que pueda reproducir discos sin número de serie.
- Los reproductores que cumplen con el estándar AAC3 deben seguir ciertas reglas en relación a las salidas por conectores analógicos, con el fin de evitar que el audio y el video sean grabados en algún otro dispositivo que no cifre estos datos, por ejemplo, un VHS común o una tarjeta capturadora de video de un computador. Estas reglas indican que la imagen que sale por los conectores analógicos debe ser degradada reduciéndose su resolución de 1920x1080 a 960x540 píxeles. Solamente las salidas autorizadas pueden transmitir la imagen en su resolución original.
- Los estudios cinematográficos pueden incluir una marca de agua a la película, es decir, un patrón de audio imperceptible por el oído humano, pero sí por los reproductores que cumplen la norma AAC3. Con esto se busca evitar la reproducción de discos con películas filmadas en el cine con una cámara de video.
- El estándar AAC3 especifica protocolos para permitir, opcionalmente, que los usuarios realicen copias de respaldo o muevan el archivo a un reproductor portátil, pero de una forma controlada. Esto requiere que el dispositivo se contacte por Internet con un servidor remoto para obtener autorización.

Según [31], AAC3 está diseñado para que el usuario no tenga control sobre los datos almacenados en este formato, lo cual impide que el usuario utilice esos datos de forma legítima. Por ejemplo, el «uso justo» («fair use»), que es una excepción a la ley de copyright de Estados Unidos, permite utilizar obras cubiertas por copyright para ciertos usos bienintencionados, tales como copiar con fines de investigación, enseñanza, periodismo, crítica, parodia y actividades realizadas por bibliotecas.

#### 3.1.4. FairPlay

FairPlay es una tecnología DRM creada por la empresa Apple. FairPlay cifra archivos de audio en formato AAC y evita que los usuarios reproduzcan estos archivos en computadores no autorizados. FairPlay está incorporado en el software multimedia QuickTime, el cual es utilizado en los reproductores de música iPod y en el software iTunes que, junto con el servicio iTunes Store, permite comprar y descargar música. Cada canción comprada en iTunes Store utilizando iTunes está codificada con FairPlay [32]. Sin embargo, en abril de 2007, Apple anunció [33] que comenzará a ofrecer música de la discográfica EMI sin cifrado FairPlay, a través de iTunes Store, con una mayor calidad y a un precio un 30 % más alto que la música con FairPlay. Sin embargo, algunos expertos en privacidad señalan [34] que, si bien estas canciones no contienen FairPlay, poseen una forma de identificar a los usuarios que las escuchan, con lo cual se atenta contra su privacidad.

FairPlay no evita que se copie el archivo, sino que solo restringe el descifrado del audio. Una archivo cifrado con FairPlay tiene actualmente las siguientes restricciones [32]:

- El archivo puede ser copiado a cualquier cantidad de reproductores de música iPod
- El archivo puede ser copiado hasta en cinco computadores autorizados simultáneamente

- El archivo puede ser copiado a un CD de audio estándar cualquier cantidad de veces  
El CD resultante no contiene DRM y sus pistas pueden ser extraídas, codificadas y reproducidas como en cualquier otro CD de audio. Sin embargo, la música en el CD contendrá artefactos o impurezas producto de la compresión, por lo que al convertirla a otro formato de compresión con pérdida, como el MP3 o el Ogg Vorbis, estas impurezas se pueden agravar causando una pérdida de calidad del audio. Una opción puede ser utilizar un formato sin pérdida al recodificar la música del CD, tal como FLAC, aunque el archivo resultante ocupará más espacio que el original.
- Una lista de canciones en iTunes que contenga alguna canción cifrada con FairPlay puede ser copiada a CD hasta siete veces. Es necesario cambiar la lista de canciones para continuar grabándola en CD.

Estas restricciones pueden cambiar en el futuro y solamente las empresas discográficas tienen el poder de definir las. Por ejemplo, se podría disminuir (incluso a cero) la cantidad permitida de copias a reproductores iPod, computadores o CD.

Una limitación intencional de FairPlay es que impide que los clientes de iTunes utilicen su música en reproductores de música no autorizados por Apple [32]. Esto provocó que un cliente de iTunes Store denunciara a Apple por conducta monopólica [35]. Una limitación no intencional de FairPlay es su incompatibilidad con la música que posea una licencia que permita la copia y la creación de obras derivadas, como una de las licencias Creative Commons. FairPlay siempre aplica las restricciones que imponen las grandes discográficas, sin importar si la licencia de una canción permite hacer algo que FairPlay impide. Es otras palabras, FairPlay, y el DRM en general, puede impedir a un usuario realizar algo a lo cual tiene derecho y que la ley le permite.

### 3.1.5. Restricciones de Windows Vista

Windows Vista incluye DRM con el objetivo de restringir la reproducción de películas de alta definición, como los discos HD-DVD y Blue-Ray. Vista impone ciertas condiciones para que la reproducción de una película se realice en alta calidad, y si no se cumplen la película se verá en baja calidad. En [36] se explica que Vista reduce la calidad de la película que está siendo reproducida por el usuario, en caso de que se detecte un dispositivo de hardware que no esté aprobado por Microsoft. Esto obliga al usuario a adquirir nuevo hardware, probablemente un nuevo computador, que sea compatible con las especificaciones que establece Microsoft para reproducir películas de alta definición. Específicamente, la tarjeta de video, la tarjeta de sonido y el monitor, con sus respectivos controladores, deben ser aprobados por Microsoft para poder reproducir películas de alta calidad. Además, en [36] se analiza el alto costo de hardware que requiere el sistema DRM de Vista para descifrar una película de alta definición. Este costo finalmente es pagado por el usuario, que tiene que pagar por un procesador más poderoso y por una tarjeta de video que posea características especiales para reproducir películas de alta definición.

En realidad, el sistema DRM de Vista no solamente restringe la visualización de películas de alta definición, sino que puede aplicarse a cualquier tipo de archivo considerado como «premium» por su autor o distribuidor.

Microsoft confirma en [37] que ha incluido este sistema de restricciones a los usuarios de Windows Vista:

«Windows Vista incluye una infraestructura de protección de contenidos diseñada específicamente para asegurar que el contenido audiovisual comercial protegido, tales como

los discos HD-DVD o Blue-Ray, pueda ser reproducido en computadores con Windows Vista»<sup>2</sup>

Microsoft también reconoce en [37] que su sistema de DRM requiere mayor uso de recursos de hardware:

*«¿Las características de protección de contenidos de Windows Vista incrementan el consumo de recursos de CPU? Sí. Sin embargo, el uso de ciclos adicionales de CPU es inevitable, pues el PC provee a los consumidores una funcionalidad adicional.»<sup>3</sup>*

La funcionalidad adicional a la que se refiere Microsoft en el texto citado, es justamente el sistema DRM que restringe la forma en que los usuarios (Microsoft los llama consumidores) reproducen archivos audiovisuales.

### 3.2. Hardware sin especificaciones técnicas

Un problema importante es la existencia de una gran cantidad de dispositivos de hardware que no poseen controladores libres y que por lo tanto no se pueden utilizar en sistemas operativos completamente libres. La causa de este problema es que la mayoría de los fabricantes de hardware no publica las especificaciones de sus dispositivos, lo cual hace imposible o muy difícil escribir un controlador para estos dispositivos. Por esta razón, en [38] Stallman identifica al hardware sin especificaciones como una amenaza.

Según [38], hay dos formas de superar el problema del hardware sin especificaciones técnicas. La primera es la ingeniería inversa, la cual permite conocer cómo funciona el hardware y así poder desarrollar un controlador. La segunda, es elegir y utilizar hardware que sea compatible con software libre.<sup>4</sup> En el primer caso se necesita un gran esfuerzo de análisis y programación, ya que la ingeniería inversa es un proceso difícil y tedioso. En el segundo caso, se necesita que los usuarios inviertan más tiempo y posiblemente más dinero a la hora de comprar hardware.

Existe la posibilidad de utilizar los controladores no libres que proveen algunos de los fabricantes de hardware, pero Stallman [38] enfatiza que utilizar controladores no libres es inaceptable porque estos niegan las libertades de los usuarios, por lo tanto no se deben utilizar.

La solución óptima, pero sobre la cual el usuario no tiene control directo, es que los fabricantes de hardware liberen las especificaciones del hardware que venden. De esta forma, la comunidad del software libre, posiblemente en conjunto con el fabricante, podrá desarrollar controladores libres para esos dispositivos. La Free Software Foundation ha realizado un llamado a los fabricantes de hardware para que publiquen las especificaciones de sus dispositivos o al menos cooperen en el desarrollo de controladores libres para sus dispositivos [39]. Uno de los argumentos utilizados en esta petición para convencer a los fabricantes de hardware de publicar sus especificaciones, es la mejora de la reputación de la empresa, el fortalecimiento de la marca y un aumento de las ventas. Una encuesta independiente [40] revela que el 89 % de los jóvenes y adolescentes estaría dispuesto a cambiar de

---

<sup>2</sup>Original en inglés: «Windows Vista includes content protection infrastructure specifically designed to help ensure that protected commercial audiovisual content, such as newly released HD-DVD or Blu-Ray discs, can be enjoyed on Windows Vista PCs.»

<sup>3</sup>Original en inglés: «*Will Windows Vista content protection features increase CPU resource consumption?* Yes. However, the use of additional CPU cycles is inevitable, as the PC provides consumers with additional functionality.»

<sup>4</sup>Se entiende por «hardware compatible con software libre», como aquel hardware que tenga un controlador que sea software libre, o que tenga publicadas sus especificaciones técnicas, de tal forma que permitan y faciliten el desarrollo de un controlador.»

una marca a otra si la segunda está asociada a una buena causa. Como se ha dicho anteriormente, respetar las libertades de los usuarios de usar, modificar y compartir el software es algo socialmente útil y ético, y por lo tanto una empresa que respete estas libertades publicando las especificaciones del hardware que vende estará apoyando una buena causa.

Además, la Free Software Foundation está elaborando un listado [41] de hardware que posee controladores libres. Este listado es útil para los usuarios que deseen adquirir nuevo hardware, pues les permite tener la seguridad de que el hardware funcionará correctamente en un sistema operativo libre.

### 3.2.1. BIOS no libre

El BIOS (Basic Input/Output System) es software que se encuentra en un chip de la tarjeta madre de un computador. Su función es iniciar el computador y preparar la carga de un sistema operativo.

Según Stallman [42], antiguamente el BIOS se encontraba en memoria ROM, es decir, no podía modificarse. Por lo tanto, no existían los problemas éticos del software, que aparecen cuando éste puede ser modificado. Actualmente, en cambio, el BIOS se almacena en memoria que puede ser modificada, y por lo tanto, el usuario puede instalar uno distinto. En este caso aparecen los temas éticos del software, pues si el BIOS no es libre, el usuario estará instalando software que no puede modificar o estudiar. De hecho, las empresas que fabrican sistemas BIOS recomiendan al usuario descargar actualizaciones e instalarlas en sus computadores, pero no ofrecen el código fuente y por lo tanto se trata de software no libre.

Además de los problemas éticos de los BIOS no libres, existen desventajas prácticas asociadas con la imposibilidad de modificarlos. Por ejemplo, en [43] se señala que los BIOS no libres que se incluyen en la mayoría de los computadores, todavía contienen código para proveer soporte a MS-DOS y que es innecesario para otros sistemas operativos, incluso para los sistemas operativos más modernos de Microsoft. Por otra parte, los BIOS no libres que incluyen algunos de los grandes vendedores de hardware pueden bloquear ciertas características del computador, como se menciona en [39]. Por ejemplo, el BIOS de algunos notebooks impide que el usuario instale tarjetas de expansión que no han sido preaprobadas por el vendedor.

Afortunadamente, existe un proyecto para crear un BIOS libre, llamado como coreboot<sup>5</sup>. Sin embargo, es difícil instalar este sistema en la mayoría de los computadores, debido a que los fabricantes de tarjetas madre también mantienen en secreto la forma de instalar o actualizar el BIOS. Para que estos proyectos tengan éxito, se necesita de la cooperación de los fabricantes de tarjetas madre, y sólo algunos de ellos están dispuestos a ayudar. Los usuarios deberían preferir los productos de aquellos fabricantes que sí cooperan con la Comunidad.

## 3.3. Formatos de archivo cerrados

El uso de formatos de archivo cerrados impide generalmente que los archivos puedan ser leídos correctamente con software libre. Por lo tanto, se debe evitar el uso de estos formatos, al menos hasta que exista software libre capaz de manejarlos correctamente. Algunos formatos de archivo cerrados utilizados ampliamente son los formatos de Microsoft Office (doc, xls, ppt), el formato de compresión de archivos RAR, y Flash.

---

<sup>5</sup><http://www.coreboot.org/>



El formato de MP3 (MPEG 2, Layer 3) es muy popular debido a que reduce considerablemente el tamaño de los archivos de audio. Sin embargo, al estar cubierto por una patente de software, la mayoría de los sistemas operativos libres no incluyen los componentes necesarios para reproducir archivos MP3, para evitar ser demandadas. Para solucionar este problema, se creó un formato de audio abierto y libre de patentes de software, llamado Ogg Vorbis. Además de Ogg Vorbis, existe Ogg Theora, un formato de video abierto y libre de patentes. La Free Software Foundation ha creado una campaña llamada PlayOgg [44] que tiene por objetivo promover el uso de los formatos Ogg. Por otra parte, la organización Xiph [45] se encarga de desarrollar y promover el uso de formatos multimedia para Internet que sean abiertos y libres de patentes, entre ellos el Ogg, y del software necesario para reproducirlos.

El formato de archivo abierto OpenDocument (ODF) que es utilizado por el paquete de ofimática libre OpenOffice, ha sido aprobado como estándar ISO [46]. Por lo tanto, al utilizar este formato, los usuarios tienen garantizado el acceso a sus datos y no dependen del software de una sola empresa para poder leerlos. Sin embargo, Microsoft ha definido su propio estándar de documentos, el Office Open XML (OOXML) y ha logrado que ISO lo apruebe a través de un proceso que ha tenido muchas irregularidades. En [47] y en [48] se critican varios puntos de OOXML: su gran complejidad, su incompletitud y su dependencia de una única empresa. Estos problemas hacen que OOXML sea tan malo como los actuales formatos cerrados de Microsoft Office en cuanto a interoperabilidad con otros programas se refiere, sean estos libres o no. Por lo tanto, se debe evitar el formato OOXML de Microsoft y preferir el formato ODF.

### 3.4. Leyes que prohíben el software libre

Existen leyes y acuerdos internacionales que prohíben ciertos programas libres. Por ejemplo, la ley estadounidense DMCA establece que es ilegal utilizar software, libre o no, para evadir los mecanismos que restringen la copia de archivos sujetos a copyright. Esta ley, por lo tanto, prohíbe un programa libre llamado libdvcss<sup>6</sup>, el cual permite descifrar el contenido de los discos de películas DVD, es decir, permite ver películas DVD en el computador. Esta prohibición puede parecer razonable si el objetivo es evitar la comercialización de copias no autorizadas de películas, pero también afecta los usos legítimos del programa libdvcss, como lo es el uso privado y no comercial de copias de películas obtenidas de forma legal.

Otro caso de legislación que prohíbe el software libre es el de la ley DADVSI aprobada en Francia [49]. Esta ley, entre otras cosas, prohíbe saltar la protección que los fabricantes introducen en los BIOS de algunos computadores, que evita instalar otro sistema operativo distinto al establecido. Según Stallman [50], esta ley efectivamente prohíbe la instalación de sistemas operativos libres en esos computadores. Técnicamente es posible cambiar el BIOS por otro que no imponga esta restricción, y que por lo tanto permita instalar un sistema operativo libre, pero la ley DADVSI lo prohíbe.

#### 3.4.1. Patentes de software

Quizás la amenaza más grande al software libre y, por lo tanto, a las libertades de los usuarios, es la ley de patentes de software. Una patente de software otorga un monopolio por un periodo de 20 años sobre una *idea* que puede ser implementada en software, no sobre la implementación en sí, como en el caso de la ley de copyright. Incluso si un programador escribe un programa desde cero, sin copiar código ni ideas a nadie, puede que esté infringiendo una o varias patentes de software. Según

---

<sup>6</sup> Disponible en <http://www.videolan.org/developers/libdvcss.html>

Stallman [51] es casi imposible desarrollar un programa no trivial que no utilice alguna patente de software. Stallman hace una analogía entre las patentes de software y un campo minado: es casi imposible avanzar sin pasar sobre alguna de ellas. Esto le puede ocurrir tanto a programadores de software libre como a programadores de software privativo. Al infringir una patente, el programador puede ser demandado por la empresa o persona que haya registrado la patente de software. Para evitar la demanda, el programador puede pagar un royalty por la patente, pero si otro programador modifica el programa, también deberá pagar un royalty. Esto en la práctica significa que la ley de patentes de software restringe la libertad de los usuarios de modificar el software.

Afortunadamente, solo unos pocos países tienen leyes de patentes de software: Estados Unidos, Inglaterra, Australia y Japón. Sin embargo, existe una fuerte presión para que otros países legislen a favor de las patentes de software. Un caso muy polémico fue el de la Unión Europea, donde en una primera instancia no se aprobó la ley que patentes de software, aunque todavía continúan los esfuerzos por parte de ciertas empresas para que se apruebe dicha ley [51].

En Chile el software no es patentable, pero Estados Unidos podría presionar a Chile para que legisle a favor de las patentes de software, pues en el Tratado de Libre Comercio que firmaron ambos países, Chile se compromete a evaluar esta posibilidad.

### 3.5. Presiones de empresas de software privativo

Algunas empresas de software privativo, como Microsoft, reconocen que el software libre es una amenaza para ellas[52]. Es lógico que estas empresas realicen presiones políticas para asegurarse que la legislación les favorezca, y harán todo lo posible para evitar que el software libre sea adoptado de forma masiva por los gobiernos, empresas y universidades.

Otra táctica que utilizan las empresas de software privativo es realizar «donaciones» de licencias o dar facilidades de pago a distintas organizaciones, sobre todo educacionales. De esta manera, las ventajas legales y económicas del software libre pueden reducirse o anularse, pues las organizaciones obtienen copias legales de su software a bajo costo. Pero incluso en estos casos se debe rechazar el software privativo y escoger el software libre, pues mantener la libertad debería ser más importante que un ahorro de dinero en el corto plazo.

### 3.6. Inercia o costumbre

Usar software libre no es necesariamente más difícil que usar software privativo. Sin embargo, el hecho de tener que cambiar el software al cual está acostumbrado resulta inconveniente para la mayoría de los usuarios. Desafortunadamente, la mayoría de los usuarios de computadores utilizan software privativo, y presumiblemente no estarán muy dispuestos a cambiar a software libre, a menos que se le den buenas razones y facilidades para hacerlo.

### 3.7. Falta de conciencia sobre las libertades de los usuarios de software

Las libertades que promueve el movimiento del software libre son desconocidas por la mayoría de las personas, y por lo tanto no se dan cuenta de que sus libertades están siendo vulneradas. Es deber de los seguidores de este movimiento difundir su filosofía para que todos los usuarios de computadores estén conscientes de sus libertades.

## Capítulo 4

# Casos de migración a software libre

En este capítulo se describirán algunos casos en que gobiernos, empresas y otras organizaciones han migrado o se han planteado migrar a software libre.

