

NOTAS Y DISCUSIONES

## Para una ética y una estética del software libre

ANDONI ALONSO Y JAVIER DEL ARCO

Center for Basque Studies Reno, Nevada - Fundación Vodafone, Madrid

**RESUMEN.** El presente artículo estudia el modo en que la comunidad hacker ha desarrollado un código propio de valores estéticos y éticos. La historia del «hackerismo» supone un paso previo para la comprensión de cómo la escritura de código se ha bifurcado en dos líneas: el software libre y el código abierto. El estudio de las dimensiones estéticas de estos movimientos puede resultar muy provechoso para correlacionar el trabajo creativo de los hackers y la manera en que éste es considerado por los movimientos de vanguardia. Desde un punto de vista ético, es difícil definir un proyecto común a todos los grupos implicados. Sin embargo, este movimiento representa, en su conjunto, la emergencia de una nueva perspectiva ética que retoma viejas ideas como la del procomún.

*Palabras clave:* ética hacker, autoría, copyright, copyleft, software libre, software de código abierto.

**ABSTRACT.** This paper deals with how an emergent community like hacker's has developed a code of aesthetic and ethical values in different levels. The history of hackerism is a previous step to understand in a very basic way how code writing has forked in two different paths: free software and open code. Aesthetics can be a way to correlate creative work made by hackers and how avant-garde movements considered their work. From an ethical point of view it is difficult to find an unified agenda for all the different groups involved with free and open software. But this movement represents the emergence of a new ethical approach, retaking older attitudes like the commons.

*Key words:* hacker ethics, authorship, copyright, copyleft, free software, open source software.

Desde finales de los noventa el movimiento del software libre ha ido adquiriendo una importancia creciente, o, dicho de otra manera, ha logrado una considerable «visibilidad social» que parece crecer según va pasando el tiempo. En su irrupción en la sociedad se ha producido un fenómeno de contagio; se oyen ahora voces que reclaman o extienden la metáfora del software libre y el código abierto a otros terrenos como el conocimiento libre, enciclopedias libres, la medicina abierta, la ciencia abierta, etc<sup>1</sup>. Las razones para que este fenómeno adquiera tan grandes dimensiones puede explicarse de forma relativamente sencilla: como la tecnología informática impregna cada vez más ámbitos sociales, todos sus movimientos, cambios, transformaciones, propuestas y evoluciones provoquen una resonancia igualmente amplia y profunda en esos sectores: podemos vivir sin centrales nucleares pero no podemos vivir sin ordenadores. A un tiempo diremos con Langdom Winner<sup>2</sup> que la tecnología encarna una forma de política y las transformaciones tanto en una como en otra acaban contagiándose. Así se puede afirmar con muchos hackers (sin necesidad de ser uno de ellos) que elegir el software libre como herramienta informática básica indica una elección política más profunda que realizarla de acuerdo exclusivamente con estándares técnicos de fiabilidad o eficacia (dicho sea de paso, estos criterios tampoco se excluyen entre sí y muchos son los que afirman la superioridad del FLOSS respecto al software propietario). Aceptado esto, el siguiente paso debe señalar la complejidad de este fenómeno, consecuencia también de su extensión por los diversos ámbitos y actividades sociales. Podría pensarse que el software libre se despliega en varios niveles donde los análisis éticos y estéticos tienen cabida junto a los políticos y económicos.

Pero antes de entrar por esos vericuetos es crucial señalar que el movimiento que apuesta por el código abierto, el software libre o el FLOSS (Free/Libre/Open Source System) no es un movimiento unificado ni con una agenda política común en todos sus objetivos (aunque quizás sí se pueda hablar de un mínimo común denominador como podría representar el acceso al código fuente). Las denominaciones tienen su importancia y en cierto sentido podríamos establecer una dicotomía entre los partidarios del código abierto y los partidarios del software libre. En cualquier caso las precisiones respecto a los grandes grupos (Stallman ¿versus? Raymond-Torvalds), son importantes aunque no hace justicia a otros grupos no tan grandes y que quizás han captado menos la atención de los usuarios y los interesados sobre este tema<sup>3</sup>. Pero en cualquier caso, parece también claro que, de acuerdo con Manuel de Landa<sup>4</sup>, este movimiento adolece de una filosofía bien estructurada (entendiendo el término *filosofía* de forma estricta), una ética suficientemente fundamentada aunque quizás no la necesita pues su importancia reside en los logros, en las cuestiones prácticas que consigue. Pero ello no impide que se puedan realizar críticas, desde el punto de vista filosófico, a ciertas declaraciones o fundamentaciones que los propios autores –Stallman o Raymond–, han manifestado.

