

# Una aproximación sociológica al fenómeno del Software Libre

Enrique Fernández Macías

Departamento de Sociología, Universidad de Salamanca

Salamanca, 27 de agosto de 2001

Versión 1.0

[efm@usal.es](mailto:efm@usal.es)

## **Introducción**

En el sector más potente y sólido de la *nueva economía*, en la industria del software, ha surgido una alternativa *desmercantilizada* al modelo de mercado imperante que está consiguiendo –por el momento– unos éxitos realmente sorprendentes: el software libre (a partir de ahora FS –por Free Software–). El relativo éxito de esta “alternativa” es sorprendente por cuanto no cuestiona el poder de una compañía en el mercado del software, ni siquiera la estructura de este mercado, sino el propio hecho de que sea mercado: lo que cuestiona (tanto a nivel moral como a nivel práctico) el FS es la legitimidad de la propiedad intelectual y del modelo de mercado para la producción de programas informáticos (FSF 1993, Stallman 2000).

La idea del FS es desarrollar todo un sistema alternativo al de mercado de programas *de libre distribución y modificación*, creados sin ánimo de lucro por programadores (profesionales o aficionados) voluntarios. Esta idea surgió (como veremos más adelante) a principios de los ochenta, pero ha sido en los últimos ocho o diez años cuando ha alcanzado un desarrollo gigantesco. El FS ha pasado de ser una alternativa anecdótica e inofensiva al modelo de mercado en la producción de software a ser una alternativa sólida y cada vez más popular, apoyada por buena parte del mundo informático y cada vez en mayor medida por el usuario medio.

El sistema de FS por excelencia es GNU/Linux, que existe desde mediados de los noventa. Para marzo de 1998, Red Hat estima el número de usuarios de Linux<sup>1</sup> en 7.500.000 (Young 1998). En el 2000, The Linux Counter da una cifra superior a 11.000.000 de usuarios<sup>2</sup>. El crecimiento en los últimos años ha sido realmente espectacular, sobre todo teniendo en cuenta que se trata de un sistema que ha partido de cero hace muchos menos años que los sistemas propietarios (Windows, Mac o Unix). Pero donde el crecimiento ha sido mayor ha sido en el mercado profesional. Un reciente estudio de IDC (una de las más importantes consultoras especializadas en nuevas tecnologías) a 800 grandes compañías estima que para principios de 2002 el gasto de estas compañías en implementar FS en sus sistemas (sobre el total de gasto en software) llegará casi al 10%. La misma consultora estimaba en un 3% la misma cifra para 1999. La mitad de las empresas encuestadas, además, decían estar usando o experimentando con GNU/Linux, por sus bajísimos costes y su fiabilidad y rendimiento. IDC también estimaba que uno de cada tres servidores de internet utilizaban GNU/Linux<sup>3</sup>.

En mercados más específicos la presencia de GNU/Linux también está creciendo espectacularmente. El mercado de producción de efectos especiales y entretenimiento, por ejemplo, está dando un vuelco hacia sistemas Linux por su bajo coste y su buen rendimiento en 3D y creación de gráficos y animaciones<sup>4</sup>. Otro ejemplo significativo lo encontramos en el mercado de sistemas operativos para equipos y herramientas

---

<sup>1</sup> Siempre es difícil de estimar el número de usuarios de un sistema de software. La estimación de usuarios de Linux de Red Hat es normalmente la más respetada.

<sup>2</sup> Estimación sacada de Linux Counter: <http://counter.li.org>

<sup>3</sup> Electronic Commerce Times, 15 de agosto de 2001 ([www.ecommercetimes.com](http://www.ecommercetimes.com)).

<sup>4</sup> ZD-Net News, 15 de agosto de 2001 ([www.zdnet.com](http://www.zdnet.com)).

industriales especializadas, donde el uso de GNU/Linux está creciendo rápidamente en los últimos años<sup>5</sup>.

Incluso muchas grandes compañías informáticas (de hardware) están apoyando de una u otra manera estas alternativas dentro de sus estrategias de mercado. La megacorporación informática por excelencia, IBM, ha anunciado para este año una inversión de 1 billón de dólares en proyectos relacionados con el sistema Linux, así como el uso de este sistema para su línea de servidores iSeries y para su línea de ordenadores especializados en usos gráfico y multimedia<sup>6</sup>.

¿Cómo es posible que, en contra de todos los pronósticos, una alternativa así haya conseguido hacerse un importante hueco en el mundo de la informática e incluso en el mercado de consumo de software? ¿Cómo puede una alternativa voluntaria y sin interés económico hacer frente sin arredrarse a la todopoderosa e increíblemente rentable industria del software? Eso es lo que trataremos de dilucidar en este artículo. Trataremos de hacer una aproximación sociológica a las causas y factores que explican el éxito del FS, y sobre esta aproximación trataremos de evaluar las posibles consecuencias del FS sobre el futuro de la industria del software.

### ***Pero ¿en qué consiste exactamente el Software Libre?***

Se trata de una *licencia de propiedad intelectual invertida*<sup>7</sup> bajo la cual se desarrollan todo tipo de programas informáticos (el modelo más típico de licencia de FS es la GPL –General Public License- de la Free Software Foundation; FSF 1991). La

---

<sup>5</sup> Linux Devices, agosto de 2001 ([www.linuxdevices.com](http://www.linuxdevices.com)).

<sup>6</sup> ZDNet News, 14 de agosto de 2001 ([www.zdnet.com](http://www.zdnet.com))

idea básica es que los programas desarrollados y distribuidos bajo una licencia de FS deben estar abiertos a cualquier tipo de modificación por parte del usuario y deben además poder copiarse y distribuirse libremente, sin ningún tipo de restricciones legales o comerciales (FSF 1991, 2001; OSI 1997). Para asegurar que el programa en cuestión esté abierto a cualquier modificación, se debe poner a disposición del usuario el código fuente del programa e incluso las instrucciones o recomendaciones necesarias para su eventual modificación.