Como ya se ha mencionado en este trabajo, el software libre debería ser utilizado como una estrategia de independencia tecnológica, para que el Estado no tenga que depender de ciertas empresas para el funcionamiento de los sistemas computacionales que manejan la información de todos los ciudadanos. Los profesionales locales del sector de las tecnologías de la información pueden capacitarse para dar soporte a cualquier software libre, adaptándolo a las necesidades específicas de cada organización, empresa o persona. En otras palabras, el software libre crea un mercado libre de servicios de soporte. El software no libre, en cambio, genera un monopolio de soporte, pues solamente la empresa que vende el software puede modificarlo y adaptarlo a las necesidades del cliente.

Los que se oponen a la independencia tecnológica utilizan el término «neutralidad tecnológica» para dejar de lado el tema de la libertad y la independencia, y centrarse sólo en la capacidad de elegir la tecnología, sin importar si ésta causa dependencia o no. Ellos defienden a las empresas que venden software privativo o software no libre, es decir, software que encadena a sus usuarios y los hace dependientes de esas empresas.

Debido a lo anterior, se debe tener claro que la mejor razón para migrar a software libre es la protección de la independencia tecnológica y las libertades de los usuarios, y no solamente la reducción de costos por conceptos de licencias de software o las posibles ventajas técnicas del software libre (como lo afirma la comunidad del «código abierto»). Aclarar esto es muy importante, pues esta postura descarta de plano cualquier oferta que pueda hacer alguna empresa de software no-libre, por muy conveniente que sea desde el punto de vista económico o técnico.

Cabe recordar que la mayor cantidad del software que se desarrolla en el mundo es software hecho a medida, diseñado según las necesidades del cliente. Por lo tanto, es software de uso privado, que no corresponde ni a software libre ni a software privativo. Dicho software no significa un problema ético, pues el cliente se queda con el código fuente y puede ejercer sus cuatro libertades.

### 4.1. Casos de migración en la administración pública

En Venezuela se decretó en diciembre de 2004 el uso prioritario de software libre y estándares abiertos en la Administración Pública [53]. Entre las razones mencionadas en dicho decreto para la adopción del software libre, se destacan el fortalecimiento de la industria de software nacional, la reducción de la brecha social y tecnológica, el mejoramiento del servicio que el Estado brinda a sus ciudadanos, la interoperabilidad y la seguridad.

En Ecuador también se estableció el uso de software libre como una política de Estado. El presidente Rafael Correa expresó lo siguiente en un mensaje emitido el 26 de abril de 2007, durante el Festival Latinoamericano de Instalación de Software Libre (FLISOL) [54]:

«...es necesario que todos adoptemos, tanto a nivel público cuanto a nivel privado, el software libre. De esa manera garantizaremos la soberanía de nuestros Estados, dependeremos de nuestras propias fuerzas, no de fuerzas externas a la región. Por eso, todos a utilizar software libre. El Gobierno ecuatoriano ya lo estableció como una política de Gobierno y de Estado. Esto será un importante paso para la integración y, por qué no decirlo, para la liberación de América Latina.»

El 11 de abril de 2008, el presidente de Ecuador, a través de un decreto, estableció el uso del software libre en la Administración Pública. Este decreto permite el uso de software privativo solamente cuando no exista una solución en software libre que cumpla con ciertos requerimientos.[55]

En la comunidad autónoma de Extremadura, España, se firmó un acuerdo para utilizar software libre y estándares abiertos en la administración pública. Se estableció el uso obligatorio de los formatos abiertos OpenDocument y PDF/A, lo cual garantiza que los documentos puedan ser leídos en el futuro. Los distintos departamentos de la Junta de Extremadura tuvieron que acostumbrarse a utilizar aplicaciones ofimáticas libres y el sistema operativo libre gnuLinEx [56]. Los motivos para esta migración son, además de económicos, de independencia tecnológica, como se puede leer en el texto oficial del Acuerdo, disponible en [57]:

«... para avanzar en la integración de la Sociedad de la Información en la Administración Pública moderna y en una sociedad global, hay que garantizar el control y gestión de aspectos tan trascendentes como independencia tecnológica, interoperabilidad entre plataformas informáticas, homogeneidad de los sistemas de información, seguridad informática en materia de sistemas de información, innovación tecnológica real y cumplimiento de estándares informáticos abiertos y libres.»

Andalucía, otra comunidad autónoma de España, está utilizando software libre en la administración pública y en la educación como herramienta para impulsar el desarrollo de una sociedad del conocimiento [58]. En esa región existe una asociación de empresas, llamada AndaLibre<sup>1</sup>, que brinda soporte a software libre. Además, se ha desarrollado una distribución de GNU/Linux local, denominada GuadaLinex<sup>2</sup>.

En Alemania, el Municipio de Múnich se encuentra en proceso de migración a software libre y GNU/Linux. Está planificado migrar el 80 % de sus 14.000 computadores para el año 2008. La decisión para migrar fue tomada en 2003, con el fin de ahorrar costos en licencias de software privativo [59].

La Oficina Federal de Empleo de Alemania migró 13.000 computadores de Windows NT a GNU/Linux [60]. Esos computadores son utilizados para brindar acceso a Internet al público. Las razones del cambio son la reducción del costo y las mejoras en seguridad. La reducción de costos se debe a que se evita el pago de licencias de software. Por otra parte, se reducen los costos de mantenimiento, pues los computadores funcionan como terminales ligeros, lo cual centraliza la configuración. Las mejoras en seguridad se refieren a un menor índice de infección por virus.

El Parlamento francés migró a software libre para reducir costos (incluso considerando el costo de migración y entrenamiento) y obtener un mayor control sobre su infraestructura de TIC al no

---

<sup>1</sup><http://www.andalibre.org/>

<sup>2</sup><http://www.guadalinux.org/que-es-guadalinux>

depender de una empresa en particular. El software utilizado es GNU/Linux, OpenOffice, Firefox y correo electrónico. [61]

La policía francesa migrará a GNU/Linux, con lo cual completará su migración a software libre, pues ya había comenzado a utilizar OpenOffice como paquete de ofimática en 2005, y Firefox y Thunderbird como navegador web y cliente de correo en 2006. En total son unos 70.000 computadores que irán migrando gradualmente a GNU/Linux en un plazo de cuatro años. Las razones para migrar son tres: diversificar los proveedores y reducir la dependencia en una única empresa, obtener el control y dominio del sistema operativo, y el costo. Al evitar el uso software privativo licenciado, la policía está ahorrando unos siete millones de euros anuales. [62]

En Italia, el Parlamento migrará a software libre y GNU/Linux sus 3.500 computadores, y además ha recomendado hacer lo mismo a las demás instituciones italianas. Los motivos del cambio son de «ahorro, pluralismo y flexibilidad». Se espera ahorrar unos tres millones de euros. [63]

En [64] se reúnen algunas leyes que buscan impulsar el uso de software libre en la administración pública de países latinoamericanos, como Argentina, Brasil, Uruguay, Perú y Venezuela.

En Cuba se está considerando adoptar software libre en la administración pública. La motivación es la soberanía tecnológica, la seguridad nacional y razones ideológicas. El objetivo es no depender de una empresa extranjera en la administración pública y especialmente en el ejército. [66]

La Marina estadounidense anunció que solamente adquirirá sistemas basados en tecnologías y estándares abiertos [78]. La decisión fue tomada principalmente para reducir los costos.

La NASA utiliza y desarrolla software libre en sus misiones, con el fin de mejorar la calidad de su software a través de la revisión por pares de la comunidad, acelerar el desarrollo de software a través de las contribuciones de la comunidad, maximizar la conciencia y el impacto de las investigaciones de la NASA, y difundir el software de la NASA para apoyar su misión educativa. [79]

En Chile también se está intentando legislar a favor del software libre. La Cámara de Diputados chilena aprobó por amplia mayoría un Proyecto de Acuerdo que pretende implementar el uso de software libre en la Administración Pública [67]. Además, este Proyecto de Acuerdo sugiere el uso de software libre en todos los establecimientos educacionales de Chile, lo cual permitiría contar con una cantidad adecuada de profesionales que entreguen soporte para el software libre en el futuro. Lamentablemente, en este Proyecto de Acuerdo sólo se mencionan las ventajas técnicas y económicas del software libre, pero no el aspecto filosófico y ético del software libre: el usuario es libre de usar, modificar y compartir el software.

A pesar de los esfuerzos por utilizar software libre en la administración pública chilena, el Ministro de Economía firmó un acuerdo con Microsoft Corporation el 9 de mayo de 2007 ([68], [69]) que, entre otros problemas, atenta contra las libertades de los usuarios, pues hace obligatorio el uso de software privativo para interactuar con las organizaciones públicas chilenas. Este acuerdo dio origen al Movimiento de Liberación Digital<sup>3</sup>, formado por ciudadanos disconformes con las consecuencias que podría tener dicho acuerdo [70].

## 4.2. Casos de migración en colegios y universidades

En Chile existe un proyecto llamado Edulinux para utilizar GNU/Linux en los colegios y liceos, con el fin de reutilizar los computadores que han quedado obsoletos, utilizando el sistema de terminales ligeros. Este sistema implica adquirir un servidor de altas prestaciones, el cual puede ser accedido desde computadores antiguos, que actúan como terminales. El costo de adquirir este servidor es

---

<sup>3</sup><http://www.liberaciondigital.org/>

menor al costo de adquirir computadores nuevos que reemplacen a los computadores antiguos.

En Rusia se instalará GNU/Linux en todos los colegios, con el objetivo de reducir la dependencia de software privativo extranjero. En el año 2008 se instalará una distribución rusa de GNU/Linux, llamada ALTLinux, en tres regiones piloto, y para el año 2009 se instalaría en todos los colegios de Rusia [71]. El motivo de esta decisión fue la multa que debió pagar el director de una escuela por instalar una copia no autorizada de Windows en su colegio, aunque también estuvo amenazado con ir a la cárcel [72].

El gobierno de Corea del Sur anunció en 2005 un plan para instalar una distribución coreana de GNU/Linux en unos 10.000 colegios de ese país. En 2007, Japón anunció que instalará GNU/Linux en los computadores de los colegios. Turquía también está realizando planes para utilizar GNU/Linux en los colegios. [73]

Si bien en Europa no hay países con políticas educacionales similares, varios colegios de los estados miembro están incursionando en el uso de software libre. En la región de Extremadura, España, hay unos 40.000 computadores con gnuLinEx, los cuales son utilizados por unos 80.000 estudiantes. En Italia, el gobierno regional de Umbría está invirtiendo 100.000 euros para promover el uso de software libre en escuelas locales. En París, unos 175.000 alumnos recibieron llaves USB o pendrives precargados con programas libres. [73]

En las escuelas de Geneva, Suiza, se migrarán unos 9.000 computadores de Windows a GNU/Linux. Una de las motivaciones es el ahorro de costos. Esta migración ocurre en el marco de un plan estatal que promueve el uso software libre en la administración. Por otra parte, se escogió software libre pues «incentiva la compartición y democratización del conocimiento, y ofrece autonomía para la adquisición de habilidades». Otra ventaja es que los estudiantes pueden trabajar en sus casas utilizando los mismos programas libres que utilizan en la escuela, lo cual «fortalece la igualdad de oportunidades». Además, se migró a software libre pues éste «evita proveer de clientes cautivos a grandes empresas» y porque utiliza formatos abiertos, lo cual fomenta la interoperatividad. [74]

La Universidad Los Ángeles de Chimbote<sup>4</sup>, en Perú, inició hace dos años un proceso de migración a software libre. Primero se modificó la malla de la carrera de Ingeniería de de Sistemas para que se utilice software libre en sus prácticas de laboratorio. Los servidores funcionan con sistemas operativos libres, al igual que casi la totalidad de los computadores de escritorio de las áreas administrativas. Se utiliza Windows solamente en casos puntuales, donde existen problemas con controladores de hardware.

### 4.3. Casos de migración en empresas

En Chile, en julio de 2006 la Sociedad de Fomento Fabril (SOFOFA), el Programa de las Naciones Unidas para el Desarrollo (PNUD) y el gobierno japonés iniciaron un proyecto para apoyar a las micro, pequeñas y medianas empresas (MIPYMEs) chilenas a través del uso de las tecnologías de información y comunicación (TIC) y el software libre [75]. En el marco de este proyecto, se realizó un estudio [76] con el fin de determinar la factibilidad de implementar software libre en las MIPYMEs y en qué medida el uso de las TIC y el software libre mejorarían el funcionamiento de las MIPYMEs. Además, bajo este proyecto se creó el sitio web *AplicaLibre*<sup>5</sup>, el cual busca difundir diferentes aplicaciones libres útiles para las MIPYMEs, además de facilitar la comunicación y colaboración entre empresas proveedoras de servicios basados en software libre y los usuarios en general. En [77] se describen varios casos exitosos de implantación de software libre en MIPYMEs y organizaciones

---

<sup>4</sup><http://www.uladech.edu.pe/>

<sup>5</sup><http://www.aplicoliberal.cl/>

gubernamentales.

Estos ejemplos son solamente algunos de los muchos casos de migración a software libre. Como se pudo apreciar, el software libre está siendo adoptado por distintos tipos de organizaciones no solamente por razones económicas o técnicas, sino porque además ofrece ventajas éticas y prácticas.

# Capítulo 5

## Plan de migración

En este capítulo se presenta la primera versión del plan de migración a software libre para la Universidad de Tarapacá. Debido a que se trata de la primera iteración, sólo incluye algunas fases del proceso de migración.

### 5.1. Objetivo

El objetivo del plan de migración es reemplazar por software libre todo el software privativo que se utiliza en las actividades oficiales de la Universidad, de tal forma que se puedan realizar las mismas tareas que se llevan a cabo actualmente con software privativo.

Excepcionalmente, se tolerará el uso de software privativo en casos específicos y debidamente justificados, donde su uso sea imprescindible. Sin embargo, esto no significa que el uso de software privativo sea éticamente aceptable, y por lo tanto se deberán hacer esfuerzos para evitar su uso.

Es importante aclarar que los programas o sistemas desarrollados por la Universidad para su uso interno, como la Intranet y los sistemas de Registraduría y Tesorería, corresponden a *software de uso privado* (ver definición en el Apéndice A), es decir, son software libre para la Universidad y por lo tanto no necesitan ser reemplazados.

### 5.2. Consideraciones

En algunos casos no será posible prescindir de un programa privativo en el corto plazo, debido a diferentes factores, los cuales se mencionan más adelante, en la sección 5.3.5. En estos casos, la migración debe realizarse en el mediano y largo plazo, realizando cambios de forma gradual.

Para evitar o reducir la resistencia al cambio por parte de los usuarios, la migración debe ser gradual, reemplazando un programa a la vez. También se debe considerar el uso de planes piloto, para realizar migraciones en menor escala.

Muchas veces un programa privativo no posee una alternativa equivalente en software libre, pero es posible utilizar varios programas libres que en su conjunto son equivalentes a dicho programa privativo.

Se seguirá el criterio de la Free Software Foundation para determinar si un programa es libre o privativo. Específicamente, se utilizará el listado de licencias disponible en [80].

Además, se debe formar o contratar a un grupo de personas con conocimientos técnicos en



administración de sistemas y redes que puedan dar soporte para software libre. Este equipo será el encargado de brindar soporte en cada una de las fases del proceso de migración. Debido a que el software libre crea un mercado libre de soporte, la Universidad puede llamar a una licitación y cualquier empresa de soporte puede postular, incluso si es una empresa de software privativo. Por lo tanto, no se estaría discriminando a ninguna empresa ni atentando contra la libre competencia.

### 5.3. Fases del proceso de migración a software libre

El proceso de migración a software libre está compuesto por varias fases, algunas de las cuales se pueden ejecutar en paralelo, mientras que otras deben llevarse a cabo de forma secuencial.

A continuación se explica cada fase del proceso de migración. Se presentan según el orden sugerido de ejecución.

#### 5.3.1. Convencer sobre la necesidad de migrar a software libre

En esta primera fase se debe convencer a las autoridades universitarias sobre la necesidad de la migración a software libre. Esta etapa es la más importante y no se debe desestimar. Si no es posible convencer sobre la necesidad de migrar a software libre, el plan de migración no tendrá la fortaleza necesaria para lograr su objetivo.

En los capítulos anteriores se han explicado las ventajas del software libre frente al software privativo. En el caso de las universidades, cobra una gran importancia la libertad para estudiar cómo funciona el software y la libertad para modificarlo según sus propósitos, tales como la investigación. El software privativo muchas veces no es adecuado para la investigación, pues al no conocer los algoritmos que implementa, un investigador no puede analizar completamente los resultados obtenidos con este tipo de software.

Económicamente, el software privativo puede ser más conveniente en el corto plazo, pero su costo en el largo plazo es considerablemente mayor, pues, a diferencia del software libre, requiere de la renovación de las licencias cada cierta cantidad de años.

Sin embargo, el uso de software libre debería ser establecido por la Universidad como una política basada en aspectos éticos y de libertad, pues sólo de esa forma el proceso de migración tendrá la fuerza suficiente para enfrentar a los obstáculos con los que se encuentre. Si se utilizan solamente criterios económicos y técnicos, el proceso de migración podría verse amenazado ante la presencia de software privativo que sea superior económica y técnicamente, pero inferior en términos de libertad.

También es importante que la comunidad universitaria en general esté informada de las razones del cambio, por una parte para reducir la resistencia, pero principalmente para que esté consciente de sus libertades en el mundo digital.

#### 5.3.2. Soporte técnico

La Dirección de Tecnología de la Información (DTI) y la Academia de Software Libre (ASL) serán las unidades encargadas de ejecutar el plan de migración. Para ello deben contar con el apoyo de todas las autoridades superiores de la Universidad.

La ASL debe evaluar las distintas alternativas libres y decidir cuáles se utilizarán, definir los programas para los cursos de capacitación, y dictar los cursos o bien buscar a los especialistas adecuados que puedan dictar otros cursos.

La DTI se encargará de brindar el soporte técnico humano y físico necesario durante el proceso de migración.

### 5.3.3. Identificar programas utilizados actualmente

En esta etapa se identifican los programas libres y privativos que se utilizan en la Universidad. Esto se puede lograr mediante encuestas o bien a través de reuniones con grupos de usuarios.

Los resultados de esta fase permitirán identificar posibles problemas, como software privativo que no tenga una alternativa libre de buena calidad. Además, se podrá saber si actualmente se está utilizando software libre en la Universidad, lo cual facilitaría el proceso de migración.

Se realizó una encuesta anónima a 120 personas, considerando académicos, funcionarios y alumnos de distintos departamentos y carreras. El objetivo de esta encuesta es determinar qué programas se utilizan actualmente, tanto libres como privativos, y cuál es la disponibilidad de los usuarios para cambiarlos. A continuación se interpretan los resultados obtenidos:

- El 72,4 % de los programas utilizados en la Universidad son privativos, mientras que el restante 27,6 % corresponde a software libre (incluyendo a la Intranet).
- La Intranet es utilizada por el 79,2 % de los encuestados.
- El 96,9 % de los funcionarios utiliza el paquete de ofimática Microsoft Office (privativo) y el 10,4 % utiliza OpenOffice (libre).
- El 53,1 % de los encuestados utiliza el cliente de correo electrónico Mozilla Thunderbird (libre) y el 10,4 % usa Outlook (privativo).
- El 50 % de los funcionarios utiliza el navegador web Firefox (libre), mientras que el 10,4 % utiliza Internet Explorer (privativo).
- Los programas privativos más utilizados son: Word (96,9 % de uso), Excel (92,7 %), PowerPoint (78,1 %), Acrobat Reader (25 %), Photoshop (24 %), Project (21,9 %) y MATLAB (20,8 %).
- De los programas privativos que tienen más de un 10 % de uso, solamente AutoCAD no posee una alternativa libre con características similares.
- Entre las actividades más frecuentes realizadas por los encuestados, en más de 5 %, solamente el CAD 3D presenta dificultades para ser realizada con software libre.
- El 38,5 % de los encuestados estaría dispuesto a cambiar de software; el 49 % solo lo haría si se realizan cursos de capacitación; y el 10,4 % no estaría dispuesto a cambiar el software que utiliza.