#### *Breve historia del software libre*

En la era del software propietario, con la existencia de una de las compañías más grandes y con mayores beneficios del mundo, resulta extraño recordar que en el principio, el software era gratuito, o, mejor dicho, ni siquiera se consideraba que fuera realmente algo importante. Neal Stephenson<sup>5</sup> recuerda cómo en los

años cincuenta hubiera sido del todo punto absurdo cobrar por el software; si se vende una máquina ésta ha de ser utilizable sin costo extra y se entiende que el software ya está incluido en el precio del hardware; equivaldría a comprar un coche pagando el motor aparte, como dos mercancías separadas. Ítem más, gran parte del software se encontraba empotrado en la máquina, en el propio hardware. La mera noción de que algo tan intangible como el software (una presencia fantasmal dentro de la circuitería) fuera realmente *algo*, y entre esos *algos* una mercancía capaz de venderse y comprarse como cualquier otra, parecía de todo punto absurda; diríamos que tenía así una ontología comercial dudosa. Por esta razón IBM ofrecía a los compradores de sus grandes máquinas una librería de programas sin costo adicional y en las que se incluía el sistema operativo, lenguajes de programación, compiladores y otros programas como procesadores de texto y así; la compañía vendía máquinas (hardware) y no instrucciones (software). Pero fue precisamente IBM la que anunció la separación del precio entre el software y el hardware: a partir del 1 de enero de 1970 tendrían precios separados. Es cierto que la capacidad de los ordenadores para realizar tareas múltiples (máquinas multipropósito) permitió el nacimiento de una industria de software ya en los sesenta, porque era necesario desarrollar aplicaciones no previstas en el diseño original del hardware; se podían hacer más cosas, la sociedad en sí se informatizaba. Y es precisamente en este divorcio entre hardware y software, entre alma y cuerpo (por así decirlo), donde se gesta el nacimiento del software libre. Sin embargo el mismo año en que IBM declara su intención de seguir esta estrategia comercial, ATT y Bell ponen a disposición de la universidad, de forma

gratuita, el código fuente de UNIX. La práctica simultaneidad de estas dos acciones contrapuestas indica la ambigüedad del momento hacia el software.

Los analistas sobre este tema coinciden en que la forma de que el movimiento a favor del software libre se ha extendido como un virus y no como una ideología común y corriente. Incluso se utiliza el nombre de «meme» que acuñó Richard Dawkins<sup>6</sup> y propagó Daniel Dennett<sup>7</sup> para sostener una curiosa teoría de la cultura. Quizás se pueda colocar su punto de partida en la emergencia de software robusto y fiable que, poco a poco fue ocupando su lugar. Cuando Richard Stallman comenzó su proyecto GNU<sup>8</sup>, su influencia apenas alcanza a un puñado de programadores que se embarcaron con él en el proyecto. Y sin embargo, la razón para el desarrollo del GNU no fue solamente una cuestión técnica –encontrar un sistema operativo más fiable– sino social y de ahí la institución de una licencia específica, la GPL (*General Public License*), que permitiese una aproximación *legal* a la autoría diferente del copyright, o, mejor, dicho, subvirtiendo el fundamento que la propia idea de copyright tiene desde su creación en el siglo XVIII<sup>9</sup>. El software libre se protege con una licencia que justamente obliga a la libre distribución a todos aquellos que realicen cualesquiera mejora del software inicial y no pueden, por decirlo así, apoderarse de su logro. Conviene no olvidar este hecho porque indica que desde el principio del nacimiento del software libre, si aceptamos que GNU fuera tal inicio, hay una preocupación ético-social evidente. Sin apenas hacerse notar y como si hubiera sido de la noche a la mañana, la mayoría de los servidores conectados a Internet empleaban *Apache* como software de conexión. *Apache*, por cierto, no es realmente software libre, sino de código abierto. Las cuestiones terminológicas, como se ha indicado

antes, son bastante más relevantes de lo que pudiera pensarse en esta cuestión.