El que un programa bajo una licencia de FS es de libre distribución queda claro: significa que se permite (de hecho, se fomenta) su copia y distribución a nivel de usuario (lo que implica que el FS es habitualmente gratuito para el usuario<sup>8</sup>). Pero para explicar su libertad de modificación (que es un aspecto fundamental, tanto como su gratuidad, para el concepto de FS) hay que explicar unos cuantos conceptos técnicos de la programación. Todo programa informático tiene dos “estados”. Primero, en su proceso de desarrollo y programación, es “código fuente”. En este momento consiste en una serie de órdenes y algoritmos en un lenguaje de programación, y no puede ser utilizado por un usuario normal. Para que pueda ser utilizado debe ser “compilado”, proceso que convierte el programa en “ejecutable” (el segundo estado): o sea, en un programa que puede ser ejecutado en cualquier ordenador compatible (con el programa). Un programa ejecutable no puede ser modificado, sólo utilizado para los

---

<sup>7</sup> En inglés, las licencias de FS utilizan toda una serie de juegos de palabras intraducibles al español para destacar la idea de propiedad intelectual invertida: “copyleft” en lugar de copyright; “all rights reversed” en lugar de all rights reserved, etc (Stallman 1999).

<sup>8</sup> Los programas en sí son gratuitos, porque nadie puede cobrar por ellos. Lo que sí permite la licencia GPL es cobrar por la distribución del programa, por el soporte de esta distribución, por los costes de envío, etc. La propia Free Software Foundation desde el principio ha enviado programas por correo a quien los solicitara a cambio de un canon. Lo cual no quiere decir que se cobre por los programas: estos

finés para los que se programó. Las empresas de software venden los programas siempre en este segundo estado, el de “ejecutable”: el código fuente nunca sale de la empresa, está protegido (por el sistema de propiedad intelectual) y es secreto. De ese modo, un programa comercial no puede ser modificado nunca por el usuario: sólo puede ser modificado por la empresa que lo produjo. Un programa con licencia de FS, por el contrario, debe distribuirse de tal modo (o directamente en código fuente o haciendo que de cualquier manera<sup>9</sup> esté disponible para el usuario) que pueda ser modificado por cualquier usuario que quiera (y tenga los conocimientos necesarios para hacerlo).

Para evitar que un programa libre sea modificado y luego patentado bajo una licencia propietaria, la licencia de FS incluye una cláusula según la cuál todo programa desarrollado utilizando FS debe a su vez ser FS (o sea, que no se puede utilizar FS para crear un programa y luego patentarlo). Lo que esta cláusula hace es una ingeniosa inversión de las restricciones del sistema de patentes utilizándolas en su favor. Utiliza la ley de propiedad intelectual para construir una alternativa a la propiedad intelectual (FSF 1993; Stallman 1999).

Pero de esta restricción también proviene una de las principales debilidades del FS, como veremos más adelante. No se puede programar FS utilizando software propietario en cualquier momento del proceso de desarrollo. Hay que tener en cuenta que el desarrollo de programas informáticos es una tarea acumulativa y extremadamente complicada. Ningún programa se desarrolla desde cero: para desarrollar un programa informático es necesario utilizar herramientas de desarrollo que son a su vez programas producidos anteriormente. E incluso todo programa informático, para poder ser usado,

---

se siguen pudiendo distribuir libremente, o descargar gratuitamente desde internet si se tiene una conexión en condiciones, etc.

<sup>9</sup> Normalmente, se pueden descargar a través de un sitio en internet.

debe utilizarse en un sistema operativo determinado, que suele ser de por sí propiedad de una empresa. Es decir, que construir cualquier programa de software libre requiere construir primero las herramientas de programación que permitan crear los programas y el sistema operativo sobre el cual se puedan ejecutar los programas. Es necesario crear una alternativa completa a los sistemas informáticos propietarios, dando la espalda prácticamente a todo el desarrollo de software llevado a cabo desde la mercantilización del software, allá por los 70. Más adelante veremos cómo se resolvió este grave problema técnico de partida, que en un principio se pensó que imposibilitaría el éxito de la alternativa del FS.

### ***Las claves del éxito del FS***

Vamos a explicar el desarrollo del FS a través de una aproximación histórico-estructural. Histórica porque se basa en el estudio de sus orígenes y evolución, y porque el tipo de exposición que seguiré es cronológica. Pero también es estructural porque la evolución histórica del FS se analiza y explica a través de conflictos y desarrollos en la estructura social del mundo informático, en la estructura de las tecnologías de la información y en la estructura económica del mercado del software.

Así, lo que haremos será exponer la evolución del FS en tres pasos sucesivos, que a mi modo de ver explican el por qué del éxito de esta alternativa desmercantizada de producción de software. En cada uno de los pasos ha habido un factor predominante(ideológico, tecnológico y económico), que ha hecho avanzar el desarrollo de la alternativa de FS tras un bloqueo del factor anterior. Pero este desarrollo es acumulativo: cada paso se ha asentado sobre el anterior, modificándolo y alterándolo. Incluso, en un momento determinado, el factor que explica un paso adelante en el

desarrollo entra en conflicto con un factor anterior. Todas esas interrelaciones y complejidades las iremos viendo a lo largo de la explicación.

## **1- El origen del FS y el fundamento ideológico**

La producción de software no siempre ha estado organizada principalmente a través del mercado. De hecho, la mercantilización del software es un hecho bastante reciente: se llevó a cabo sobre todo a finales de los setenta y principios de los ochenta, al popularizarse los ordenadores y convertirse en un sector potencialmente muy rentable económicamente.

