

El Linuxita

Tribuna electrónica al servicio de los usuarios de software libre

<<http://eroj.org/linux>>

<linux@eroj.org>

ALEGATO A FAVOR DEL LINUX

Lorenzo Peña

En las cosas humanas sucede frecuentemente que lo que empieza siendo una opción instrumental se convierte en una preferencia intrínseca. Lo que llamamos ‘arte’ suele haber seguido esa vicisitud. Empiézanse construyendo inmuebles para guarecerse de la intemperie y luego la arquitectura adquiere el rango de un arte y (para algunos humanos) se convierte en finalidad en sí misma. Pueden multiplicarse los ejemplos.

La computación es una de esas actividades. Muchos son aquellos que han empezado por dedicarle un poquito de tiempo —como mero instrumento para realizar otras tareas— pero para quienes, a la postre, el instrumento se ha trocado en un fin en sí mismo. Para muchos se ha convertido en un modo de vivir; para muchos más en uno de los principales pasatiempos; para no pocos es una pasión. Igual que se puede querer el arte por el arte, puede quererse (y a menudo se quiere) la computación por la computación.

Las consideraciones que voy a presentar aquí a favor de un sistema operativo particular, el Linux, son (espero) asumibles independientemente de que el lector se encuentre entre quienes dedican algo de tiempo a la computación sólo a título de mero instrumento en aras de bienes intrínsecamente deseables o entre quienes se han aficionado a ella hasta el punto de desearla como una finalidad en sí misma.

Creo que hay motivos para optar por el Linux desde ambas perspectivas. Quienes vean en la computación una finalidad podrán tener motivos adicionales para adoptar una u otra opción (motivos que no serán sólo los de cuánto sirvan los programas cuyo uso adopte uno para obtener tales o cuales resultados, sino que incluirán también aspectos estéticos y similares).

Para no adentrarme en terrenos en los que es mucho lo que depende del gusto personal, me limitaré a considerar el uso de sistemas operativos desde el punto de vista meramente instrumental, como si la única razón para dedicar algún tiempo a actividades de computación fuera el poder, con ella, alcanzar resultados que son extrínsecos al mero manejo de computadoras (resultados como pueden ser: preparar manuscritos; editar folletos o libros; imprimir; hacer diapositivas o transparencias; obtener resultados de cálculos complicados; llevar una contabilidad; poner —a través de la *Web*— ciertos documentos al alcance de un amplio público; hacer trabajos de diseño; escrutar textos impresos y utilizar lo así escrutado para alguna de las tareas recién indicadas; reproducir discos compactos con álbumes de fotos, para ponerse a salvo de pérdida de datos por rotura o extravío del soporte; etc.).

Para efectuar tales trabajos, necesita una computadora (normalmente un PC), un sistema operativo y unos programas de *software* adecuados para las tareas que se proponga realizar. Cuando digo que hace falta una computadora entiendo que la misma lleve, como anejos, ciertos periféricos adaptados a las necesidades: impresora, escrutadora, lectora o grabadora de CD-Rom, etc. O sea, todo el *hardware*.

Cae fuera del ámbito de este artículo —y también de la competencia de quien lo escribe— considerar alternativas al *PC* propiamente dicho (el IBM-compatible). Sea como fuere, y hasta donde puedo juzgar, la gran mayoría de los usuarios de computadoras se han decantado por *PC*'s (IBM-compatibles), ya sea por el precio, ya sea por las prestaciones, ya sea por imitación de lo que otros hacen a su alrededor o por lo que sea. Vamos a dar por admitida esa opción ampliamente mayoritaria, aunque no podremos desentendernos de la necesidad de buscar en nuestro trabajo la mayor compatibilidad con los instrumentos usados por la gente del *Mac* o por los usuarios de terminales UNIX.

Una vez que hemos optado por tener una computadora personal de tipo IBM (un *PC*), nos toca dar el segundo paso, y seleccionar un sistema operativo (para *PC*). Hay muchos.

No vamos aquí a considerarlos todos. Igual que, cuando uno se plantea en la vida si se va a mudar y va a ir a buscar empleo, amor, u hogar a otro sitio no se sienta ante un globo terráqueo dándole vueltas para ver dónde se posará el dedo, tampoco normalmente hace uno una lista de todos los sistemas operativos disponibles para sopesar sus respectivos *pros* y *contras* hasta ver por cuál se decantará. En abstracto eso sería racional, pero en la práctica no lo es, porque demandaría un esfuerzo demasiado grande. Lo normal es considerar unas pocas alternativas, porque son aquellas con las que se ha topado uno, las que le han salido al paso o no andan lejos.

Así, entre los sistemas operativos para computadoras personales hay algunos que reúnen una serie de ventajas (y de inconvenientes), pero que, por la razón que sea, casi nadie usa, como el OS/2 de la casa IBM, o algunas versiones comerciales de UNIX para PC (p.ej. el SCO). Hay también diversas versiones del DOS (Microsoft no tiene la

exclusiva: están el PC-DOS de la IBM, el DR-DOS de Caldera, versiones gratuitas como el FREE-DOS, etc). Aunque voy a centrar la discusión en torno a dos grandes alternativas —el Linux y el Windows de Microsoft—, buena parte de tales consideraciones valen también si, entre las alternativas, barajamos a algunos de tales sistemas (valen en unos casos a favor de los mismos y en otros casos en contra, según).

Ciñendo, pues, nuestro debate a dos grandes alternativas (el Linux y el Windows de Microsoft, que a su vez se divide en DOS+Windows3.1, Windows95, Windows98, Windows-NT), voy a ofrecer dos grandes grupos de argumentos a favor del Linux: un primer grupo de argumentos que podemos llamar ‘egoístas’ y un segundo grupo de argumentos ‘altruistas’.



Argumentos que miran al provecho propio

1º.— Mientras que Windows es de pago, el Linux es gratuito.

Tal aserto ha de matizarse: uno puede comprar una distribución del Linux a una casa comercial por el dinero que quiera hacer pagar esa casa (justamente porque el Linux es lo que técnicamente se denomina ‘*free software*’ a cualquier casa comercial le es lícito hacer una distribución del mismo con fines lucrativos y por el precio que le dé la gana).¹

Mas, a diferencia del *software* no-libre (como el Windows de Microsoft), el Linux —y en general todo el *software* libre— sólo puede venderse en condiciones que no restrinjan los derechos del comprador. El vendedor no vende ni el *software* ni siquiera el derecho de uso del *software* (esto último es lo que se vende y se compra en el caso del *software* no-libre), sino un objeto material en el cual se ha reproducido el *software* en cuestión junto con lo que la casa comercial añade.

La principal diferencia práctica estriba en que, si —p.ej.— Ud compra Windows, ni está comprando (claro) el Windows (que sigue siendo propiedad de Microsoft) ni meramente un CD-Rom en el cual se haya estampado una copia del Windows, sino que está adquiriendo un derecho de uso del Windows en las condiciones que fije Microsoft, condiciones que, en el *software* no-libre en general, son éstas: Ud no puede averiguar el código fuente del programa; no puede usarlo más que en una sola máquina; no puede copiarlo ni prestarlo a un amigo para que lo copie.

En cambio, si adquiere Ud *software* libre (como el Linux), la casa comercial sólo le ha vendido un soporte físico, por el precio que hayan acordado entre ella y Ud, mas no ha restringido para nada los ulteriores derechos de Ud con relación al *software* libre

¹. Por si puede servirle de algo al lector, reproduzco aquí de una revista especializada esta lista (parcial e incompleta) de distribuidores comerciales en España (ésta, si es publicidad, es gratuita; mas no puedo garantizar lo verídico de la información): ABC Analog, <www.abcnet.es>; Kender, <www.kender.es>; Encis-Net, <www.encis.es>.

estampado en ese soporte: Ud puede averiguar el código fuente (es más: normalmente resulta obligatorio que se incluya tal código en la distribución); puede usarlo en tantas máquinas como quiera, copiarlo, regalarlo, y hasta modificarlo y vender las copias, modificadas o no (con la única excepción de que no puede restringir los derechos de terceros —o sea: será nulo cualquier contrato que Ud les haga firmar y que restrinja su derecho a usar el *software* con el mismo margen de libertad de que ha disfrutado Ud).

De todas esas ventajas del *software* libre para el usuario terminal, la principal, sin embargo, es el precio. Aunque un distribuidor comercial del Linux puede poner a sus discos estampados el precio que le dé la gana, nadie se los compraría si les pusiera un precio alto, ya que el Linux se consigue por cero P_t sin más que bajárselo uno de cualquiera de los sitios de acceso gratuito donde está colocado en el internet.

Así, mientras que puede costar mucho dinero una nueva versión del Windows de Microsoft (especialmente de la gama alta, Windows NT), las distribuciones comerciales del Linux suelen ser baratitas.

Y están las distribuciones no comerciales, tanto aquellas de las que se encargan asociaciones sin ánimo de lucro (que no hacen pagar casi nada más que el coste del material y una tarifa módica por costos de funcionamiento) cuanto las que están libremente accesibles en el internet.

Pero, en segundo lugar, eso de la gratuidad del Linux —frente al costo del *software propietario* (entendiendo por tal un *software* poseído, en propiedad, por una gran compañía con ánimo de lucro)— significa que, al adquirir el Linux, no está Ud contrayendo ningún compromiso de usarlo o no-usarlo, mientras que, cuando Ud compra [el derecho de usar] un sistema operativo propiedad de una multinacional, está firmando un contrato que le prohíbe una serie de cosas ya enumeradas —entre ellas copiar el disco y usar el programa en más de una computadora a la vez; y seguramente también (¡léase bien el contrato de compraventa, incluida la letra menuda!) vendrá Ud obligado a borrar o destruir versiones viejas si la nueva la adquiere a título de actualización. No invento nada. Antes bien, cito textualmente un extracto de uno de esos «contratos de licencia de uso» que le endosan a uno cuando —como vulgarmente se dice— se «compra» uno un paquete de soft propietario (en este caso un sistema operativo). (Omito sólo el nombre de la multinacional en cuestión), sin que naturalmente haya tenido ocasión de negociar tal «contrato»):

Nuevas versiones: Si Vds han adquirido el programa contenido en este paquete para actualizar una versión anterior del mismo programa u otro relacionado con él, contenido en un paquete adquirido por Vds en ocasión anterior, la presente versión actualizada será la única que tienen derecho a utilizar, y (a menos que reciban otra autorización de ...) tendrán que destruir o cancelar cualquier material del programa y documentación relacionada, incluidas todas las copias y modificaciones referentes al paquete anterior.

Con lo cual, si respeta Ud ese contrato, al instalar la nueva versión sobre la vieja, le está prohibido conservar o regalar los disquetes de ésta última (o tirarlos a la

basura sin haberlos previamente triturado); así que viene Ud obligado a quemar las naves irreparablemente; de suerte que, aunque luego no se acostumbre Ud a la nueva versión —o se dé cuenta de que ésta no reconoce su peculiar *hardware*, o que no se adapta a sus necesidades, o que tiene *bugs* o incompatibilidades con unos u otros programas que Ud estuviera ya usando—, así y todo ya no podrá volverse atrás. Si quiere retornar a la versión anterior, habrá de pagar nuevamente por ella si es que está todavía disponible en el mercado. Si no, aguantarse y resignarse.