A partir de estos resultados se puede concluir que la búsqueda de alternativas libres y la resistencia al cambio por parte de los usuarios no deberían ser grandes obstáculos para el proceso de migración a software libre. Sin embargo, tampoco se deben subestimar estos factores.

### 5.3.4. Buscar alternativas libres

En esta etapa se buscan programas libres que puedan realizar las mismas funciones que los programas privativos identificados anteriormente.

En la web existen listados de software libre que pueden ser utilizados para buscar programas libres que realicen una determinada función o que reemplacen a un determinado programa privativo. Algunos recursos en la web para buscar programas libres son:

- El Directorio de Software Libre (<http://directory.fsf.org/>) de la Free Software Foundation
- Tabla de software libre para Windows (<http://www.gnu.org/software/for-windows.html>) elaborada por el proyecto GNU
- El sitio Alternativas Libres (<http://alts.homelinux.net/>) organiza los programas libres según la tarea que realizan y según los programas privativos a los cuales pueden reemplazar
- Graziano Sorbaioli mantiene un listado ([http://www.sorbaioli.org/software\\_pc.php](http://www.sorbaioli.org/software_pc.php)) de software libre para GNU/Linux, Windows y MacOS.
- El proyecto OpenApps (<http://www.aclibre.org/openapps.gsol>) recopila software libre para Windows

Como el objetivo del plan de migración es reemplazar, en la medida de lo posible, cada software privativo utilizado en la Universidad por software libre, se tendrá que reemplazar cada programa privativo por una alternativa libre.

El procedimiento para buscar alternativas libres es el siguiente. Se escoge un programa no libre que aparezca en la encuesta, se investiga sobre ese programa y se determinan sus principales funcionalidades. Después se buscan programas libres que cumplan con cada una de esas funcionalidades, si existen.

En algunos casos, especialmente con algunos paquetes de software privativo que incluyen una gran cantidad y variedad de funciones, es necesario un conjunto de programas libres para reemplazar a un programa privativo. Esto se debe a que en el software libre, generalmente, cada programa se enfoca en una única función, y no siempre posee una interfaz gráfica. Sin embargo, cada vez se desarrollan más entornos gráficos que permiten acceder más fácilmente a las funcionalidades de uno o varios programas libres.

En el Apéndice B se describen los programas privativos identificados anteriormente, y se presentan sus respectivas alternativas en software libre, si existen.

### 5.3.5. Identificar posibles obstáculos para la migración

En esta fase se deben identificar los factores que pueden dificultar el proceso de migración y proponer una forma de eliminarlos o disminuirlos. A continuación se mencionan algunos factores que podrían dificultar la migración a software libre:

- La resistencia natural de las personas frente al cambio, en general, y frente al cambio del software que utilizan, en particular. Sin embargo, ya han ocurrido cambios en el software utilizado sin que se produzca un rechazo por parte de los usuarios. Como ejemplos de estos cambios podemos mencionar a Windows y Office, los cuales han introducido cambios importantes en su interfaz entre versiones, y sin embargo los usuarios se han adaptado y continúan utilizándolos.
- Falta de capacidad técnica para dar soporte a software libre. A pesar de que cualquier técnico informático puede capacitarse para dar soporte para software libre, aún no hay una oferta tan amplia de soporte para software libre.

- Falta de software libre con capacidades y facilidad de uso similares al software privativo utilizado actualmente. Cada día el software libre está mejorando para ofrecer más funcionalidades y mayor facilidad de uso. Sin embargo, aun hay una brecha con respecto al software privativo en algunas áreas. Una forma de solucionar este problema es colaborando con los proyectos que buscan desarrollar alternativas libres para ciertas tareas, ya sea donando dinero, contratando a programadores que trabajen en esos proyectos, orientando el trabajo académico para ayudar en el desarrollo de dichos proyectos, o de alguna otra forma.
- Indiferencia o ignorancia frente al tema de la libertad digital. Para esto se puede formar un grupo de activistas que promuevan la importancia del software libre y de una sociedad libre en la era digital.
- Existencia de un convenio con un tercero que haga obligatorio el uso de software privativo. Por ejemplo, Microsoft realiza donaciones de computadores con la condición de que ejecuten únicamente software privativo especificado por esa empresa.
- Una reciente adquisición de licencias de software privativo. La administración podría tener la sensación de haber desperdiciado el dinero gastado en licencias de software privativo. En este caso se podría destacar el hecho de que una migración a software libre evitaría futuros costos de licencias.

### 5.3.6. Utilizar software libre en los servidores

El software libre para servidores se destaca por su calidad, seguridad, eficiencia y estabilidad. Además, para los usuarios que acceden a servicios como el correo electrónico o a la Intranet les resulta transparente el hecho de que los servidores utilicen software libre o privativo. Por lo tanto, se recomienda comenzar la migración en los servidores de la Universidad.

El personal que trabaje en esta fase de la migración ganará experiencia y estará mejor capacitado para brindar soporte en las siguientes fases, en las cuales se tendrá que instalar y configurar software libre en los computadores de los funcionarios.

La Universidad ya está migrando a software libre en los servidores. Se han reemplazado las versiones privativas de Unix, como Solaris, por el sistema operativo libre GNU/Linux, específicamente las distribuciones Debian y RedHat.

Técnicamente, tanto RedHat como Debian son apropiados para su uso en servidores, pero desde el punto de vista de la libertad, no son muy adecuados, pues incluyen software privativo en sus repositorios y componentes no libres (denominados *binary blobs*) en el núcleo Linux. Como se ha explicado en capítulos anteriores, el software privativo posee intrínsecamente un problema de seguridad y de privacidad, pues su código fuente no está disponible, el cual se hace más grave cuando se utiliza en servidores que manejan datos e información personal de una gran cantidad de funcionarios y alumnos.

gNewSense<sup>1</sup>, BLAG<sup>2</sup> y Ututo<sup>3</sup> son distribuciones de GNU/Linux 100 % libres, pero no están diseñadas específicamente para servidores, por lo que pueden requerir de un esfuerzo adicional para configurarlas para este fin. Además, no cuentan con tantos voluntarios como en el caso de Debian, ni con el soporte técnico comercial que posee RedHat. Como alternativa a estas distribuciones 100 % libres, se puede utilizar cualquier distribución de GNU/Linux pero sin instalar software privativo.

---

<sup>1</sup><http://www.gnewsense.org/>

<sup>2</sup><http://www.blagblagblag.org/>

<sup>3</sup><http://ututo.org/>

Es relativamente fácil evitar instalar programas privativos, pero es difícil evitar el firmware no libre, pues viene incrustado en el núcleo Linux. Afortunadamente, existe un proyecto llamado Linux-libre<sup>4</sup>, cuyo objetivo es eliminar el firmware privativo de Linux.

Se recomienda utilizar gNewSense en los servidores, pues es una distribución 100 % libre que está basada en Ubuntu, que a su vez es un derivado de Debian, y por lo tanto su configuración es similar. Como segunda opción, se sugiere utilizar Debian con Linux-libre en los servidores, sin activar los repositorios de software privativo (*non-free*) y el de software libre que depende de software privativo (*contrib*).

Por otra parte, en la Universidad se utiliza el gestor de bases de datos Oracle, que es software privativo. Se sugiere cambiarlo por algún gestor de bases de datos que sea libre, como PostgreSQL, Firebird o MySQL.

### 5.3.7. Utilizar software libre sobre Windows

En esta etapa los funcionarios utilizarán aplicaciones libres disponibles para Windows, tales como OpenOffice, Firefox y Thunderbird. El objetivo es que los usuarios se acostumbren a utilizar programas libres sin tener que cambiar de sistema operativo.

Para apoyar a los funcionarios se deberán realizar cursos de capacitación sobre estos programas libres. Los cursos deben ser gratuitos y certificados, pero no obligatorios, ya que algunas personas pueden aprender por sí mismas a utilizar programas libres. En el apéndice C se propone un programa para un curso de capacitación en OpenOffice.

El uso generalizado de Microsoft Office es un factor importante a considerar en el proceso de migración, pues se deberá capacitar a la mayoría de los funcionarios en el uso de OpenOffice. Sin embargo, estos cursos de capacitación deberían ser breves, pues al ser bastante similares ambos paquetes de software, solo sería necesario explicar sus diferencias.

Por otra parte, el amplio porcentaje de uso del cliente de correo libre Thunderbird, elimina o reduce la necesidad de realizar cursos de capacitación para programa. Lo mismo ocurre con el navegador web libre Firefox.

### Utilizar formatos de archivo abiertos

En esta etapa también se establecerá el uso de formatos de archivo abiertos, tales como el formato OpenDocument (ODF), en los documentos digitales generados al interior de la Universidad.

Se debe evitar el uso de formatos cerrados, como los formatos DOC, PPT y XLS de Microsoft Office. También se debe evitar el formato Microsoft Office Open XML (MSOOXML), pues, si bien Microsoft publicó sus especificaciones, éstas poseen varios defectos, como se explica en [47].

En la Sección 2.1.4 se explicaron las ventajas de los formatos de archivo abiertos. El uso de formatos abiertos permite que la información generada por la Universidad pueda ser leída sin problemas en el futuro, sin la necesidad de contar con un computador que utilice el mismo programa con el que fueron creados los documentos. Además, el uso de formatos abiertos permite que la información pueda ser leída correctamente con software libre, lo cual facilita la migración.

---

<sup>4</sup><http://directory.fsf.org/project/linux/>

### 5.3.8. Cambiar Windows por GNU/Linux con algunos componentes privativos

En esta etapa se cambiará el sistema operativo no libre utilizado actualmente por un sistema libre: GNU/Linux. Sin embargo, se permitirá el uso de ciertos programas y controladores privativos, tales como el reproductor de Flash y controladores privativos de hardware.

El objetivo de esta fase es que los usuarios se habitúen a un sistema operativo libre, pero que puedan recurrir a ciertos componentes no libres para funciones específicas.

La mayoría de las distribuciones de GNU/Linux incluyen o facilitan la instalación de controladores y aplicaciones privativos, por lo tanto cualquiera de ellas puede ser utilizada en esta etapa. Sin embargo, se recomienda utilizar Debian o Ubuntu, pues su uso facilitará la posterior utilización de gNewSense, una distribución 100 % libre.

La Universidad actualmente utiliza GNU/Linux en los terminales de la Biblioteca Central y en algunos laboratorios de computación. Por lo tanto, muchos alumnos ya están habituados a este sistema operativo, lo cual favorece el proceso de migración.

En esta etapa se realizarán cursos de capacitación para que los funcionarios y alumnos se habitúen a la interfaz gráfica del sistema operativo libre GNU/Linux. Existen dos interfaces gráficas populares para GNU/Linux, GNOME y KDE, por lo que se deberá escoger una de ellas.

En general, los cursos deben estar orientados a nivel de usuario, y no a nivel de administrador del sistema. Las tareas de administración del sistema, como la instalación del sistema operativo, instalación de software y configuración de hardware, serán realizadas por la unidad encargada de brindar soporte para las TIC en la universidad, a menos que un funcionario desee realizarlas por cuenta propia. También se deben ofrecer cursos a nivel de administración del sistema para los usuarios que estén interesados en mantener sus propios computadores.

La Biblioteca Central realiza cursos breves para que los alumnos aprendan a utilizar GNU/Linux y aplicaciones libres como OpenOffice, los cuales están instalados en los terminales de la Biblioteca. El alto nivel de uso de los terminales demuestra que una breve explicación es suficiente para habituarse al entorno gráfico de GNU/Linux.

### 5.3.9. Adquirir hardware compatible con software libre

En esta etapa la Universidad comprará hardware que sea compatible con software libre, es decir, que no requiera de controladores ni firmware privativos para funcionar. No es suficiente, desde el punto de vista de la libertad, que un dispositivo de hardware funcione en GNU/Linux con un controlador privativo, como es el caso de algunas tarjetas de video, tarjetas de red inalámbrica y cámaras web, entre otros. Por lo tanto, se deberá elaborar una especificación que le permita a la Universidad adquirir computadores cuyo hardware funcione con GNU/Linux y controladores libres. Para crear esta especificación se utilizará como referencia el listado de hardware compatible con software libre de la Free Software Foundation [41].

Además, si es posible, se escogerán placas madre que puedan funcionar con un BIOS libre, como coreboot<sup>5</sup>. Sin embargo, la instalación de un BIOS libre es difícil y requiere la intervención del hardware. Existen algunas placas madres vienen con un BIOS libre preinstalado, pero son pocas y es difícil encontrarlas.

Existen algunas empresas que venden computadores preparados especialmente para funcionar

---

<sup>5</sup>En [http://www.coreboot.org/Supported\\_Motherboards](http://www.coreboot.org/Supported_Motherboards) se puede ver un listado de placas madre que soportan coreboot

con software libre. Los Alamos Computers<sup>6</sup> vende computadores y notebooks con gNewSense (una distribución de GNU/Linux 100 % libre) preinstalado. Silicon Mechanics<sup>7</sup> vende servidores con un BIOS libre, coreboot, preinstalado.

### 5.3.10. Utilizar GNU/Linux sin componentes privativos

En esta etapa se utilizará únicamente software libre. No se soportará el uso de aplicaciones ni controladores privativos.

En esta fase se deberá utilizar alguna distribución de GNU/Linux que tenga por objetivo no incluir aplicaciones, controladores y firmware privativos, tales como gNewSense<sup>8</sup>, BLAG<sup>9</sup> y Ututo<sup>10</sup>. Se recomienda utilizar gNewSense, pues está basada en Ubuntu, una distribución popular para uso de escritorio, la cual a su vez está basada en Debian, que es la distribución que aquí se sugiere para los servidores.

Como alternativa, también se podrá utilizar cualquier distribución de GNU/Linux, pero sin instalar software privativo y utilizando el núcleo Linux-libre<sup>11</sup>, que no posee firmware privativo. Si no se desea utilizar gNewSense, se recomienda como segunda opción utilizar Debian con el núcleo Linux-libre, sin el repositorio de software privativo (*non-free*) ni el de software libre que depende de software privativo (*contrib*).

Al llegar a esta etapa, se habrá logrado el objetivo del plan. Sin embargo, se deberá continuar con los esfuerzos para evitar el uso de software privativo.

### 5.3.11. Desarrollo de software libre

En esta fase se busca fomentar el desarrollo de software libre, especialmente de aquél que puede servir como alternativa al software privativo que aun se utiliza en la Universidad.

Como se ha mencionado, no todos los programas privativos poseen una alternativa en software libre que ofrezca las mismas funcionalidades. En este plan, estos programas se consideran excepciones, y se permite su uso hasta que se desarrolle un programa libre que pueda reemplazarlo.

En general, los proyectos de software libre aceptan la colaboración de voluntarios y la donación de dinero. La Universidad podría, y debería, ayudar al desarrollo de software libre.

Una forma de colaborar con estos proyectos puede ser mediante actividades académicas, especialmente del área de la informática. Por ejemplo, se podría definir como taller de un curso, o como tema de tesis, la implementación de una funcionalidad necesaria en algún programa libre. Además, los académicos podrían investigar programas libres y proponer formas para mejorarlos.

También es posible colaborar mediante la donación de dinero. Se debe recordar que para utilizar software libre no se requiere el pago de licencias. Por lo tanto, se puede destinar parte del dinero que se gastaría en licencias de software privativo, en donaciones para proyectos de software libre que se utilizan o que se podrían utilizar en la Universidad.

---

<sup>6</sup><http://laclinux.com/>

<sup>7</sup><http://siliconmechanics.com/>

<sup>8</sup><http://www.gnewsense.org/>

<sup>9</sup><http://www.blagblagblag.org/>

<sup>10</sup><http://ututo.org/>

<sup>11</sup><http://directory.fsf.org/project/linux/>

## Capítulo 6

# Conclusión y trabajo futuro

En este trabajo se han expuesto los problemas éticos, de seguridad y de privacidad intrínsecos al software privativo. Luego se explicó qué es el software libre y cómo logra superar estos problemas del software privativo, destacando sus ventajas éticas, económicas y legales, y en seguridad, privacidad e independencia tecnológica. También se explicaron algunos factores que amenazan al software libre y a las libertades de los usuarios de computadores, como el DRM (Gestión Digital de “Derechos” o Restricciones), el hardware sin especificaciones, los formatos de archivo cerrados, las patentes de software, las leyes anti-copia, y las presiones de las grandes empresas de software privativo. Luego se expusieron algunos casos de migración a software libre en distintas partes del mundo. Finalmente, se propuso un plan de migración a software libre para la Universidad de Tarapacá de Arica, considerando los programas que se utilizan actualmente y sus respectivas alternativas en software libre.

Según los resultados obtenidos a partir de las encuestas y la búsqueda de alternativas libres, se puede concluir que la migración a software libre es factible. Si la Universidad se lo propone, puede utilizar únicamente software libre y obtener así independencia tecnológica, es decir, la libertad para controlar y desarrollar sus sistemas computacionales de la forma en que estime conveniente según sus propios intereses, y no según los intereses de las empresas de software privativo.

Como analizamos anteriormente, el objetivo del software libre es proteger las libertades de los usuarios de computadores. Para lograrlo, además de desarrollar software que respete estas libertades, el movimiento del software libre busca crear conciencia en la sociedad, es decir, en los habitantes de todos los países del mundo, para que rechace el software que no respete sus libertades. Stallman [82] considera que la medida más importante para que los países utilicen software libre, es la migración en colegios y universidades. Estas instituciones educativas, cuando escogen entre software privativo, que subyuga al usuario y lo hace dependiente, y software libre, que otorga auto-suficiencia, están decidiendo el futuro de la sociedad. Por lo tanto, está mal que los colegios y universidades enseñen a utilizar software privativo. El autor de la tesis, así como el profesor guía, concuerdan plenamente con lo expuesto anteriormente. La sociedad, y en particular las instituciones de educación superior, deben defender la libertad de aprender y de compartir el conocimiento, y por lo tanto deben utilizar y promover el software libre, pues garantiza las libertades anteriormente expuestas en el ámbito del software.

Como trabajo futuro se propone la ejecución del plan de migración y la generación de nuevas versiones de éste. En particular, se debe planificar con mayor detalle la migración en unidades específicas. Por ejemplo, se debe analizar cuál es la mejor opción para reemplazar el sistema de biblioteca Aleph y se debe diseñar un procedimiento para trasladar las bases de datos y volver a entrenar a los funcionarios que utilizan este sistema, entre otras consideraciones.



El plan de migración a software libre puede ser extendido de varias formas. Se podrían incorporar programas libres específicos para apoyar la enseñanza de distintas disciplinas y la investigación en diferentes áreas, para aprovechar las posibilidades que ofrece el software libre, y evitar o reducir la adopción de software privativo en actividades que aún no se realizan, pero que se podrían realizar. Por otra parte, sería conveniente formalizar una metodología para la migración a software libre adaptada a las universidades chilenas.

