

El Informe NERA analizado

Ricardo Galli
Universitat de les Illes Balears*

4 de septiembre de 2003

Resumen

Este artículo es un análisis crítico del *dossier* de David Evans y Bernard J. Reddy: “Government Preferences for Promoting Open-Source Software: A Solution in Search of A Problem” [1, 2]. El informe ha sido realizado por encargo de Microsoft con el objetivo de usarlo en sus campañas de *lobby* contra el fomento del código abierto, especialmente el software libre y la licencia GPL. Dicho informe está siendo entregado a legisladores y miembros de gobiernos, especialmente europeos, con el objetivo de detener o minimizar la promulgación de leyes que discriminen positivamente al código abierto. Como era de esperar, el informe es muy parcial y está lleno de falsas hipótesis y evidencias.

1. Introducción

Microsoft está llevando a cabo una inmensa tarea global de *lobbying* para convencer a los gobiernos y legisladores de que no elaboren leyes que favorezcan al código abierto (*open source*) y al software libre, en particular a los liberados bajo licencia GPL. Está tan alarmado Microsoft con el crecimiento y apoyo social al software libre que presionó furiosamente¹ a través de la ya conocida BSA (*Business Software Alliance*) para anular una reunión de la Organización Mundial de la Propiedad Intelectual (WIPO) que debía realizarse en Suiza con el objetivo de debatir acerca de las implicaciones del software libre [3].

Una de las acciones emprendidas por Microsoft es enviar una carpeta con una serie de informes a los políticos europeos que están vinculados con acciones legislativas a favor de una discriminación positiva del código abierto.

Uno de los informes más conocidos, o al menos que tiene más impacto, es el informe titulado *Government Preferences for Promoting Open-Source Software: A Solution in Search of a Problem*². La primera versión, y la más divulgada, de este documento es de mayo de 2002³ [1] (Figura 1). Una segunda versión —con cambios mínimos, especialmente para adecuarla mejor a un formato artículo científico-técnico— fue publicada en el volumen 9 (invierno de 2003) de *Michigan Telecommunications and Technology Law Review* [2].

* Agradezco a los miembros de la lista de Bulma (<http://bulmalug.net>), especialmente a Xosé Otero, Paco Ros, Francico J. Martínez y Giles A. Radford por sus correcciones y sugerencias de último momento.

¹Parece haber ganado, porque al día de hoy la reunión está suspendida indefinidamente.

²“Preferencia de los gobiernos para promover el software de código abierto: una solución en búsquedas de un problema”.

³Curiosamente en las mismas fechas sale publicado otro artículo en la misma línea titulado *Public Subsidies for Open Source? Some Economic Policy Issues of the Software Market*, de Klaus Schmidt y Monika Schnitzer. Los autores agradecen a Evans y Reddy por su ayuda y a la financiación de NERA.

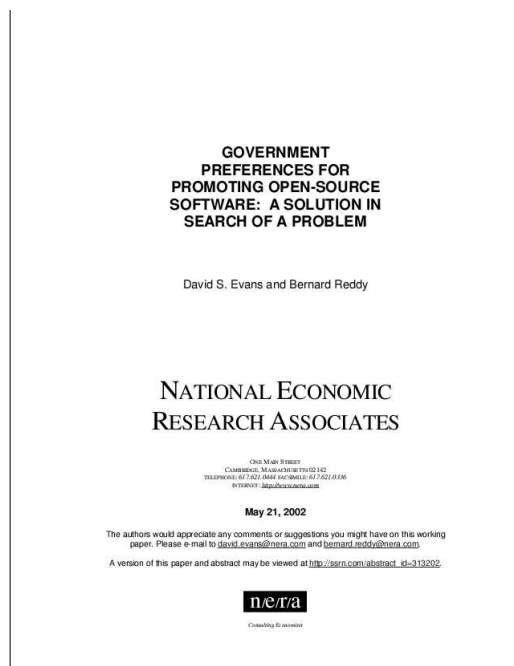


Figura 1: Informe Evans-Reddy

Como se puede observar en la Figura 1, el informe está elaborado por la empresa NERA (*National Economic Research Associates*) y escrito por David Evans —vicepresidente de NERA— y Bernard Reddy. Aunque el informe tiene la apariencia y estilo de un estudio independiente, en realidad es un informe pagado por Microsoft, tal como lo aclaran en la primera nota al pie. Como ellos mismos explican en www.neramicrosoft.com (Figura 2), NERA es una empresa que trabaja casi exclusivamente para Microsoft en áreas económicas y de monopolio. Durante todo el juicio anti-monopolio del Estado (USA) contra Microsoft, los dos autores han trabajado elaborando informes y declarando a favor de Microsoft.