Linus Torvalds, apoyándose en el trabajo previo de la FSF, el software desarrollado entonces como GNU, lanza un proyecto de trabajo en Internet, a fin de completar el requerido por GNU y nace de esta manera GNU/Linux en 1991. Este hecho alcanza gran significado en la historia del software porque todo un sistema operativo, perfectamente utilizable y cada vez más robusto, según se le van añadiendo mejoras, nace de un trabajo comunal y no es el producto de una compañía de software concreta. Se estima que el desarrollo de un sistema operativo completo cuesta aproximadamente unos quinientos millones de pesetas y esa podría ser una de las principales razones para la ausencia de compañías competidoras. La inversión es demasiado elevada desde el punto de vista empresarial como para arriesgarse a un fracaso, más en la actual situación de semi-monopolio. Contemplada los aspectos meramente económicos de un desarrollo como el GNU/Linux es impresionante. No existe un paralelo histórico desde el punto de vista empresarial. Como consecuencia, el creciente monopolio de los sistemas operativos comienza a resquebrajarse, primero ligeramente porque su uso requiere un conocimiento profundo del software y luego cada vez más generalizadamente, tan pronto como aparecen distribuciones comprensibles para usuarios no iniciados en el software. El software propietario comienza a recibir cada vez más críticas, especialmente por su deslizamiento hacia un monopolio y su marcada obsolescencia, es decir, a un producto no demasiado bien hecho que requiere continuos «parches» (*patches*). Respecto a la tendencia hacia el monopolio –las numerosas demandas judiciales son una prueba de ello– difundieron en los medios una imagen no demasiado

solvente desde el punto de vista moral. Una era caracterizada, en los países de economía pujante, hacia la diversificación de la oferta se encuentra con una homogenización radical en el terreno informático. Añadamos a ello la reciente lucha contra la propia noción de copyright y la aceptación, más o menos explícita de prácticas cercanas a lo ilegal sobre el respeto del copyright. Napster y E-Mule son los ejemplos más recientes de una sociedad informatizada que no acepta de buen grado la existencia de una propiedad intelectual. Respecto a la segunda, se trata de una industria basada en la aceleración y en la difusión de productos no demasiado bien terminados (piénsese en los innumerables parches y actualizaciones de los diversos sistemas operativos y programas propietarios) ha creado una sensación de desconfianza en el usuario medio.

Podemos hablar de una «humillación del usuario», en el sentido de que la complejidad e inestabilidad de los sistemas operativos propietarios los continuos fallos –atribuibles a razones tan distintas como la mala factura del sistema operativo, la inexperiencia del usuario, la complejidad del sistema, etc.– provocan una sensación de hastío. Así que, desde luego, existe una reticencia, cada vez más generalizada, hacia la (s) grande(s) compañía(s) de sistemas operativos propietarios. Y curiosamente esta desconfianza respecto a sistemas que no funcionan o son demasiado complejos puede contaminar a otras alternativas. La separación tan enorme entre el productor de software y el público genera una desconfianza que, muchas veces se emplea precisamente contra el software libre. Paradójicamente se le acusa de inestabilidad, incompatibilidad a este último, cuando claramente tiene mayores ventajas técnicas. Es curioso que el hecho de ser gratuito suponga en ocasiones una

dificultad para ganarse la confianza del usuario no especializado.

Pero también en este momento comienzan a producirse divisiones. Algunos interpretan que Linux no tiene por qué adherirse al espíritu evangelizador, militante de Stallman, sino que se contempla como una posibilidad *diferente* de hacer negocio, de encontrar una ética del software que sea abierta pero que al mismo tiempo permita hacer dinero; básicamente no vendiendo software sino servicio sobre ese software de código abierto. El término código abierto nace en 1998<sup>10</sup> durante una conferencia a raíz de la decisión de Netscape para liberar el código y colocarlo a disposición de quien quisiera emplearlo. Stallman recibió noticia de este nuevo término, obra de Eric Raymond pero decidió no emplearlo porque se apartaba bastante de su propia noción de software libre. De hecho las cosas fueron empeorando al punto en el que Raymond declaró que la licencia GPL era ya innecesaria provocando una reacción incendiaria de Stallman. Así que en la actualidad puede afirmarse que existen dos corrientes diferenciadas básicamente por el aspecto comercial del software libre o, mejor dicho, del open source. Raymond entiende como un gran éxito la colaboración con Netscape que le llevó a dialogar directamente con varias de las compañías que se encuentran en *Fortune 500*<sup>11</sup>