Y no crea que se atiene Ud a la ley si destruye los disquetes pero conserva la documentación —llevado a ello tal vez por la insuficiencia de documentación de la nueva versión, o por la pésima traducción de la misma. También es delictivo —y por lo tanto punible— guardar vieja documentación, o tirarla (sin haberla destrozado) a un contenedor de papeles.

Igualmente, si instala una nueva versión de Windows en su computadora doméstica y luego (usando el mismo soporte) en la del trabajo (porque en ésta lo que le ha puesto su empleador es una versión ya obsoleta), sepa que está violando la ley y puede venir demandado y hasta perseguido por la policía y los jueces. Y no crea que sería el primer caso, ni que su condición modesta lo pone a salvo de esos avatares. Un vecino con el que se haya peleado por las basuras podría, en venganza, denunciarlo. Y de hecho hay pobres muchachos encarcelados por acusaciones de «piratería», que es como denominan los mercaderes de *software* no-libre a la copia de sus mercancías.

2º.— El Linux es UNIX. En estricto rigor, eso no es del todo verdad. UNIX es una familia de sistemas operativos para computadoras grandes (no personales), y es una marca registrada. Hay una serie de variantes del UNIX que se usan en grandes empresas, en centros de investigación científica, en Universidades, etc. Y, por razones legales, el Linux no pudo surgir diciendo que era UNIX. Así que surgió como un no-UNIX que, a todos los efectos, es [como si fuera] UNIX. ¡Vamos, que, para lo práctico, es UNIX!

Su estructura interna o subyacente no será UNIX, pero lo que presenta al usuario es indistinguible del UNIX. Eso significa que, si aprende Ud a manejar el Linux, ha aprendido a manejar (un) UNIX, y viceversa (salvo que, en este último paso, tendrá que aprender una pequeña adaptación complementaria).

Bueno, bien, y ¿qué ventaja tiene que sea UNIX? Tiene la ventaja de que UNIX es lo que se emplea en las grandes redes, en los grandes centros de investigación, en las Universidades. Aunque ahora Microsoft trata de irle comiendo terreno al UNIX también en ese ámbito y aunque ha registrado triunfos en esa guerra mercantil, así y todo el mundo de las redes de comunicación y de las grandes instalaciones sigue siendo todavía un mundo UNIX. Y nadie sabe cuál será el desenlace de la pelea. Si Microsoft parecía llevar las de ganar, antes o después, por la superioridad de sus medios y la astucia y decisión de sus tácticas, tal vez le esté empezando a pasar aquello de que la avaricia

rompe el saco. Muchos imperialismos han sido a la postre víctimas de su propio y ávido expansionismo.

3°.— Además de que usar el Linux conlleva familiarizarse con UNIX (lo cual para muchísimos usuarios puede ser en sí indiferente porque no van a relacionarse directamente con ese mundo informático-electrónico), el UNIX en general (y, por lo tanto, también el Linux en particular) posee ventajas propias que lo hacen superior no sólo a los sistemas operativos uni-tarea (el DOS) sino también a otros sistemas operativos multi-tarea (Windows 95, 98, NT; OS/2; etc).

3°.1.— Lo primero que hay que señalar en este punto es que el Linux hace muy bien la multitarea, mientras que algunas versiones de Windows hacen, sí, multitarea, pero mal, o mucho peor que el Linux. Así Windows 3.1 y Windows 95 apenas son genuinos sistemas multitarea: cuando un usuario lanza una aplicación teniendo abiertas otras, éstas se quedan medio congeladas. Probablemente Windows NT (¿tal vez Windows 98?) sea en eso mejor (el autor de este artículo no lo ha probado). Aunque no sin fallos, el Linux cumple bastante bien esa función de multitarea.

3°.2.— Pero, en segundo lugar, las ventajas del UNIX no se limitan a eso de la multitarea, sino que incluyen los tres siguientes factores.

[1] UNIX (y por lo tanto el Linux) asignan a cada fichero un dueño, un grupo de usuarios (siendo el dueño del fichero miembro de tal grupo) y unos permisos de lectura, escritura y ejecución diferenciados para el dueño, el grupo y los demás. Eso hace que UNIX (y el Linux) ponga una barandilla que evita en muchos casos los desastres o percances. En el Linux los virus difícilmente pueden surgir y pulular. No digo (como dicen otros) que sea imposible que surjan, mas lo seguro es que ese sistema de permisos interpone un obstáculo nada fácil de franquear contra la diseminación de los virus.² También el sistema de permisos salvaguarda la privacidad de los ficheros de cada usuario y evita que los errores de un usuario novel o inexperto se salden en destrucción de datos de otros usuarios.

Desde luego todo eso tiene sus límites. Si los errores los comete el administrador de la máquina (el superusuario o 'root' [raíz]), entonces el sistema de permisos no servirá de protección. Pero es más frecuente que los errores los cometan otros usuarios. Si Ud compra un *Pentium II* que van a usar Ud mismo más su compañero/a, su hermano y su sobrino —alternativamente—, esperemos que Ud, como superusuario, sea más cauto

² La verdad es que existen sospechas —difundidas y seguramente fundadas, al menos en parte— de que los principales originadores o propagadores de virus son las grandes empresas multinacionales de *software* propietario, las cuales: (1º) los han introducido en sus discos CD y disquetes de modo que salten a las computadoras en las que se proceda a una copia de tal producto *soft* (nada «soft» en este caso, de ser eso cierto) cuando se trate de una copia n+1 (será la segunda, o la tercera o la que sea); y (2º) los han propagado para que los usuarios tengan pánico a las copias no autorizadas que dizque esparcirían los virus (cuando justamente eso no está demostrado, mas funciona en el plano de la ideología y la propaganda de esas casas comerciales). En el Linux no puede haber ningún motivo de esa índole, y efectivamente el hecho es que no ha habido ninguna epidemia vírica.

y más experto; los errores o las imprudencias de los otros usuarios no provocarán el descalabro que causarían con el Windows.

[2] Una segunda ventaja del UNIX (y por lo tanto del Linux) estriba en la flexibilidad de configuración. En el DOS, el OS/2 y el Windows la estructura de ficheros es ésta: cada fichero tiene un solo nombre y está en un solo directorio; cada directorio es a lo sumo subdirectorio de un directorio superior; cada directorio que no sea subdirectorio es una división de un conducto (*drive*) o unidad, que es un disco o una partición de disco; los conductos o unidades son: A:, B:, C:, D:, ... En cambio, en el UNIX existe un procedimiento que es el de los *links* o enlaces, por el cual, a todos los efectos, un fichero puede tener varios nombres y estar en varios directorios; un directorio puede ser subdirectorio de varios directorios; eliminase en cambio la noción de conducto o unidad: cualquier paquete de ficheros —tenga el soporte físico que tenga, sea un CDROM, un disquete o lo que fuere— se «monta» como un directorio.

Verdad es que el DOS —y en menor medida el OS/2 y el Windows— trataron de paliar la rigidez del sistema de nombre único y ubicación única con las órdenes ‘join’, ‘assign’, ‘subst’, ‘append’. Eran arreglijos que no siempre podían funcionar, que a menudo resultaban incompatibles con nuevas implementaciones (sobre todo en el Windows) y que nunca solucionaban lo esencial.

En cambio, supongamos que usa Ud el WordPerfect 5.1 y tiene unas macros en las que le va a buscar sistemáticamente el programa las plantillas de fusión en un directorio «c:\Uriel\hobbies\recursos\plantill». Reorganiza todo (compra una nueva computadora, respalda y restaura el disco duro, reparticiona, introduce nuevos directorios, hace hueco para las cosas de su hermana etc). El resultado es que tales plantillas están ahora en el directorio «e:\hobbies\recursos\plantill». Tendrá que rehacer sus macros, lo cual es tarea ardua. O tendrá que hacer asignaciones con las órdenes ‘subst’ etc (si se lo permite el sistema operativo, porque en eso el primitivo DOS era mejor). O bien use el Linux y tendrá todas las opciones que Ud quiera (gracias a la flexibilidad del propio Linux cuanto a la que en esto añade el programa Linux de emulación de terminales DOS, a saber el DOSEMU). Así, entre otras, podrá poner sus plantillas en

```
/home/Uriel/cosas_particulares/distracciones/hobbies/recursos/plantillas
```

y llamar también a ese directorio de otro modo; y podrá, en su emulador DOS, usar 24 letras de conducto para llamar a sendos directorios. El resultado es que, con unos pequeños toquecitos en ficheros ejecutables de texto ASCII —transparentes y cuyo manejo no requiere ningún conocimiento de informática—, podrá Ud dejar en paz las macros del WordPerfect, el cual seguirá buscando y hallando las plantillas en lo que para él seguirá siendo «c:\Uriel\hobbies\recursos\plantill».

[3] Una tercera ventaja del UNIX es que restringe menos que otros sistemas operativos la libertad de llamar a los ficheros y directorios como a uno le dé la gana. Está claro que alguna restricción tiene que haber (por razones sintácticas), pero a uno

suele gustarle que sean las menos posibles. El DOS impuso la absurda restricción 8.3 (cada nombre de fichero tenía un máximo de ocho caracteres de nombre propiamente dicho más un máximo de tres caracteres de extensión separada del nombre por un punto); y encima había muchas restricciones en cuanto a qué caracteres podían usarse. El paso al OS/2 y al Windows95 y NT permitió dejar atrás el constreñimiento 8.3, pero en lo tocante a caracteres admisibles se instaló una situación oscilante entre el caos, la incompatibilidad de plataformas y lo imprevisible. Al optar por ANSI, en lugar de ASCII, el Windows dio lugar a serios problemas con nombres de fichero que tuvieran caracteres acentuados u otros signos que no fueran ASCII estándar. Aunque el Linux no soluciona satisfactoriamente todos esos problemas ni garantiza perfecta compatibilidad (lo cual es imposible por otro lado), sí concede más libertad, al menos para que cada uno haga de su capa un sayo (en su computadora). Puede Ud llamar a un fichero así:

```
Mi4ªcarta_de_amor(la_que_escribí_con_más[ansiosas]expectativas):a_MªJesús;borr.2º.a.1
```

Por poder, puede hasta poner retornos de línea y saltos de pág^a en nombres de fichero; y usar en tales nombres símbolos reservados, como el asterisco. (No resulta aconsejable, porque hará que algunos programas se vuelvan locos con eso.) Como a nadie obligan, puede Ud seguir usando nombres como ‘MaJesus.car’ (y, gracias a los *links*, llamar a la carta de ambos modos, o ponerla, con diferente nombre, en dos directorios).

4º.— Otra ventaja del Linux respecto de sistemas operativos de propiedad de empresas con ánimo de lucro (como Windows de Microsoft) es que el Linux viene, en cualquier distribución, acompañado de un enorme cúmulo de utilidades. Los equivalentes a esas utilidades para Windows etc puede que estén disponibles, puede que no. Habría que ver en cada caso. Habiendo sido durante años usuario de uno de esos sistemas operativos, puedo asegurar que no siempre se hallan utilidades así, por más que uno busque.

