

1. *Porqué los conocimientos científicos evolucionan tan deprisa*

Poco antes del verano se celebraban, creo recordar que en Barcelona, unas jornadas sobre la figura de Einstein tanto desde el punto de vista científico como desde el humano; en alguno de los comentarios vertidos en la prensa se destacaba a Einstein como una de las más grandes figuras científicas de todos los tiempos, superando ampliamente a personajes de la talla de Newton. Tras pensar algún tiempo sobre esta afirmación llegué a la conclusión de que no podía sostenerse objetivamente; en efecto, ambos personajes no son comparables toda vez que cuando Einstein desarrollo sus teorías no partía de la nada, las propuestas de Newton y las numerosas experiencias científicas realizadas desde entonces, unas veces, las más, confirmando la teoría general de la gravedad, y otras mostrando que había ciertos casos que no se ajustaban a la misma, fueron un amplio bagaje científico que el científico del siglo pasado supo aprovechar bien; muy distinto fue el panorama y los medios con los que Newton pudo desarrollar su investigación.

Para poder decir si uno fue mejor científico que el otro deberíamos tener la posibilidad de que hubieran investigado en las mismas condiciones (mismos medios, mismos conocimientos de referencia, etc.). Esta es una de las grandezas de la ciencia, el trabajo de un científico a través de sus publicaciones, conferencias, participaciones en congresos, etc. se hace accesible al resto de la humanidad; de este modo los avances son enormes, pues un científico novel puede empezar su investigación en el punto en que la han dejado sus predecesores; al gran Leonardo da Vinci se le atribuye la famosa frase: <<mal discípulo es aquel que no supera a su maestro>> que se presta perfectamente a la situación comentada.

2. *¿Y qué pasa con las aplicaciones informáticas?*

Frente a este modo de proceder muy asumido en el mundo de la investigación, llama la atención que en uno de los instrumentos que cada día se utilizan más en la misma, la informática, el desarrollo de los programas de ordenador más usuales (sistema operativo y aplicaciones) se haya movido en las últimas décadas dentro del mundo de las patentes, la ocultación del código fuente y los intereses puramente comerciales. Los precios alcanzados por algunas de estas aplicaciones son realmente astronómicos, de modo que lo único que se puede hacer en muchas ocasiones es olvidarse de ellos u obtener copias "ilegales" de los mismos.

Sin embargo el tema no parece preocupar mucho a los organismos oficiales; el ejemplo más claro, la reciente convocatoria de becas de la Fundación Séneca, dependiente de los órganos de gobierno de la Comunidad Autónoma de Murcia; en la misma se podía leer (página web: <http://www.f-integra.org/seneca/convocatorias/ayuda.htm>):

<<Becas de formación investigadora; Programa Séneca 2002: rellenar los formularios...

LOS FORMULARIOS SE DEBEN RELLENAR EN MICROSOFT WORD 6.0 O WORD 97.

NO SE ADMITIRÁN CONVERSIONES DE OTROS PROCESADORES>>¹

¡Esta sí que es buena!; es decir, para poder optar a una beca el solicitante debe ser propietario de una licencia de MicrosoftOffice (que vale lo suyo), por supuesto no pirateada, pues desde un organismo oficial no se va a fomentar esa ilegalidad, y sólo así podrá entregar una solicitud que no sea rechazada de antemano. ¿Y por qué no le indican también el modelo de ordenador a usar, la marca de coche con la que deberá acercarse a Séneca para hacer entrega de los formularios, la colonia que deberá usar en el día de la entrega.....?. Estamos llegando a extremos increíbles en la imposición de monopolios informáticos y sin embargo la sociedad española lo acepta como si fuera la única opción (¡me gustaría saber cómo reaccionarían los españolitos de a pie si al comprarse un coche le obligaran a llenar combustible únicamente en unas determinadas marcas de gasolineras o llevarlos a lavar a un sitio fijo u obligarles a pasar las vacaciones en una determinada zona!).

¿La única opción?; ¿realmente no queda nada de aquellos primeros programadores de la década de los 70 del siglo pasado que creaban programas informáticos, generalmente en el entorno universitario, y los compartían con la comunidad, intercambiando códigos y mejorándolos a base de aportar cada uno su sapiencia en el tema, lo que supuso avances gigantescos en la informática de la época?; ¿ya no es posible obtener el código fuente de un programa que nos interesa y adaptarlo a nuestras necesidades?; entonces, ¿debemos de ser nosotros los que nos adaptemos a las aplicaciones informáticas y no al revés?.