Finalmente, un área de reciente discusión en la comunidad del software libre [83] es el asunto de la libertad y la privacidad en las aplicaciones o servicios web, tema no tratado en esta tesis. Las aplicaciones web son, básicamente, programas que se ejecutan en un servidor y no en el computador del usuario, por lo tanto no se le aplican las cuatro libertades propuestas por Stallman. Sin embargo, existe un movimiento que busca aplicar el mismo concepto del software libre a las aplicaciones web. La licencia Affero GPL se puede utilizar para hacer obligatoria la distribución del código fuente de un programa que es utilizado como servicio web. En el presente trabajo no se consideró el uso de aplicaciones web libres. Se podría extender el plan de migración para incorporar el uso de aplicaciones web libres. Por ejemplo, se podrían desplegar aplicaciones web libres para trabajo en grupo, gestión de proyectos, correo electrónico y compartición de documentos.

# Bibliografía

- [1] Naciones Unidas; «Declaración Universal de los Derechos humanos», 1948-12-10, disponible en <http://www.un.org/spanish/aboutun/hrights.htm>
- [2] Stallman, Richard M.; «Overview of the GNU System», 1996, disponible en <http://www.gnu.org/gnu/gnu-history.html>, accedido en 2008-08-27
- [3] Gates, William H.; «An Open Letter to Hobbyists», 1976-2-3, disponible en <http://www.blinkenlights.com/classiccmp/gateswhine.html>
- [4] Stallman, Richard M.; «La Definición de Software Libre», 1996, disponible en <http://www.gnu.org/philosophy/free-sw.es.html>, accedido en 2008-08-27
- [5] Stallman, Richard M.; «El manifiesto de GNU», 1985, disponible en <http://www.gnu.org/gnu/manifesto.es.html>, accedido en 2008-08-27
- [6] Stallman, Richard M.; «Por qué el software no debe tener propietarios», 1994, disponible en <http://www.gnu.org/philosophy/why-free.es.html>, accedido en 2008-08-27
- [7] Wheeler, David A.; «Why Open Source Software / Free Software (OSS/FS, FLOSS, or FOSS)? Look at the Numbers!», revisión del 16 de abril de 2007, disponible en [http://www.dwheeler.com/oss\\_fs\\_why.html](http://www.dwheeler.com/oss_fs_why.html)
- [8] Acero, Fernando; «Mentiras arriesgadas (y gobiernos que se dejan seducir por ellas)», 2005-3-14, disponible en <http://www.kriptopolis.org/node/496/print>
- [9] Scolnik, Hugo; «La criptografía debe pensar en la gente común para aplicar la seguridad», 2004-2-12, disponible en <http://www.belt.es/noticias/2004/marzo/10/entrevista.htm>
- [10] «El programa de Microsoft para mostrar el código de Windows a gobiernos», Hispalinux, 2003-2-5, disponible en <http://www.hispalinux.es/node/430/print>
- [11] Entrevista a Diego Saravia, Software Libre Chile, 2006-8-30, disponible en <http://www.softwarelibre.cl/drupal//?q=node/768/print>
- [12] «Communications Assistance for Law Enforcement Act», Electronic Privacy Information Center, disponible en [http://www.epic.org/privacy/wiretap/calea/calea\\_law.html](http://www.epic.org/privacy/wiretap/calea/calea_law.html), accedido en 2008-9-16
- [13] Hinestroza, Jorge et al.; «El sabotaje tecnológico de PDVSA y los derrames petroleros en el Lago de Maracaibo», junio 2003, disponible en <http://www.mindfully.org/Plastic/Polyvinylchloride/PDVSA-Petroleros-Maracaibo19jun03.htm>

- [14] Hineostroza, Jorge; «Terrorismo ambiental aplican en el Lago de Maracaibo saboteadores petroleros», 2003-1-10, disponible en <http://www.aporrea.org/actualidad/n3830.html>, accedido en 2008-9-16
- [15] Schenier, Bruce; «Secrecy, Security, and Obscurity», 2002-5-15, disponible en <http://www.schneier.com/crypto-gram-0205.html#1>
- [16] Stallman, Richard M.; «The Free Software Movement and the Future of Freedom», 2006-3-9, disponible en <http://fsfeurope.org/documents/rms-fs-2006-03-09.en.html>, accedido en 2008-9-16
- [17] «Software Abierto: Factibilizando la neutralidad tecnológica para Chile», Fundación País Digital, 2005, disponible en <http://www.paisdigital.org/files/software%20abierto.pdf>
- [18] Zeitschriften, Heise; «WGA notification just doesn't stop», 2007-3-6, disponible en <http://www.heise-security.co.uk/news/86294>
- [19] Evers, Joris; «Microsoft's antipiracy tool phones home daily», ZDNet, 2006-6-8, disponible en [http://news.zdnet.com/2100-3513\\_22-6081286.html](http://news.zdnet.com/2100-3513_22-6081286.html)
- [20] «OpenPGP Message Format», Internet Engineering Task Force, noviembre 1998, disponible en <http://www.ietf.org/rfc/rfc2440.txt>
- [21] Greve, Georg; «Why should you care?», disponible en <http://drm.info/fsfe/why>, accedido en 2007-6-12
- [22] Stallman, Richard M.; «Opposing Digital Rights Mismanagement», 2006, disponible en <http://www.gnu.org/philosophy/opposing-drm.html>, accedido en 2007-6-12
- [23] «Advanced Access Content System», Wikipedia, disponible en [en.wikipedia.org/wiki/Advanced\\_Access\\_Content\\_System](http://en.wikipedia.org/wiki/Advanced_Access_Content_System), accedido en 2007-6-15
- [24] Torvalds, Linus; «Flame Linus to a crisp!», 2003-4-23, disponible en [http://groups.google.com/group/fa.linux.kernel/browse\\_frm/thread/2f93510ff38d9089/](http://groups.google.com/group/fa.linux.kernel/browse_frm/thread/2f93510ff38d9089/), accedido en 2007-6-15
- [25] Marson, Ingrid; «UK Linux guru [Alan Cox] backs GPL 3», ZDNet UK, 2006-1-31, disponible en <http://news.zdnet.co.uk/software/0,1000000121,39249913,00.htm>
- [26] «Transcript of Eben Moglen at the 3rd international GPLv3 conference», Free Software Foundation Europe, 2006-6-22, disponible en <http://www.fsfeurope.org/projects/gplv3/barcelona-moglen-transcript.en.html#drm>
- [27] «GPLv3 Discussion Draft FAQ», Free Software Foundation, disponible en <http://gplv3.fsf.org/dd3-faq>, accedido en 2007-6-15
- [28] Marson, Ingrid; «Trusted Computing comes under attack», ZDNet UK, 2006-1-27, disponible en <http://news.zdnet.co.uk/security/0,1000000189,39249368,00.htm>, accedido en 2007-7-7
- [29] Schneier, Bruce; «Palladium and the TCPA», Crypto-Gram Newsletter, 2002-8-15, disponible en <http://www.schneier.com/crypto-gram-0208.html#1>, accedido en 2007-7-7

- [30] «Trusting Computing», Wikipedia, disponible en [http://en.wikipedia.org/wiki/Trusted\\_computing](http://en.wikipedia.org/wiki/Trusted_computing), accedido en 2007-7-7
- [31] Correa, Carlos M.; «Uso justo en la era digital», noviembre 2006, ISSN:1697-073X, disponible en <http://www.gestioncultural.org/boletin/2006/bgc15-CCorrea.pdf>
- [32] «FairPlay (DRM)», Wikipedia, disponible en [http://en.wikipedia.org/wiki/FairPlay\\_\(DRM\)](http://en.wikipedia.org/wiki/FairPlay_(DRM)), accedido en 2007-6-16
- [33] «Apple Unveils Higher Quality DRM-Free Music on the iTunes Store», Apple, 2007-4-2, disponible en <http://www.apple.com/pr/library/2007/04/02itunes.html>
- [34] Kaplan, Dan; «Privacy advocates concerned with iTunes' DRM-free music», SC Magazine UK, 2007-6-7, disponible en <http://scmagazine.com/uk/news/article/662875/privacy-advocates-concerned-itunes-drm-free-music/>
- [35] Cohen, Peter; «iTunes user sues Apple over FairPlay DRM», disponible en <http://www.macworld.com/news/2005/01/06/slattery/index.php?pf=1>, accedido en 2007-6-18
- [36] Gutmann, Peter; «A Cost Analysis of Windows Vista Content Protection», disponible en [http://www.cs.auckland.ac.nz/~pgut001/pubs/vista\\_cost.html](http://www.cs.auckland.ac.nz/~pgut001/pubs/vista_cost.html), accedido en 2007-6-25
- [37] White, Nick; «Windows Vista Content Protection - Twenty Questions (and Answers)», 2007-1-20, disponible en <http://windowsvistablog.com/blogs/windowsvista/archive/2007/01/20/windows-vista-content-protection-twenty-questions-and-answers.aspx>, accedido en 2007-6-25
- [38] Stallman, Richard M.; «The GNU Project», 1998, disponible en <http://www.gnu.org/gnu/thegnuproject.html>, accedido en 2007-6-25
- [39] «The road to hardware free from restrictions: How hardware vendors can help the free software community», Free Software Foundation, febrero 2007, disponible en [http://www.fsf.org/resources/hw/how\\_hardware\\_vendors\\_can\\_help.html](http://www.fsf.org/resources/hw/how_hardware_vendors_can_help.html), accedido en 2007-5-15
- [40] Panepento, Peter; «Young People Buy Products Tied to Social Causes, but Doubt Their Value», disponible en <http://www.philanthropy.com/premium/articles/v19/i03/03002601.htm>, accedido en 2007-6-25
- [41] «Hardware Devices that Support GNU/Linux», Free Software Foundation, disponible en <http://www.fsf.org/resources/hw/>, accedido en 2007-6-25
- [42] Stallman, Richard M.; «The Free Software Foundation's Campaign for Free BIOS», 2005-2-26, disponible en <http://www.fsf.org/news/freebios.html>, accedido en 2007-6-25
- [43] Seebach, Peter; «Open BIOSes for Linux, Modern systems need not be held back by a legacy boot process», 2006-8-31, disponible en <http://www-128.ibm.com/developerworks/linux/library/l-bios.html>, accedido en 2007-6-25
- [44] «Getting started with Ogg and installing Ogg plugins», Free Software Foundation, disponible en <http://www.fsf.org/resources/formats/playogg>, accedido en 2007-6-26

- [45] «About Xiph», disponible en <http://xiph.org/about/>, accedido en 2007-6-26
- [46] «Open Document Format for Office Applications (OpenDocument) v1.0», International Organization for Standardisation (ISO), disponible en <http://www.iso.org/iso/en/CatalogueDetailPage.CatalogueDetail?CSNUMBER=43485&scopelist=PROGRAMME>, accedido en 2007-7-7
- [47] «Say NO to the Microsoft Office format as an ISO standard», disponible en <http://www.nooxml.org/petition>, accedido en 2007-6-26
- [48] Hiser, Sam; «Achieving Openness: A Closer Look at ODF and OOXML», 2007-6-14, disponible en <http://www.onlamp.com/pub/a/onlamp/2007/06/14/achieving-openness-a-closer-look-at-odf-and-ooxml.html>
- [49] «DADVSI», Wikipedia, disponible en <http://en.wikipedia.org/wiki/DADVSI>, accedido en 2007-6-25
- [50] Stallman, Richard M; Intervención sobre el software libre y la ley DAVDSI en la Univerité d'été UDF, 2006, video disponible en <http://video.google.ca/videoplay?docid=6502696231837859029>, accedido en 2007-6-25
- [51] Stallman, Richard M.; «Fighting Software Patents - Singly and Together», 2004, disponible en <http://www.gnu.org/philosophy/fighting-software-patents.html>, accedido en 2007-7-7
- [52] «MSFT Annual Report 2006», Microsoft, disponible en [http://www.microsoft.com/msft/reports/ar06/staticversion/10k\\_fr\\_bus\\_07.html](http://www.microsoft.com/msft/reports/ar06/staticversion/10k_fr_bus_07.html)
- [53] «Decreto N° 3.390», Gobierno de Venezuela, 2004-12-23, disponible en <http://www.gobiernoenlinea.ve/docMgr/sharedfiles/Decreto3390.pdf>, accedido en 2007-8-7.
- [54] «El Presidente llama a utilizar software libre en un mensaje a 17 países del continente», Presidencia de Ecuador, 2007-4-26, disponible en <http://www.presidencia.gov.ec/noticias.asp?noid=9267>, accedido en 2007-8-7
- [55] «El Presidente Correa autoriza la utilización del Software Libre en la Administración Pública Central», Prensa Presidencial, Gobierno de la República del Ecuador, 2008-4-11, disponible en <http://www.presidencia.gov.ec/noticias.asp?noid=13318>
- [56] «La Junta de Extremadura adaptará todos sus ordenadores al software libre LinEx en un año», Junta de Extremadura, 2006-7-28, disponible en <http://www.juntaaldia.es/noticias/detalle.aspx?id=1724>, accedido en 2008-4-7
- [57] «Acuerdo para la implantación de programas informáticos libres en los ordenadores personales de la Junta de Extremadura», Junta de Extremadura, 2006-7-25, disponible en [http://www.estandaresabiertos.com/index.php?option=com\\_content&task=view&id=56&Itemid=38](http://www.estandaresabiertos.com/index.php?option=com_content&task=view&id=56&Itemid=38), accedido en 2008-4-7
- [58] Decreto 72/2003, Medidas de Impulso de la Sociedad del Conocimiento en Andalucía, Junta de Andalucía, 2003-3-21, disponible en <http://www.andaluciajunta.es/SP/AJ/CDA/Ficheros/ArchivosPdf/DecretoConocimiento.pdf>, accedido en 2008-11-30

- [59] «Aprobado el plan de migración de Munich», 2004-6-17, disponible en <http://softlibre.barrapunto.com/article.pl?sid=04/06/17/1231210>, accedido en 2008-4-6
- [60] «Federal Employment Office switches to Linux», Heise Online, 2008-1-22, disponible en <http://www.heise.de/english/newsticker/news/102218>, accedido en 2008-4-4
- [61] «French MPs dump Windows for Linux», ZDNet France, 2006-11-27, disponible en <http://www.zdnetasia.com/news/software/0,39044164,61970345,00.htm>, accedido en 2008-4-7
- [62] «French police deal blow to Microsoft», 2008-1-30, disponible en [http://afp.google.com/article/ALeqM5iU4Lq7tOR\\_WV0JLZ3IeRaIH03x6w](http://afp.google.com/article/ALeqM5iU4Lq7tOR_WV0JLZ3IeRaIH03x6w), accedido en 2008-4-7
- [63] «El Parlamento italiano migra de Windows a [GNU/]Linux», 2007-7-13, disponible en <http://www.kriptopolis.org/parlamento-italiano-migra-a-linux>, accedido en 2008-4-7
- [64] «Legislación sobre el uso del software libre en la administración pública», disponible en <http://www.cpsr-peru.org/swlibre/swlibre/>, accedido en 2008-4-7
- [65] Carta Iberoamericana de Gobierno Electrónico, 2007-11-10, disponible en <http://www.clad.org.ve/cartagobelec.pdf>, accedido en 2008-4-12
- [66] «Cuba busca adoptar el software libre», 2007-2-19, disponible en <http://www.laflecha.net/canales/softlibre/noticias/cuba-busca-adoptar-el-software-libre>, accedido en 2008-4-4
- [67] «Proyecto de acuerdo N° 368: Solicita implementar en Presidencia, Ministerios y Servicios de la Administración del Estado el sistema de software libre», Cámara de Diputados de Chile, 2007-6-6, disponible en <http://www.camara.cl/pacuerdo/docpacuerdo.aspx?prmID=2036>, accedido en 2008-4-4
- [68] Leal, Christian; «El día que Chile se vendió a Microsoft», 2007-7-23, disponible en <http://www.elfrancotirador.cl/2007/07/23/el-dia-que-chile-se-vendio-a-microsoft/>, accedido en 2008-4-6
- [69] «Microsoft y el gobierno de Chile firman acuerdo de colaboración», Microsoft, 2007-5-9, disponible en <http://www.microsoft.com/chile/prensa/2007/may/acuerdo/>, accedido en 2008-4-6
- [70] «Por qué los chilenos rechazan el acuerdo Microsoft-Gobierno de Chile», enero 2008, disponible en [http://www.liberaciondigital.org/material/por\\_que\\_no\\_al\\_acuerdo.pdf](http://www.liberaciondigital.org/material/por_que_no_al_acuerdo.pdf), accedido en 2008-4-6
- [71] «Russian OS to be installed in every school», cnews, 2007-10-14, disponible en <http://eng.cnews.ru/news/top/indexEn.shtml?2007/09/14/266177>, accedido en 2008-4-7
- [72] «Rusia toma venganza de Microsoft y apoya al software libre», INFOBAE, 2008-1-31, disponible en <http://tecnologia.infobaeprofesional.com/notas/60917-Rusia-toma-venganza-de-Microsoft-y-apoya-al-software-libre.html?cookie>, accedido en 2008-4-4

- [73] «Russia to install GNU/Linux in every school», 2007-9-25, disponible en <http://ec.europa.eu/idabc/en/document/7219/531>, accedido en 2008-4-7
- [74] «9000 PCs in Swiss schools migrating to GNU/Linux in September», Free Software Daily, 2007-4-5, disponible en [http://www.tdg.ch/pages/home/tribune\\_de\\_geneve/1\\_actu/sciences\\_hi\\_tech/detail\\_hi\\_tech/\(contenu\)/211880](http://www.tdg.ch/pages/home/tribune_de_geneve/1_actu/sciences_hi_tech/detail_hi_tech/(contenu)/211880), accedido en 2008-4-10
- [75] «Proyecto SOFOFA-PNUD potenciará el uso de la computación e Internet en MIPYMES y Municipalidades», 2006-7-18, disponible en <http://www.pnud.cl/prensa/noticias-2006/18-07-2006.asp>, accedido en 2008-4-4
- [76] Velásquez, Juan; Fuentes, Alejandro; Jiménez, Ángel; «Mejoramiento de la Gestión y Uso de TIC's de las MIPYMES y Gobiernos Locales a través de Software Libre», Universidad de Chile, mayo 2007, disponible en [http://www.pnud.cl/publicaciones/05-07-2007-Informe\\_Pyme.pdf](http://www.pnud.cl/publicaciones/05-07-2007-Informe_Pyme.pdf), accedido en 2008-4-4
- [77] Casos de éxito de AplicaLibre, disponible en [http://www.aplicalibre.cl/index.php?option=com\\_content&task=view&id=14&Itemid=28](http://www.aplicalibre.cl/index.php?option=com_content&task=view&id=14&Itemid=28), accedido en 2008-4-4
- [78] «Navy to focus only on open systems», 2008-3-6, disponible en <http://www.fcw.com/online/news/151858-1.html>, accedido en 2008-3-4
- [79] «NASA Open Source Software», disponible en <http://opensource.arc.nasa.gov/>, accedido en 2008-4-11
- [80] Stallman, Richard M.; «Various Licenses and Comments about Them», disponible en <http://www.gnu.org/licenses/license-list.html>, accedido en 2007-8-20
- [81] da Rosa, Fernando; Heinz, Federico; «Guía práctica sobre Software Libre. Su selección y aplicación local en América Latina y el Caribe», UNESCO, 2007, ISBN 92-9089-103-3, disponible en <http://fedaro.info/?p=49>, accedido en 2008-4-12
- [82] Entrevista a Richard Stallman, PRAGOTI, 2008-7-17, disponible en <http://www.pragoti.org/node/1646>, accedido en 2008-4-12
- [83] «Freedom for Web Services», Free Software Foundation, 2008, disponible en <http://www.fsf.org/news/FreedomForWebServices>, accedido en 2008-08-11
- [84] Página principal del Proyecto GNU, disponible en <http://www.gnu.org/>, accedido en 2007-1-26
- [85] «GNU General Public License», Free Software Foundation, disponible en <http://www.gnu.org/licenses/gpl.html>, accedido en 2007-1-26
- [86] «Los Peligros de las patentes de software: Innovación», disponible en <http://www.nosoftwarepatents.com/es/m/dangers/innovation.html>, accedido en 2007-1-26
- [87] «Digital Millennium Copyright Act», Wikipedia, disponible en <http://en.wikipedia.org/wiki/DMCA>, accedido en 2007-1-26
- [88] Galeano, Carlos; Mantilla, Juan; Duque, Carlos et al. «Herramientas de Software con Licencia Pública General para el Modelado por Elementos Finitos» Dyna online, 2007, vol.74, no.153, p.313-324, ISSN 0012-7353, disponible en [http://www.scielo.org.co/scielo.php?script=sci\\_arttext&pid=S0012-73532007000300031&lng=pt&nrm=iso](http://www.scielo.org.co/scielo.php?script=sci_arttext&pid=S0012-73532007000300031&lng=pt&nrm=iso), accedido en 2008-4-1.