Mas lo importante aquí es que, aunque existan tales utilidades, las firmas propietarias del sistema operativo no tienen interés en incorporarlas a la mercancía que venden. Primero porque no podría ser en las mismas condiciones (las utilidades son a menudo *software* libre, y en cualquier caso no son propiedad de la firma que posee el sistema operativo). En segundo lugar, porque, cuando esas utilidades sean *software* no-libre, la firma que vende derechos de uso del sistema operativo prefiere no mermar su propia ganancia pagando regalías a otras firmas.

El resultado es que (al menos hasta donde ha podido comprobar, por su experiencia, el autor de este artículo), cuando uno adquiere un derecho de uso de un sistema operativo no-libre no puede (mientras no se busque la vida por ahí) ni siquiera descomprimir archivos comprimidos .zip, ni visualizar ficheros de impresión .ps (PostScript) ni convertirlos a formatos imprimibles adecuados a la impresora propia, ni usar discos duros removibles o cintas de respaldo. Las distribuciones del Linux suelen

incluir muchas de esas utilidades, y cada vez incluyen más (cada uno instala lo que quiere, claro).

Al tratarse en general de *software* libre, no hay razones mercantiles que restrinjan su uso; antes bien, cada distribuidor (más si quiere vender su distribución) querrá incorporar más virguerías que hagan la compra de su soporte más ventajosa que la de sus competidores.³

5º.— Una quinta ventaja del Linux respecto al Windows y a otros sistemas operativos similares estriba en la documentación disponible. A menudo el Windows y ese tipo de sistemas operativos que son poseídos por grandes empresas comerciales se ofrecen sin apenas documentación.

Las razones son varias. Una es, posiblemente, que la empresa transnacional dueña del *software* en cuestión no tiene interés en facilitar demasiadas pistas que permitan averiguar sus *intrínquilis*, limitándose a dar al usuario un mínimo de pautas (casi casi de obligado cumplimiento) para el uso y manejo del programa. Otra posible razón estriba en que frecuentemente grupos próximos (por interés, negocios o lo que sea) a tales empresas venden sus propios manuales, y que el propietario-distribuidor del sistema operativo no quiere hacerles competencia (él mismo vive de esa *claque*, de ese coro sin el cual poco se venderían sus mercancías). Otra razón puede estibar en la concepción opaca y reservada de los dueños de la empresa: ellos y sus programadores son los sabios, y al común de los mortales con pocas palabras y explicaciones basta (y esas palabras, pocas, suelen hallarse en las ayuditas «en línea»). Claro que no es de descartar una razón más, puramente crematística, y es que los manuales cuestan (aunque no costaría prácticamente nada un manual en fichero de texto que viniera en el propio CD-ROM, como sucede con el Linux; mas eso sería posiblemente un desdoro para Microsoft y cía, empresas de postín que no van a rebajarse a ofrecer un vulgar fichero de texto fácil de copiar y distribuir gratuitamente).

El resultado es que —por muy presuntamente amigables que sean esos sistemas operativos no-libres (cada uno contará la feria según le haya ido en ella, y a quien esto escribe esa amigabilidad del Windows le parece puramente legendaria)—, a la hora de habérselas con dificultades imprevistas no sabe uno qué hacer ni tiene documentación adonde acudir.

Con el Linux sucede casi lo opuesto (y puede que caiga en el otro extremo): puede uno siempre consultar miles de págs de documentación gratuita. El único problema es la escasez de tiempo; que, si no, uno acabaría resolviendo todos los problemas —a diferencia de lo vendido que se siente uno frente al hermetismo de Microsoft y demás empresas de ese estilo.

³. Reconocer eso no excluye la crítica a las distribuciones existentes. Están bien pero podrían y deberían estar mejor, incluyendo más piezas de *software*.

6°.— Además de que cada usuario del Linux tiene a su disposición un gran cúmulo de documentación gratuita, puede llamar en su auxilio a la comunidad linuxita, formada por muchos entusiastas, con probabilidades de que alguno le eche una mano. En cambio poca ayuda desinteresada (pero competente) puede uno encontrar para configurar el Windows. Y el soporte de las empresas es seguramente otro mito. Ya se sabe: puede uno llamar en tales horas durante tanto tiempo; lo intenta sin éxito y, cuando al fin se entabla la comunicación, ya se ha pasado el tiempo de soporte gratis, y hay que pagar la consulta (que, por otra parte, en general, se revela muy decepcionante). Con el agravante de que creía uno, equivocadamente, que disponía de una redcilla de seguridad.

7°.— Otra ventaja estriba en las actualizaciones. Al comprarle a una empresa de *software* el derecho de uso de un sistema operativo, Ud sólo adquiere ese derecho. A lo sumo podrá la empresa prometerle (en términos vagos) condiciones favorables para actualizaciones futuras. En cambio, al instalar un sistema operativo libre, como el Linux, Ud sabe que las versiones futuras seguirán siendo libres, y por lo tanto, aunque Ud haya comprado a la empresa tal su disco de distribución, puede luego actualizarse a futuras versiones directamente desde el internet, o pasándose a otra distribución. No está obligado a comprar de nuevo esa misma distribución determinada ni ninguna otra (aunque lo módico del precio hace que tampoco esa opción resulte prohibitiva).

8°.— Una ventaja del Linux sobre el Windows es que, mientras éste último sólo puede funcionar con un interfaz gráfico [GUI, *graphic user interface*] (sin el cual Windows 95 queda reducido al MS-DOS 7, perdiéndose la [ya escasa y problemática] multitarea que posibilita ese sistema), el Linux en principio funciona en modo texto, con consolas virtuales no-gráficas. Hay, sí, distribuciones del Linux orientadas al modo gráfico (p.ej. la de Red Hat, y en menor medida otras), lo cual las ha hecho muy populares, ya que esos menús gráfico-ventaneros son del gusto de bastantes usuarios, quienes piensan que facilitan el trabajo para los neófitos.

No sé cuánto haya de fundamento en eso. Lo que sé es que, como dice el refrán, de gustos no hay nada escrito (o sea: estipulado) —o más bien de gustos no cabe debatir.

A muchos les encantan los entornos gráficos. Para ellos Linux tiene entornos gráficos despampanantes, vistosos, cromos sensacionales. A otros nos gustan los entornos de modo carácter o texto (porque en éstos los caracteres son mucho más visibles y porque —con razón o sin ella— nos parecen más prácticos y rápidos para el trabajo efectivo). Quienes así sentimos podemos usar Linux en multi-tarea plena sin tener que correr ningún entorno gráfico.⁴

⁴. Hay determinadas tareas para las cuales a algunos (no sé si somos muchos o somos pocos) nos parece que el entorno gráfico hace perder demasiado tiempo sin que el resultado sea estéticamente mejor. Me explico. Para acceder a páginas *web* que se busquen únicamente por su contenido de información, un programa de acceso en modo texto —como el Lynx— permite abrir y leer las páginas solicitadas con una velocidad muchas veces mayor que la que se alcanza con un visualizador gráfico, sea el que fuere; y además el texto que así se lee queda muchísimo más bonito (si se tiene el Lynx para Linux configurado con

(Es posible —aunque no estoy seguro de que sea la explicación correcta— que sea por razones mercantiles por lo que han concedido un privilegio abusivo a los entornos gráficos las casas de *software* propietario. La razón —de ser cierta mi hipótesis [que, insisto, emito a título de conjetura nada más]— sería que eso vende más, no en el sentido de que los entornos de texto se vendan mal o carezcan de demanda, ni siquiera de clientes solventes, sino de que se saca más beneficio vendiendo sólo entornos gráficos, más caros, y obligando así a que, aunque sea a regañadientes, se pase al modo gráfico el usuario no adicto a tales entornos [por tener defectos de visión, como más de tantas dioptrías, o por inclinación a lo práctico y al resultado, o por lo que sea].)

No todo es cuestión de mero gusto en este asunto de cómo comunicar (si por gráficos o por mensajes lingüísticamente acuñados). Hay también un problema de eficacia. Según en qué contextos y para qué fines será mejor, o peor, utilizar un sistema de señales gráficas o un código de representación lingüística como el alfabético. En nuestra sociedad hemos pasado recientemente de un predominio del código lingüístico-alfabético a los códigos de señales gráficas (tal vez sobre todo por causa de los problemas que acarrea la circulación de vehículos automóviles).

Los sistemas de representaciones icónicas se han difundido también para trascender fronteras idiomáticas. (Si va Ud a Budapest y busca los baños, la palabra puede serle incomprensible, mientras que un dibujo puede darle más fácilmente una pauta.)

Sin embargo, sería menester un estudio científico para saber si —cuenta habida de todo— resulta mejor, para el trabajo con una computadora, una interfaz icónica o una lingüística. A favor de la primera cabe aducir su más fácil aprendizaje (al menos para quien no hable el idioma). A favor de una interfaz lingüística pueden alegarse consideraciones de rapidez y eficacia en el manejo. Es posible que haya personas para cuyas características resulte más idónea una interfaz o la otra.

Sea ello como fuere, es preferible dejar abiertas las opciones y no encerrarse irremediabilmente en una vía, como la del predominio de las señalizaciones icónicas, que tal vez a la postre no se revele el mejor camino.

9º.— La última de las razones egoístas que voy a invocar es una cuestión de calidad. Una serie de observaciones empíricas y de generalizaciones inductivas (falibles y tal vez no contrastadas en estudios científicamente rigurosos, pero no por ello desechables) llevan a la conclusión de que Windows 95 es un producto chapucero —seguramente por el ansia de copar pronto el mercado y adelantarse a la competencia,

colores adecuados), ya que el color destaca y atrae al ojo más que la figura (y en el Lynx hay muchas combinaciones posibles de colores para resaltar, de diferentes maneras, las diversas partes de un texto HTML). Para acceder a páginas *web* orientadas a la información, al contenido, el entorno gráfico nos parece un estorbo. Por supuesto, un programa de acceso gráfico está en su lugar para acceder a una pinacoteca virtual, p.ej.

desbancando especialmente a la IBM que no había sabido promocionar publicitariamente su OS/2—.

Así, p.ej., a pesar de que casi todo el *hardware* que compra uno para computadoras personales viene hoy con soporte de *software* para Windows 95 (y sólo para Windows 95, o a lo sumo para Windows NT también), a pesar de ello muchas veces el *soft* no es reconocido, el *hard* no se instala, y eso que está dizque «optimizado para Windows 95». ¿Qué pasará? ¿Por qué cada vez que entra uno se obstina el Windows 95 en sostener que la pantalla y tarjeta gráfica están mal configuradas, que el ratón no funciona bien, que hay que rebutear por enésima vez, que el disquete con soporte de *soft* no contiene nada etc etc?