Antes, la programación se llevaba a cabo sobre todo en departamentos de investigación (universitarios o empresariales), que era donde se utilizaban ordenadores, siguiendo un modelo a medio camino entre la producción científica y la artesanal (Raymond, 2000; Stallman 1999). Los programadores eran más un gremio (o una comunidad, como gustan decir ellos mismos) que individuos competidores en el mercado. El modelo productivo del software era parecido al del mundo académico-científico: la contribución al desarrollo y a la innovación se premian más con el reconocimiento de la comunidad que con recompensas de tipo económico<sup>10</sup>.

Pero como hemos dicho, entre finales de los setenta y principios de los ochenta, el sistema predominante de producir software pasa a ser el mercado (Newman 1999). El paso de un modelo productivo a otro es también el paso de una estructura social a otra (con sus diferentes jerarquías y posiciones relativas), lo que inevitablemente produce una dislocación social. El dinero desplaza al prestigio como mecanismo principal de

---

<sup>10</sup> Por supuesto, este reconocimiento puede acabar por convertirse en recursos económicos si permite acceder a un mejor trabajo (lo que Bourdieu llamaría conversión de capital social en capital económico; Bourdieu, 1999). Pero el que el incentivo a la producción no sea directamente monetario sino

incentivos y de asignación de posiciones en la estructura social, sustituyendo la cooperación por la competencia como principio de organización productiva, y –lo que tal vez sea más importante aún- desplazando en las posiciones de la cúspide de esta estructura social los programadores expertos en marketing a los programadores expertos en programación. Son los desplazados<sup>11</sup> por el nuevo modelo social de mercado los que se oponen, desde el campo de la informática, al nuevo modelo de mercado<sup>12</sup>. Estos desplazados crean el proyecto GNU<sup>13</sup> y la FSF<sup>14</sup> (en 1983), que son el origen del software libre y que consiguen desarrollar en los años ochenta no sólo potentes herramientas informáticas sino una ingeniosa alternativa legal a la propiedad intelectual (son los que crean el sistema de licencias libres; FSF 1991) y una crítica al modelo de mercado en el desarrollo de software que proporcionará el sustrato ideológico a todo el movimiento del FS.

La crítica de la FSF al modelo de mercado para la producción de software y a la propiedad intelectual deja traslucir el origen que venimos comentando. Esta crítica tiene

---

de reconocimiento de la comunidad tiene importantes implicaciones, tal vez la más importante el que se fomenta (premia) más la actitud y el comportamiento cooperativo.

<sup>11</sup> No todos los antiguos gurús quedan desplazados en el nuevo modelo de mercado: algunos se consiguen adaptar al modelo de mercado formando sus propias empresas (así se creó por ejemplo Sun Microsystems). No obstante, la verdadera cúspide de la programación de mercado la ocupan programadores y programas muy discutibles en cuanto a su competencia técnica (el ejemplo paradigmático es Bill Gates y su sistema operativo Windows) pero indiscutibles en cuanto a estrategias de mercado.

<sup>12</sup> Aquí el ejemplo más claro es el gurú por excelencia del FS, Richard M. Stallman, uno de los más reputados programadores de la época “heroica” del software pre-mercado, antiguo programador del laboratorio de Inteligencia Artificial del MIT, y desde los ochenta el más acérrimo defensor y líder del proyecto del Software Libre.

<sup>13</sup> Las siglas significan *GNU's Not Unix* (se trata de un acrónimo recurrente, una broma usual entre informáticos), y designan el proyecto de crear un sistema informático completo libre, tipo Unix (Unix es un sistema operativo propietario muy respetado entre informáticos por sus virtudes técnicas y porque es compatible entre ordenadores de prácticamente cualquier tipo).

dos componentes, uno moral y otro técnico. Aunque el argumento moral es importante, es la crítica desde una perspectiva técnica (que también tiene un fondo ideológico) la más poderosa y convincente en la ideología del FS. El argumento ideológico-moral se basa en que todo usuario de un programa informático debería ser libre para usarlo, modificarlo, copiarlo o prestarlo si así lo desea (FSF 1993; Stallman 2000). Este argumento se contrapone al argumento moral habitual de la propiedad intelectual (el derecho del creador a la propiedad de su obra –y a comerciar con ella, que es lo verdaderamente importante–; Worlock 2000) y como éste, es indiscutible salvo a nivel puramente moral: la cuestión es si ha de predominar el derecho del productor o el del consumidor<sup>15</sup>.

La mayoría de las defensas de los derechos de propiedad intelectual se basan más, no obstante, en un argumento *pragmático* que en el argumento moral del derecho del autor a su obra (Weckert 1997; Strong 1994): la propiedad intelectual y el mercado son necesarios para incentivar la innovación y el progreso (para dar a los innovadores una razón para esforzarse en innovar; Fernández Enguita 1999: 141). La alternativa del FS también se basa más en un argumento “pragmático” que en el argumento moral: simplemente, el modelo de mercado bloquea y daña la innovación en el desarrollo de software (FSF 1993; Stallman 1999). Aquí la argumentación se basa en gran medida en la experiencia de la programación en la época pre-mercantilizada. En esa época, sobre la base de incentivos no monetarios (el reconocimiento de la comunidad de programadores y el propio placer del trabajo bien hecho) que aseguraban el desarrollo cooperativo de software, la producción era mucho más eficiente e innovadora que en el modelo de

---

<sup>14</sup> Free Software Foundation.

<sup>15</sup> La FSF no se opone en sí al derecho del autor al reconocimiento de la autoría de su obra, sino a su derecho a comerciar con ella, a poner barreras –comerciales o de cualquier tipo- a su uso.

mercado (Stallman 2000). El modelo de mercado, al imposibilitar la modificación y mejora del código fuente de los programas por parte de la comunidad de programadores, bloquea la mejora de los programas, la innovación y el desarrollo. El modelo de mercado y de propiedad intelectual no es eficiente ni innovador técnicamente, por muy rentable que sea en términos económicos. La alternativa propuesta es resucitar el viejo modelo de producción libre y cooperativa por la comunidad de programadores<sup>16</sup>.