La sensación mayoritaria que podemos obtener hoy en día es que ordenador y aplicaciones informáticas son todo uno, te compras el ordenador que, por supuesto, funciona con el correspondiente Windows (95, después 98, ahora XP...) y como no el Microsoft Office con su Word, Excel,; para los que usan los "extraños Mac", la cosa no cambia mucho, un sistema operativo por defecto y los programas de oficina y otros a los que debemos adaptarnos.

¹ Es curioso, pero seguro que muchos de los sufridos lectores de Bacteria han tenido con MsWord el típico problema de "fichero corrupto" que no puede ser recuperado; pues bien, muchos de esos los recupera perfectamente el procesador de textos de OpenOffice, además de no verse afectado por los conocidos virus de macro que pululan por los ficheros de MsWord (disponible para Windows, Linux, Solaris y Mac); ¡pero no se puede usar para rellenar los formularios de Séneca!!!!!!.

Pero, ¿qué hacen realmente esos programas cuyo código fuente no conocemos?; ¿sabéis que el gobierno alemán ha abandonado Windows por que tiene serios indicios de que este sistema hace más de lo Microsoft dice que hace y envía información de nuestro ordenador a no se sabe, aunque se sospecha, dónde?. Porque, si no conocemos el código de los programas que están funcionando en nuestro ordenador (bueno, seguro que la mayoría de nosotros aún conociéndolo poco podríamos utilizarlo, pero muchos programadores detectarían de inmediato qué hace cada fragmento del código) ¿cómo podemos saber que no realiza cosas que no están bajo nuestro control?; además, a lo que íbamos en el inicio de este artículo, si fuéramos programadores, ¿no sería una gran ventaja para nosotros el poder adaptar un código ya conocido a nuestras necesidades o las de nuestro cliente y no partir de cero para realizar tal trabajo?; esto entroncaría muy bien con la forma de desarrollarse la ciencia, usando los resultados de los que nos precedieron, y aportándolos a los que nos sucederán.

¿Una utopía?; pues no, además del conocidísimo software comercial o <<propietario>>, existen comunidades de programadores que comparten el código, usando el que previamente han realizado otros y aportando el suyo propio (programas completos o programas modificados a partir del creado por otros) a toda la comunidad. Es el llamado <<software libre>> o de <<código fuente abierto>>, en el cual el programador ofrece su trabajo en forma de código original, con ciertas licencias particulares que permiten a otros transformar las aplicaciones pero reconociendo explícitamente que son también de código abierto y ofreciéndolas bajo las mismas condiciones al resto del mundo. En unos casos son sistemas operativos completos (Linux, FreeBSD, etc.), en otras son aplicaciones informáticas que pueden funcionar bajo un sistema operativo de código abierto o no, el conjunto constituye hoy día una alternativa muy clara al software propietario, pero que es sorprendentemente desconocida incluso en el ámbito universitario; no es la primera vez que he visto quedarse con la boca abierta a algún compañero ante comentarios sobre “Linux, OpenOffice.....”; ¡como si les sonara a palabras alienígenas!

3. *Cómo funciona eso del software libre*

El mundo del software libre es muy dinámico, como consecuencia de haber nacido directamente en Internet, hay una gran cantidad de foros, listas de correo y páginas web en las que se ofrecen las aplicaciones, las dudas más frecuentes sobre cómo instalarlas y usarlas, actualizaciones, etc. En el desarrollo de los programas suelen ofrecerse muchas personas como voluntarios para probarlos, detectar fallos y comentarlos con el autor o los autores. De hecho una de las experiencias más enriquecedoras que he tenido en este mundo particular fue la primera vez que probando una versión de desarrollo de un programa (*Xine*, visualizador de películas en DVD y otros formatos), ante problemas para su correcta compilación e instalación y teniendo una consola abierta donde estaba tratando inútilmente de hacer funcionar el mismo, abrí otra consola y entré en un IRC (chat) donde el autor me fue indicando lo que debía hacer, ante los fallos me proponía cambiar de táctica y al final acabé con el programa funcionando en una parte de la pantalla y en la consola del chat tenía al autor de *Xine* agradecido porque había detectado un problema y la forma de solucionarlo (infinitamente más agradecido estaba yo, que ya no necesitaba comprar un programa propietario para ver las películas en DVD y había aprendido muchísimas cosas interesantes en un rato); a las pocas horas de arreglado mi problema ya se podía descargar una versión del programa *Xine* con este y otros arreglos implementados. Situaciones similares las he vivido en numerosas ocasiones en mis ya dos años y medio usando el sistema operativo Linux (¡qué lejos ese 1985 en el que me inicié en el mundo de la informática con mi viejo Amstrad CPC6128, que aún me funciona, y Jorge de Costa me pasaba programas que había logrado copiar desde cintas a disquetes, que eran el no va más de la informática de la época!); cosas como la traducción al castellano de algún programa o correcciones de pequeños fallos de presentación de texto son muy fáciles de realizar ante un código al que tienes acceso. Y si no puedes hacerlo tú mismo, hay un nutrido grupo de gente dispuesta a hacerlo por ti y generalmente la solución del problema llega en segundos (chat) o en unas pocas horas a través del correo electrónico.