Sin duda son fallos de programación que le pueden pasar a cualquiera. Los programadores de M\$ ni son más tontos que otros, ni es verosímil que trabajen menos bajo el látigo del patrón (quien se juega en eso muchísimos miles de millones de dólares). Pero nada prueba que el ser humano haga mejores cosas cuando trabaja en esas condiciones de emulación y presión. Hay estudios que parecen mostrar lo contrario:⁵ que algunos, o muchos, producen mejores cosas cuando no se juegan tanto en lo que están haciendo, sino que lo hacen por mor de la calidad de su producción como un fin en sí mismo. Y el Linux es principalmente producido, sin ánimo de lucro, por universitarios cuyos emolumentos no van a cambiar automáticamente porque el producto sea mejor o peor, ni, todavía menos, porque sea mejor o peor aceptado en el mercado.⁶

Teniendo en cuenta que al Windows se lo dan todo hecho, y al Linux nada, no resulta arriesgada ni peregrina la conjetura de que el Linux es un producto de más calidad.⁷



⁵. Véase el Anejo Nº 1 de este artículo: «Studies Find Reward Often No Motivator: Creativity and Intrinsic Interest Diminish if Task is Done for Gain» por Alfie Kohn, reproducido del *Boston Globe*.

⁶. Véase el suelto «Studies Find Reward Often No Motivator» por Alfie Kohn, que se reproduce como anejo al presente artículo. Desde luego esos estudios no son concluyentes y puede que ni ahora ni nunca quepa decir la última palabra. Pero lo que es seguro es que no siempre los incentivos producen mejores resultados que el trabajo en aras del producto o de la labor en sí; aún más seguro es que no siempre los incentivos económicos son los más eficaces. Muchas de las mejores creaciones de la mente humana (la filosofía de Platón o la de Leibniz, la música de Vivaldi, la novela de George Eliot o la de Dostoievski, la poesía de Espronceda) no existirían si sus autores hubieran orientado básicamente sus vidas según el principio de hacer lo que más beneficio económico les reportara.

⁷. Claro que, entre tanto, ya está ahí el Windows 98; sobre él hay opiniones encontradas, desde las de quienes insisten en que lo han probado y es una «patata» hasta las de quienes aducen que se han superado los fallos que tuvo en su lanzamiento y que no es malo el producto resultante. La verdad, no lo sé. Ni tengo ganas de averiguarlo tras mi experiencia con el tan aclamado Windows 95. Y es que, aun suponiendo que, efectivamente, Microsoft hubiera, ¡al fin!, ofrecido un producto de calidad aceptable, quedan en pie todas las otras razones aquí aducidas.

Argumentos independientes del provecho propio

10°.— Lo malo de que prevalezca en demasía el *software* propietario es que con ello se cercenan las posibilidades de desarrollo de alternativas. Por los mecanismos usuales del mercado (el juego «limpio» y el menos limpio), se van formando oligopolios, y se cierran los mercados a cuanto salga del patrón impuesto por los oligopolistas. Ahora bien, la experiencia muestra que nunca hay demasiada imaginación, nunca demasiada inventiva, nunca demasiada iniciativa. Aun pagando el precio de disonancias, discordancias, disparidades, es mejor para el progreso humano que pululen y proliferen implementaciones alternativas a fin de que la sociedad tenga nuevas opciones abiertas.

De entre los oligopolios o casi-monopolios a pocos les estará sobrando el ‘casi’ tanto como a Microsoft, cuyo Windows (en una u otra variante) es utilizado hoy por más del 90% de los usuarios de *PC*'s. Si uno quiere promover diversidades estimulantes, ayudar a que se abran vías alternativas, evitar el callejón de una opción única, lo mejor es secundar a alternativas menos estándar. Pero, para que sea eficaz tal esfuerzo, hay que no dedicarse a secundar algo que parezca definitivamente arrinconado, obsoleto o fracasado. Y, descartados el producto monopolístico más los que parecen tener a lo sumo pasado (no futuro), lo que queda es el Linux.

11°.— Cada uno es muy dueño de estar a favor del actual sistema socio-económico (el capitalismo) o en contra del mismo (a favor de alguna estructura de tipo colectivista).

Ahora bien, si se está en contra del sistema, seguramente será porque uno piense que es injusto —y concretamente porque se juzgue injusta la distribución de la riqueza (o más bien su falta de distribución). De ser así, no parece muy razonable contribuir a agravar la desigualdad de riquezas, contribuir a que el más rico del planeta sea todavía más rico.

Claro que uno puede, como principio, abogar por un sistema de distribución de riquezas radicalmente distinto del actual, pero a la vez sostener que —dentro del actual sistema injusto— da lo mismo que el margen de desigualdad sea un poco mayor o un poco menor; o, dicho de otro modo, que, si bien es injusto que exista la actual desigualdad en la distribución de riquezas —el abismo entre ricos y pobres—, dado ese sistema da lo mismo que la distancia entre el más rico y el más pobre sea un factor multiplicativo de 9^9 unidades o que sea de 9^9+1 .

Eso se puede sostener y seguramente no es lógicamente incongruente. Sin embargo, ¿cuál es entonces la razón que lo lleva a uno a repudiar el actual sistema socio-económico y su distribución? Pueden alegarse varias cosas. Una es que tal sistema no satisface las necesidades básicas de la mayoría de la población del globo. Sin embargo, esa razón por sí sola difícilmente lleva a abogar por una alternativa distributiva

radical, a menos que se demuestre que el actual patrón distributivo es incompatible con la toma de medidas que alivien la situación de los más desfavorecidos.

En general, sin embargo, esa razón en sí no constituye una motivación igualitaria para repudiar el actual sistema, ni por lo tanto acarrea forzosamente, de suyo, un rechazo radical de ese sistema. Lo que sí constituye un motivo para repudiar el sistema (con o sin paliativos) es una posición que articula uno u otro tipo de igualitarismo.

Mas, aceptado el igualitarismo, acéptase la tesis de que es mala la desigualdad económica, al menos aquella que supere cierta cota o cierto umbral. Si eso es así, el único modo concebible para sostener que da lo mismo que la desigualdad económica máxima sea un poco mayor o un poco menor es afirmar que, rebasado cierto umbral, la desigualdad es mala sin importar ya grados; y que el aumento marginal de desigualdad ya no incrementa el grado de maldad de la desigualdad resultante. Sino que es moralmente indiferente.

Mas eso parece escasamente razonable. Si es mala la desigualdad (o la desigualdad excesiva), entonces lo único que puede hacer que una desigualdad de $n+1$ no sea peor que una de n es que haya una raya, una línea de demarcación tal que, más acá de esa línea, no se dé (en absoluto, tal vez) maldad en la desigualdad; pero más allá la maldad sea total y absoluta. Y eso es increíble. Es como si hubiera una línea tajante de demarcación en la resistencia física tal que el deportista que la rebase, por poco que sea, sea un deportista totalmente bueno, al paso que el que se quede un poquitín por debajo (por poco que sea) sea un deportista totalmente malo. O que el alumno que tiene una media de 5,001 sea óptimo, mientras que el que obtiene una media de 4,999 sea pésimo.

Descartado ese antigradualismo difícilmente creíble del todo o nada, lo normal será pensar que una mayor desigualdad es peor que otra menor. Y, por consiguiente, que es malo incrementarla. Y que, por lo tanto, es malo contribuir a que el más rico del planeta sea todavía más rico, y esté a más distancia del más pobre. ¡Que por uno no quede!

12°.— Si es uno partidario del sistema capitalista, seguramente verá en éste ventajas de éstas que Adam Smith atribuía al mercado —el cual, como una mano invisible, haría que, a la postre, el egoísmo de cada quien resulte en el bien y la prosperidad generales. No entra en mis propósitos debatir aquí de esa paradoja, que juzgo muy equivocada. Lo que sí creo oportuno decir al respecto es que, si uno atribuye esas virtudes al capitalismo, a la economía de mercado, entonces parece razonable que favorezca medidas antimonopolistas. Y una medida antimonopolista hoy es que haya un poco de variedad y competencia en el mercado de los sistemas operativos para *PC*'s. De las cosas que puede uno hacer para lograrlo, la que, a primer vista por lo menos, parece menos distante de conseguir algún resultado es optar por Linux en lugar de Windows.

13º.— Otro motivo no-egoísta para no colaborar al mayor poderío de los oligopolios del *software* es protestar contra la represión que sufren los mal llamados ‘piratas’:⁸ jóvenes que han comprado un paquete de *software* pero que lo han usado en más de una máquina, o que han recibido de un amigo unos disquetes y han probado su contenido, y que ahora están en presidio por eso como si hubieran matado a alguien. Sabemos quiénes no están en la cárcel habiendo hecho lo que han hecho.⁹

En realidad eso de la piratería es un cuento absurdo.¹⁰ Los contratos que se firman son inicuos y en su negociación una de las partes es impotente y la otra

⁸. El *Diccionario jurídico* de Raymond Guillien & Jean Vincent (trad. español, Temis, Bogotá 1996) da esta definición: ‘**Pirata**. Individuo fuera de la ley, que, al contrario del corsario, no tenía carta de contramarca y se entregaba, por su propia cuenta, a acciones de pillaje en el mar.’ ‘**Piratería**. *Der. Pen.* Delito que consiste, respecto de toda persona que se encuentra a bordo de una aeronave en vuelo, de un buque en el mar o de cualquier otro medio de transporte colectivo, en apoderarse de ese medio o en ejercer su control por medio de violencia o de amenaza de violencia (C.P., art. 462)’. ¿Qué tiene que ver la piratería con un uso u otro de ciertos productos materiales que uno no ha robado —ni con violencia ni sin ella— [p.ej. una colección de disquetes] o, si se quiere, de las ideas plasmadas o grabadas en esos productos? Puede la ley prohibir ciertos usos de los productos adquiridos, limitando así el derecho de propiedad material; puede hasta prohibir que uno haga ciertas operaciones intelectuales o informáticas si ha comprado tales productos o incluso si sólo los ha comprado otra persona que luego se los ha prestado a uno; por poder, puede prohibir que uno se levante antes de las 7 de la mañana salvo con autorización del alcalde. Es abusivo. Y cabe alegar lo oscuro y problemático de la noción de «uso» de un bien intelectual. J. Miguel Rodríguez Tapia & Fernando Bondía Román, en sus *Comentarios a la Ley de Propiedad Intelectual* (Civitas, Madrid 1997, p.371), señalan a este respecto: «... Se consagra, pues, [en la ley] un nuevo derecho (*rectius* facultad) en materia de programas de ordenador: el derecho de uso. Derecho que, en sentido estricto, no es propiamente un derecho de autor. La noción de derecho de uso es de por sí algo extraño a la propiedad intelectual pues el autor de cualquier otra obra intelectual protegida no puede impedir al adquiriente del soporte en el que está incorporada la obra que la use o utilice.» Sin embargo peor incluso que eso es que se tilde de ‘piratescas’ a actuaciones que no conllevan violencia ni sobre las personas ni sobre las cosas, que no comportan ningún hurtar o sustraer, que sólo consisten en un uso que la ley ha dado en prohibir, aunque esa prohibición no se siga para nada de la doctrina jurídica tradicional. No se usan las palabras al buen tuntún. Su empleo no es anodino. No hemos de acostumbrarnos a llamar ‘piratas’ a los usuarios cuyo uso infrinja alguno de los privilegios que otorga a los propietarios el vigente —y contingente— texto legal. Porque, haciéndolo, estamos empezando a justificar la represión, y el clima policiaco que atizan las grandes multinacionales del *software*, con la obsequiosa colaboración de políticos, altos dignatarios y personeros de las fuerzas de vigilancia y orden público. El Código Penal (cito por su versión anterior, mas los cambios no son de fondo) dispone en su art. 126: «En caso de infracción o cuando exista temor racional y fundado de que ésta va a producirse de modo inminente, la autoridad judicial podrá decretar, a instancia de los titulares de los derechos reconocidos en esta ley, las medidas cautelares que, según las circunstancias, fuesen necesarias para la protección urgente de tales derechos...» Al amparo de lo cual se han creado brigadas policiales para la represión preventiva del «pirateo». No está nadie a salvo de que, esgrimiendo un decreto judicial, vengan, se lo pongan a uno todo patas arriba, se incauten cautelarmente de su computadora para ver si hay en ella una copia ilegal de un *software*, aunque a la postre —tras haber sido copiado el contenido del disco duro— uno acabe obteniendo la restitución. Mas, si es uno justamente un inventor, ya se ha quedado sin sus secretos de inventor. No se los devolverán.