- [89] «Corel Draw», Wikipedia, disponible en [http://en.wikipedia.org/wiki/Corel\\_Draw](http://en.wikipedia.org/wiki/Corel_Draw),  
accedido en 2008-03-13
- [90] «Dreamweaver», Wikipedia, disponible en <http://en.wikipedia.org/wiki/Dreamweaver>,  
accedido en 2008-03-15
- [91] «Sun Opens Java», Sun Microsystems, 2006-11-13, disponible en  
<http://www.sun.com/2006-1113/feature/story.jsp>
- [92] «Sun begins releasing Java under the GPL», Free Software Foundation, 2006-11-15, disponible  
en <http://www.fsf.org/news/fsf-welcomes-gpl-java.html>
- [93] Stallman, Richard M.; «Free But Shackled - The Java Trap», 2004-4-12, disponible en  
<http://www.gnu.org/philosophy/java-trap.html>
- [94] «Comparison of computer algebra systems», Wikipedia, disponible en  
[http://en.wikipedia.org/wiki/Comparison\\_of\\_computer\\_algebra\\_systems](http://en.wikipedia.org/wiki/Comparison_of_computer_algebra_systems), accedido en  
2008-03-30
- [95] Biondi, Philippe; Desclaux, Fabrice; «Silver Needle in the Skype», disponible en [http://www.  
blackhat.com/presentations/bh-europe-06/bh-eu-06-biondi/bh-eu-06-biondi-up.pdf](http://www.blackhat.com/presentations/bh-europe-06/bh-eu-06-biondi/bh-eu-06-biondi-up.pdf),  
accedido en 2008-06-14
- [96] Granneman, Scott; «Skype security and privacy concerns», 2005-09-22, disponible en  
<http://www.securityfocus.com/columnists/357>
- [97] «Skype Reads Your BIOS and Motherboard Serial Number», disponible en  
<http://www.pagetable.com/?p=27>, accedido en 2008-06-14
- [98] «Skype Extras plug-in manager», disponible en  
[http://share.skype.com/sites/security/2007/02/skype\\_extras\\_plugin\\_manager.html](http://share.skype.com/sites/security/2007/02/skype_extras_plugin_manager.html),  
accedido en 2008-06-14
- [99] «Skype snoop agent reads mobo serial numbers», disponible en  
[http://www.theregister.co.uk/2007/02/11/skype\\_bios\\_snoop/](http://www.theregister.co.uk/2007/02/11/skype_bios_snoop/), accedido en 2008-06-14



# Apéndice A

## Glosario de términos

A continuación se definen algunos conceptos importantes que son utilizados en este trabajo.

### A.1. Definiciones generales

**Aplicación:** software orientado al usuario final. Por ejemplo, un procesador de texto, una planilla electrónica o un navegador web.

**Código fuente:** conjunto de instrucciones legibles por humanos, que define el modo en que funciona un software. Sin acceso al código fuente de un programa es muy difícil modificarlo.

**Controlador:** software cuya finalidad es hacer que un dispositivo de hardware funcione en un sistema operativo.

**Estándares abiertos y cerrados:** se refieren a la forma en que los programas se comunican entre sí para intercambiar información. Los estándares abiertos son aquellos que tienen una especificación pública, y que pueden ser implementados por cualquier programador en cualquier programa. Los estándares cerrados, en cambio, son aquellos que son secretos; nadie sabe cómo interactuar o compartir datos con un programa que utiliza estándares cerrados, excepto los creadores de ese formato cerrado.

**Firmware:** software que se ejecuta en dispositivos de hardware, no en el procesador principal del computador. El firmware puede estar en memoria de sólo lectura (ROM) o en memoria programable (PROM). Ejemplos de dispositivos que utilizan firmware son: routers, reproductores de música portátiles, tarjetas de red inalámbrica y cámaras web.

**Freeware:** tipo de software que permite ser utilizado y distribuido sin pago. El freeware no es software libre ya que generalmente incluye restricciones al uso, como prohibir su uso para fines comerciales, prohíbe su modificación, no incluye el código fuente y puede restringir la distribución del software.

**GNU:** proyecto iniciado en 1984 con el objetivo de crear un sistema operativo completo similar a Unix completamente libre: el sistema operativo GNU. Las variantes del sistema operativo GNU que utilizan el núcleo llamado Linux son utilizadas ampliamente en la actualidad; aunque a menudo estos sistemas se les llama Linux, deben ser llamados GNU/Linux o GNU+Linux [84]. GNU es un acrónimo recursivo que significa GNU's Not Unix (GNU No es Unix).

**Linux:** núcleo del sistema operativo GNU/Linux. Un núcleo es un componente un sistema operativo, que provee una interfaz entre el hardware y el resto del sistema operativo.

**Open source o código abierto:** término utilizado para referirse al «software libre», pero omitiendo los aspectos éticos y de libertad.

**Shareware:** tipo de software similar al freeware, pero que además posee algunas funcionalidades bloqueadas, con el fin de convencer al usuario de comprar la versión completa. Generalmente muestra mensajes molestos indicando al usuario que su copia del software no está registrada. El shareware no es software libre, por razones similares a las del freeware (véase *Freeware*).

**Software:** programa informático. Plural: software, programas.

**Software comercial:** es aquel software que es utilizado con fines comerciales. Equivocadamente se utiliza este término como lo opuesto a software libre. El software libre puede ser utilizado con fines comerciales, por lo tanto también puede ser software comercial.

**Software de uso privado:** es aquel software que es creado y utilizado por una persona o una organización para fines personales o propios de la organización y que no es publicado. El software de uso privado es software libre para quien lo usa, ya que él puede ejercer sus cuatro libertades básicas.

**Software gratuito:** El software gratuito es aquel que se puede utilizar sin pagar. El Software gratuito no necesariamente es software libre, y el software libre no necesariamente es gratuito, ya que se puede pagar por obtener un software libre, y se puede obtener un software no libre de forma gratuita. Gratuidad y libertad son dos conceptos distintos.

**Software libre:** es aquel software cuya licencia respeta las cuatro libertades básicas de los usuarios de software, definidas en [4], es decir, es el software que puede ser usado, estudiado, modificado, copiado y distribuido sin restricciones.

**Software privativo o no libre:** es aquel software que no respeta simultáneamente las cuatro libertades de los usuarios del software, definidas en [4]. El software que no es libre también se conoce como «software privativo», pues priva a los usuarios de una o más de sus libertades básicas.

## A.2. Definiciones legales

El software libre, al igual que el software privativo, está sujeto a las leyes que regulan su uso, modificación y distribución.

**Copyleft:** concepto construido sobre la base del copyright, pero invirtiendo su propósito. El copyleft otorga el derecho a los usuarios de hacer copias y modificaciones a una obra o programa y le añade una cláusula, mediante una licencia, que prohíbe quitar este derecho en obras o programas derivados. Es decir, todos los derivados de una obra o programa con copyleft, deben ser copyleft, y pueden ser copiados y modificados.

**Copyright:** ley creada con el propósito de promover la innovación. El copyright otorga al autor de una obra el monopolio sobre su obra por un tiempo limitado. En el caso del software, el copyright prohíbe realizar copias de un software, tanto en forma binaria como en código fuente.

**Dominio Público:** obra que no tiene copyright o cuyo copyright expiró. Una obra de dominio público puede ser utilizada con cualquier fin. Sin embargo, una obra derivada de una obra de dominio público puede estar sujeta a copyright y, por lo tanto, tener restricciones de uso, copia, modificación y distribución.

**GNU General Public License, GNU GPL:** licencia de software libre con copyleft, creada por la Free Software Foundation para el proyecto GNU. Otorga al usuario los derechos de usar, modificar, copiar y distribuir el software. El texto completo de esta licencia se encuentra en [85].

**Licencia de software:** documento que define derechos y restricciones sobre el uso de un software. Una licencia tiene validez legal, y el que otorga dicha licencia puede demandar a su titular en caso de que no cumpla con los términos establecidos en ella.

**Marca registrada:** nombre de un producto o empresa que no puede ser utilizado por ningún otro producto o empresa del mismo rubro.

**Patente de software:** documento legal que otorga un monopolio sobre una idea que puede ser implementada en software.

### A.3. Definiciones técnicas

**CAD:** Diseño Asistido por Computador (Computer-Aided Design).

**CAE:** Ingeniería Asistida por Computador (Computer-Aided Engineering)

**CAM:** Fabricación Asistida por Computador (Computer-Aided Manufacturing).

**Gráficos de mapas de bits:** representan imágenes como una matriz de píxeles, como se utiliza generalmente para la representación de imágenes fotográficas.

**Gráficos vectorizados o vectoriales:** representan imágenes utilizando primitivas geométricas, tales como puntos, líneas, curvas y polígonos, las cuales están basadas en ecuaciones matemáticas.

**IDE:** Entorno Integrado de Desarrollo (Integrated Development Environment)

## Apéndice B

# Listado de programas libres y privativos

### B.1. Introducción

Existen sitios web con listados de software libre para diferentes sistemas operativos, en los cuales se pueden encontrar programas libres que pueden reemplazar a determinados programas privativos:

- La Free Software Foundation mantiene un directorio de software libre<sup>1</sup> que corre sobre sobre sistemas operativos libres, en el cual los programas se organizan según categorías.
- En la página del proyecto GNU hay un listado<sup>2</sup> de alternativas libres para aplicaciones privativas en Windows.
- El sitio Alternativas Libres<sup>3</sup> organiza los programas libres según la tarea que realizan y según los programas privativos a los cuales pueden reemplazar.
- En el sitio de Graziano Sorbaioli<sup>4</sup> hay un listado de software libre organizado según la tarea que realizan, y además se pueden descargar imágenes de CD con software libre para Windows y Mac OS.
- Los proyectos OpenApps<sup>5</sup> y TheOpenDisc<sup>6</sup> recopilan software libre para Windows, y ofrecen imágenes de CD para descargar.

Se trató de incluir solamente software libre que cumpla con el criterio de la Free Software Foundation, es decir, que respete las cuatro libertades de los usuarios de software. Sin embargo, para verificar si un programa es libre, se puede revisar la licencia de cada programa y buscarla en el listado de licencias libres que mantiene la Free Software Foundation, el cual está disponible en [80].

Se consideraron programas libres que funcionan en GNU/Linux, en Windows o en ambos. No se ha tomado en cuenta el sistema operativo Mac OS debido a su escasa presencia en la Universidad de Tarapacá. Sin embargo, en la mayoría de los casos el software libre es multiplataforma, es decir, se puede compilar y ejecutar en distintos sistemas operativos. También se debe tener en cuenta que

---

<sup>1</sup><http://directory.fsf.org/>

<sup>2</sup><http://www.gnu.org/software/for-windows.html>

<sup>3</sup><http://alts.homelinux.net/>

<sup>4</sup><http://www.sorbaioli.org/free-software>

<sup>5</sup><http://www.aclibre.org/openapps/>

<sup>6</sup><http://www.theopendisc.com/>

existe Cygwin<sup>7</sup>, un software libre que facilita el funcionamiento de programas para GNU/Linux en Windows. Por otra parte, Wine<sup>8</sup> es un software libre que permite ejecutar algunos programas para Windows (libres o no) en GNU/Linux.

Existen alternativas libres que están escritas en Java. La mayoría de ellas requiere de la máquina virtual Java de Sun para funcionar, la cual hasta el momento de realizar este trabajo no era completamente libre. Solo unas pocas aplicaciones Java pueden funcionar con alguna máquina virtual de Java libre. En el caso de que un programa libre esté escrito en Java, se indicará.

A continuación se describen los programas privativos que, según la encuesta realizada, son utilizados en la Universidad de Tarapacá. Para cada uno de ellos se proponen programas libres que pueden servir de alternativa. Luego se mencionan los programas libres que ya se utilizan en la Universidad.

## B.2. Programas privativos utilizados en la Universidad

Muchas de las siguientes descripciones de programas privativos fueron traducidas de los artículos correspondientes de la Wikipedia en inglés. Otras descripciones fueron obtenidas de los sitios web oficiales de los programas.

### 3D Studio Max

3D Studio Max es un programa para realizar gráficos y animaciones tridimensionales.

Página web: <http://www.autodesk.com/3dsmax>

Alternativas libres:

- Blender (<http://www.blender.org/>) es un modelador 3D que permite realizar animaciones y programas interactivos
- Art of Illusion (<http://www.artofillusion.org/>) es un modelador 3D sencillo que requiere Java
- Wings 3D (<http://www.wings3d.com/>) es un modelador 3D basado en subdivisiones

### Adobe Acrobat

Adobe Acrobat es un software que permite crear y editar archivos PDF.

Página web: <http://www.adobe.com/products/acrobat/>

Alternativas libres:

- PDFCreator (<http://www.pdfforge.org/products/pdfcreator>) permite crear archivos PDF desde distintas aplicaciones para Windows
- OpenOffice (<http://es.openoffice.org/>) puede exportar documentos en formato PDF de forma nativa
- Los entornos de escritorio GNOME y KDE permiten generar archivos PDF desde cualquier aplicación

---

<sup>7</sup><http://www.cygwin.com/>

<sup>8</sup><http://www.winehq.org/>

- KWord (<http://www.koffice.org/kword/>) permite editar archivos PDF
- PDFedit (<http://sourceforge.net/projects/pdfedit>) permite editar archivos PDF

## Adobe Flash Player

Adobe Flash Player es un reproductor de Flash. Flash es un formato de animación para la web, basado en gráficos vectoriales, que ha sido extendido para incluir sonido y video, y programas escritos en el lenguaje ActionScript.

Página web: <http://www.adobe.com/es/products/flashplayer/>

Alternativas libres:

- gnash (<http://www.gnu.org/software/gnash/>) es un reproductor de Flash
- swfdec (<http://swfdec.freedesktop.org/wiki/>) es un reproductor de Flash

## Adobe Reader

Adobe Reader es un visualizador de archivos PDF, no puede editar ni crear archivos PDF. Antes era llamado Adobe Acrobat Reader.

Página web: <http://www.adobe.com/products/reader/>

Alternativas libres:

- Los entornos GNOME y KDE incluyen visualizadores de archivos PDF
- SumatraPDF (<http://blog.kowalczyk.info/software/sumtrapdf/sumatra.css>) es un visualizador de archivos PDF para Windows

## Aleph Integrated Library System

Aleph es un sistema integrado para gestionar los recursos y operaciones de una biblioteca.

Página web: <http://www.exlibris.co.il/>

Alternativas libres: Algunos sistemas de gestión de biblioteca libres son:

- OpenBiblio (<http://obiblio.sourceforge.net/>)
- PhpMyLibrary (<http://directory.fsf.org/project/PhpMyLibrary/>)
- Koha (<http://directory.fsf.org/project/koha/>)
- Evergreen (<http://www.open-ils.org/>)

## ALGOR

ALGOR es un software de análisis de elementos finitos de propósito general. Se distribuye en diferentes paquetes para aplicaciones específicas, como simulación de eventos mecánicos y dinámica de fluidos computacional. El análisis de elementos finitos es una técnica de simulación computacional utilizada en el análisis de ingeniería. Utiliza una técnica numérica llamada método de elementos

finitos, la cual sirve para encontrar soluciones aproximadas de ecuaciones diferenciales parciales y de ecuaciones integrales.

Página web: <http://www.algor.com/>

Alternativas libres: En el directorio de software libre de la FSF se mencionan varios programas libres para el análisis de elementos finitos, tales como:

- deal.II (<http://directory.fsf.org/project/tochnog/>)
- Elmer (<http://directory.fsf.org/project/Elmer/>)
- Figura (<http://directory.fsf.org/project/figura/>)
- GetDP (<http://directory.fsf.org/project/GetDP/>)
- Gmsh (<http://directory.fsf.org/project/gmsh/>)
- keyFE2 (<http://directory.fsf.org/project/keyFE2/>)
- OOFEM (<http://directory.fsf.org/project/OOFEM/>)
- Tochnog (<http://directory.fsf.org/project/tochnog/>)

En [88] se realiza un análisis muy completo de herramientas GPL para el modelado de elementos finitos, tales como Calculix, CodeAster, Tochnog, FreeFem, FElT, Z88, SLFFEA, OpenFoam, OpenFlower, SLFCDF, Impact, NETGEN, NGSolve. También se hacen pruebas y se comparan resultados con un software privativo, Ansys.

## Antivirus

Programas antivirus para Windows.

Alternativas libres:

- ClamWin (<http://www.clamwin.com/>) es un antivirus libre para Windows. No puede, por sí mismo, revisar los archivos en tiempo real.
- Winpooch (<http://winpooch.free.fr/>) es un programa que brinda protección contra spyware y troyanos. También se puede asociar con ClamWin para brindar protección contra virus en tiempo real.
- Nixory (<http://nixory.sourceforge.net/>) es un programa anti spyware para Mozilla Firefox.
- ClamAV (<http://www.clamav.net/>) es un antivirus para GNU/Linux. Se utiliza principalmente en servidores de correo y servidores de archivos, para evitar que los computadores que utilicen Windows se infecten.

## ArcView

ArcView es un sistema de información geográfico (GIS, Geographical Information System). Un GIS es un sistema que permite capturar, almacenar, analizar y gestionar datos que están referenciados espacialmente con la Tierra.

Página web: <http://www.esri.com/software/arcview/>

Alternativas libres:

- Quantum GIS (<http://www.qgis.org/>)
- GRASS GIS (<http://grass.itc.it/>)

## AutoCAD

AutoCAD es un software CAD para el diseño y dibujo técnico en dos y tres dimensiones.