A lo largo de los años ochenta y principios de los noventa, la FSF consigue considerables éxitos teniendo en cuenta la dificultad de la tarea: se trata de construir, desde cero, todo un sistema de software alternativo a los sistemas existentes en el mercado y al menos igual de bueno. Siguen una organización de la producción de software al estilo de la de los departamentos universitarios (de hecho, trabajan en locales cedidos por el MIT), y consiguen desarrollar todo un conjunto libre de herramientas de programación de gran potencia, que es utilizado por muchísimos programadores incluso de fuera del mundillo del FS por su calidad y gratuidad (Newman 1999). Pero en un momento dado, el desarrollo se estanca por falta de recursos suficientes como para crear una alternativa global a los sistemas propietarios (Raymond 2000). Concretamente, se atascan en el desarrollo del elemento fundamental para conseguir plantear una alternativa viable a los sistemas propietarios: el sistema operativo. La fuerza de trabajo de la que dispone la FSF (compuesta fundamentalmente por unos cuantos programadores entusiastas que trabajan en el desarrollo de FS en sus

---

<sup>16</sup> ¿De qué vivirían los programadores en el modelo preconizado por la FSF? De su oficio de programadores (como programadores contratados por las empresas que necesitaran de sus servicios), pero no (directamente, porque indirectamente sí que sería posible) de los programas que desarrollarían. De todos modos, en el propio modelo de mercado la grandísima mayoría de los programadores son pagados en salarios, no en royalties (los royalties los cobra la empresa; Browne 2000).

ratos libres) no pueden competir con el enorme potencial en recursos –sobre todo humanos- de las grandes empresas del software.

## **2- Internet, Linux y el fundamento tecnológico**

La alternativa del FS estaba a principios de los 90 bloqueada por un problema de recursos. La tarea de desarrollar una alternativa global al sistema de software propietario parecía técnicamente imposible. Es entonces cuando las fenomenales consecuencias sociales de una nueva tecnología permiten a la alternativa del FS superar este escollo, y convertirse en una alternativa técnicamente equiparable e incluso superior al sistema de software propietario. Esta nueva tecnología es, por supuesto, internet.

Los sistemas de redes de comunicaciones informáticas venían desarrollándose en EEUU y Europa desde los 70, pero es a mediados de los 90 cuando se convierten en un verdadero medio de comunicación de masas, con la “explosión” de internet (Leiner et al. 2000). Las características técnicas de este nuevo medio de comunicación permiten una comunicación ya no sólo bidireccional, sino incluso multidireccional; global e inmediata; y desde el principio, casi completamente desregulada<sup>17</sup> (Castells 1999).

---

<sup>17</sup> La relación entre la desregulación y la “libertad” de internet aún deben ser estudiadas en profundidad, sin caer en los tópicos al uso. La desregulación gubernamental de por sí no generó la extraordinaria libertad que han gozado las comunicaciones en internet, sobre todo en sus comienzos. Más bien esta libertad se implantó en la época en que internet era un medio de comunicación entre departamentos de investigación y en el momento de *anarquía* que siguió a la privatización de internet (en la primera mitad de los 90, Newman 1999), mientras se libraban feroces luchas entre grandes corporaciones por controlar el nuevo medio. Según el mercado se ha ido desarrollando, concentrándose el poder en internet en unas pocas compañías, los espacios de libertad se han ido reduciendo y las compañías han ido creando su propia *regulación*. Incluso los propios gobiernos están regulando (ya no la propiedad de internet, sino los límites a la libertad de expresión o cuestiones de copyright) internet, presionados por las necesidades y agendas de estas megacorporaciones de la nueva economía (véase el

Estas características “técnicas” del nuevo medio, más la extraordinaria homogeneidad social –transcontinental- de sus usuarios (las clases medias-altas a nivel mundial) favorecieron el desarrollo en internet de *comunidades virtuales* con una ideología (que se solía llamar no hace mucho cyberpunk) de tipo libertaria muy cercana a los valores de la FSF. Los entusiastas usuarios de internet se sienten parte de una comunidad libre y desregulada, en la que las cosas se comparten y son gratuitas (así era en cierta medida internet en sus comienzos –hoy ya no tanto-). Estos usuarios son en gran medida profesionales hipercualificados (sobre todo de sectores de nuevas tecnologías) y con ganas de colaborar y contribuir al desarrollo de la utopía de internet. El nacimiento de comunidades virtuales efectivas (foros, mailing lists, etc.) da a este grupo potentes canales de comunicación inmediatos y multidireccionales y con ello consolida un sistema de valores comunes, una ideología y, en fin, una estructura social. En este tipo de comunidades se suelen desarrollar unas pautas de relación en las que el prestigio (no hay otro recurso en juego) se otorga a aquél que más comparte, al que más aporta al grupo (Raymond 1998; Kollock 1999).

Es de una comunidad virtual de este tipo (concretamente, de un foro de informáticos) de donde surge Linux, que conseguirá hacer del FS una alternativa viable definitivamente. Hacia 1992, Linus Torvalds, un estudiante de informática finlandés, escribe el núcleo de un sistema operativo tipo UNIX como un ejercicio de sus estudios. En lugar de guardarlo para sí, tiene la feliz idea de enviarlo a un foro de informática de internet para que los usuarios del foro lo vean y comenten (Torvalds 1998). Desde el principio, el sistema operativo de Linus (que estaba en su primera versión incompleto y plagado de errores) es recibido con entusiasmo en el foro, y cientos de informáticos de

---

caso Napster y las crecientes regulaciones de la propiedad intelectual en las nuevas tecnologías como ejemplo).

todo el mundo lo prueban, analizan y envían sus sugerencias y modificaciones de nuevo al foro. Desde el primer momento, Linus Torvalds aplica la licencia GPL de la FSF a su sistema operativo (al que llama Linux), que pasa por tanto a ser parte del proyecto de FS. De este modo comienza un proceso de desarrollo descentralizado y en gran medida desorganizado (el propio Linus hizo las veces de coordinador), que es tan increíblemente eficiente, que para 1993 LINUX ya se había convertido en un sistema operativo completo, capaz de competir en prestaciones técnicas con cualquier sistema propietario. Este proceso de desarrollo sigue a cada vez mayor velocidad (según se va corriendo la voz hay más colaboradores), hasta el punto de que hacia mediados de los 90 se convierte en el sistema operativo más potente y robusto técnicamente de los existentes, el preferido por los informáticos de todo el mundo. De este modo, de la noche a la mañana, Linux se convierte en el sistema operativo –y como veremos, en el modelo de desarrollo- de toda la alternativa del FS.