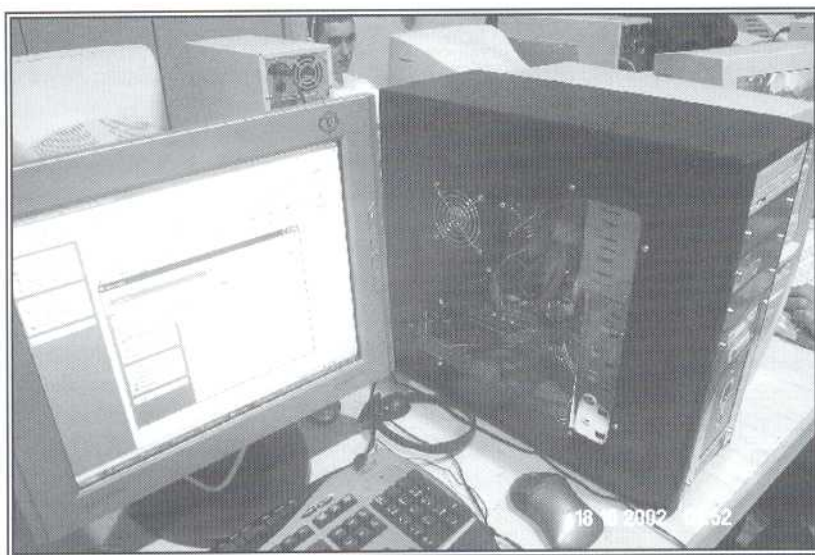
En ocasiones se detectan fallos importantes en los programas libres que pueden afectar a la seguridad de tus datos, pero generalmente podemos decir que detectado y arreglado, ya que a las pocas horas estamos en condición de instalar la nueva versión con el error corregido; en las aplicaciones comerciales tras detectar algún error en el programa habrá que esperar a la futura nueva versión, por la que además habrá que pagar (¡qué ironía, pagar más a quien te ha vendido un programa defectuoso!).

Es muy frecuente en el mundo del software libre la creación de programas de forma modularl, me explicaré. Imaginaos que soy un programador que quiero hacer una aplicación que descargue ficheros Mp3 por Internet o los cree a partir de un Cd de audio, corrija defectos de sonido y luego los pase a un formato a partir del cual se pueda crear un nuevo Cd de audio utilizable en cualquier aparato de música con lector de cdrom; podría partir de cero, pero es más fácil, rápido y da mejores resultados recoger aplicaciones, si existen, que hagan muy bien algunas de estas tareas, añadirle las que completen lo que falte y una de dos, o juntarlas todas en un programa o crear simplemente una aplicación que vaya llamando a las anteriores según se requiera. Así podemos aprovechar programas informáticos que realizan tareas muy concretas pero que por la misma razón han podido ser probados y mejorados hasta el último detalle, por lo que en muchas ocasiones las realizan con gran perfección; de este modo en el mundo del software libre se han conseguido programas que trabajan con una perfección que envidian los autores de muchos programas propietarios que valen muchísimos euros.