Página web: <http://www.autodesk.com/autocad>

Algunas alternativas libres para AutoCAD son:

- QCad Community Edition (<http://www.ribbonsoft.com/qcad.html>) es un programa CAD en 2D.
- CADEMIA Community Edition (<http://www.cademia.org/>) es un programa CAD en 2D escrito en Java. Requiere JRE 5.0 para funcionar.
- FreeCAD (<http://free-cad.sourceforge.net/>) es un programa de modelamiento CAD en 3D enfocado principalmente en la ingeniería mecánica.
- SagCAD (<http://sagcad.sourceforge.jp/>) es un programa CAD en 2D.
- Varkon (<http://varkon.sourceforge.net/>) es un programa CAD en 2D y 3D. Soporta modelado paramétrico e incluye un lenguaje de programación.
- Figura (<http://directory.fsf.org/project/figura/>) es un sistema CAD distribuido para ingeniería mecánica.
- BRL-CAD (<http://my.brlcad.org/>) es un sistema de modelado sólido.

## Banner

Banner es un software para DOS que permite crear e imprimir carteles de múltiples hojas.

Alternativas libres: OpenOffice Draw y GIMP permiten imprimir dibujos y fotografías en múltiples páginas.

## Borland C++

Borland C++ es un compilador y un entorno integrado de programación (IDE) para el lenguaje C++.

Alternativas libres:

- Dev-C++ (<http://www.bloodshed.net/devcpp.html>) es un entorno de programación para Windows
- Anjuta (<http://anjuta.sourceforge.net/>) y KDevelop (<http://www.kdevelop.org/>) son entornos integrados de desarrollo para GNU/Linux
- GCC (<http://gcc.gnu.org/>), la Colección de Compiladores de GNU, incluye un compilador de C++
- MinGW (<http://www.mingw.org/>) provee varias utilidades de programación GNU para Windows. Incluye compiladores, un depurador y otras herramientas.



## Catia

Catia es un programa de CAD/CAM/CAE (diseño, manufactura e ingeniería asistidos por computador). Proporciona apoyo desde la concepción del diseño (CAD) hasta la producción (CAM) y el análisis (CAE) de productos.

Página web: <http://www.ibm.com/software/applications/plm/catiav5/>

Alternativas libres: Ver alternativas libres para AutoCAD.

## Cisco Packet Tracer

Packet Tracer es una herramienta para la enseñanza de redes de computadores, a través de métodos visuales e interactivos. Es posible realizar simulaciones de redes y observar cómo funcionan internamente. El objetivo es entrenar a los estudiantes antes de trabajar con equipos reales.

Página web: [http://www.cisco.com/web/learning/netacad/landing/Packet\\_Tracer.html](http://www.cisco.com/web/learning/netacad/landing/Packet_Tracer.html)

Alternativas libres:

- ns-2, The Network Simulator (<http://www.isi.edu/nsnam/ns/>)
- Java Network Simulator (<http://jns.sourceforge.net/>)
- adevs (<http://www.ornl.gov/~1qn/adevs/index.html>)

## CleanUp!

CleanUp! es un programa gratuito que elimina archivos temporales de Windows y de navegadores web. CleanUp! además permite eliminar de manera segura los archivos, para que sea imposible recuperarlos utilizando software especializado.

Alternativas libres:

- No se han encontrado programas libres que eliminen archivos temporales de Windows
- Nixory (<http://nixory.sourceforge.net/>) es un antispyware para Firefox
- Wipe (<http://wipe.sourceforge.net/>) es una utilidad para eliminar archivos de forma segura

## CorelDRAW

CorelDRAW es un editor de gráficos vectorizados. CorelDRAW también es el nombre de una suite de aplicaciones gráficas de Corel que incluye varios componentes:

1. CorelDRAW: edición de gráficos vectorizados
2. Corel PHOTO-PAINT: edición de gráficos de mapas de bits
3. Corel PowerTRACE: convierte imágenes de mapas de bits a gráficos vectorizados
4. Pixmantec RawShooter Essentials: soporta el formato de archivo RAW, el cual es utilizado para almacenar la información no procesada del sensor de las cámaras digitales.

Página web: <http://coreldraw.com/>

Algunas alternativas libres para cada uno de los componentes del paquete CorelDraw mencionados son:

1. Alternativas libres para CorelDRAW:
  - Inkscape (<http://www.inkscape.org/>)
  - OpenOffice Draw (<http://www.openoffice.org/product/draw.html>)
  - Xara Xtreme (<http://www.xaraxtreme.org/>)
2. Alternativas libres para Corel PHOTO-PAINT:
  - GIMP (<http://www.gimp.org/>)
  - Krita (<http://www.koffice.org/krita/>)
  - Paint.Net (<http://www.getpaint.net/>)
3. Alternativas libres para Corel PowerTRACE:
  - potrace (<http://potrace.sourceforge.net/>)
  - Autotrace (<http://autotrace.sourceforge.net/>)
  - Inkscape (<http://www.inkscape.org/>)
4. Alternativa libres para Pixmantec RawShooter Essentials:
  - UFRaw (<http://ufraw.sourceforge.net/>)

## Dreamweaver

Dreamweaver es un editor de páginas web que permite diseñarlas sin tener conocimiento de código HTML. Esta forma de editar páginas web, y documentos en general, se denomina modo WYSIWYG (What You See Is What You Get, lo que vé es lo que obtiene). También ofrece ayuda a los diseñadores que escriben directamente el código HTML de sus páginas.

Página web: <http://www.adobe.com/products/dreamweaver/>

Alternativas libres:

- Kompozer (<http://www.kompozer.net/>) es un editor de páginas WYSIWYG
- Nvu (<http://www.nvu.com/>) es un editor de páginas WYSIWYG
- Quanta Plus (<http://quanta.kdwebdev.org/>) es un editor de páginas WYSIWYG
- Amaya (<http://www.w3.org/Amaya/>) es un editor de páginas WYSIWYG
- Bluefish (<http://bluefish.openoffice.nl/>) es un editor de HTML, no tiene modo WYSIWYG

## Electronics Workbench

Electronics Workbench es un software de diseño (captura de esquemas) y simulación de circuitos electrónicos basado en SPICE. También ofrece herramientas de diseño de placas de circuitos integrados (PCB). Posee características orientadas a la enseñanza, como cuestionarios integrados.

Página web: <http://www.electronicworkbench.com/>

Alternativas libres:

- Oregano (<http://arrakis.gforge.lug.fi.uba.ar/>) es un software de diseño (captura de esquemas) y simulación de circuitos electrónicos. Puede utilizar gnuicap o ngspice para realizar las simulaciones.
- gEDA (<http://www.geda.seul.org/>) es un conjunto de herramientas para la automatización de diseño electrónico:
  - gschem - diseño de circuitos (captura de esquemas)
  - gnetlist - listado de conexiones (*netlisting*)
  - ngspice - simulador de circuitos basado en SPICE
  - gnuicap - simulador de circuitos
  - gspiceui - interfaz gráfica para ngspice y gnuicap
  - pcb - diseño de placas de circuitos impresos (PCB)
  - gerbv - visor de archivos Gerber, los cuales se utilizan en la fabricación de placas de circuitos impresos (PCB)
  - Icarus Verilog - simulador de circuitos analógicos y digitales basado en el lenguaje de descripción de hardware (HDL) Verilog
  - GTKWave - visor de ondas digitales
  - gwave - visor de ondas analógicas
  - wcalc - análisis de líneas de transmisión y estructuras electromagnéticas
- Kicad ([http://www.lis.inpg.fr/realise\\_au\\_lis/kicad/](http://www.lis.inpg.fr/realise_au_lis/kicad/)) es un conjunto de herramientas para el diseño de esquemas de circuitos y placas de circuitos impresos (PCB)
- Electric (<http://www.gnu.org/software/electric/electric.html>) es un sistema CAD que puede manejar varias formas de diseño de circuitos, incluyendo diseño de esquemas de circuitos integrados para aplicaciones específicas (ASICs), dibujo esquemático, lenguaje de descripción de hardware y diseño híbrido electro-mecánico.

## Encarta

Encarta es una enciclopedia general con elementos multimedia e interactivos.

Página web: <http://www.microsoft.com/products/encarta/default.msp>

Alternativa libre: Wikipedia (<http://es.wikipedia.org/>)

## Filter Design

Filter Design Toolbox es un conjunto de herramientas para MATLAB que permite diseñar, simular y analizar filtros digitales. Es una extensión del Signal Processing Toolbox, el cual es requerido y se debe conseguir de forma separada.

Página web: <http://www.mathworks.com/products/filterdesign/>

Alternativa libre: Fiview (<http://uazu.net/fiview/>)

## Fireworks

Fireworks es un editor de gráficos de mapas de bits y vectoriales que está orientado a diseñadores de páginas web. Permite optimizar las imágenes para la web y crear prototipos interactivos de páginas web, entre otras funciones.

Página web: <http://www.adobe.com/products/fireworks/>

Alternativas libres: no se han encontrado alternativas exactas para Fireworks, aunque es posible utilizar en su lugar editores de imágenes de propósito general, como GIMP o Inkscape.

## FlashGet

FlashGet es un gestor de descargas. Permite descargar una lista de archivos, dividir un archivo en varias partes y descargar un archivo desde distintas fuentes.

Página web: [http://www.flashget.com/index\\_en.htm](http://www.flashget.com/index_en.htm)

Alternativas libres:

- DownThemAll! (<http://www.downdthemall.net/>) es un gestor de descargas para Firefox
- FlashGot (<http://flashgot.net/>) es un gestor de descargas para Firefox
- Free Download Manager (<http://www.freownloadmanager.org/>) es un gestor de descargas para Windows
- Gwget (<http://www.gnome.org/projects/gwget/>) es un gestor de descargas para GNU/Linux

## FreeHand

FreeHand es un programa de publicación de escritorio (*desktop publishing*) basado en gráficos vectorizados. FreeHand ya no se encuentra en desarrollo, pues la empresa que lo desarrolló fue comprada por Adobe, que posee su propio programa de diseño gráfico, Adobe Illustrator.

Página web: <http://www.adobe.com/products/freehand/>

Alternativas libres:

- Scribus (<http://www.scribus.net/>) es un software de publicación de escritorio
- Passepartout (<http://www.stacken.kth.se/project/pptout/>) es un software de publicación de escritorio
- Inkscape (<http://www.inkscape.org/>) es un editor de gráficos vectoriales

- OpenOffice Draw (<http://www.openoffice.org/product/draw.html>) es un editor de gráficos vectoriales
- Xara Xtreme (<http://www.xaraxtreme.org/>) es un editor de gráficos vectoriales

## FrontPage

Frontpage es un editor de páginas web que formaba parte de Microsoft Office y que permitía editar páginas web sin manipular directamente el código HTML. Actualmente ya no está en desarrollo.

Página web: <http://www.microsoft.com/frontpage/>

Alternativas libres: Ver alternativas libres para Dreamweaver.

## Internet Explorer

Internet Explorer es el navegador web que viene integrado y preinstalado con Windows desde que apareció Windows 98. Debido a que es parte de Windows, no es posible desinstalarlo. Esto hace que Internet Explorer sea el navegador web más utilizado.

Internet Explorer es criticado, además por ser software privativo, por no respetar los estándares de la web. Lamentablemente, muchas páginas web son desarrolladas para ser visualizadas solamente en Internet Explorer, por lo cual no funcionan bien con otros navegadores que sí respetan los estándares.

Página web: <http://www.microsoft.com/windows/products/winfamily/ie/default.msp>

Alternativas libres:

- Firefox (<http://www.mozilla.com/firefox/>) es un navegador web
- IceCat (<http://www.gnu.org/software/gnuzilla/>) es un navegador web que remueve algunos componentes privativos de Firefox
- Epiphany (<http://www.gnome.org/projects/epiphany/>) es el navegador web del entorno GNOME
- Konqueror (<http://www.konqueror.org/>) es el navegador web del entorno KDE

## IrfanView

IrfanView es un visor de imágenes para Windows que permite ver, editar y convertir archivos de imágenes y reproducir archivos de audio/video. Posee pocas funcionalidades para editar imágenes, pero en cambio es muy rápido y soporta una gran cantidad de formatos de archivo.

Página web: <http://www.irfanview.com/>

Alternativas libres:

- Los entornos GNOME y KDE incluyen programas para visualizar archivos de imágenes, audio y video
- cornice (<http://wxglade.sourceforge.net/extra/cornice.html>) es un visor de imágenes que también funciona en Windows
- imgv (<http://imgv.sourceforge.net/>) es un visor de imágenes que también funciona en Windows

- GIMP (<http://www.gimp.org/>) es un editor de imágenes que puede convertir archivos de imágenes en diferentes formatos
- ImageMagick (<http://imgv.sourceforge.net/>) es un software para aplicar filtros y efectos a imágenes, que posee varias interfaces gráficas

## iTunes

iTunes es un programa para reproducir y organizar archivos de música y video. El programa también es una interfaz para gestionar los archivos multimedia de los reproductores iPod y del iPhone. Además, iTunes puede conectarse con la tienda iTunes a través de Internet para comprar y descargar música, videos musicales, programas de televisión, juegos para el iPod, audiolibros, *podcasts*, películas y *ringtones*. La tienda iTunes se puede acceder únicamente utilizando el software iTunes.

Página web: <http://www.apple.com/itunes/>

Alternativas libres:

- Algunos programas para organizar y reproducir música, audiolibros, *podcasts* y radios de Internet son:
  - Amarok (<http://amarok.kde.org/>)
  - Rhythmbox (<http://www.gnome.org/projects/rhythmbox/>)
  - aTunes (<http://www.atunes.org/>)
  - musikCube (<http://www.musikcube.com/>)
- Se puede utilizar clientes de redes *peer to peer* para descargar música, videos, programas de televisión, juegos, audiolibros, películas y ringtones. Compartir archivos es éticamente correcto, y en principio también es legal si los archivos son utilizados con fines personales y no comerciales. Algunos de estos programas son:
  - eMule (<http://www.emule-project.net/>) y aMule (<http://www.amule.org/>)
  - Azureus (<http://azureus.sourceforge.net/>)
  - Transmission <http://www.transmissionbt.com/>
  - Ares Galaxy (<http://aresgalaxy.sourceforge.net/>)
  - Shareaza (<http://shareaza.sourceforge.net/>)
- Desde los sitios web de Jamendo (<http://www.jamendo.com/>) y Magnatune (<http://www.magnatune.com/>) se puede descargar o escuchar música de forma gratuita. Algunos programas, como Rhythmbox y Amarok permiten seleccionar y escuchar música de esos sitios. Además, Amarok facilita la compra de música de la tienda Magnatune.

## Java Development Kit y Java Runtime Environment

La tecnología Java se compone básicamente por: un lenguaje de programación, un compilador de código intermedio (bytecode), una máquina virtual que interpreta el código intermedio (bytecode), una biblioteca de clases y un *plugin* para ejecutar programas Java (applets) en navegadores web.

El paquete de desarrollo de Java de Sun (Java Development Kit, JDK) es el entorno de desarrollo más utilizado, pero no es el único. Existen varios compiladores, máquinas virtuales y bibliotecas de clases de Java, tanto libres como privativos.

Stallman explica en [93] que no es conveniente escribir software libre utilizando bibliotecas privativas o una plataforma de programación no libre, pues sería necesario que los usuarios instalen software privativo para ejecutarlo. Afortunadamente, esto ya no se aplica a la plataforma Java de Sun, pues ha sido liberada bajo la licencia GPL [92]. Sin embargo, la versión liberada, llamada OpenJDK o JDK 7, aún no está lista para su uso en producción y todavía incluye componentes no libres.

Sun anunció en una charla en la conferencia FOSDEM 2007<sup>9</sup> que OpenJDK incluirá componentes privativos de terceros para mantener la compatibilidad con las versiones anteriores del JDK, hasta que estén listos los reemplazos libres que están desarrollando junto con la comunidad. Por lo tanto, si bien la liberación de la plataforma Java de Sun es algo extremadamente positivo, aún no se puede considerar software libre completamente, debido a dichos componentes privativos.

Página Web de Java de Sun: <http://www.sun.com/software/opensource/java/>

Alternativas libres:

- GNU Compiler for Java (<http://gcc.gnu.org/java/>) es un compilador y un intérprete para el lenguaje de programación Java. El compilador puede generar bytecode o código de máquina nativo.
- GNU Classpath (<http://www.gnu.org/software/classpath/>) es un conjunto de bibliotecas de clases que pueden ser usadas por máquinas virtuales y compiladores del lenguaje de programación Java. Algunas máquinas virtuales Java que utilizan las bibliotecas de GNU Classpath son:
  - Kaffe (<http://www.kaffe.org/>)
  - SableVM (<http://www.sablevm.org/>)
  - CACAO (<http://www.cacaojvm.org/>)
  - JamVM (<http://jamvm.sourceforge.net/>)
- IcedTea ([http://icedtea.classpath.org/wiki/Main\\_Page](http://icedtea.classpath.org/wiki/Main_Page)) tiene por objetivo crear reemplazos libres para los componentes privativos de OpenJDK, la plataforma Java que liberó Sun, para que ésta sea completamente software libre.
- gcjwebplugin (<http://www.nongnu.org/gcjwebplugin/>) es un *plugin* para ejecutar programas Java (applets) en navegadores web.

## Jcreator

JCreator es un IDE (entorno integrado de programación) para Java que funciona en Windows. Posee una interfaz similar a Microsoft Visual Studio. Está programado en C++, por lo cual no requiere de una máquina virtual Java para funcionar, como es el caso de muchos otros IDE para Java. Esto lo hace comparativamente más rápido.

Página web: <http://www.jcreator.com/>

Alternativas libres:

---

<sup>9</sup>Liberating Java, <http://video.fosdem.org/2007/FOSDEM2007-Liberating-Java.ogg>

- Eclipse (<http://www.eclipse.org/>) es un entorno de programación para Java y otros lenguajes
- NetBeans (<http://www.netbeans.org/>) es un entornode de programación para Java y otros lenguajes
- Anjuta (<http://anjuta.sourceforge.net/>) en un entorno de programación para GNOME
- KDevelop (<http://www.kdevelop.org/>) es un entono de programación para KDE

## LabVIEW

LabVIEW es una plataforma y entorno de desarrollo basado en un lenguaje de programación visual. LabVIEW es utilizado para adquisición de datos, control de instrumentos y automatización industrial. Está disponible para varios sistemas operativos, incluyendo Windows, GNU/Linux, Mac OS y Unix.

El lenguaje de programación utilizado en LabVIEW, llamado G, es un lenguaje de flujo de datos. La ejecución es determinada por la estructura de un diagrama de bloques gráfico, en el cual el programador conecta diferentes nodos de función dibujando alambres. Estos alambres propagan las variables y cualquier nodo puede ejecutarse tan pronto como tenga disponibles todas sus entradas de datos. Como esto puede ocurrir con varios nodos a la vez, G es inherentemente capaz de ejecución en paralelo. El hardware multi-procesador y multi-hilo es aprovechado automáticamente.

Página web: <http://www.ni.com/labview/>

Alternativas libres: no se encontraron.

## LINDO/LINGO

LINGO es una herramienta para construir y resolver modelos de optimización lineal, no lineal e integral. LINGO provee un lenguaje para expresar los modelos de optimización, un entorno para contruir y editar problemas, y un conjunto de solucionadores. LINDO API es un conjunto de solucionadores que pueden ser utilizados desde otros programas.