Es este factor “tecnológico” (sería más apropiado decir las consecuencias sociales de un avance tecnológico) el que permite, pues, que el sistema de FS supere el primer escollo de los recursos y alcance una madurez técnica equiparable o incluso superior a cualquier sistema de software propietario. Las potencialidades tecnológicas y el tipo de sistemas sociales que favorecen las redes informáticas (internet) son las que explican que se desarrolle de manera “espontánea” el modelo de organización de la producción de software que siguió Linux, que supera indiscutiblemente (en cuanto modelo organizativo) al modelo que se sigue en la producción para el mercado. Hay que remarcar que es este modelo organizativo el responsable del éxito a nivel técnico del FS, más que ningún otro aspecto. Como vimos más atrás, el proyecto GNU (que hoy sigue el mismo modelo de desarrollo que Linux) se quedó estancado por falta de recursos. Siguiendo un modelo parecido al de las grandes corporaciones de software

(desarrollo centralizado y organizado en equipo) el proyecto de FS no podía competir con el mercado por falta de recursos. Pero con la llegada del modelo de Linux, de golpe, esta situación se invierte: ahora son las grandes corporaciones las que no pueden competir con la velocidad de desarrollo, capacidad de innovación y solidez técnica de la alternativa del FS.

Lo que caracteriza el modelo organizativo de Linux es: 1º- El bajo coste (para los desarrolladores) y rapidez de la colaboración en las comunidades virtuales de internet, y la gran cantidad de usuarios-colaboradores (Raymond 1998); 2º- La eficiencia (auto)organizativa de las estructuras de redes de las comunidades virtuales (Raymond 1998bis; Browne 1998); 3º- El modelo social de estas comunidades virtuales, en las que un sistema de incentivos de reconocimiento de la comunidad (status) favorece el comportamiento cooperativo, la descentralización y la horizontalidad de la cooperación (Raymond 1998bis; Torvalds 1998), aspectos todos ellos extremadamente beneficiosos para la innovación y el desarrollo.

Este modelo de desarrollo es adoptado por la mayoría de proyectos de FS, que conocen así un desarrollo descomunal en estos años. El espacio de desarrollo por excelencia de FS deja de ser el laboratorio informático y pasa a ser internet, donde tienen su sede la mayor parte de estos proyectos (cada uno suele tener su página web y su foro de discusión, incluso en muchos casos más de uno). Se crean en este momento infinidad de proyectos, existiendo así hoy una alternativa “libre” para casi cualquier programa o tipo de programa propietario existente<sup>18</sup>.

Pero en este momento, el FS se encuentra con un nuevo escollo, este ~~nenos~~ fácilmente reconocible: el contacto con el consumidor medio. El FS es ya (desde la

---

<sup>18</sup> Por poner un ejemplo conocido para sociólogos, el *clon* libre del SPSS se llama PSPP, y se viene desarrollando (dentro del proyecto GNU) desde hace tres o cuatro años.

irrupción de Linux) una alternativa potente y fiable a los sistemas propietarios, pero también es tremendamente complicada para el usuario medio, con conocimientos medios (o sea, bajos) de informática. Todo el sistema de FS no pasa aún de ser un sistema hecho por y para hackers<sup>19</sup>.

### **3- La lucha contra Microsoft, el fundamento de mercado y la iniciativa Open Source**

Desde el momento mismo de su constitución como tal, el mercado de software se ha caracterizado por una fuerte tendencia a la concentración, hasta el punto de que desde mediados de los 90, en casi todos los ámbitos del software hay un monopolio de hecho de una sola compañía: Microsoft<sup>20</sup>. Esta exagerada tendencia al monopolio deriva de dos características del mercado del software: 1º- La existencia de economías de escala especialmente importantes: los costes se concentran casi por completo en el desarrollo del producto, mientras que la distribución tiene unos costes unitarios irrisorios<sup>21</sup>; 2º- La inexistencia de un estándar compatible: al ser un campo tan desregulado (o lo que es lo mismo, tan regulado por un mercado competitivo) no existen estándares que permitan la compatibilidad y la posibilidad de compartir trabajo o herramientas de trabajo entre usuarios de programas creados por distintas compañías. Como la compatibilidad y posibilidad de compartir trabajo son absolutamente imprescindibles para que el software cumpla su utilidad, la desregulación ha llevado inevitablemente al monopolio: monopolio de la compañía que ha conseguido imponer

---

<sup>19</sup> Uso hackers en la acepción original de “programadores muy expertos e ingeniosos”, y no en la hoy habitual de piratas informáticos.

<sup>20</sup> Microsoft tiene el monopolio casi absoluto en sistemas operativos, suites ofimáticas y navegadores, probablemente los tres ámbitos más importantes del software de consumo, junto con los juegos (que es la última apuesta fuerte de Microsoft).

<sup>21</sup> Esta característica es común a otros mercados de bienes intelectuales, como la música.

su sistema como estándar. Esto es exactamente lo que ocurrió con Microsoft. Primero se hizo estándar como sistema operativo, lo que luego utilizó para ir extendiendo su dominio a otros productos (suites ofimáticas, navegadores, juegos,...).