#### *Hacia una estética del software libre*

El software libre consiste finalmente en un conjunto de algoritmos que permiten administrar las tareas de un ordenador. En cuanto tal implica la articulación de una serie de estrategias, habilidades y conocimientos. Y del mismo modo en que se puede hablar de belleza en otras ramas de la ingeniería –la belleza de un puente, por ejemplo–, se puede asignar

lo mismo de la escritura de software. Existe software elegante y aquel que no lo es. De hecho esta elegancia en la escritura así como la capacidad de resolver problemas de forma original son virtudes apreciadas dentro del mundo hacker. Así que podríamos hablar de elegancia, originalidad, sencillez, economía, novedad, entre otros, como atributos propios de la escritura de código de. Pero ser capaz de reconocer tales atributos estéticos posiblemente necesite un conocimiento muy profundo del software. Sin embargo Stephenson realiza un curioso paralelismo entre la apariencia (estética) del software libre y la de la maquinaria. El arranque de Linux es una sucesión en serie de comandos que muestran al usuario lo que realmente está ocurriendo, los procesos que se están produciendo para que el sistema operativo funcione. En ese enorme listado aparecen también los posibles errores o partes del sistema que no se han acoplado. Por el contrario, otros sistemas operativos comienzan ocultando todos estos procesos, ejecutando, y nunca mejor dicho, una suerte de magia que termina cuando el escritor aparece ante el usuario. Stephenson recuerda haber visto un tipo de taladro enorme, que se manipulaba con las dos manos y que a primera vista daba la impresión de una máquina temible que había que manejar con cuidado. Frente o tras éste aparece el taladro «amigable», con forma de pistolita y que parece inofensivo pero no lo es; sigue siendo una máquina que hay que manejar con cuidado. Aunque en apariencia la Pero estéticamente ese peligro no se refleja. En cualquier caso esta estética propugnada por Stephenson, fundamentada en cierta ética –hay que decir o mostrar la verdad de lo que ocurre con los ordenadores– no la seguiría todo el mundo porque finalmente muchos usuarios quieren simplemente emplear los ordenadores para tareas concretas y

no para saber más de los ordenadores. Podemos hablar así de una estética hacker que se manifiesta también en una forma más o menos común de entender y producir el software y que no tiene por qué atraer a otro tipo distinto de individuos.

Sin embargo hay otra forma de entender el aspecto estético-artístico del software libre. Es interesante anotar, tal como señala Javier Echeverría, que gran parte del trabajo científico durante el siglo XVIII y XIX se realizó de forma cooperativa<sup>12</sup>, un precedente también del software libre. El trabajo de Michael Century y de Geert Lovink incide precisamente en este sentido porque gran parte de las corrientes artísticas del siglo XX han hecho hincapié precisamente en la creatividad compartida, en la idea de grupo y en el prestigio entre los propios colegas. Century observa la construcción del GNU/Linux en paralelo a diversos aspectos de la práctica artística. Fundamentalmente hay tres campos en los que se establece este paralelismo: la noción de autoría, la modularidad y transparencia de los elementos empleados y el deslizamiento de un control individual del proceso creativo hacia una actividad colectiva. De hecho existen poéticas en el siglo XX que apuestan por una visión diferente del arte, como la de Umberto Eco y su *Obra Abierta*<sup>13</sup>. Eco preconizaba en los setenta la existencia de un tipo de arte que se presentaba como una propuesta para el espectador, antes que como una obra perfectamente acabada. Los espectadores *actualizaban* la obra al contemplarla, así que se trataba de un trabajo comunal donde la autoría resultaba ser compartida: la interpretación era un paso inexcusable para la obra. Con este fondo teórico, para algunos críticos la tecnología ha puesto en cuestión que la noción tradicional de autor artístico,

heredada del siglo XIX, como ocurre por ejemplo con el hipertexto<sup>14</sup>. El net-art ha hecho de la interactividad, en ocasiones, un aspecto crucial de la obra artística y de nuevo se puede ver. Hasta qué punto son ciertos estos ilustres precedentes no tiene demasiada importancia. Asimismo ciertas teorizaciones como las de Martha Woodmansee<sup>15</sup> hacen ver cómo es posible que la noción tradicional de autor no sea semejante a la establecida a finales del siglo XIX y en realidad se parezca más a la autoría de los hackers, donde es una comunidad con sus tradiciones y sus normas las que realmente realizan la obra.