⁹. Hasta se han creado unidades especiales para la represión del denominado ‘delito informático’. En la abrumadora mayoría de los casos éste sólo consiste en, o bien copiar programas de *software* comercial (sin robarles ni quitarles nada a sus propietarios), o bien husmear —sin causar daño alguno— en los vericuetos de la red planetaria de telecomunicaciones (y, de haber daño, sería únicamente la pérdida de algún secreto armamentístico, siendo jugoso que un país que se dice independiente se ponga a disposición de potencias extranjeras para reprimir a súbditos suyos que se hayan entregado a ese devaneo de curiosidad sin ánimo de explotación militar ni económica). No tengo estadísticas del número de jóvenes que han pasado por los calabozos de esas fuerzas represivas, pero todos hemos leído una serie de noticias al respecto.

¹⁰. Es una lástima que, cuando se piden tantas despenalizaciones —incluyendo las de tráfico lucrativo de productos dañinos para la salud—, no se alcen voces para pedir la despenalización de uso o difusión de copias de información no-contractualmente autorizadas. La despenalización no acarrearía forzosamente su legitimación legal, igual que no es legítimo abstenerse de pagar una deuda. Simplemente han de ser asunto del derecho civil y no penal.

prepotente. Mas aun así: la ruptura de una condición contractual podría ser motivo para un juicio civil y hasta —si se demuestra empobrecimiento de la parte presuntamente lesionada— para el pago de una multa o indemnización. Lo que es absurdo y cruel es tratar eso como delito y someterlo al código penal.

Hoy juzgamos cruel e inhumano que en otro tiempo hubiera prisión por deudas; pero esto no es menos malo. Al revés, es todavía peor, porque el deudor había causado, al recibir el dinero, un empobrecimiento relativo del acreedor, al paso que el mal llamado «pirata» sólo ha causado (suponiendo a mucho suponer) un no-enriquecimiento del propietario del *software* en cuestión. Y no-enriquecimiento no es empobrecimiento. El que el patrimonio de uno no se incremente no significa que disminuya.

Las grandes empresas de *software* (y en primer lugar M\$) no dudan en utilizar los aparatos coercitivos de los Estados, sus pistolas y sus rejas, para reprimir y encerrar a los disidentes del sistema que —sin hacerle mal a nadie— corren en sus computadoras programas de diseño, bases de datos, hojas de cálculo, lo que sea, y cuya única falta es que no han enriquecido aún más al más rico del mundo (o a otro que no le ande tan lejos).

Creo que es sensato que aquellos a quienes eso parezca injusto lo manifiesten colaborando —sin salirse del marco legal para nada— a que prosperen alternativas viables frente al *software* que simboliza la prepotencia de esos magnates tan implacables.



Objeciones

1^a.— Es insignificante lo que uno puede contribuir a un resultado así. Será verdad eso de que un grano, sin hacer granero, ayuda al compañero. Pero no lo es menos que, para mí, usuario individual, una decisión así es importante (tengo que hacer cambios en mi manejo de las máquinas, en mis costumbres, etc), al paso que es insignificante el efecto indirecto de mi opción particular.

Respuesta.— Razonando así nunca votaría uno. El peso que tiene un votante indio en el resultado político es como del orden de 0,0000003 %. El que tiene un comprador de Windows en el incremento del poderío de M\$ es muchísimo mayor.

Además, puede uno no sólo abstenerse de comprar un producto y adquirir (¡gratis!) un producto rival, sino también expresar a otros sus razones (sin necesidad de usar lenguaje ofensivo ni incurrir en alharacas o aspavientos). 5 ó 10 por acá, otros tantos por allá, eso va creando un ambiente. Minoritario, sí, mas no desdeñable. (De hecho ya hay compañías contrincantes de M\$ que se han decidido a portar al Linux su *software*, como Caldera, Netscape y Corel. Ya se sabe: el enemigo de mi enemigo es mi amigo.)

2ª.— Cualquiera que sea la importancia de estos asuntos para quienes viven en eso o de eso, no vale la pena, para quienes tienen preocupaciones mayores, perder tiempo en tales bagatelas.

Respuesta.— ¡Tal vez sea así! Pero, por las mismas, uno empeñado en preocupaciones o tareas más altas no perdería nunca tiempo haciendo el amor, dando de comer a los patos junto al estanque, llevando a sus niños a pasear, montando en bicicleta, nadando o mirando una película. Está bien que algunos estén tan enfrascados en actividades valiosas y hasta sublimes que no les quede tiempo alguno para estas otras cosas más ramplonas y de menor vuelo. Otros, sin embargo, pueden dedicarles algún tiempito, aunque sea un poco a título de diversión o entretenimiento, mientras descansan un momento de quehaceres más arduos e importantes.

3ª.— La opción por o contra un sistema operativo o un tipo de *software* (propietario o libre) no tiene nada que ver con cuestiones sociales, con la alternativa entre colectivismo y capitalismo o individualismo.

Respuesta.— Es verdad que cabe, por una u otra razón, preferir el *software* propietario aun siendo uno partidario de un sistema colectivista, y preferir el *software* libre aun si uno ama al capitalismo (entre otras cosas tal será el sentir de quien se lucre distribuyendo *software* libre —mientras que se arruinaría si se acabara de imponer monopolísticamente el *software* propietario). Pero parece incongruente que alguien abogue por una sociedad sin propiedad privada de bienes materiales y piense que, no obstante, en tal sociedad haya de haber *software* propietario, en el sentido apuntado. Podrá creer que, aun sin propiedad privada, los autores de un producto intelectual tienen ciertos derechos especiales sobre el uso de tal producto. Mas esos derechos no podrán ser los de propiedad material sobre el producto, ni sobre su soporte, ni sobre los bienes materiales ganados gracias a ese producto. Tal vez será un derecho a cierta recompensa, del tipo que sea, por haber hecho una contribución especial al bien de la sociedad, mas eso es harina de otro costal.

4ª.— Todos esos alegatos anti-Windows son los de los fracasados. Windows ha triunfado en buena liz. Por algo será. Nadie es más sabio que el comprador, ya que cien millones de compradores no pueden equivocarse.

Respuesta.— Desgraciadamente pueden equivocarse cien millones de compradores, y aun doscientos. Aunque el proceso que ha llevado a la hegemonía de Windows hubiera sido menos objetable (en lo tocante al recurso a propaganda sesgada, bombo y platillo, prácticas mercantiles cuestionables, etc), ningún indicio serio permite aseverar que lo que se impone en el mercado es siempre lo mejor. Son infinitamente complejos los resortes de la acción humana, siendo uno de ellos el efecto bola de nieve («¿Dónde vas Vicente?»). Una razón que puede llevar a A a comprar X es el mero hecho de que A piense que B, C, D, etc, van a comprar la mercancía X (máxime si eso repercute de algún modo en las opciones que tenga A en el caso de comprar otra mercancía a la hora

de hacer intercambios con B, C, D, etc.). Y todavía más que A piense que B, C etc piensan que A va a comprar X y van a ajustar sus actuaciones y opciones a tal expectativa. Una vez desencadenada una serie de tales mecanismos, poco influye la calidad del producto, que muchas veces el comprador no tiene posibilidad de comprobar (máxime si ni siquiera tiene conocimiento de productos alternativos).

5^a.— ¿No es algo incongruente que un *software* producido, sin ánimo de lucro, por los programadores que lo inventan sea utilizado para el lucro de empresas que venden distribuciones comerciales del mismo, como pasa con el Linux?

Respuesta.— No, no es incongruente. Estamos en una sociedad capitalista, en una economía de mercado. Quienes producen *software* sin ánimo de lucro pueden imponer al uso y distribución de sus productos la condición de que no se haga con ánimo de lucro (tal es el caso del *software* gratuito usual, que uno puede usar y copiar pero no distribuir comercialmente, o sólo con la restricción de hacer pagar una suma muy módica).

Los originadores del complejo GNU-Linux han optado por otra opción, la del *software libre*, o sea uno con el cual uno puede hacer lo que quiera siempre y cuando el resultado de tal uso no restrinja derechos de terceros.

Esa opción es perfectamente razonable en el marco de esta sociedad. En primer lugar porque ofrece un aliciente a otros para, a cambio, sufragar el desarrollo ulterior de ese *software* libre (sin que eso lo transforme en *software* propietario en el sentido técnico, ya que el sufragante o subvencionador no poseerá derechos monopolísticos sobre el producto, mientras que Microsoft posee un derecho absoluto y exclusivo sobre cualquier desarrollo ulterior del Windows). En segundo lugar, entre el equipo de tales originadores de *software* unos tendrán unas opciones sociales y otros otras; la fórmula del *software* libre es neutral, acomodándose a las actuales estructuras sociales sin empero darles un aval o un respaldo, y sin contribuir al predominio de los más poderosos.

6^a.— En el mundo del Linux hay una pléyade de distribuidores. No hay un solo Linux sino Línuxes para todos los gustos. Eso causa un caos y lo deja a uno perplejo sin saber a qué carta quedarse ni por qué distribución decantarse.

Respuesta.— Es verdad que se da tal variedad y que acarrea algunas consecuencias desagradables. Mas no hay que exagerar. La mayor parte de los problemas y de las soluciones son comunes. Los Linuxitas hablan el mismo lenguaje informático. Hay bastante comunicación entre ellos. Y la variedad tiene también mucho de positivo: da un margen al gusto, la preferencia, lo cual tiene un aspecto estimulante. Lo de la perplejidad se resuelve fácilmente: entérese Ud un poco, otee, y luego sepa que cualquier distribución del Linux es mejor que ninguna.

7^a.— Con Linux no puede el usuario esperar el socorro de una empresa fuerte que le dé soporte en caso de avería o disfunción grave.