La posición tan exageradamente dominante de Microsoft en el mercado, así como sus agresivas estrategias, le han creado muchos enemigos en el mundo de la informática. No sólo entre las compañías de software competidoras, sino (lo que es más importante) entre los principales productores de hardware, que se han visto obligados a utilizar en sus equipos los sistemas de Microsoft a cambio de licencias millonarias (que Microsoft podía imponer por su posición monopólica) que reducían sus márgenes de beneficio.

El creciente éxito y reconocimiento técnico de la alternativa del FS en el mundo de la informática hizo que estos enemigos en el mercado de Microsoft se fijaran en las posibilidades que el FS les brindaba para mejorar su posición de mercado (o lo que es lo mismo, para hacer frente a Microsoft). Las más importantes compañías de software desplazadas por Microsoft en su imparable monopolización de los productos informáticos han apoyado en los últimos años de un modo u otro la iniciativa del FS. Corel, propietaria de la suite ofimática WordPerfect (la número uno antes de que Microsoft barrierá con su Office), o Netscape, la gran perdedora (frente a Microsoft Internet Explorer) de la “Guerra de los Navegadores”, son buenos ejemplos: ambas compañías han empezado a utilizar de un modo u otro el FS dentro de su estrategia comercial<sup>22</sup>.

---

<sup>22</sup> Corel ha lanzado su propia versión de Linux, siguiendo una licencia tipo Open Source (más abajo veremos en qué consiste el Open Source), así como versiones para Linux de sus principales programas, algunos de ellos distribuidos gratuitamente a través de su sitio web. Netscape, por su parte, adoptó el modelo de desarrollo descentralizado de Linux para su navegador Navigator, en el proyecto llamado Mozilla, que también se distribuye gratuitamente.

Muchos fabricantes de hardware también empiezan a apoyar a finales de los noventa el FS. Por una parte, preocupándose por hacer sus productos compatibles con el nuevo sistema (lo que en el mundo informático es equivalente a un decidido apoyo); por otra, utilizando el FS como sistema operativo en sus productos<sup>23</sup>. La propia IBM ha incorporado recientemente programas y sistemas bajo licencia de FS a su estrategia comercial, como ya vimos más atrás.

Este repentino (y paradójico, como veremos a continuación) apoyo por parte de muchas importantes compañías informáticas ha dado al FS el último empujón para alcanzar el éxito que tratamos de explicar en este artículo. Por un lado, proporciona ingentes recursos económicos al proyecto. Pero por otro lado –y esto es lo realmente importante- el apoyo por parte de la industria informática consigue por fin acercar el FS al usuario medio. Las compañías que incluyen FS en sus estrategias comerciales se preocupan de darle publicidad y de convertirlo (centrando sus esfuerzos de desarrollo en hacer más fácil su uso) en un objeto de consumo masivo. Además, esto sirve de acicate para que la propia comunidad del FS se centre cada vez más, desde finales de los 90, en convertir la alternativa del FS en algo más manejable para el usuario medio.

Es evidente que hay una cierta paradoja en todo esto. La ideología del FS (la desarrollada por la FSF) es abiertamente contraria al sistema de mercado en el desarrollo de software. La entrada de compañías informáticas en el mundo del FS tenía tarde o temprano que chocar con el fundamento ideológico del movimiento del FS de un modo u otro. Este choque se produjo, y tuvo tal fuerza, que dividió el movimiento del FS en dos grupos enfrentados: por una parte, bajo el liderazgo de la FSF (y Richard

---

<sup>23</sup> Hasta ahora, Linux ha sido utilizado como sistema operativo preinstalado sobre todo para servidores de red y ordenadores de uso específico (más arriba poníamos el ejemplo de ordenadores para uso gráfico), así como para algunos Hand-Held PCs.

Stallman), quedó el grupo de los *irreductibles*, que siguieron manteniendo sus principios antimercado y oponiéndose a muchas de las iniciativas de las empresas que utilizaron el FS dentro de sus estrategias comerciales; por otra parte, se creó (bajo las siglas de la Open Source Initiative) un movimiento nuevo que agrupaba a todos los que creían que había que ser más tolerante con los principios de mercado y con el interés económico de las empresas de software, aunque ello supusiera abandonar parte de los principios de libertad de uso, modificación y distribución que habían constituido el núcleo de la alternativa del FS (Scoville 2001).

El acontecimiento que marcó la entrada a gran escala de las grandes empresas informáticas en el movimiento del FS fue el anuncio de Netscape en 1998 de que iba a distribuir en su sitio de internet el código fuente de su navegador (que pasaría a llamarse proyecto Mozilla) y experimentar con el modelo de desarrollo seguido por el FS. Una serie de programadores relacionados con el movimiento del FS (liderados por Eric Raymond, carismático programador y teórico del Open Source, que había sido contratado por Netscape para su proyecto Mozilla) fundaron la Open Source Initiative como respuesta a este anuncio. Según sus propios fundadores, la razón para crear esta alternativa al FS fue evitar la “actitud confrontacional” (OSI 1999) hacia el mercado de software asociada al concepto de Software Libre, para así conseguir aproximarse al mundo corporativo y hacerles ver “la superioridad del proceso abierto de desarrollo” (OSI 1999). La FSF no apoyó la iniciativa y surgió así un cisma en el movimiento del FS que cada vez parece resultar más conflictivo.

Entre el Open Source (en adelante, OS) y el FS no existen diferencias en cuanto a metodología de desarrollo de software (abierta y descentralizada). Pero de ahí en adelante, les diferencia todo. Mientras que el FS es una alternativa global al sistema de mercado en el desarrollo de software, el OS es sólo una alternativa al sistema de

organización de la producción de software (Barr 2001). El OS sí que permite programas propietarios (propiedad de una empresa) y restricciones a su uso (por ejemplo, a su uso comercial) o modificación. Básicamente lo que hace el OS es adaptar las ventajas del sistema de organizar la producción de software que seguía el FS a la producción de software en el mercado.