Es tan estrecha la relación entre ambas empresas, o la dependencia de una respecto a la otra, que NERA no ha tenido dudas en ligar el nombre de Microsoft y su imagen corporativa a la de NERA, como se puede observar en el sitio web expresamente desarrollado (Figura 2).

Es decir, el informe NERA aparenta ser un *dossier* independiente pero está *escrito a pedido y pagado por Microsoft* con objetivos muy concretos: convencer a los legisladores de que cualquier norma que favorezca al “código abierto” es perjudicial para la economía⁴.

2. Razonamiento y conclusiones del informe NERA

El documento está redactado con un estilo innecesariamente confuso pero aparentemente bien fundamentado, sobre todo por el uso excesivo de términos técnicos económicos⁵, y la repetición de suposiciones e interpretaciones subjetivas para que parezcan hechos comprobados⁶. Desde el punto de vista de la estructura, el documento está bastante desbalanceado y mal organizado —hasta en la numeración de las secciones— que hace que al

⁴Aunque no especifica si es malo para la economía del país del legislador, para la economía de Estados Unidos o para la economía de Microsoft y NERA.

⁵Lo que los anglo-sajones llaman *mumbo jumbo*.

⁶La licencia GPL es mala, el “código abierto” no es innovador, la industria es muy innovadora...

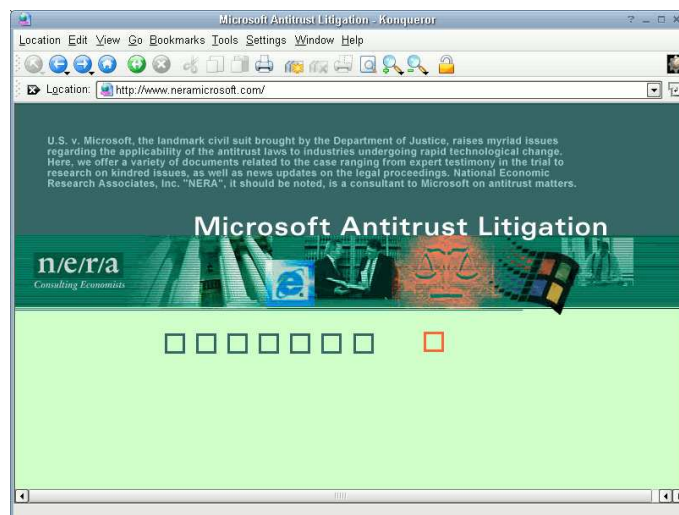


Figura 2: NeraMicrosoft.com

lector le resulte difícil ubicarse en el contexto. Aunque pretende ser un informe analítico, en realidad combina hechos con estadísticas parciales y opiniones para obtener hipótesis que luego son usadas como verdades irrefutables⁷.

Las conclusiones formales del informe pueden resumirse en:

- No hay problemas en el mercado que justifiquen la intervención del gobierno.
- La industria del software es particular, pero es igual a todas las industrias basadas en propiedad intelectual.
- El software propietario ha sido muy innovador y beneficioso para los usuarios.
- El código abierto ha tenido algunos éxitos, pero el software GPL ha tenido muy pocos.
- Los gobiernos no saben como hacerlo, por lo tanto no deben intervenir en el mercado del software.
- No creemos en las “externalidades”⁸, pero si los gobiernos las consideran es mejor que inviertan más en I+D de software, pero que no financien el desarrollo de software GPL.

¿Como llegan a todas estas conclusiones?

⁷Parece que este tipo de razonamiento ya fue descrito por Ramon Llull como “razonamiento circular” en su *Ars Magna*. Sus estudios para evitar este tipo de razonamientos le hizo concebir lo que algunos científicos consideran el antepasado primigenio de la ciencia informática.