Página web: <http://www.lindo.com/>

Posibles alternativas libres:

- GLPK (<http://www.gnu.org/software/glpk/glpk.html>), o Kit de programación lineal de GNU, resuelve problemas de programación lineal, programación lineal entera mixta y otros problemas relacionados
- lp\_solve (<http://lpsolve.sourceforge.net/>) resuelve de problemas de programación lineal y enteros

## Lotus 1-2-3

Lotus 1-2-3 fue una planilla electrónica para MS-DOS, aunque más tarde se publicaron versiones para Windows. Además de ser una planilla electrónica, podía hacer gráficos y tenía algunas funcionalidades rudimentarias de base de datos. Luego formó parte del paquete Lotus SmartSuite de IBM. Ya no está en desarrollo.

Página web: <http://www.ibm.com/software/lotus/products/123/>

Alternativas libres:



- OpenOffice Calc (<http://www.openoffice.org/product/calc.html>)
- Gnumeric (<http://www.gnome.org/projects/gnumeric/>)
- KSpread (<http://www.koffice.org/kspread/>)

## Lotus Word Pro

Lotus Word Pro es un procesador de palabras que formó parte del paquete de ofimática Lotus SmartSuite. Ya no está en desarrollo.

Página web: <http://www.ibm.com/software/lotus/products/smartsuite/wordpro.html>

Alternativas libres:

- OpenOffice Writer (<http://www.openoffice.org/product/writer.html>)
- AbiWord (<http://www.abisource.com/>)
- KWord (<http://www.koffice.org/kword/>)

## Maple

Maple es un paquete de software matemático de propósito general. Los usuarios pueden escribir fórmulas matemáticas en la notación matemática tradicional. Maple puede realizar cálculo numérico y cálculo simbólico, además de graficar. Maple incorpora un lenguaje de programación de muy alto nivel. También ofrece interfaces para otros lenguajes, como C, Fortran, Java, Matlab y Visual Basic, además de una interfaz para Excel.

Página web: <http://www.maplesoft.com/products/Maple11/professionals/index.aspx>

Alternativas libres:

- Cálculo simbólico y numérico de propósito general:
  - Maxima (<http://maxima.sourceforge.net/es/>)
  - YaCaS (<http://yacass.sourceforge.net/>)
  - SAGE (<http://modular.math.washington.edu/sage/>)
- Cálculo numérico, matrices y programación:
  - GNU Octave (<http://www.gnu.org/software/octave/>)
- Trazado de gráficos:
  - KmPlot (<http://edu.kde.org/kmplot/>)
  - KAlgebra (<http://kalgebra.berlios.de/>)
  - GraphCalc (<http://www.graphcalc.com/>)
  - Gnuplot (<http://www.gnuplot.info/>)
- Rutinas de cálculo numérico que se pueden utilizar en programas C:
  - GNU Scientific Library (<http://www.gnu.org/software/gsl/>)

- Análisis estadístico:
  - R (<http://www.r-project.org/>)
  - PSPP (<http://www.gnu.org/software/pspp/>)

En [94] hay una lista de sistemas de cálculo simbólico libres y privativos, en donde se indica el área de aplicación de cada uno.

## Mathcad

Mathcad permite realizar y documentar cálculos científicos e ingenieriles. Incluye algunas capacidades de un sistema de álgebra computacional (o cálculo simbólico), pero se mantiene orientado en aplicaciones numéricas de ingeniería. Su interfaz se centra en una hoja de trabajo, en la cual las ecuaciones y expresiones se muestran gráficamente, en vez de en texto plano.

Aunque Mathcad está orientado principalmente a los usuarios no-programadores, también es utilizado en proyectos más complejos para visualizar los resultados del modelado matemático que utilizan computación distribuida y lenguajes de programación tradicionales.

Página web: <http://www.ptc.com/>

Alternativas libres: Ver alternativas libres para Maple.

## Mathematica

Mathematica es un software matemático y científico que permite realizar cálculo simbólico y numérico, aritmética de precisión arbitraria, procesamiento de datos y trazado de gráficos. Posee un lenguaje de programación que soporta los paradigmas de programación funcional y procedural. Su interfaz, llamada *notebook*, permite visualizar, crear, manipular y combinar programas, gráficos, fórmulas matemáticas, texto y ventanas de diálogo. También provee un depurador.

Página web: <http://www.wolfram.com/products/mathematica/index.html>

Alternativas libres: Ver alternativas libres para Maple.

## MathType

MathType es una herramienta que permite escribir fórmulas matemáticas en distintos tipos de documentos, como procesadores de texto, páginas web, presentaciones, T<sub>E</sub>X L<sup>A</sup>T<sub>E</sub>X y MathML. MathType se puede instalar como reemplazo del editor de ecuaciones de Microsoft Office.

Página web: <http://www.dessci.com/en/products/mathtype/>

Alternativas libres: El paquete de ofimática OpenOffice incluye un editor de ecuaciones muy completo, aunque para utilizarlo se requiere el manejo de un lenguaje de texto, por lo que no resulta tan fácil para alguien acostumbrado al editor de ecuaciones de Microsoft Office.

## MATLAB

MATLAB es un entorno para cálculo numérico y un lenguaje de programación. Puede manipular matrices, graficar funciones y datos, implementar algoritmos, crear interfaces de usuario, y enlazarse con programas en otros lenguajes. Aunque solamente realiza cálculo numérico, existe un *toolbox* opcional para acceder al sistema de cálculo simbólico de Maple.

MATLAB posee un IDE que incluye un intérprete de comandos, un editor de texto con resaltado de sintaxis, ayuda en línea y datos de depuración.

Alternativas libres:

- GNU Octave (<http://www.gnu.org/software/octave/>) utiliza un lenguaje que en principio es compatible con MATLAB. Trabaja junto con Gnuplot para realizar gráficos. No posee un IDE propio, solamente tiene una interfaz de línea de órdenes, que es equivalente a la ventana de órdenes de MATLAB. Sin embargo, existen otros proyectos que ofrecen un IDE para GNU Octave, tales como:
  - kOctave (<http://athlone.ath.cx/~matti/kde/k octave/>)
  - Octave Workshop (<http://www.ics.es.yamanashi.ac.jp/mirror/octave-workshop/>)
- FreeMat (<http://freemat.sourceforge.net/>) es un entorno de desarrollo interpretado y orientado a matrices, similar a MATLAB, que posee interfaz gráfica.
- SAGE (<http://modular.math.washington.edu/sage/>) es un sistema matemático que combina varias utilidades existentes a través de una interfaz gráfica

## Messenger

Microsoft Network Messenger es un cliente de mensajería instantánea para la red MSN. En las versiones más recientes pasa a llamarse Windows Live Messenger. Permite comunicarse por texto, voz y video.

Página web: <http://get.live.com/general/home>

Alternativas libres:

- Pidgin, <http://www.pidgin.im/>
- aMSN, <http://www.amsn-project.org/>
- Miranda IM, <http://www.miranda-im.org/>
- Kopete, <http://kopete.kde.org/>
- emesene, <http://www.emesene.org/>
- Adium, <http://www.adiumx.com/>

Algunas de estas alternativas, como aMSN y Kopete, soportan la transmisión y recepción de video. Para realizar videoconferencias y llamadas de voz, véase WengoPhone (<http://www.openwengo.org/>) y Ekiga (<http://ekiga.org/>).

## Microsoft Access

Microsoft Access es un sistema gestor de bases de datos relacional con interfaz gráfica. Además, se puede utilizar para crear aplicaciones.

Página web: <http://office.microsoft.com/en-us/access/default.aspx>

Alternativas libres:

- OpenOffice Base (<http://www.openoffice.org/product/base.html>) es un sistema gestor de bases de datos de escritorio, similar a Access
- Kexi (<http://www.kexi-project.org/>) es una utilidad para la administración de bases de datos Access
- mdbtools (<http://mdbtools.sourceforge.net/>) permite acceder a bases de datos Access y exportarlas a otros formatos de bases de datos, como MySQL y PostgreSQL

### Microsoft Fortran PowerStation

Power Station es entorno integrado de desarrollo para Fortran. Ya no está en desarrollo.

Alternativas libres:

- GFortran (<http://gcc.gnu.org/wiki/GFortran>) es un compilador de Fortran que está incluido en GCC, la colección de compiladores de GNU.

### Microsoft Office

Microsoft Office es un paquete de software de ofimática, que incluye un procesador de textos (Word), una planilla electrónica (Excel), una herramienta para realizar presentaciones (PowerPoint), y una aplicación de correo electrónico (Outlook)

Página Web: <http://www.microsoft.com/office/>

Alternativas libres:

- OpenOffice (<http://es.openoffice.org/>) es un paquete de software de ofimática similar a Microsoft Office, que incluye un procesador de textos (Writer), una planilla electrónica (Calc) y un programa para realizar presentaciones (Impress).
- KOffice (<http://www.koffice.org/>) es un paquete de software de ofimática. Incluye un procesador de textos (Kword), una planilla electrónica (KSpread) y un programa para realizar presentaciones (KPresenter).
- AbiWord (<http://www.abisource.com/>) es un procesador de textos que posee menos requerimientos de hardware que OpenOffice Writer.
- Gnumeric (<http://www.gnome.org/projects/gnumeric/>) es una planilla electrónica que posee menos requerimientos de hardware que OpenOffice Calc
- Thunderbird (<http://www.mozilla-world.org/es/products/thunderbird/>) es un cliente de correo electrónico.
- Evolution (<http://www.gnome.org/projects/evolution/>) es un cliente de correo electrónico que incluye calendario y lista de tareas.

### Microsoft Paint

Microsoft Paint es un editor de gráficos de mapas de bits simple que se incluye con Microsoft Windows.

Alternativas libres: Gpaint (<http://www.gnu.org/software/gpaint/>) y KoulourPaint (<http://kolourpaint.sourceforge.net/>) son dos editores de gráficos simples, mientras que GIMP (<http://www.gimp.org/>), Krita (<http://www.koffice.org/krita/>) y Paint.Net (<http://www.getpaint.net/>) ofrecen una mayor cantidad de herramientas.

## Microsoft Project

Microsoft Project es un programa de gestión de proyectos. Permite crear planificaciones, asignar recursos a tareas, hacer seguimiento de los avances, gestionar presupuestos y analizar cargas de trabajo.

Página web: <http://www.microsoft.com/project>

Alternativas libres:

- Planner (<http://live.gnome.org/Planner>, <http://winplanner.sourceforge.net/>)
- OpenProj (<http://openproj.org/>)
- dotProject (<http://www.dotproject.net/>)
- TaskJuggler (<http://www.taskjuggler.org/>)
- KPlato (<http://www.koffice.org/kplato/>)

## Microsoft Visio

Microsoft Visio es un programa para realizar diagramas. Incluye un conjunto de símbolos de diferentes tipos, los cuales se pueden insertar y conectar entre sí para formar diagramas.

Página web: <http://office.microsoft.com/en-us/visio/>

Alternativas libres:

- OpenOffice Draw (<http://www.openoffice.org/product/draw.html>)
- Dia (<http://live.gnome.org/Dia>)
- Kivio (<http://www.koffice.org/kivio/>)

## MuPAD

MuPAD es un sistema de álgebra computacional (CAS), es decir, permite manipular fórmulas de manera simbólica. Posee módulos para álgebra lineal, ecuaciones diferenciales, teoría de números y estadísticas. Además, incluye un lenguaje de programación y puede hacer gráficos interactivos y animaciones. El núcleo de MuPAD se incluye en Scientific Notebook, Scientific Workplace y en versiones recientes de MathCAD.

Página web: <http://www.mupad.de/>

Alternativas libres: Existen sistemas de álgebra computacional libres, tales como:

- Maxima (<http://maxima.sourceforge.net/es/>)
- YaCaS (<http://yacass.sourceforge.net/>)

## Nero Burning ROM

Nero Burning ROM es un programa para grabar CD y DVD.

Página web: <http://www.nero.com/>

Alternativas libres:

- Brasero (<http://www.gnome.org/projects/brasero/>) para GNU/Linux
- K3b (<http://k3b.plainblack.com/>) para GNU/Linux
- InfraRecorder (<http://infrarecorder.sourceforge.net/>) para Windows

## Oracle Database

Oracle Database, comúnmente llamado Oracle, es un sistema de gestión de base de datos relacional (RDBMS). Es considerado como uno de los sistemas de bases de datos más completos, destacando por su soporte de transacciones, estabilidad y escalabilidad.

Página web: <http://es.wikipedia.org/wiki/Oracle>

Alternativas libres:

- PostgreSQL (<http://www.postgresql.org/>)
- Firebird (<http://www.firebirdsql.org/>)
- MySQL (<http://www.mysql.com/>)

## Origin

Origin es un software de análisis y visualización de datos para científicos. Entre las herramientas de análisis de datos se incluye análisis estadístico, regresión, procesamiento de señales, procesamiento de imágenes y ajuste de curvas. Se pueden crear distintos tipos de gráficos.

Página web: <http://www.originlab.com/>

Alternativas libres:

- QtiPlot (<http://soft.proindependent.com/qtiplot.html>) está diseñado específicamente para ser una alternativa libre para Origin.
- Fityk (<http://www.unipress.waw.pl/fityk/>) es un programa de ajuste de curvas, análisis y visualización de datos.

## Paint Shop Pro

Paint Shop Pro es un editor de gráficos de mapas de bits y vectoriales.

Página web: <http://www.corel.com/servlet/Satellite/us/en/Product/1184951547051>

Alternativas libres:

- GIMP (<http://www.gimp.org/>)
- Krita (<http://www.koffice.org/krita/>)
- Paint.Net (<http://www.getpaint.net/>)

## POV-Ray

POV-Ray es un programa para crear imágenes tridimensionales mediante la técnica de trazado de rayos, la cual produce una iluminación realista. Los objetos se definen utilizando un lenguaje de programación o un programa modelador externo.

La licencia de POV-Ray permite la libre distribución del programa y del código, pero restringe la distribución comercial e impone ciertas restricciones en la modificación del programa. Por lo tanto, POV-Ray no es software libre.

Página web: <http://www.povray.org/>

Alternativas libres:

- Panorama (<http://panorama.sourceforge.net/>)
- YafRay (<http://www.yafRay.org/>)
- Sunflow (<http://sunflow.sourceforge.net/>)

## PowerBuilder

PowerBuilder es un sistema de desarrollo rápido de aplicaciones (RAD). Incluye herramientas para diseñar la interfaz de usuario, definir reportes y acceder a bases de datos. Las herramientas forman parte de un entorno integrado de desarrollo.

Página web: <http://www.sybase.com/products/internetappdevttools/powerbuilder>

Alternativas libres:

- Gambas <http://gambas.sourceforge.net/> es un entorno de desarrollo basado en un intérprete de Basic con que soporta objetos, que permite diseñar la interfaz del programa y acceder a bases de datos.
- Glade (<http://glade.gnome.org/>) es un programa para diseñar interfaces gráficas de usuario y una biblioteca para acceder a ellas desde diferentes lenguajes de programación.
- Lazarus (<http://www.lazarus.freepascal.org/>) es un sistema RAD para los lenguajes Pascal y Object Pascal, similar a Delphi.

## PowerDesigner

PowerDesigner es una herramienta colaborativa de modelado de aplicaciones empresariales. Incluye herramientas para el modelado de procesos de negocio, generación de código, modelado de datos y datawarehouse, modelado de objetos, generación de reportes, repositorio, análisis de requerimientos, modelado XML.

Página web: <http://www.sybase.com/products/modelingmetadata/powerdesigner>

Alternativas libres:

- ArgoUML (<http://argouml.tigris.org/>) es una herramienta de modelado UML.
- DBDesigner4 (<http://www.fabforce.net/dbdesigner4/>) es una herramienta visual de diseño de bases de datos.
- Umbrello UML (<http://uml.sourceforge.net/>) es un editor de diagramas UML.

- Dia (<http://www.gnome.org/projects/dia/>) es un editor de diagramas que incluye plantillas para realizar diagramas UML y diagramas entidad-relación. Existen programas externos que pueden generar código a partir de diagramas Dia:
  - dia2code (<http://dia2code.sourceforge.net/>) es un generador de código que lee diagramas UML realizados con Dia y los convierte en archivos C, C++, Java, Ada, PHP, Python, SQL, C# y Shapefile.
  - tedia2sql (<http://tedia2sql.tigris.org/>) es un programa que convierte diagramas Dia en archivos SQL compatibles con diferentes motores de bases de datos.
- Druid Database Manager (<http://druid.sourceforge.net/>) es una herramienta gráfica para construir y gestionar bases de datos. Además, Druid puede generar código SQL, documentación, y código en C, C++ y Java.

## Pro/ENGINEER

Pro/ENGINEER es una programa CAD/CAM/CAE (diseño, manufactura e ingeniería asistidos por computador) 3D para ingeniería y diseño mecánico. Utiliza el concepto de modelado sólido paramétrico basado en características, en vez de utilizar solamente valores dimensionales.

Página web: <http://www.ptc.com/>

Alternativas libres: Ver alternativas libres para AutoCAD. Entre ellas, Varkon (<http://varkon.sourceforge.net/>) soporta modelado paramétrico.

## Proteus

Proteus es un software para la simulación de microprocesadores, captura de esquemas y diseño de tarjetas de circuitos impresos (PCB).

Página web: <http://www.labcenter.co.uk/>

Alternativas libres: Ver alternativas para Electronics Workbench.

## PSIM

PSIM es un software de simulación diseñado específicamente para electrónica de potencia y control de motores. Incluye una herramienta para diseñar esquemas de circuitos, un simulador y un visualizador de ondas.

Página web: <http://www.powersys.fr/psimpresent.php>

Alternativas libres: Ver alternativas libres para Electronics Workbench.

## PSpice

PSpice es un software de simulación de circuitos electrónicos basado en SPICE. Pspice es parte de OrCAD, un paquete de software para la automatización de diseño electrónico, que permite diseñar placas de circuitos impresos (PCB), y realizar diagramas y simulaciones de circuitos.

Página web: <http://www.cadence.com/products/orcad/>

Alternativas libres: Ver alternativas libres para Electronics Workbench.



## Publisher

Publisher es una aplicación de publicación de escritorio (*desktop publishing*). Los usuarios pueden crear disposiciones de páginas (*layout*), combinando textos, gráficos e imágenes, generalmente para crear material publicitario.

Página web: <http://www.microsoft.com/office/publisher>

Alternativas libres:

- Scribus (<http://www.scribus.net/>)
- Passepartout (<http://www.stacken.kth.se/project/pptout/>)

## Scientific Workplace

Scientific Workplace es un programa de procesamiento de texto y fórmulas matemáticas que utiliza el sistema de preparación de documentos  $\text{\LaTeX}$ . Incluye un sistema de álgebra computacional, MuPAD, que permite resolver expresiones matemáticas y realizar gráficos en dos y tres dimensiones.

Scientific Word y Scientific Notebook son programas similares a Scientific Workplace, pero que tienen menos funcionalidades. Ambos son procesadores de texto y fórmulas matemáticas. Scientific Word soporta la preparación de documentos  $\text{\LaTeX}$  pero no posee un sistema de álgebra computacional. Scientific Notebook, por el contrario, incluye un sistema de álgebra computacional pero no soporta la preparación de documentos  $\text{\LaTeX}$ .