Las licencias OS, por tanto, no son contrarias a la propiedad intelectual, ni tampoco a los principios de mercado en la producción de software. Un programa bajo licencia OS puede ser propiedad de una empresa (por ejemplo, Mozilla es propiedad de Netscape), y esta empresa puede utilizar como quiera sus derechos de propiedad, estableciendo restricciones a su uso (por ejemplo, no cobrando por su uso a particulares pero sí a otras empresas), rentabilizándolo como medio de publicidad (como también ocurre con el caso de Mozilla) o convirtiéndolo en software propietario estándar cuando le interese o resulte más rentable<sup>24</sup>. En cambio, un programa bajo licencia de FS no es propiedad de nadie y no se puede imponer ningún tipo de restricción a su uso, modificación o distribución.

Además de todo esto, hay que destacar un último aspecto fundamental en las licencias OS. Como vimos más arriba, el desarrollo de software es una tarea acumulativa: un programa normalmente se apoya en muchos otros programas anteriores (herramientas de programación o versiones previas del mismo programa). Como la

---

<sup>24</sup> La distribución gratuita inicial de un producto no es una estrategia de mercado nueva: así se consigue una buena base de usuarios a los que luego se podrá cobrar cuando parezca oportuno. Así lo hizo por ejemplo Napster, un buen ejemplo de producto semilibre al estilo de los programas OS: partió como un sistema de compartir música entre usuarios de internet y fue saludado como el origen de un nuevo tipo de comunidad virtual. Pero desde el primer momento no fue más que una estrategia de mercado, sólo que especialmente agresiva: en cuanto hubiera suficientes usuarios, se empezaría a cobrar por el servicio. El problema es que Napster abrió la caja de Pandora: desde el principio surgieron

licencia OS sí que permite la propiedad intelectual, cualquier programa desarrollado utilizando una herramienta o programa bajo licencia OS puede ser patentado. Ya dijimos que todos los programas bajo licencia GPL (la original del FS) están protegidos contra la propiedad intelectual por una cláusula que no permite patentar ningún programa desarrollado utilizando software libre (FSF 1991). Esto evitaba la posibilidad de que una empresa se apropiara de un programa desarrollado libremente y comerciara con él. Pero en el caso del OS, esto no es así: en la mayoría de los casos, se puede patentar un programa desarrollado con software bajo licencia OS. Queda claro, por todos estos aspectos, que el OS no es una alternativa al sistema de mercado, sino la adaptación de los métodos de desarrollo de software del FS al sistema de mercado.

La gran mayoría de las empresas que se han aproximado al movimiento del software libre lo han hecho utilizando el modelo de OS, no el de la FSF. Ello explica que en los últimos años, el desarrollo del OS haya sido espectacular, probablemente mayor que el del FS. Dado que en gran medida han sido estas empresas las que han popularizado los sistemas GNU/Linux últimamente, los nuevos usuarios suelen acercarse a estos sistemas a través de la alternativa OS, no del FS tradicional. Este hecho puede determinar que el OS acabe por imponerse al FS.

### ***Conclusiones***

La crisis provocada en el mercado de software por la irrupción del FS es demasiado reciente como para poder evaluar aún sus consecuencias. Hay muchas razones para pensar, no obstante, que estas consecuencias pueden ser realmente de mucho calado: como hemos visto, lo que se ha puesto en cuestión no es el liderazgo de

---

programas alternativos dentro del modelo de FS como Gnutella (que sí es software libre), con la misma

Microsoft o la legitimidad de una estructura determinada de mercado del software, sino el propio hecho de que sea mercado, la legitimidad y eficiencia del mercado como modo de organización de la producción de software. El que una alternativa de este tipo haya conseguido mantenerse e incluso crecer a costa de los sistemas de mercado es de por sí un fenómeno excepcional, que por fuerza tendrá importantes implicaciones (sean más o menos visibles).

El FS, por una parte, parece que ya ha conseguido uno de sus objetivos de partida: resucitar un modo de producir y usar programas libre y cooperativo –frente al modo propietario y competitivo imperante desde los setenta-. Esto parece indiscutible desde el surgimiento de Linux. Pero ¿qué implicaciones ha tenido para este objetivo la popularización del FS a finales de los 90? La utopía de comunidad de usuarios-creadores de software libre no puede superar el ámbito de los expertos en informática, por el simple hecho de que no todo el mundo es experto en informática. En este sentido, la popularización del FS se ha tenido que llevar a cabo rebajando los conocimientos de informática necesarios para utilizar el FS (o sea, haciendo más fácil su uso), por lo que cada vez más usuarios de FS no pueden hacer efectiva la “libertad” de modificar el software, aunque sí la de usarlo o compartirlo. En cierto modo, se rompe una de las características fundamentales de la antigua “comunidad” del FS, la de que los usuarios son también programadores. Pero ello no quiere decir que se pierdan las ventajas del FS, ni que este se haga menos *libre*: la antigua comunidad de usuarios-programadores sigue existiendo, sólo que ahora hay una comunidad aún mayor de simples usuarios, que se benefician de la libertad de uso y distribución del software según el modelo libre. Por ello, la popularización del software no tiene por qué dañar los objetivos del movimiento del FS.

---

funcionalidad de Napster pero no son software propietario y utilizan un modelo de red descentralizado.