#### *Hacia una ética del software libre*

Antes se indicaba que debe existir una especie de mínimo común denominador respecto a las motivaciones morales de los diversos grupos y comunidades que desarrollan tanto FLOSS como código abierto. Y tal vez la forma más general de incluir a estos grupos sea la ya legendaria denominación de «hacker». De hecho Pekka Himanen, uno de los teóricos del código abierto, titula su libro *la ética de hacker*, donde fundamenta el comportamiento hacker en una sugerente relectura del clásico *La ética protestante y el espíritu del capitalismo*. Eric Raymond considera que el movimiento por el código abierto es la «venganza de los hackers»<sup>16</sup> y el propio Stallman se considera a sí mismo como el último hacker. El término *hacker*, de acuerdo con el diccionario Webster, tiene dos acepciones referidas a cuestiones informáticas: usuario o programador informal pero entusiasta de los ordenadores y aquél que trata de acceder a datos informáticos de forma ilegal<sup>17</sup>. La primera de las acepciones es la más usual en los últimos años aunque durante los ochenta y gran parte de los noventa, se consideraba

equivalente a criminal informático. En cualquier caso, el término refiere originariamente a una comunidad de estudiantes en el prestigioso MIT (Massachusetts Institute of Technology) que se dedicaban a gastar novatadas de gran dificultad técnica o de gran ingenio, como hacer subir un coche a la gran cúpula del MIT o hacer desaparecer el despacho del rector<sup>18</sup>. Esta comunidad que se perpetúa desde la fundación del MIT está rodeada de leyendas y supuestos héroes, por lo que resulta difícil dar cuenta de los verdaderos intereses de sus miembros; en todo caso existe algo así como un acuerdo tácito de lo que es apropiado y lo que no. Además sería más correcto hablar de una moral hacker que de una ética<sup>19</sup>. Aun así hay autores que han tratado de explicitar la ética hacker como Steven Levy<sup>20</sup> basada en una serie de principios, en cierta axiología un tanto naif. Junto a verdaderos principios éticos como «el acceso a ordenadores o a cualquier cosa que pueda enseñar como funciona el mundo deber ser permitido» o «la información debe ser libre» añade otros como «se puede crear belleza en un ordenador» o «el ordenador puede cambiar la vida<sup>21</sup>».

En términos generales se puede afirmar así que existen dos sensibilidades distintas y a veces contrapuestas, tal como se ha señalado anteriormente. La ética defendida por Stallman sería una ética «a priori»; basada en un principio como es el derecho inalienable de todo ser humano a acceder a la información relevante. El software propietario sería así una negación de tal derecho y por lo tanto algo cercano al delito. Según el propio Stallman: «El hecho de firmar una licencia típica de software implica traicionar a su vecino: ``Prometo privar a mi vecino de este programa de forma que yo sea capaz de tener una copia para mí.'' Las personas que toman estas decisiones sien-

ten presión psicológica interior que los justifica, a cambio de degradar la importancia de ayudar a su vecino— así que el espíritu público sale perjudicado. Se trata de un daño psicosocial asociado con el daño material provocado por el desincentivo de usar el programa»<sup>22</sup>. Así que Stallman considera un *imperativo moral*, de carácter universal, y aplicable no sólo a cuestiones como el software libre, sino también a ciertas patentes biológicas<sup>23</sup>. Estas patentes, a las que denomina como biopiratería son un ejemplo de cómo se puede estrangular el bienestar de países enteros cuando se niega el acceso al conocimiento. Esta postura realmente se abre más allá de la escritura del software. Pero a la vez Stallman considera necesario insistir en la pureza de la actitud ética ante el software. Ésta es la razón por la que afirma: «la postura de aquellos que defienden el código abierto es simplemente una actitud pragmática de cómo producir más código y realizar negocios con ello<sup>24</sup>. De acuerdo con Lovink, las diferencias entre FLOSS y código abierto van desapareciendo paulatinamente<sup>25</sup>. Lo interesante es entender la política y ética detrás de esa anomalía que ha representado el software libre.