⁸*Efecto red*. Explica el informe que gracias al efecto red los usuarios se benefician del monopolio de Microsoft, por ejemplo, todos los usuarios pueden intercambiar ficheros Word. Pero si el gobierno interviene para crear aplicaciones distintas ese beneficio se diluiría y perjudicaría a los usuarios. Obviamente intentan confundir “aplicación” con “formato de intercambio de datos”. Si hubiese diversidad de aplicaciones, además de disminuir riesgos de propagación de virus y de dar libertad a los usuarios para elegir la “aplicación”, Microsoft finalmente empezaría a hacer los filtros que hace años dejó de desarrollar. O empezaría a documentar perfectamente sus formatos propietarios o confusos y variables.

El informe tiene 79 páginas, analizarlo para explicar las contradicciones en cada párrafo necesitaría otro documento de igual o mayor longitud, así que nos centraremos en las líneas que consideramos más importantes del informe.

Aunque no está exactamente organizado de esta forma, las ideas centrales son:

1. La industria informática no está concentrada.
2. La calidad del software se incrementó, pero los precios bajaron.
3. Por **1** y **2** no existen problemas en el mercado.
4. Durante los 90 se incrementaron el número de patentes de software registradas.
5. Debido a **4**, hay innovación en el mercado del software.
6. La licencia GPL es mala para los negocios y
7. el “código abierto” no es innovador, aún menos el software desarrollado con licencia GPL.
8. Los gobiernos no deben tomar decisiones políticas, sólo técnicas y económicas.
9. Por **3**, **5**, **6**, **7** y **8** los gobiernos no deben financiar a proyectos de software GPL.

Varios de los hechos e hipótesis que presentan son —como mínimo— discutibles. Seguramente los lectores iniciados en temas de software libre, patentes y monopolios informáticos ya se están haciendo las siguientes preguntas:

- Con la evidencia del monopolio de Microsoft⁹, ¿cómo se puede justificar que la industria informática no está concentrada?
- Con el número creciente de virus que se propagan rápidamente¹⁰ por errores de diseño y programación de programas básicos como el navegador de Internet o el lector de correo electrónico más utilizado, ¿cómo pueden decir que la calidad del software se ha incrementado notablemente en los últimos años?
- Con un tema tan complejo y controvertido como es el de las patentes de software y la relación entre ellas y la innovación. Con los estudios y análisis de científicos reconocidos que presentan evidencias contrarias¹¹ ¿cómo pueden extrapolar directamente el aumento de patentes de software con mayor innovación? ¿No podría ser que antes no existían las patentes de software —o eran más complicadas de obtenerlas¹²— y por eso en los últimos años se han incrementado?
- Si la licencia GPL es relativamente joven pero ha permitido el desarrollo de una cantidad inmensa de software —disponibles de forma gratuita—, además con nuevas empresas que basan su negocio en software liberado con esta licencia ¿cómo pueden asegurar que la GPL es mala para los negocios?, ¿cuántas pequeñas y medianas empresas —hasta grandes empresas como Google, O’Reilly y Amazon— tienen toda su infraestructura funcionando con software libre y Linux?

⁹93% del Windows, entre 90% y 95% del Microsoft Office.

¹⁰MSBlast y SoBig en menos de dos semanas. Además de los problemas de seguridad de los sistemas afectados se debe al “efecto red”: la mayoría de los usuarios usan la misma aplicación.

¹¹<http://ffii.org>

¹²Los criterios se han relajado a principios de los 90.

- Si Internet, el correo electrónico, los estándares web, los sistemas operativos avanzados como Unix, editores de texto, compiladores, lenguajes de programación, etc. se han desarrollado en entornos abiertos y de “compartir el código fuente”, respetando estándares y siguiendo los principios científicos fundamentales como la revisión de los pares, la construcción acumulativa sobre los avances anteriores, ¿cómo afirman y justifican que el “código abierto” no es innovador?¹³
- Si la tarea y responsabilidad fundamental de los cargos políticos electos es precisamente tomar decisiones políticas para asegurar el bienestar general ¿cómo se puede afirmar que los gobiernos no saben y no deben tomar decisiones políticas sobre industrias y tecnologías que afectan a todas las actividades de un país desarrollado?. ¿Cómo una empresa norteamericana —condenada en su propio país por actividades monopolistas predatorias¹⁴ en el área informática— pretende que los gobiernos europeos no legislen sobre el mismo tema para evitar las desigualdades y actividades predatorias? ¿Cómo pueden sugerir que los gobernantes deben tomar decisiones como si fuesen empresas? Es decir, sólo maximizar ganancias y minimizar gastos¹⁵.