Página web: <http://www.mackichan.com/>

Alternativas libres:

- $\text{\LaTeX}$  (<http://www.latex-project.org/>) es un sistema de preparación de documentos. Existen varias distribuciones de  $\text{\LaTeX}$  que facilitan su instalación:
  - Tex Live (<http://www.tug.org/texlive/>) es una distribución de  $\text{\LaTeX}$  para GNU/Linux y Windows
  - MiKTeX (<http://miktex.org/>) y proTeXt (<http://www.tug.org/protext/>) son distribuciones de  $\text{\LaTeX}$  para Windows
- Programas con interfaz gráfica que facilitan la escritura de documentos  $\text{\LaTeX}$ :
  - Kile (<http://kile.sourceforge.net/>)
  - Texmaker (<http://www.xmlmath.net/texmaker/>)
  - TeXnicCenter (<http://www.toolscenter.org/>)
- Maxima (<http://maxima.sourceforge.net/es/>) es un sistema de álgebra computacional y puede realizar gráficos en dos y tres dimensiones
- Algunos programas para realizar gráficos de funciones matemáticas:
  - KmPlot (<http://edu.kde.org/kmplot/>) es un programa para realizar gráficos de funciones matemáticas en dos dimensiones
  - KAlgebra (<http://kalgebra.berlios.de/>) es un programa para realizar gráficos de funciones matemáticas en dos y tres dimensiones
  - GraphCalc (<http://www.graphcalc.com/>) es una calculadora gráfica para Windows que realiza gráficos en dos y tres dimensiones

## Scilab

Scilab es un paquete de software de cálculo numérico. Utiliza un lenguaje de alto nivel similar al de MATLAB, pero no son compatibles. Incluye varias funciones matemáticas y posee estructuras de datos de alto nivel, como listas, polinomios, funciones racionales y sistemas lineales.

Algunos sitios web erróneamente consideran a Scilab como software libre u «open source». Sin embargo, la licencia de Scilab prohíbe el uso comercial de versiones modificadas, lo cual viola las libertades 0 y 1, es decir, la libertad de ejecutar el software con cualquier propósito y la libertad de modificar el software.

Página web: <http://www.scilab.org/>

Alternativas libres: Ver alternativas libres para MATLAB.

## SketchUp

SketchUp es un programa de modelamiento 3D. Según sus desarrolladores, posee una interfaz más simple e intuitiva que otros programas de CAD en 3D. Se puede utilizar para diseñar edificios que luego pueden agregarse a Google Earth.

Página web: <http://www.sketchup.com/>

Alternativas libres:

- Wings 3D (<http://wings.sourceforge.net/>)
- Art of Illusion (<http://www.artofillusion.org/>)
- Blender (<http://www.blender.org>)

## Skype

Skype es un programa que permite a los usuarios realizar llamadas telefónicas a través de Internet. Las llamadas a otros usuarios del servicio son gratuitas, mientras que las llamadas a teléfonos fijos o celulares tienen un costo. Otras funcionalidades de Skype son mensajería instantánea, transferencia de archivos y videoconferencias.

Existen algunos problemas de privacidad con Skype, según [95], [96], [97], [98] y [99]. Las llamadas pueden ser interceptadas y la empresa propietaria de Skype, eBay, ha afirmado que va más allá de lo que pide la ley cuando la policía solicita los datos de sus usuarios. Además, Skype accede a información del BIOS para identificar al computador y así proveer medidas de gestión digital de restricciones (DRM).

Página web: <http://www.skype.com/>

Alternativas libres:

- WengoPhone (<http://www.openwengo.org/>)
- Ekiga (<http://ekiga.org/>)

## SolidWorks

SolidWorks es un software de diseño mecánico tridimensional.

Página web: <http://www.3ds.com/products-solutions/solidworks/>

Alternativas libres: Ver alternativas libres para AutoCad.

## SPSS

SPSS es un programa de análisis estadístico para investigadores de *marketing*, investigadores de salud, compañías de encuestas, gobiernos, investigadores de educación, y otros. Además de análisis estadístico, permite la gestión de datos y documentación de datos (metadatos).

Página web: <http://www.spss.com/es/>

Alternativas libres:

- PSPP (<http://www.gnu.org/software/pspp/>) es un software de análisis estadístico similar a SPSS
- R (<http://www.r-project.org/>) es una implementación del lenguaje de programación para estadísticas S

## SPSS AMOS

SPSS AMOS es un programa de modelado de ecuaciones estructurales (*structural equation modelling*), que permite descubrir relaciones causales.

Página web: <http://www.spss.com/la/productos/amos/amos.htm>

Alternativas libres: No se encontraron.

## Software para PLC

Un controlador lógico programable (PLC, programmable logic controller) es un dispositivo electrónico que se utiliza en la automatización industrial.

Alternativas libres:

- EMC o Enhanced Machine Controller (<http://www.linuxcnc.org/>) es un sistema para el control en tiempo real de máquinas, tales como robots, herramientas y sensores.

## SurfCam

SurfCam es un software de diseño y manufactura asistidos por computador (CAD/CAM)

Página web: <http://www.surfware.com/>

Alternativas libres:

- Para CAD, ver alternativas libres para AutoCAD
- Para CAM:
  - GCAM o GNU Computer Aided Manufacturing (<http://gcam.js.cx/>) es un sistema que soporta varias máquinas, tales como tornos y fresadoras.
  - EMC o Enhanced Machine Controller (<http://www.linuxcnc.org/>) también permite controlar tornos y fresadoras.

## SWiSH Max

SWiSH Max es un software para realizar animaciones y presentaciones utilizando Flash, un formato de animación para la web, basado en gráficos vectoriales, que ha sido extendido para incluir sonido y video, y programas escritos en el lenguaje ActionScript.

Página web: <http://swishzone.com/>

Alternativas libres:

- OpenOffice.org Impress (<http://www.openoffice.org/>) es un software para realizar presentaciones, las cuales pueden ser exportadas en formato Flash.
- Synfig (<http://synfig.org/>) es un software de animación en dos dimensiones que utiliza gráficos vectoriales y que puede exportar animaciones en formato Flash.
- KToon (<http://ktoon.toonka.com/>) es un software de animación en dos dimensiones que puede exportar animaciones en formato Flash.

## Tina

Tina es un software de simulación de circuitos y diseño PCB. Soporta circuitos analógicos, digitales y mixtos.

Página web: <http://www.tina.com/>

Alternativas libres: Ver alternativas libres para Electronics Workbench.

## Toad

Toad es una herramienta la administración de bases de datos y el desarrollo en SQL.

Página web: <http://www.toadsoft.com/>

Alternativas libres:

- pgAdmin (<http://www.pgadmin.org/>) es un software de administración de bases de datos PostgreSQL.
- MySQL Navigator (<http://sourceforge.net/projects/mysqlnavigator/>) es un software de administración de bases de datos MySQL.
- phpPgAdmin (<http://phppgadmin.sourceforge.net/>) es una aplicación web para la administración de bases de datos PostgreSQL.
- phpMyAdmin (<http://www.phpmyadmin.net/>) es una aplicación web para la administración de bases de datos MySQL.
- KNoda (<http://www.knoda.org/>) es un software de administración de bases de datos DBase, Firebird, MS Access, MySQL, Paradox, PostgreSQL, SQLite y ODBC.
- DBDesigner 4 (<http://www.fabforce.net/dbdesigner4/>) es un software de diseño, modelado, creación y mantenimiento de bases de datos.
- Quantum (<http://sourceforge.net/projects/quantum/>) es un plugin para Eclipse para el desarrollo en SQL. Soporta varios sistemas gestores de bases de datos. Requiere Java.

- GNOME-DB (<http://directory.fsf.org/project/gnomedb/>) es un framework para desarrollar aplicaciones orientadas a bases de datos. Puede acceder a bases de datos PostgreSQL, MySQL y Oracle, y a fuentes de datos ODBC.
- Ferret (<http://www.gnuferret.org/>) es una herramienta para realizar diagramas de entidad-relación, generar diagramas con las tablas y generar las sentencias SQL correspondientes.
- Dia2Code (<http://dia2code.sourceforge.net/>) es un programa que genera código en distintos lenguajes, incluyendo SQL, a partir de diagramas UML realizados con Dia.

## UltraEdit

UltraEdit es un editor de textos que posee herramientas para programadores, tales como macros, resaltado de sintaxis, gestión de proyectos y búsqueda con expresiones regulares.

Página web: <http://www.ultraedit.com/products/ultraedit.html>

Alternativas libres:

- gedit (<http://www.gnome.org/projects/gedit/>) es un editor de texto sencillo que forma parte del entorno GNOME.
- Kwrite (<http://www.kate-editor.org/>) es un editor de texto sencillo que forma parte del entorno KDE.
- Kate (<http://kate-editor.org/>) es un editor de texto avanzado que forma parte del entorno KDE.
- Emacs (<http://www.gnu.org/software/emacs/>) es un editor de texto avanzado.
- Vim (<http://www.vim.org/>) es un editor de texto avanzado.
- Notepad++ (<http://notepad-plus.sourceforge.net/es/site.htm>) es un editor de código fuente para Windows.

## Visual Studio

Visual Studio es un entorno integrado de desarrollo (IDE) de Microsoft. Incluye herramientas para diseñar interfaces gráficas de usuario, páginas web y esquemas de bases de datos.

Página web: <http://msdn.microsoft.com/en-us/vstudio/default.aspx>

- Eclipse (<http://www.eclipse.org/>) es un IDE extensible. Funciona con una máquina virtual Java libre.
- KDevelop (<http://www.kdevelop.org/>) es un IDE para el entorno KDE.
- Anjuta (<http://anjuta.sourceforge.net/>) es un IDE para el entorno GNOME.
- Dev-C++ (<http://www.bloodshed.net/devcpp.html>) es un IDE para Windows.
- Gambas (<http://gambas.sourceforge.net/>) es un IDE basado en un intérprete de Basic. Es similar a Visual Basic pero no busca ser compatible con dicho programa.
- Glade (<http://glade.gnome.org/>) es un diseñador de interfaces gráficas de usuario.

- QT Designer (<http://trolltech.com/products/qt/features/tools/designer>) es un diseñador de interfaces gráficas de usuario.
- Umbrello (<http://uml.sourceforge.net/>) permite realizar diagramas UML.
- Dia (<http://www.gnome.org/projects/dia/>) es un editor de diagramas genéricos que también soporta diagramas UML.
- SharpDevelop (<http://www.sharpdevelop.com>) es un IDE para la plataforma .NET
- Mono ([www.go-mono.org](http://www.go-mono.org)) es una plataforma de desarrollo .NET
- Monodevelop (<http://www.monodevelop.com/>) es un IDE para la plataforma .NET
- DotGNU (<http://www.dotgnu.org/>) es un software para el desarrollo de aplicaciones web y programación en C#.

## WinAmp

WinAmp es un reproductor de música y video.

Página web: <http://www.winamp.com/>

Alternativas libres:

- Amarok (<http://amarok.kde.org/>) es un reproductor de música para el entorno KDE
- Rhythmbox (<http://www.gnome.org/projects/rhythmbox/>) es un reproductor de música para el entorno GNOME
- VLC (<http://www.videolan.org/vlc/>) es un reproductor de video y audio.
- aTunes (<http://www.atunes.org/>) es un reproductor de música para Windows
- musikCube (<http://www.musikcube.com/>) es un reproductor de música para Windows
- Los entornos GNOME y KDE incluyen programas para reproducir archivos de música y video

## Windows Media Player

Windows Media Player es un reproductor de música y video.

Página web: <http://www.microsoft.com/windows/windowsmedia/es/>

Alternativas libres: ver alternativas libres para WinAmp.

## Winplot

Winplot es un software de trazado de gráficos de propósito general.

Página web: <http://math.exeter.edu/rparris/winplot.html>

Alternativas libres:

- KmPlot (<http://edu.kde.org/kmplot/>) es un programa para realizar gráficos de funciones matemáticas en dos dimensiones

- KAlgebra (<http://kalgebra.berlios.de/>) es un programa para realizar gráficos de funciones matemáticas en dos y tres dimensiones
- GraphCalc (<http://www.graphcalc.com/>) es una calculadora gráfica para Windows que realiza gráficos en dos y tres dimensiones
- Gnuplot (<http://www.gnuplot.info/>) es un trazador de gráficos de propósito general que funciona desde la línea de órdenes, pero también existen algunas interfaces gráficas para utilizarlo más fácilmente, como XGFE.

## WinRAR

WinRAR es un gestor de archivos comprimidos. Soporta RAR y ZIP y otros formatos de archivo.

Página web: <http://www.rarlab.com/>

Alternativas libres:

- 7-Zip (<http://www.7-zip.org/>) es un gestor de archivos comprimidos para Windows
- PeaZip (<http://peazip.sourceforge.net/>) es un gestor de archivos comprimidos para Windows y GNU/Linux
- Los entonos GNOME y KDE incluyen programas gestores de archivos comprimidos.

## WinZip

WinZip es un gestor de archivos comprimidos. Soporta ZIP, RAR y otros formatos de archivo.

Página web: <http://www.winzip.com/>

Alternativas libres: Ver alternativas libres para WinRAR

## B.3. Programas libres utilizados en la Universidad

El siguiente es un listado de los programas libres que, según la encuesta realizada, son utilizados en la Universidad de Tarapacá. Se incluye también la dirección web y una breve descripción de cada programa.

- 7-zip (<http://www.7-zip.org/es/>): gestor de archivos comprimidos
- Amaya (<http://www.w3.org/Amaya/>): editor de páginas web
- aMSN (<http://www.amsn-project.net/>): cliente de mensajería instantánea
- Anjuta (<http://anjuta.sourceforge.net/>): entorno de programación
- Apache (<http://www.apache.org/>): servidor web
- Bison (<http://www.gnu.org/software/bison/>): generador de analizadores sintácticos
- Blender (<http://www.blender.org/>): modelador 3D
- Bluefish (<http://bluefish.openoffice.nl/>): editor de páginas web

- Cygwin (<http://www.cygwin.com/>): entorno GNU para Windows
- Dev-C++ <http://www.bloodshed.net/devcpp.html>: entorno de programación
- Dia (<http://www.gnome.org/projects/dia/>): editor de diagramas
- Eclipse (<http://www.eclipse.org/>): entorno de programación
- Emesene (<http://www.emesene.org/>): cliente de mensajería instantánea
- Epiphany (<http://www.gnome.org/projects/epiphany/>): navegador web
- Evolution (<http://www.gnome.org/projects/evolution/>): cliente de correo electrónico y agenda
- Firefox (<http://www.mozilla.com/firefox/>): navegador web
- Flex (<http://www.adobe.com/products/flex/>): biblioteca para desarrollar aplicaciones Flash. ADVERTENCIA: para ejecutar las aplicaciones desarrolladas con Flex puede ser necesario utilizar el reproductor privativo de Flash de Adobe.
- Gambas (<http://www.mozilla.com/firefox/>): entorno de programación
- GCC (<http://gcc.gnu.org/>): colección de compiladores
- gedit (<http://www.gnome.org/projects/gedit/>): editor de textos
- Ghostscript (<http://www.ghostscript.com/>): intérprete de PostScript y PDF
- GhostView (<http://pages.cs.wisc.edu/~ghost/>): visor de archivos PostScript y PDF para GNU/Linux
- GIMP (<http://www.gimp.org/>): editor de imágenes
- GNU Octave (<http://www.gnu.org/software/octave/>): lenguaje de alto nivel para computación numérica similar a MATLAB
- GNU Pascal (<http://www.gnu-pascal.de/gpc/h-index.html>): compilador de lenguaje Pascal
- Gnuplot (<http://www.gnuplot.info/>): trazador de gráficos
- Icecast (<http://www.icecast.org/>) es un software para la transmisión de audio y video mediante *streaming*
- ImageMagick (<http://www.imagemagick.org/>): herramientas de manipulación de imágenes
- Inkscape (<http://www.inkscape.org/>): editor de gráficos vectoriales
- JBoss (<http://www.jboss.org/>): servidor de aplicaciones web
- Kile (<http://kile.sourceforge.net/>): entorno de edición LaTeX
- Maxima (<http://maxima.sourceforge.net/>): sistema de álgebra computacional
- MikTeX (<http://miktex.org/>): sistema LaTeX



- Moodle (<http://moodle.org/>): sistema de gestión de cursos para aprendizaje en línea
- Mozilla (<http://www.seamonkey-project.org/>): navegador web
- MySQL (<http://www.mysql.com/>): sistema gestor de bases de datos
- NetBeans (<http://www.netbeans.org/>): entorno de programación
- Nvu (<http://www.nvu.es/>): editor de páginas HTML
- OpenOffice.org (<http://es.openoffice.org/>): paquete de software de ofimática
- Otter (<http://www.cs.unm.edu/~mccune/otter/>): demostrador de teoremas para lógica relacional
- PDF Creator (<http://sourceforge.net/projects/pdfcreator/>): impresora virtual para crear archivos PDF desde distintas aplicaciones para Windows
- pgAdmin (<http://www.pgadmin.org/>): gestión de bases de datos PostgreSQL
- PHP (<http://www.php.net/>): lenguaje de programación para la web
- Pidgin (<http://www.pidgin.im/>): cliente de mensajería instantánea
- PLT DrScheme (<http://www.plt-scheme.org/software/>): intérprete de Scheme
- PostgreSQL (<http://www.postgresql.org/>): sistema gestor de bases de datos
- PuTTY (<http://www.chiark.greenend.org.uk/~sgtatham/putty/>): cliente de Telnet y SSH para Windows
- Python (<http://www.python.org/>): lenguaje de programación interpretado
- Quanta Plus (<http://quanta.kdewebdev.org/>): editor de páginas web
- Sistemas desarrollados por la Universidad: Estos sistemas corresponden a la categoría de software de uso privado, por lo tanto son libres para la Universidad, pues es libre de usarlos y modificarlos, y de copiarlos y distribuirlos si lo desea:
  - Sistema académico y financiero
  - Intranet
  - UtaMed
  - Exact (mecánica)
  - Telar, Puzzle (programa educativo desarrollador por alumnos de computación para arqueología)
- SWI-Prolog (<http://www.swi-prolog.org/>): compilador de PROLOG
- TexnicCenter (<http://www.texniccenter.org/>): entorno de edición LaTeX
- Thunderbird (<http://www.mozilla.com/thunderbird/>): cliente de correo electrónico
- WinDjView (<http://windjview.sourceforge.net/>): visor de imágenes en formato DejaVu
- WinSCP (<http://winscp.net/>): cliente para transferencia de archivos mediante SFTP, FTP y SSH para Windows

## Apéndice C

# Programa para curso de capacitación en OpenOffice

OpenOffice es un paquete de ofimática libre que puede utilizarse como un reemplazo de Microsoft Office, el paquete de ofimática privativo utilizado por la mayoría de los funcionarios y alumnos en la Universidad.

A continuación se presenta un programa para un curso de capacitación en OpenOffice de 8 horas:

- Breve introducción a OpenOffice. ¿Qué es OpenOffice? ¿Qué diferencias hay con Microsoft Office? ¿Cómo obtener ayuda?
- OpenOffice Writer
  - Propiedades de la página. Columnas, encabezados y pies de página, tamaño de página.
  - Insertar objetos. Imágenes, tablas, diagramas, campos, hiperenlaces.
  - Revisión ortográfica Plantillas.
  - Uso de estilos para mantener un formato consistente.
  - Creación automática de índices.
- OpenOffice Impress
  - Consejos para realizar diapositivas. Colores, contraste, tamaño del texto, uso de imágenes.
  - Patrones de diapositivas. Editar el estilo de todas las diapositivas.
  - Transiciones
- OpenOffice Calc
  - Gráficos
  - Ordenar y filtrar datos
  - Funciones