Merece la pena realizar un ejercicio de fundamentación ética para el software libre, desde el punto de vista filosófico. Quizás sea Luciano Floridi quien ha argumentado la posibilidad de una ética informacional extendida en tres niveles interrelacionados<sup>26</sup>: como recurso, como producto y como objetivo. El primero refiere a la necesidad de conocer suficientemente como para adoptar una decisión correcta; la ignorancia es así un atenuante. En segundo lugar los individuos pueden producir información que puede ser utilizado por otros de forma ética. La falta sería entonces proporcionar información falsa que lleve a confusiones en la decisión ética. Finalmente existe la búsqueda de

información como actividad ética y en este caso el intrusismo o la falta de respeto a la privacidad serían los causantes del fallo ético. Floridi argumenta que cualquier tratamiento de una ética informacional ha de tener en cuenta estos tres niveles porque la información se caracteriza por ser un entorno, un ambiente donde los individuos actúan entre sí, transforman y planifican. Desde luego el software libre se ajusta a los tres: proporciona los medios para buscar información, al permitir acceso al código fuente, es un producto de información para la mejora de la sociedad y finalmente sirve para obtener información de modo lícito. Si entendemos que la información es un entorno, un medio ambiente, es verdad que el FLOSS cumple todas las condiciones para una adecuada práctica ética. En este sentido todo el movimiento del Software libre/Código Abierto significa reabrir un debate sobre las prácticas buenas en la programación pero desde un punto de vista positivo; no lo que se debe o no se debe hacer sino cómo hacer el bien de manera sustancial con esta tecnología.

*Conclusión: para una ética y una estética del software libre y más allá*

El software libre, código abierto o FLOSS, el tiempo señalará cuál de estas denominaciones tiene éxito, ha supuesto una novedad empresarial, tecnológica y filosófica. Ésta es una de las ventajas de la diversidad, que la tendencia futura será la resultante de diversas fuerzas que empujarán el desarrollo del software libre en diversas direcciones. Parece importante subrayar la necesidad de que exista una pluralidad de voces en este mundo porque tiene que dar lugar a otro, a uno que difunda el conocimiento libre como muestra el proyecto que ha fundado la licencia «*creative commons*»<sup>27</sup>. La

idea, tal como sostienen sus fundadores, es un uso adecuado de ciertos derechos que animen a la contribución voluntaria y libre, con carácter libertario pero sin una prédica amenazante sobre la necesidad vital de luchar contra el software propietario. Y quizás la estrategia no haya de ser combativa contra el mundo cerrado del copyright. Tal vez la estrategia correcta sea la creación de un mundo cultural paralelo al que detenta el copyright. Las posibilidades éticas y políticas no se acaban solamente en el software son mayores de lo que parecen, tal como el trabajo de Stallman sobre la biopiratería demuestra.

Por otro lado, resulta extraordinariamente sorprendente la resurrección de una nueva forma de entender la economía en el espacio de la *high-tech*, que hunde sus raíces en la tradición, tales como la ciencia del siglo XVIII, la autoría previa a este siglo, las tácticas de *potlach* y de la economía del regalo y del prestigio, las técnicas creativas de las primeras vanguardias, etc. Todas estas tendencias o formas de organizar la creatividad y la economía, distintas hasta cierto punto del capitalismo tradicional, sugieren que existen otras posibilidades legadas por el pasado y que inadvertidamente han sido actualizadas en el presente por otros medios. También merece la pena indicar que, a diferencia de otras éticas, la ética del software libre, su teorización se realiza a posteriori, esto es, una vez que se ha establecido esta corriente de software alternativo. Desde el punto de vista teórico supone toda una novedad y una tarea filosófica apasionante, porque revitaliza un campo, el ético, que parecía cada vez más arrinconado. En estos tiempos es saludable escuchar de nuevo la palabra «libertad» en contextos concretos como los indicados aquí y esperar su expansión a otros ámbitos.