Respuesta.— Ya hablamos de eso más arriba. La experiencia de quien esto escribe lo hace muy escéptico respecto al socorro que quepa esperar de las grandes empresas electrónico-informáticas. Tal vez a otros les haya ido mejor en eso y sean más optimistas. En todo caso, y sea como fuere, las casas comerciales que distribuyen Linux ofrecen soporte en condiciones y a precios que desafían cualquier oferta tentadora de los mercaderes de *software* propietario. Y está el ancho mundo del ciberespacio linuxita, donde habitan muchas almas caritativas.

8ª.— El Linux es complicado y difícil.

Respuesta.— No es más complicado ni más difícil que el Windows, ni está menos documentado. Lo que pasa es que una tarea hecha entre dos es más fácil que hecha por uno solo. Y por tres más fácil. Y así sucesivamente. Como la mayoría usan Windows, suele ser más fácil obtener una ayudica de un vecino, hermano, proveedor, compañero de trabajo etc cuando uno usa el Windows. Además de que ya la computadora que uno compra vendrá normalmente con Windows. En eso, sólo en eso, estriba la facilidad. Y es un factor real, que explica muchas cosas, muchas opciones. Eso sí, cuando el Windows empieza a fallar y no reconoce la tarjeta de vídeo o el conducto de disco removible o la escrutadora; cuando los mensajes de error se repiten y hasta empieza a sonar una alarma que sólo se calma apagando la máquina; cuando no hay cómo solventar nada de eso con la documentación impresa adquirida con el sistema operativo; cuando esas cosas suceden, u otras por el estilo, ¡olvídese Ud del socorro del fabricante del producto, del del proveedor, del de sus colegas en el trabajo, del de su vecino! Está Ud solo ante el peligro y arrégleselas como pueda. O, si no, pásese al Linux.

9ª.— La sociedad no puede avanzar si no remunera a los inventores. El que tales productos sean marcas registradas, propiedad de ciertas empresas, es simplemente una implementación de ese principio de patentes, o sea: es la manera de asegurar que los inventores (en este caso los programadores) puedan beneficiarse de su invención y estén así estimulados —ellos y otros— para llevar a cabo nuevas invenciones.

Respuesta.— Con los actuales oligopolios del *software* (o de lo que sea) la situación es inversa. M\$ tiene un puñado de programadores en su nómina (serán muchos, mas constituyen una ínfima minoría de entre los programadores listos y capaces, sea en EE.UU o en el planeta Tierra). Lo inventado por esos programadores no es propiedad de ellos, sino de la empresa multinacional Microsoft.¹¹ Los beneficios de M\$ van a sus

¹¹. Es éste un tema para otro artículo que tratará (o trataría) de la propiedad intelectual del empleador sobre lo inventado por sus empleados. En el pasado el marido era propietario de los bienes intelectuales inventados, descubiertos o creados por su mujer; las primeras investigadoras científicas y mujeres de letras del siglo XIX y comienzos del XX no poseían ni co-poseían esos bienes intelectuales, propiedad de sus cónyuges, así no hubieran aportado nada. Hoy la ley hace que quien tenga la propiedad intelectual sea el empleador y no el empleado que crea un programa de computadora (incluyendo la documentación del mismo —lo cual, por extensión, puede llegar a ser casi toda la producción intelectual, ya que un libro será, cada vez, más, un libro electrónico —accesible al usuario mediante cierta interfaz ejecutable—, y por ende un programa de computadora). En el caso de España, copiando los textos legislativos extranjeros, lo ha estipulado la *Ley de Propiedad Intelectual* (Véase su edición en Unidad de Programación Editorial, Madrid 1996). El artículo 97 dispone: «1. Será considerado autor del programa

propietarios, y sólo indirectamente a los programadores a sueldo de esos propietarios (quienes se beneficiarán en las condiciones que fije la empresa, que es la que está en posición de fuerza a la hora de regatear en la contratación laboral).

Pero es que, aunque las ganancias de M\$ fueran íntegramente a los bolsillos de los programadores a sueldo de M\$, la abrumadora mayoría de los inventores de *software* no se beneficiarían lo más mínimo de tales ganancias (al revés: con esa ascendencia de M\$ ven mermadas sus posibilidades de recibir, también ellos, alguna compensación o retribución por sus propios inventos).

Al paso que la utilización —aunque sea no onerosa— de *software* libre ayuda a que la promoción de tal *software* sea ulteriormente auspiciada y sufragada por diversas entidades (Universidades, empresas privadas, instituciones de administración pública,

de ordenador la persona o grupo de personas naturales que lo hayan creado, o la persona jurídica que sea contemplada como titular de los derechos de autor en los casos expresamente previstos por esta Ley. [...] 4. Cuando un trabajador asalariado cree un programa de ordenador, en el ejercicio de las funciones que le han sido confiadas o siguiendo las instrucciones de su empresario, la titularidad de los derechos de explotación correspondientes al programa de ordenador así creado, tanto el programa fuente como el programa objeto, corresponderán, exclusivamente, al empresario, salvo pacto en contrario.» ¡Más claro, agua! El empleador es, no sólo el titular de los derechos de propiedad intelectual sobre el *software*, sino que recibe la consideración legal de **autor** a todos los efectos. ¡El autor de un programa es una empresa multinacional, poseída por unos pocos magnates y por una cohorte de especuladores bursátiles! Claro que la doctrina jurídica ha expresado bastante malestar ante esos textos legales. Así J. Miguel Rodríguez Tapia & Fernando Bondía Román, *Comentarios a la Ley de Propiedad Intelectual* (Civitas, Madrid 1997) comentan al respecto (pp. 356-7): «a partir de la Ley 16/1993... se reconoce que el autor de un programa de ordenador pueda ser también una persona jurídica... : 'o la persona jurídica que sea contemplada como titular de los derechos de autor en los casos expresamente previstos por esta Ley'.... Importante, pues, la brecha que se abre en el sistema general de la propiedad intelectual con motivo de la autoría de los programas de ordenador, lo que supone una apreciable disminución del nivel de protección para los autores de tales obras... Una cosa es que en algún determinado supuesto previsto por la Ley la persona jurídica pueda beneficiarse de la protección otorgada al autor (art. 5.2), y otra muy distinta que a la persona jurídica se la considere como autor (art. 97.1).». Y añaden (p.35): «Así pues, cuando un programa merezca la calificación de obra colectiva la autoría sobre el mismo podrá corresponder, salvo pacto en contrario, a una persona jurídica, lo que conlleva que la totalidad de los derechos de autor o de propiedad intelectual —tanto los morales como los patrimoniales— se le atribuirán *ab initio* a la persona jurídica. Ello supone no sólo una considerable disminución del nivel de protección que reciben los auténticos autores-personas físicas de los programas de ordenador frente a la que reciben los autores de otras obras intelectuales, sino una carencia absoluta de protección a través de la propiedad intelectual pues, en el supuesto de que hayan hecho abdicación de su autoría (supuesto más que probable frente a las casas productoras de *software*) no les quedará más que, en su caso, la contraprestación que se derive del vínculo jurídico (no necesariamente contractual) que les ligue a la persona jurídica como consecuencia de la realización de su aportación.»

asociaciones sin ánimo de lucro); con lo cual el resultado será que muchos más inventores recibirán recompensa adecuada por su contribución al progreso humano, y éste se verá así más propiciado.¹²

Lorenzo Peña

25-07-1998

(actualizado 13-08-1999)



Anejo N° 1

STUDIES FIND REWARD OFTEN NO MOTIVATOR

Creativity and intrinsic interest diminish if task is done for gain

By Alfie Kohn¹

Special to the *Boston Globe*²

In the laboratory, rats get Rice Krispies. In the classroom the top students get A's, and in the factory or office the best workers get raises. It's an article of faith for most of us that rewards promote better performance.

But a growing body of research suggests that this law is not nearly as ironclad as was once thought. Psychologists have been finding that rewards can lower performance levels, especially when the performance involves creativity.

A related series of studies shows that intrinsic interest in a task — the sense that something is worth doing for its own sake — typically declines when someone is rewarded for doing it.

¹². Es importantísimo recalcar que el informático que produce *software* libre en una Universidad, o en cualquier otro lugar, no contrae con la empresa que eventualmente subvencione su investigación ninguna obligación similar a la que liga con su empleador al empleado de una firma de *software* propietario. Ni el departamento universitario en cuestión viene vinculado por nada comparable tampoco. Si el empresario es vivo, está sufragando esa labor porque ve que va a salir ganando. Mas no adquiere ningún derecho exclusivo de carácter crematístico, ninguna marca registrada. Eso sí, obtiene una magnífica promoción publicitaria; y, sobre todo, ha calculado que saldrá beneficiado usando (aun sin derechos exclusivos) el *software* libre. En eso hace como las multinacionales del automóvil que sufragan en parte la construcción de autopistas (mediante propinas a los responsables políticos para que manden hacerlas con presteza), aun sin adquirir ninguna exclusiva para sus coches. La diferencia estriba en que el soborno es ilegal e ilegítimo, al paso que las subvenciones a la investigación son legales y legítimas.

¹. Alfie Kohn, a Cambridge, MA writer, is the author of «No Contest: The Case Against Competition,» recently published by Houghton Mifflin Co., Boston, MA. ISBN 0-395-39387-6.

². Reprinted from the Monday 19 January 1987 *Boston Globe*.

If a reward — money, awards, praise, or winning a contest — comes to be seen as the reason one is engaging in an activity, that activity will be viewed as less enjoyable in its own right.

With the exception of some behaviorists who doubt the very existence of intrinsic motivation, these conclusions are now widely accepted among psychologists. Taken together, they suggest we may unwittingly be squelching interest and discouraging innovation among workers, students and artists.

The recognition that rewards can have counter-productive effects is based on a variety of studies, which have come up with such findings as these: Young children who are rewarded for drawing are less likely to draw on their own than are children who draw just for the fun of it. Teenagers offered rewards for playing word games enjoy the games less and do not do as well as those who play with no rewards. Employees who are praised for meeting a manager's expectations suffer a drop in motivation.

Much of the research on creativity and motivation has been performed by Theresa Amabile, associate professor of psychology at Brandeis University. In a paper published early last year on her most recent study, she reported on experiments involving elementary school and college students. Both groups were asked to make «silly» collages. The young children were also asked to invent stories.

The least-creative projects, as rated by several teachers, were done by those students who had contracted for rewards. «It may be that commissioned work will, in general, be less creative than work that is done out of pure interest,» Amabile said.

In 1985, Amabile asked 72 creative writers at Brandeis and at Boston University to write poetry. Some students then were given a list of extrinsic (external) reasons for writing, such as impressing teachers, making money and getting into graduate school, and were asked to think about their own writing with respect to these reasons. Others were given a list of intrinsic reasons: the enjoyment of playing with words, satisfaction from self-expression, and so forth. A third group was not given any list. All were then asked to do more writing.