Pero si la popularización del FS en sí no tiene por qué dañar ese ideal de producción y utilización libre y cooperativa de conocimiento, la entrada en escena de las grandes empresas que han popularizado los sistemas libres sí que conlleva riesgos evidentes para los objetivos del FS. Ya hemos visto cómo la presión de estas empresas para hacer estos objetivos más compatibles con el sistema de mercado han resultado en una escisión de la comunidad del FS y en la entrada de criterios más mercantilistas y próximos a la propiedad intelectual en los planteamientos de buena parte de esa comunidad. Si esa tendencia (la Open Source Initiative) es la que definitivamente se impone y tiene éxito (lo que no parece nada difícil hoy en día), sus implicaciones serán muy diferentes a las que tendría el éxito del ideal tradicional de FS. Probablemente forzaría una crisis en el mercado de software y un cambio en la estructura de ese mercado, pero en absoluto su desaparición total como pretendía el FS original. El mercado de software se haría más competitivo (la abundancia de herramientas de programación gratuitas y la existencia de un sistema operativo gratuito estandarizado disminuirían los costes de desarrollo permitiendo a más empresas entrar en el mercado), volviendo a una situación en cierto modo parecida a la de sus inicios. Con lo que la consecuencia última del FS (del Open Source) sería, paradójicamente, un reforzamiento de los mecanismos de mercado en la producción de software –haciéndolo simplemente más competitivo-.

De todos modos ¿hasta qué punto se pueden transplantar los principios que han hecho tan exitoso el desarrollo del FS a un sistema de producción de software de mercado? En principio, como demuestra la iniciativa Open Source, parece que algunos aspectos del modelo se pueden adaptar a la producción de software comercial. La descentralización de la producción y el aprovechamiento de la rapidez y los bajos costes de la colaboración a través de sistemas de redes son elementos que en principio se

pueden adaptar perfectamente a la organización del trabajo en una empresa privada. Lo que parece más difícil de mantener en ese caso es la productividad derivada del sistema de incentivos a la cooperación “desinteresada” que se consigue con el modelo de comunidad de desarrolladores voluntarios del FS. Tal vez esto fuera el elemento más novedoso y la verdadera clave del éxito del desarrollo de GNU/Linux. Si así fuera, todas las ventajas del sistema sólo serían posibles en la producción de software fuera del mercado y del sistema de copyright. Esto daría una ventaja comparativa al FS frente al software propietario e incluso frente al modelo OS que podría a largo plazo darle la victoria en su batalla<sup>25</sup>.

### ***Bibliografía***

- Bourdieu, P. (1999) *Razones Prácticas*, Madrid: Anagrama.
- Barr, J. (2001) ‘Live and let license’, en *LinuxWorld*, 5/23/01.
- Browne, C. (1998) ‘Linux and Decentralized Development’, *First Monday* (firstmonday.org), vol 3, nº 3.
- Browne, C. (2001) ‘The Economics of Free Software’, Internet: <http://www.cbbrowne.com/info/freeeconviews.html>.
- Castells, M. (1999) *La era de la información: 1. La sociedad red*, Madrid: Alianza.

---

<sup>25</sup> Otra cuestión es que pasaría una vez ganada esa batalla. Un aspecto que puede ser de gran importancia, aunque muy difícil de evaluar, es la motivación y fuerza que ha otorgado al proyecto de FS el simple hecho de ser la alternativa “libre” al modelo de mercado (debo esta sugerencia a Inés Calzada). Todos los problemas de legitimidad y valoración negativa del modelo de mercado se convierten así en legitimidad y valoración positiva *a priori* del modelo alternativo, “libre”. ¿Perdería entonces productividad y capacidad de innovación el modelo del FS sin un la motivación de un enemigo contra el que luchar?

- Fernández Enguita, M. (1999) *El engranaje de la desigualdad*, Barcelona: Ariel.
- Free Software Foundation (1991) *GNU General Public License*, Boston: FSF.
- FSF (1993) *El manifiesto GNU*, Boston: FSF.
- FSF (2001) 'The Free Software Definition', Boston, FSF.
- Kollock, P. (1999) 'The Economies of Online Cooperation: Gifts and Public Goods in Cyberspace, en Smith, M y Pollock, P. (eds.) *Communities in Cyberspace*, London: Routledge.
- Leiner, B. M et al. (2000) 'A Brief History of the Internet', Internet: Internet Society ([www.isoc.org](http://www.isoc.org)).
- Newman, N. (1999) 'The Origins and Future of Open Source Software, a NetAction White Paper', San Francisco: NetAction ([www.netaction.org](http://www.netaction.org)).
- Open Source Initiative (1998) 'The Open Source Definition', Internet: [www.opensource.org](http://www.opensource.org).
- OSI (1999) 'The history of the OSI', Internet: [www.opensource.org](http://www.opensource.org).
- Raymond, E. S. (1998) 'Homesteading the Noosphere', en *First Monday* ([firstmonday.org](http://firstmonday.org)), vol 3, nº 3.
- Raymond, E. S. (1998bis) 'The Cathedral and the Bazaar', en *First Monday* ([firstmonday.org](http://firstmonday.org)), vol 3, nº 10.
- Raymond, E. S. (2000) 'A Brief History of Hackerdom', Internet: <http://www.tuxedo.org/~esr/writings/hacker-history/hacker-history.html>.
- Scoville, T (2001) 'Whence the Source: Untangling the Open Source/Free Software Debate', Internet: O'Reilly ([opensource.oreilly.com](http://opensource.oreilly.com)).
- Stallman, R. (1999) 'El proyecto GNU', Boston: FSF.

-Stallman, R. (2000) 'Copyright versus Community in the Age of Computer Networks', Boston: FSF.

-Strong, W. S. (1994) 'Copyright in the New world of Electronic Publishing', Washington DC: Annual Meeting of the Association of American University Presses.

-Torvalds, L. (1998) 'What Motivates Free Software Developers?' (entrevista), en *First Monday* (firstmonday.org), vol. 3, n° 3.

-Weckert, J. (1997) 'Intellectual Property Rights of Computer Software', en *Business Ethics*, vol. 6, n° 2, pp. 102-109.

-Worlock, D. R. (2000) 'After Content: the Business of Information in the Post-Copyright Age', en *Business Information Review*, vol. 17, n° 4, pp. 191-197.

-Young, R. F. (1998) 'Sizing the Linux Market', Internet: Red Hat paper ([www.redhat.com](http://www.redhat.com)).