4. Fuera tópicos:

SOFTWARE LIBRE no significa DIFÍCIL DE INSTALAR Y DE UTILIZAR

¡Ya, ya!, pero.... ¡esos sistemas libres son muy difíciles de instalar y de usar!; ¿os suena este comentario?; ciertamente que al nacer en un entorno de personas que dominaban la informática los sistemas operativos y las aplicaciones informáticas libres solían hacer pocas concesiones para los novatos, no abundaban los programas con una presentación gráfica atractiva ni botones para hacer las tareas con un simple "clic" de ratón, pero el panorama ha cambiado muchísimo y así, por ejemplo en el caso de Linux, los últimos cuatro años han visto un vertiginoso avance en el desarrollo de los entornos gráficos, el cual sólo ha sido posible gracias a la disponibilidad del código fuente. Por ejemplo, mientras que en Windows podemos estar más que contentos con cambiar el fondo de la pantalla, buscar unos iconos más originales y poco más, en Linux podemos elegir entre numerosos entornos diferentes (Kde, Gnome, Fluxbox, Enlightenment, Sawfish, Icewm, Windowmaker, etc.); algunos son sumamente configurables, otros consumen muy poca memoria y son más apropiados para equipos con más reducidos recursos de memoria o tarjetas gráficas más antiguas. Además, en el caso extremo de ordenadores muy antiguos podemos optar por hacer muchas cosas desde la línea de comandos y olvidarnos de cargar un entorno gráfico, o aprovechar las posibilidades de estos sistemas operativos nacidos en Internet para arrancar el sistema de ventanas y los programas en otro ordenador remoto (un servidor) mejor "acondicionado", pudiendo así salvar equipos en los que ya no se puede ni instalar un Windows98 y no digamos un WindowsXP, y así invertir los generalmente limitados recursos económicos en cualquier otra cosa más necesaria.



En lo que respecta a la facilidad de instalación desde cero (son muchas las personas a las que le han entregado el ordenador con su versión de Windows instalada y afirman lo de la facilidad cuando nunca han tenido que vérselas con las dificultades de poner el Windows desde el principio), os puedo decir que es un mito; en los numerosos casos en que he tenido ordenadores con alguna versión de Windows y alguna distribución de Linux funcionando, he necesitado más "cedes" con controladores específicos para Windows que para Linux (las distribuciones que he probado de Linux son, sobre todo Mandrake, pero también RedHat, Suse, Lindows y Linex). ¡Ya!, ¿pero dónde está el Explorer, el Word instalado, el Excel, y....?. ¡Pero bueno!, ¿es que el Word es tan imprescindible para un ordenador como el teclado para las antiguas máquinas de escribir?.

Hay muchísimas aplicaciones informáticas que igualan o incluso superan a las de Microsoft, algunas son comerciales (StarOffice, HandOffice, WordPerfect, Lotus, etc.) y otras son de código fuente libre (OpenOffice, Koffice, etc.). Esto se puede aplicar a cualquier otro tipo de aplicación informática que se os ocurra: hojas de cálculo, programas de gráficos, de dibujo, de música, de vídeo, de cálculo numérico, de autoedición, etc.; además, muchos de ellos graban los datos en archivos con formatos libres, muchas veces en modos de sólo texto que ocupan muy poco espacio en disco y así no estás sujeto a un software propietario que incluso puede hacer, para obligarte a comprar las nuevas versiones del programa, que los formatos de los archivos de datos sean incompatibles de una versión a otra más reciente (¿a que os suena a ciertos programas comerciales?).

5. Reflexiones finales

Así pues, hay alternativas al software propietario, sistemas operativos y programas libres, de código abierto, con formato de los ficheros de datos libre, que pueden ser adaptados a las necesidades de cada cual de manera relativamente fácil, en los que los errores se corrigen nada más ser detectados; además estos sistemas operativos y programas son frecuentemente gratuitos o se ofrecen a precios muy reducidos, muchas veces centrados en la asesoría de la instalación y de la adaptación a las necesidades del usuario. El código abierto permite una evolución rápida del software y una más fácil adaptación a las necesidades de cada usuario, en la línea de lo mencionado para el desarrollo de la Ciencia. Además se trata de sistemas mucho más seguros, con unas pocas costumbres de uso basadas en el sentido común y la actualización de los paquetes en cuanto se detectan fallos podemos olvidarnos de los virus, mientras que el sistema de trabajo multiusuario evita que el error de uno de los usuarios o la infección por un virus afecten a todo el sistema y, por tanto, a los demás usuarios.