The results were clear. Students given the extrinsic reasons not only wrote less creatively than the others, as judged by 12 independent poets, but the quality of their work dropped significantly. Rewards, Amabile says, have this destructive effect primarily with creative tasks, including higher-level problem-solving. «The more complex the activity, the more it's hurt by extrinsic reward,» she said.

But other research shows that artists are by no means the only ones affected.

In one study, girls in the fifth and sixth grades tutored younger children much less effectively if they were promised free movie tickets for teaching well. The study, by James Gabarino, now president of Chicago's Erikson Institute for Advanced Studies in Child Development, showed that tutors working for the reward took longer to communicate ideas, got frustrated more easily, and did a poorer job in the end than those who were not rewarded.

Such findings call into question the widespread belief that money is an effective and even necessary way to motivate people. They also challenge the behaviorist assumption that any

activity is more likely to occur if it is rewarded. Amabile says her research «definitely refutes the notion that creativity can be operantly conditioned.»

But Kenneth McGraw, associate professor of psychology at the University of Mississippi, cautions that this does not mean behaviorism itself has been invalidated. «The basic principles of reinforcement and rewards certainly work, but in a restricted context» — restricted, that is, to tasks that are not especially interesting.

Researchers offer several explanations for their surprising findings about rewards and performance.

First, rewards encourage people to focus narrowly on a task, to do it as quickly as possible and to take few risks. «If they feel that 'this is something I have to get through to get the prize,' they're going to be less creative,» Amabile said.

Second, people come to see themselves as being controlled by the reward. They feel less autonomous, and this may interfere with performance. «To the extent one's experience of being self-determined is limited,» said Richard Ryan, associate psychology professor at the University of Rochester, «one's creativity will be reduced as well.»

Finally, extrinsic rewards can erode intrinsic interest. People who see themselves as working for money, approval or competitive success find their tasks less pleasurable, and therefore do not do them as well.

The last explanation reflects 15 years of work by Ryan's mentor at the University of Rochester, Edward Deci. In 1971, Deci showed that «money may work to buy off one's intrinsic motivation for an activity» on a long-term basis. Ten years later, Deci and his colleagues demonstrated that trying to best others has the same effect. Students who competed to solve a puzzle quickly were less likely than those who were not competing to keep working at it once the experiment was over.

Control plays role

There is general agreement, however, that not all rewards have the same effect. Offering a flat fee for participating in an experiment — similar to an hourly wage in the workplace — usually does not reduce intrinsic motivation. It is only when the rewards are based on performing a given task or doing a good job at it — analogous to piece-rate payment and bonuses, respectively — that the problem develops.

The key, then, lies in how a reward is experienced. If we come to view ourselves as working to get something, we will no longer find that activity worth doing in its own right.

There is an old joke that nicely illustrates the principle. An elderly man, harassed by the taunts of neighborhood children, finally devises a scheme. He offered to pay each child a dollar if they would all return Tuesday and yell their insults again. They did so eagerly and received the money, but he told them he could only pay 25 cents on Wednesday. When they returned, insulted him again and collected their quarters, he informed them that Thursday's rate would be just a penny. «Forget it,» they said — and never taunted him again.

Means to and end

In a 1982 study, Stanford psychologist Mark L. Lepper showed that any task, no matter how enjoyable it once seemed, would be devalued if it were presented as a means rather than an end. He told a group of preschoolers they could not engage in one activity they liked until they first took part in another. Although they had enjoyed both activities equally, the children came to dislike the task that was a prerequisite for the other.

It should not be surprising that when verbal feedback is experienced as controlling, the effect on motivation can be similar to that of payment. In a study of corporate employees, Ryan found that those who were told, «Good, you're doing as you /should/» were «significantly less intrinsically motivated than those who received feedback informationally.»

There's a difference, Ryan says, between saying, «I'm giving you this reward because I recognize the value of your work» and «You're getting this reward because you've lived up to my standards.»

A different but related set of problems exists in the case of creativity. Artists must make a living, of course, but Amabile emphasizes that «the negative impact on creativity of working for rewards can be minimized» by playing down the significance of these rewards and trying not to use them in a controlling way. Creative work, the research suggests, cannot be forced, but only allowed to happen.



Anejo N° 2

THE GNU MANIFESTO¹

The GNU Manifesto which appears below was written by Richard Stallman at the beginning of the GNU project, to ask for participation and support. For the first few years, it was updated in minor ways to account for developments, but now it seems best to leave it unchanged as most people have seen it.

Since that time, we have learned about certain common misunderstandings that different wording could help avoid. Footnotes added in 1993 help clarify these points.

For up-to-date information about the available GNU software, please see the latest issue of the GNU's Bulletin. The list is much too long to include here.

What's GNU? Gnu's Not Unix!

GNU, which stands for *Gnu's Not Unix*, is the name for the complete Unix-compatible software system which I am writing so that I can give it away free to everyone who can use it.² Several other volunteers are helping me. Contributions of time, money, programs and equipment are greatly needed.

So far we have an Emacs text editor with Lisp for writing editor commands, a source level debugger, a yacc-compatible parser generator, a linker, and around 35 utilities. A shell (command interpreter) is nearly completed. A new portable optimizing C compiler has compiled itself and may be released this year. An initial kernel exists but many more features are needed to emulate Unix. When the kernel and compiler are finished, it will be possible to distribute a GNU system suitable for program development. We will use TeX as our text formatter, but an nroff is being worked on. We will use the free, portable X window system as well. After this we will add a portable Common Lisp, an Empire game, a spreadsheet, and hundreds of other things, plus on-line documentation. We hope to supply, eventually, everything useful that normally comes with a Unix system, and more.

GNU will be able to run Unix programs, but will not be identical to Unix. We will make all improvements that are convenient, based on our experience with other operating systems. In particular, we plan to have longer file names, file version numbers, a crashproof file

¹ Copyright © 1985, 1993 Free Software Foundation, Inc. Permission is granted to anyone to make or distribute verbatim copies of this document, in any medium, provided that the copyright notice and permission notice are preserved, and that the distributor grants the recipient permission for further redistribution as permitted by this notice. Modified versions may not be made.

² The wording here was careless. The intention was that nobody would have to pay for **permission** to use the GNU system. But the words don't make this clear, and people often interpret them as saying that copies of GNU should always be distributed at little or no charge. That was never the intent; later on, the manifesto mentions the possibility of companies providing the service of distribution for a profit. Subsequently I have learned to distinguish carefully between «free» in the sense of freedom and «free» in the sense of price. Free software is software that users have the freedom to distribute and change. Some users may obtain copies at no charge, while others pay to obtain copies — and if the funds help support improving the software, so much the better. The important thing is that everyone who has a copy has the freedom to cooperate with others in using it.

system, file name completion perhaps, terminal-independent display support, and perhaps eventually a Lisp-based window system through which several Lisp programs and ordinary Unix programs can share a screen. Both C and Lisp will be available as system programming languages. We will try to support UUCP, MIT Chaosnet, and Internet protocols for communication.

GNU is aimed initially at machines in the 68000/16000 class with virtual memory, because they are the easiest machines to make it run on. The extra effort to make it run on smaller machines will be left to someone who wants to use it on them.

To avoid horrible confusion, please pronounce the ‘G’ in the word ‘GNU’ when it is the name of this project.

Why I Must Write GNU

I consider that the golden rule requires that if I like a program I must share it with other people who like it. Software sellers want to divide the users and conquer them, making each user agree not to share with others. I refuse to break solidarity with other users in this way. I cannot in good conscience sign a nondisclosure agreement or a software license agreement. For years I worked within the Artificial Intelligence Lab to resist such tendencies and other inhospitalities, but eventually they had gone too far: I could not remain in an institution where such things are done for me against my will.

So that I can continue to use computers without dishonor, I have decided to put together a sufficient body of free software so that I will be able to get along without any software that is not free. I have resigned from the AI lab to deny MIT any legal excuse to prevent me from giving GNU away.

Why GNU Will Be Compatible with Unix

Unix is not my ideal system, but it is not too bad. The essential features of Unix seem to be good ones, and I think I can fill in what Unix lacks without spoiling them. And a system compatible with Unix would be convenient for many other people to adopt.

How GNU Will Be Available

GNU is not in the public domain. Everyone will be permitted to modify and redistribute GNU, but no distributor will be allowed to restrict its further redistribution. That is to say, proprietary modifications will not be allowed. I want to make sure that all versions of GNU remain free.

Why Many Other Programmers Want to Help

I have found many other programmers who are excited about GNU and want to help.

Many programmers are unhappy about the commercialization of system software. It may enable them to make more money, but it requires them to feel in conflict with other programmers in general rather than feel as comrades. The fundamental act of friendship among programmers is the sharing of programs; marketing arrangements now typically used essentially forbid programmers to treat others as friends. The purchaser of software must choose between

friendship and obeying the law. Naturally, many decide that friendship is more important. But those who believe in law often do not feel at ease with either choice. They become cynical and think that programming is just a way of making money.

By working on and using GNU rather than proprietary programs, we can be hospitable to everyone and obey the law. In addition, GNU serves as an example to inspire and a banner to rally others to join us in sharing. This can give us a feeling of harmony which is impossible if we use software that is not free. For about half the programmers I talk to, this is an important happiness that money cannot replace.

How You Can Contribute

I am asking computer manufacturers for donations of machines and money. I'm asking individuals for donations of programs and work.

One consequence you can expect if you donate machines is that GNU will run on them at an early date. The machines should be complete, ready to use systems, approved for use in a residential area, and not in need of sophisticated cooling or power.

I have found very many programmers eager to contribute part-time work for GNU. For most projects, such part-time distributed work would be very hard to coordinate; the independently-written parts would not work together. But for the particular task of replacing Unix, this problem is absent. A complete Unix system contains hundreds of utility programs, each of which is documented separately. Most interface specifications are fixed by Unix compatibility. If each contributor can write a compatible replacement for a single Unix utility, and make it work properly in place of the original on a Unix system, then these utilities will work right when put together. Even allowing for Murphy to create a few unexpected problems, assembling these components will be a feasible task. (The kernel will require closer communication and will be worked on by a small, tight group.)

If I get donations of money, I may be able to hire a few people full or part time. The salary won't be high by programmers' standards, but I'm looking for people for whom building community spirit is as important as making money. I view this as a way of enabling dedicated people to devote their full energies to working on GNU by sparing them the need to make a living in another way.

Why All Computer Users Will Benefit

Once GNU is written, everyone will be able to obtain good system software free, just like air.³

This means much more than just saving everyone the price of a Unix license. It means that much wasteful duplication of system programming effort will be avoided. This effort can go instead into advancing the state of the art.

³ This is another place I failed to distinguish carefully between the two different meanings of «free». The statement as it stands is not false — you can get copies of GNU software at no charge, from your friends or over the net. But it does suggest the wrong idea.

Complete system sources will be available to everyone. As a result, a user who needs changes in the system will always be free to make them himself, or hire any available programmer or company to make them for him. Users will no longer be at the mercy of one programmer or company which owns the sources and is in sole position to make changes.

Schools will be able to provide a much more educational environment by encouraging all students to study and improve the system code. Harvard's computer lab used to have the policy that no program could be installed on the system if its sources were not on public display, and upheld it by actually refusing to install certain programs. I was very much inspired by this.