3. Análisis detallado

En las siguientes seis secciones se elaboran análisis más detallados de lo que consideramos el núcleo importante del informe NERA.

En la sección 3.1 se analiza si realmente el software propietario ha sido fuente de tantas innovaciones en los últimos años, y sobre todo si esa innovación está demostrada por el hecho de que se hayan incrementado el número de patentes de software.

En la sección 3.2 se analiza si tal como dice el informe NERA, realmente existe poca concentración en la industria del software y el porqué de la contradicción de esa supuesta baja concentración con el monopolio de Microsoft en los sistemas y aplicaciones de escritorio.

El informe asegura que en los últimos años la calidad del software ha aumentado y los precios han caído, sobre todo gracias al “monopolio natural” de Microsoft. Lo analizamos en la sección 3.3.

En la sección 3.4 se analiza la falsedad —repetida a lo largo de todo el informe NERA— de que la licencia GPL es intrínsecamente mala para los negocios.

Así como el informe asigna la innovación como característica casi exclusiva del software propietario, también argumenta la falta de innovación existente en el software libre. Estos últimos son analizados en la sección 3.5.

Finalmente la sección 3.6 analiza las exigencias para que los gobernantes no tomen decisiones políticas, ni intervengan en los mercados, ni financien proyectos de desarrollo de software GPL. Y si lo hacen, que sea BSD.

3.1. Hay más patentes, por lo tanto más innovación

El estudio de Microsoft (página 7, 20), establece una relación directa entre el número de patentes de software y la innovación, investigación y desarrollo: “como en la década de los 90 hubo un incremento de registros de patentes de software, entonces hay más innovación

¹³Hay estudios que aseguran que las tecnologías actuales fueron desarrolladas en gran parte durante los 80.

¹⁴Y con otro juicio y posible multa pendiente en la Unión Europea.

¹⁵Página 67, en la crítica a Lessig.

y la industria es floreciente”. Dado que existen estudios e informes que indican lo contrario, como mínimo debería considerarse dicha afirmación como controvertida.

El informe compara el número de patentes registradas en Estados Unidos durante la década de los 80 con las registradas en los 90. La diferencia es considerable, y de ello inferen que la innovación se ha incrementado. Pero no mencionan que los *criterios de patentabilidad se han relajado considerablemente a principios de los 90*. El más significativo de los cambios fue la modificación de las pruebas para determinar la obviedad de una invención, los nuevos criterios toman más en cuenta factores secundarios —como el éxito comercial— para calificar de “no obvia” a una invención [33].

Es como eliminar las pruebas prácticas de conducción para obtener la licencia de conducir y años después —ante el incremento más que probable del número de licencias otorgadas— inferir que existe una mejora de la educación vial de la población.

La relación entre patentes e innovación es un tema tan complejo y controvertido, que hasta Alan Greenspan —presidente de la Reserva Federal de Estados Unidos— tiene dudas de cuál es el sistema adecuado para proteger las invenciones más basadas en las ideas que en objetos físicos [10].

Carl Shapiro, especialista reconocido en temas de patentes, también se cuestiona y pregunta si el sistema legal de patentes norteamericano está correctamente diseñado para promover la investigación mas que desalentarla [9]. Shapiro también menciona que importantes investigaciones están demostrando la tendencia de las compañías a registrar nuevas patentes y elaborar estrategias muy agresivas¹⁶. El informe de Shapiro acaba afirmando que el sistema de patentes norteamericano está llevando a una situación potencialmente peligrosa, especialmente en los campos de biotecnología, semiconductores, software informático y comercio electrónico.

Es especialmente relevante un estudio muy reciente de Bessen y Hunt [8]. Según dicho estudio, si las patentes promoviesen la investigación se podría demostrar que son complementarias a la I+D. Sin embargo el estudio demuestra que en la década de los 90 el *registro de patentes ha pasado a ser un sustituto de la I+D*. Claramente las *patentes de software no son un incentivo (o actividad complementaria) de la investigación y desarrollo*.

Según Bessen y Hunt, las patentes de software en USA significan el 15% de todas las patentes, la mayoría adquiridas por grandes empresas y de manufacturación.

- El sector de manufacturación posee el 69% de las patentes de software, pero sólo emplea al 10% de los programadores.
- Por otro lado, todas las empresas del sector de “publicación de software y servicios” (que incluye a IBM) solo poseen el 16% de las patentes de software, pero emplea al 42% de los programadores y analistas.