Entonces, si estos sistemas libres son tan potentes y su desarrollo sigue las positivas estrategias de la compartición de datos que tan bien le han ido a la Ciencia y en general al desarrollo de la civilización, ¿por qué no se conocen y usan más?. La respuesta no es sencilla, pues incluye aspectos tan variados como la inercia y la resistencia al cambio, la inexistencia en el software libre de las costosas campañas publicitarias que rodean al software propietario, la escasez de ordenadores que se venden con alguno de estos sistemas ya instalado, el desconocimiento de la existencia de aplicaciones alternativas a las que "todos usan", etc.

Yo mismo dudé mucho antes de, ante la insistencia de José Antonio Palazón, lanzarme a “ver de qué iba aquello”; dos años después apenas uso Windows y ya en el curso pasado en todas las clases de teoría que impartí me ayudé de las presentaciones por ordenador de OpenOffice. Entre el laboratorio de investigación y el despacho que ocupo tenemos 5 ordenadores de sobremesa y un portátil; el portátil y uno de los ordenadores de sobremesa, que usamos como servidor principal, ya sólo tienen instalado Linux; un viejo 486 estamos a punto de montarlo como una terminal para trabajar con Linux pero arrancado desde el servidor de forma remota, los otros tres comparten dos sistemas operativos (Windows 98 y Linux), pero la mayor parte del tiempo funcionan bajo el segundo.

Sobre las ventajas del software libre merece la pena leer la carta de respuesta del congresista peruano Edgard David Villanueva Núñez al Gerente General de Microsoft en Perú ante sus críticas al *Proyecto de Ley Software Libre en la Administración Pública* de ese país; se puede ver en la siguiente página web: http://pimientolinux.com/peru2ms/villanueva_a_ms.html

6. Mis preferencias en software libre

En el cuadro 1 se recogen algunos de los ejemplos de aplicaciones libres que utilizamos en el laboratorio o que uso en casa, donde tengo más desarrollado el aspecto multimedia y lúdico. La distribución Linux que preferentemente uso es la correspondiente a Mandrake (ahora ya por la 9.0).

¡Ah!, este documento se entregó a la redacción de *EUBACTERIA* en formato “doc” de Microsoft, pero ha sido realizado con OpenOffice que puede grabar, si no hay más remedio, sus archivos de datos en ese formato propietario (¡a ver si toman nota estos del reino de los procariotas!), completa suite ofimática que está disponible para Linux, Solaris, PowerPc (Macs) y Windows.

<i>Características o tipo de uso</i>	<i>Aplicaciones libres</i>
Astronomía	Kstars, Openuniverse
Autoedición	Scribus
Bases de datos	MySQL, MySQL + OpenOffice
Bibliografía	Tkbibtex, Pybliographic
Cálculo numérico y representaciones gráficas de funciones matemáticas	R, Gnuplot, Mathplot, Octave
Correo electrónico	Mozilla, Kmail
Descarga de ficheros multimedia	Edonkey, Gnutella, Lobster
Dibujo	Gimp
Digitalización (escáner, cámara digital, diapositivas, etc.)	Xsane-gimp, Kooka
Grabación de “cedés”	Xcdroasts, Cdrdao, Cdrecord, Mkisofs
Hojas de cálculo	Gnumeric
Intercambio de archivos	Samba, Gftp, Iglooftp, Ssh
IRC (Chat)	Xchat
Música	Kscd, Tcd (curioso, en formato no gráfico), Xmms
Navegadores en Internet	Mozilla, Galeon, Konqueror, Nautilus
Paso de películas dvd a cds	Dvdrip + Transcode
PDA (conexión con agendas electrónicas)	Jpilot, Kpilot
Preparación de discos de audio a partir de Mp3	Grip + Xmms
Procesadores de texto	Emac, Nedit, Kwrite, Mc
Proceso de textos profesional	Tetex + Kile
Refrescadores de pantalla	Xscreensaver (¡decenas de refrescadores; magnífico!)
Sistema operativo / distribución	Linux / Mandrake
Sistemas de información geográfica	Grass
Suites ofimáticas (procesador de textos + hoja de cálculo + dibujo + presentaciones + ...)	OpenOffice, Koffice
Televisión codificada (Cablecrypt): Ono	Cabletv
Televisión codificada (Nagravisión): Canal +	Xawdecode, Taztv
Televisión no codificada	Xawtv, Zapping, Kwintv
Uso de equipos remotos	Ssh, Xwrapper, Kdm, Xdm
Visualización de películas	Xine, Videolan, Aviplay, Mplayer

Hasta la próxima y bienvenidos al curso 2002-2003.