*Nota bibliográfica para  
no iniciados*

A continuación se ofrece una serie mínima de textos para comprender el movimiento FLOSS y software de código abierto. En España el software libre ha tenido bastante importancia, especialmente en Extremadura donde el apoyo gubernamental supuso un espaldarazo histórico a esta forma de entender el software. La Junta de Extremadura ha «transmitido el virus del FLOSS» a otras administraciones autonómicas y también europeas. La redacción de las dos versiones del Libro Blanco del Software Libre (vid. <http://www.libroblanco.com/html/index.php>) se ha realizado desde esa comunidad y es un buen exponente del estado de la cuestión en nuestro país (existe edición impresa de 2004 y 2005). El libro de Pekka Himanen, *La ética hacker* es quizá el texto más original para intentar dar cuenta del software libre. Y los textos de Stallman junto con los artículos que se ofrecen en la página web de la Free Software Foundation muestran un buen panorama de la línea «dura» del software libre; Traficantes de Sueños publicó de Stallman *Software libre para una sociedad libre* (2004) donde se recoge una antología de varios de sus artículos. Sin embargo el libro de Eric Raymond, *The Cathedral and the Bazaar* está sin publicar, aunque hay una versión on-line <http://es.tldp.org/Otros/catedral-bazar/cathedral-es-paper-00.html>. Raymond, que se considera a sí mismo un «etnógrafo del mundo hacker» representa la otra gran corriente del software libre y su posible proyección comercial. Y para saber más de los aspectos técnicos y legales del FLOSS y código abierto, es recomendable el libro *Perspectives on Free and Open Source Software* (Feller, Fitzgerald y Lakhani, MIT, Massachusetts, 2005).

BIBLIOGRAFÍA  
COMPLEMENTARIA

- DAWKINS, RICHARD (1989) *The Selfish Gene*, Oxford University Press, Oxford.
- DENNETT, DANIEL (2000) *La peligrosa idea de Darwin*, Galaxia Gutenberg, Madrid.
- ECHEVERRÍA, JAVIER, «Ética en Internet», *El País*, Babelia, 13-4-2002, p.12.
- ECO, UMBERTO (1990), *Obra Abierta*, Ariel, Barcelona.
- FLORIDI, LUCIANO, *Ética de la Información, Su Naturaleza y Alcance*, en esta misma publicación.
- HIMANEN, PEKKA (2003) *La ética de hacker*, Destino, Barcelona.
- LANDOW, GEORGE, P (1992) *Hypertext: The Convergence of Contemporary Critical Theory and Technology*. Baltimore: John Hopkins University Press.
- LANDOW, GEORGE, P (1994), *Hyper/Text/Theory*. Baltimore: John Hopkins University Press.
- LANGDOM WINNER (1983) «Do Artifacts Have Politics?» en: D. MacKenzie et al. (eds.), *The Social Shaping of Technology*, Open University Press, Philadelphia. Hay trad. española de Mario Francisco Villa en <http://www.campus-oei.org/salactsi/winner.htm>
- LEVY, STEVEN (1984) *Hackers, the True Computer Heroes*, Bantam, New York.
- LOVINK, GEERT (2003), *My First Recession*, Nai, New York.
- MANUEL DE LANDA, *Open Source Movement in Search of a Philosophy* en Raymond, Eric *The Revenge of the Hackers*, <http://www.oreilly.com/catalog/openources/book/raymond2.html>
- SANTOS UNAMUNO, ENRIQUE (2002) *Laberintos de papel: Jorge Luis Borges*

*e Italo Calvino en la era digital*, Universidad de Extremadura, Cáceres.  
WOODMANSEE, MARTHA, *The Genius and the Copyright: Economic and Legal Conditions of the Emergence of the 'Author'*, Eighteen Century Studies, Vol. 17, Nº 4, Verano, 1984, Baltimore.

## WEBGRAFÍA

<http://creativecommons.org/about/history>

<http://www.cddc.vt.edu/host/delanda/pages/opensource.htm>  
<http://www.gnu.org/gnu/initial-announcement.es.html>  
<http://www.gnu.org/philosophy/should-be-free.es.html>  
<http://www.opensource.org/docs/history.html>  
<http://www.stallman.org/articles/bioperacy.html>  
<http://www.univ-orleans.fr/EXT/ASTEX/astex/doc/fr/adrut/html/acse12.htm>

## NOTAS

<sup>1</sup> Vid. <http://www.libre.org/fundacion> que propone llevar el conocimiento a una situación similar al del software libre. Entre las enciclopedias on-line wikipedia ha supuesto una revolución comunal tanto en la producción (cualquiera puede enviar una entrada) como en la posibilidad de acceso libre a un creciente repertorio de todo tipo de conocimientos ([www.wikipedia.org](http://www.wikipedia.org)) y existen versiones especializadas como la de la medicina abierta <http://www.openmedicine.com/>, la ciencia abierta, <http://biocc.ngic.re.kr/FreeHonestScienceMove/> etc.