Finally, the overhead of considering who owns the system software and what one is or is not entitled to do with it will be lifted.

Arrangements to make people pay for using a program, including licensing of copies, always incur a tremendous cost to society through the cumbersome mechanisms necessary to figure out how much (that is, which programs) a person must pay for. And only a police state can force everyone to obey them. Consider a space station where air must be manufactured at great cost: charging each breather per liter of air may be fair, but wearing the metered gas mask all day and all night is intolerable even if everyone can afford to pay the air bill. And the TV cameras everywhere to see if you ever take the mask off are outrageous. It's better to support the air plant with a head tax and chuck the masks.

Copying all or parts of a program is as natural to a programmer as breathing, and as productive. It ought to be as free.

Some Easily Rebutted Objections to GNU's Goals

1.— «Nobody will use it if it is free, because that means they can't rely on any support.»

«You have to charge for the program to pay for providing the support.»

If people would rather pay for GNU plus service than get GNU free without service, a company to provide just service to people who have obtained GNU free ought to be profitable.⁴

We must distinguish between support in the form of real programming work and mere handholding. The former is something one cannot rely on from a software vendor. If your problem is not shared by enough people, the vendor will tell you to get lost.

If your business needs to be able to rely on support, the only way is to have all the necessary sources and tools. Then you can hire any available person to fix your problem; you are not at the mercy of any individual. With Unix, the price of sources puts this out of consideration for most businesses. With GNU this will be easy. It is still possible for there to be no available competent person, but this problem cannot be blamed on distribution arrangements. GNU does not eliminate all the world's problems, only some of them.

⁴ Several such companies now exist.

Meanwhile, the users who know nothing about computers need handholding: doing things for them which they could easily do themselves but don't know how.

Such services could be provided by companies that sell just hand-holding and repair service. If it is true that users would rather spend money and get a product with service, they will also be willing to buy the service having got the product free. The service companies will compete in quality and price; users will not be tied to any particular one. Meanwhile, those of us who don't need the service should be able to use the program without paying for the service.

2.— «You cannot reach many people without advertising, and you must charge for the program to support that.» «It's no use advertising a program people can get free.»

There are various forms of free or very cheap publicity that can be used to inform numbers of computer users about something like GNU. But it may be true that one can reach more microcomputer users with advertising. If this is really so, a business which advertises the service of copying and mailing GNU for a fee ought to be successful enough to pay for its advertising and more. This way, only the users who benefit from the advertising pay for it.

On the other hand, if many people get GNU from their friends, and such companies don't succeed, this will show that advertising was not really necessary to spread GNU. Why is it that free market advocates don't want to let the free market decide this?⁵

3.— «My company needs a proprietary operating system to get a competitive edge.»

GNU will remove operating system software from the realm of competition. You will not be able to get an edge in this area, but neither will your competitors be able to get an edge over you. You and they will compete in other areas, while benefiting mutually in this one. If your business is selling an operating system, you will not like GNU, but that's tough on you. If your business is something else, GNU can save you from being pushed into the expensive business of selling operating systems.

I would like to see GNU development supported by gifts from many manufacturers and users, reducing the cost to each.⁶

4.— «Don't programmers deserve a reward for their creativity?»

If anything deserves a reward, it is social contribution. Creativity can be a social contribution, but only in so far as society is free to use the results. If programmers deserve to be rewarded for creating innovative programs, by the same token they deserve to be punished if they restrict the use of these programs.

⁵ The Free Software Foundation raises most of its funds from a distribution service, although it is a charity rather than a company. If *no one* chooses to obtain copies by ordering from the FSF, it will be unable to do its work. But this does not mean that proprietary restrictions are justified to force every user to pay. If a small fraction of all the users order copies from the FSF, that is sufficient to keep the FSF afloat. So we ask users to choose to support us in this way. Have you done your part?

⁶ A group of computer companies recently pooled funds to support maintenance of the GNU C Compiler.

5.— «Shouldn't a programmer be able to ask for a reward for his creativity?»

There is nothing wrong with wanting pay for work, or seeking to maximize one's income, as long as one does not use means that are destructive. But the means customary in the field of software today are based on destruction.

Extracting money from users of a program by restricting their use of it is destructive because the restrictions reduce the amount and the ways that the program can be used. This reduces the amount of wealth that humanity derives from the program. When there is a deliberate choice to restrict, the harmful consequences are deliberate destruction.

The reason a good citizen does not use such destructive means to become wealthier is that, if everyone did so, we would all become poorer from the mutual destructiveness. This is Kantian ethics; or, the Golden Rule. Since I do not like the consequences that result if everyone hoards information, I am required to consider it wrong for one to do so. Specifically, the desire to be rewarded for one's creativity does not justify depriving the world in general of all or part of that creativity.

6.— «Won't programmers starve?»

I could answer that nobody is forced to be a programmer. Most of us cannot manage to get any money for standing on the street and making faces. But we are not, as a result, condemned to spend our lives standing on the street making faces, and starving. We do something else.

But that is the wrong answer because it accepts the questioner's implicit assumption: that without ownership of software, programmers cannot possibly be paid a cent. Supposedly it is all or nothing.

The real reason programmers will not starve is that it will still be possible for them to get paid for programming; just not paid as much as now.

Restricting copying is not the only basis for business in software. It is the most common basis because it brings in the most money. If it were prohibited, or rejected by the customer, software business would move to other bases of organization which are now used less often. There are always numerous ways to organize any kind of business.

Probably programming will not be as lucrative on the new basis as it is now. But that is not an argument against the change. It is not considered an injustice that sales clerks make the salaries that they now do. If programmers made the same, that would not be an injustice either. (In practice they would still make considerably more than that.)

7.— «Don't people have a right to control how their creativity is used?»

«Control over the use of one's ideas» really constitutes control over other people's lives; and it is usually used to make their lives more difficult.

People who have studied the issue of intellectual property rights carefully (such as lawyers) say that there is no intrinsic right to intellectual property. The kinds of supposed intellectual property rights that the government recognizes were created by specific acts of legislation for specific purposes.

For example, the patent system was established to encourage inventors to disclose the details of their inventions. Its purpose was to help society rather than to help inventors. At the time, the life span of 17 years for a patent was short compared with the rate of advance of the state of the art. Since patents are an issue only among manufacturers, for whom the cost and effort of a license agreement are small compared with setting up production, the patents often do not do much harm. They do not obstruct most individuals who use patented products.

The idea of copyright did not exist in ancient times, when authors frequently copied other authors at length in works of non-fiction. This practice was useful, and is the only way many authors' works have survived even in part. The copyright system was created expressly for the purpose of encouraging authorship. In the domain for which it was invented — books, which could be copied economically only on a printing press — it did little harm, and did not obstruct most of the individuals who read the books.

All intellectual property rights are just licenses granted by society because it was thought, rightly or wrongly, that society as a whole would benefit by granting them. But in any particular situation, we have to ask: are we really better off granting such license? What kind of act are we licensing a person to do?

The case of programs today is very different from that of books a hundred years ago. The fact that the easiest way to copy a program is from one neighbor to another, the fact that a program has both source code and object code which are distinct, and the fact that a program is used rather than read and enjoyed, combine to create a situation in which a person who enforces a copyright is harming society as a whole both materially and spiritually; in which a person should not do so regardless of whether the law enables him to.

8.— «Competition makes things get done better.»

The paradigm of competition is a race: by rewarding the winner, we encourage everyone to run faster. When capitalism really works this way, it does a good job; but its defenders are wrong in assuming it always works this way. If the runners forget why the reward is offered and become intent on winning, no matter how, they may find other strategies — such as, attacking other runners. If the runners get into a fist fight, they will all finish late.

Proprietary and secret software is the moral equivalent of runners in a fist fight. Sad to say, the only referee we've got does not seem to object to fights; he just regulates them («For every ten yards you run, you can fire one shot»). He really ought to break them up, and penalize runners for even trying to fight.

9.— «Won't everyone stop programming without a monetary incentive?»

Actually, many people will program with absolutely no monetary incentive. Programming has an irresistible fascination for some people, usually the people who are best at it. There is no shortage of professional musicians who keep at it even though they have no hope of making a living that way.

But really this question, though commonly asked, is not appropriate to the situation. Pay for programmers will not disappear, only become less. So the right question is, will anyone program with a reduced monetary incentive? My experience shows that they will.

For more than ten years, many of the world's best programmers worked at the Artificial Intelligence Lab for far less money than they could have had anywhere else. They got many kinds of non-monetary rewards: fame and appreciation, for example. And creativity is also fun, a reward in itself.

Then most of them left when offered a chance to do the same interesting work for a lot of money.

What the facts show is that people will program for reasons other than riches; but if given a chance to make a lot of money as well, they will come to expect and demand it. Low-paying organizations do poorly in competition with high-paying ones, but they do not have to do badly if the high-paying ones are banned.

10.— «We need the programmers desperately. If they demand that we stop helping our neighbors, we have to obey.»

You're never so desperate that you have to obey this sort of demand. Remember: millions for defense, but not a cent for tribute!

11.— «Programmers need to make a living somehow.»

In the short run, this is true. However, there are plenty of ways that programmers could make a living without selling the right to use a program. This way is customary now because it brings programmers and businessmen the most money, not because it is the only way to make a living. It is easy to find other ways if you want to find them. Here are a number of examples.

A manufacturer introducing a new computer will pay for the porting of operating systems onto the new hardware.

The sale of teaching, hand-holding and maintenance services could also employ programmers.

People with new ideas could distribute programs as freeware, asking for donations from satisfied users, or selling hand-holding services. I have met people who are already working this way successfully.

Users with related needs can form users' groups, and pay dues. A group would contract with programming companies to write programs that the group's members would like to use.

All sorts of development can be funded with a Software Tax

Suppose everyone who buys a computer has to pay x percent of the price as a software tax. The government gives this to an agency like the NSF to spend on software development.

But if the computer buyer makes a donation to software development himself, he can take a credit against the tax. He can donate to the project of his own choosing — often, chosen because he hopes to use the results when it is done. He can take a credit for any amount of donation up to the total tax he had to pay.

The total tax rate could be decided by a vote of the payers of the tax, weighted according to the amount they will be taxed on.

The consequences

- * The computer-using community supports software development.
- * This community decides what level of support is needed.
- * Users who care which projects their share is spent on can choose this for themselves.

In the long run, making programs free is a step toward the post-scarcity world, where nobody will have to work very hard just to make a living. People will be free to devote themselves to activities that are fun, such as programming, after spending the necessary ten hours a week on required tasks such as legislation, family counseling, robot repair and asteroid prospecting. There will be no need to be able to make a living from programming.

We have already greatly reduced the amount of work that the whole society must do for its actual productivity, but only a little of this has translated itself into leisure for workers because much nonproductive activity is required to accompany productive activity. The main causes of this are bureaucracy and isometric struggles against competition. Free software will greatly reduce these drains in the area of software production. We must do this, in order for technical gains in productivity to translate into less work for us.