<sup>2</sup> Langdom Winner (1983) «Do Artifacts Have Politics?» en: D. MacKenzie et al. (eds.), *The Social Shaping of Technology*, Open University Press, Philadelphia. Hay trad. esp de Mario Francisco Villa en <http://www.campus-oei.org/salactsi/winner.htm>

<sup>3</sup> Vid el trabajo de campo de Geert Lovink sobre el grupo alemán Oekonux (una mezcla entre oekonomik y Linux, <http://www.oekonux.org/>) en *My First Recession*.

<sup>4</sup> Ver Manuel de Landa, *Open Source Movement in Search of a Philosophy*

<sup>5</sup> Stephenson, Neal, *En el principio fue la línea de comandos*

<sup>6</sup> Dawkins, Richard *The Selfish Gene*

<sup>7</sup> Dennett, Daniel, *La peligrosa idea de Darwin*

<sup>8</sup> El anuncio histórico del lanzamiento del proyecto GNU se encuentra en <http://www.gnu.org/gnu/initial-announcement.es.html>

<sup>9</sup> Se trata del estatuto de la reina Ana de 1710 y pensado para animar a escritores y creadores en general con la perspectiva de una recompensa para su talento y esfuerzo.

<sup>10</sup> ver <http://www.opensource.org/docs/history.html>

<sup>11</sup> Raymond, Eric *The Revenge of the Hackers*,

<sup>12</sup> J. Echeverría: «Ética en Internet», (recensión de Pekka Himanen, La ética del hacker y el espíritu de la era de la información)

<sup>13</sup> Eco, Umberto, *Obra Abierta*, Uno de los méritos de Eco es haber empleado el término «abierto» bastante antes de la avalancha de cosas «abiertas» en las que vivimos actualmente.

<sup>14</sup> Landow, George, P *Hypertext: The Convergence of Contemporary Critical Theory and Technology*. Baltimore: John Hopkins University Press y *HyperText/Theory*. Ver asimismo el trabajo imprescindible de Enrique Santos Unamuno (2002) *Laberintos de papel: Jorge Luis Borges e Italo Calvino en la era digital*, donde se hace un serio seguimiento de la hipertextualidad y su relación con Borges y Calvino, dos de los emblemas de la «obra abierta hipertextual».

<sup>15</sup> Woodmansee, Martha, *The Genius and the Copyright: Economic and Legal Conditions of the Emergence of the 'Author'* Es interesante comprobar cómo antes del siglo XVIII, las obras literarias se consideraban como la aplicación de una serie de normas literarias legadas por los antiguos o eran resultado de la inspiración divina, pero en ningún caso resultado de la personalidad individual del autor.

<sup>16</sup> Himanen, Pekka. *La ética de hacker*.

<sup>17</sup> Ésta es la entrada del Oxford America's Dictionaries:

1 informal an enthusiastic and skillful computer programmer or user.

- a person who uses computers to gain unauthorized access to data.

2 a person or thing that hacks or cuts roughly.

<sup>18</sup> Aquí se pueden ver las bromas o hacks en el MIT durante varias décadas <http://hacks.mit.edu/Hacks/>

<sup>19</sup> En el sentido de que los individuos que componen la comunidad hacker se adhieren a una serie de valores por medio de acciones y decisiones prácticas, o dicho de otra manera,

comparten un espíritu común, un conocimiento práctico tácito.

<sup>20</sup> Levy, Steven, *Hackers, the True Computer Heroes*.

<sup>21</sup> Ibid. pp. 39-50

<sup>22</sup> <http://www.gnu.org/philosophy/shouldbe-free.es.html>

<sup>23</sup> <http://www.stallman.org/articles/biopiracy.html>

<sup>24</sup> <http://www.univ-orleans.fr/EXT/ASTEX/astex/doc/fr/adrut/html/acse12.htm>

<sup>25</sup> Lovink, op. cit.

<sup>26</sup> Vid. Floridi, Luciano, *Ética de la Información, Su Naturaleza y Alcance*, en este mismo número

<sup>27</sup> «Usamos derechos privados para crear bienes públicos, obras creativas liberadas para ciertos usos. Como el software libre y el código abierto, nuestros fines son cooperativos y para la comunidad, pero nuestros medios son voluntarios y libertarios. Trabajamos para ofrecer a los creadores lo mejor de los dos mundos,, una forma de proteger su trabajo al tiempo que animamos ciertos usos». En <http://creativecommons.org/about/history>