Ello coincide con la opinión de la mayoría de ejecutivos de empresas de software que afirman que las patentes tienen poco valor para la actividad de sus empresas¹⁷ [11].

Las patentes de software pueden haber sido complementarias de la I+D en la década de los 80, cuando los estándares para obtener patentes eran relativamente altos, pero eso ha cambiado totalmente en la década de los 90, cuando las patentes se han convertido en sustitutas de la investigación y desarrollo. Según los cálculos del estudio, *la actividad de I+D hubiese sido hasta un 10% ó 15% más alta sin la sustitución de la investigación y desarrollo por las estrategias de patentes*.

¹⁶Como *patent mining* o *submarine patents*.

¹⁷Hay que destacar que William Neukom —de Microsoft— fue uno de los que más ha defendido la influencia positiva de las patentes de software para la investigación e innovación: http://www.jameshuggins.com/h/tek1/software_patent_microsoft.htm

El mayor uso de patentes de software está asociado con intensidades más bajas de I+D.

Las patentes de software han sido registradas principalmente por empresas que se dedican a construir “matorrales de patentes”¹⁸, situación que preocupa especialmente a Shapiro [9] por las dificultades de negociación de patentes a la hora de introducir un nuevo producto al mercado.

Finalmente, aunque los autores del Informe NERA utilizan el recurso de contabilizar números de patentes para convencer que la innovación florece en la industria del software, más adelante (página 54) ellos mismos ponen en entredicho todo el sistema de patentes de software cuando resumen los problemas expuestos por tres premios Nobel de Economía¹⁹:

El sistema de patentes es un método imperfecto del gobierno para remediar esos fallos [falta de diseminación del conocimiento] —el gobierno asegura a los inventores un monopolio temporal sobre sus invenciones a cambio de una *exposición completa*²⁰ [de las invenciones].

Puede afirmarse entonces que no hay evidencias de que el sistema de patentes de Estados Unidos sea el adecuado. No hay evidencias de que ese mismo sistema “relajado” de patentes de software sea complementario a las actividades de I+D en la industria del software. Por lo tanto tampoco se puede deducir que la industria del software esté en una época de innovación floreciente basados solamente en un análisis cuantitativo de patentes. Probablemente está ocurriendo exactamente lo contrario.

3.2. No hay problemas en el mercado informático

Desde el inicio los autores del informe dejan clara su postura respecto de la intervención de los gobiernos en los mercados. Ya en el segundo párrafo de la página 2 exponen:

En los últimos 20 años, la mayoría de los gobiernos han elegido incrementar su confianza en las fuerzas del mercado para gobernar la producción y distribución de bienes y servicios.²¹

Seguramente se refieren, entre otras cosas, a la liberalización de los años 80 promovida por la *Federal Communications Commission* (FCC) de USA. O al proyecto de Ley (*Billtext*) HR776 de 1992 del Congreso Norteamericano para incrementar la competencia y mejorar la “eficiencia de la producción y distribución de energía”. Hechos recientes, como los cortes generalizados de luz en todo el nordeste norteamericano [6], escándalos como Enron —la principal productora de energía de USA—, o el gran fracaso de ofrecer telefonía móvil celular compatible y con cobertura global, tal como se hizo en Europa [7]²² — que luego de casi 10 años todavía lleva una ventaja considerable a USA— plantean serias dudas de

¹⁸*Patent thickets*

¹⁹George Akerlof, Joseph Stiglitz, Michael Spence.

²⁰Cabe recordar que el software propietario no obedece —porque no está obligado— a la exposición o revelado del secreto —código fuente— para obtener una patente.

²¹Lo expresado por los primeros autores está en concordancia con la evolución de la “Escuela de Chicago”, que hasta la década de los 80 defendió la intervención de la FCC para evitar los monopolios como el de AT&T, pero ahora siguen los argumentos de “integración vertical como forma de optimizar las ‘externalidades’”. David Evans explica en [4] que la división de AT&T en las “Baby Bells” fue beneficiosa porque no innovaba —se olvidó de la invención del transistor de los laboratorios Bell, los premios Nobel, o del desarrollo del mismo UNIX—. En coherencia con la “post Escuela de Chicago” ahora defiende lo contrario: el monopolio de Microsoft es beneficioso.

²²Notar además que uno de los autores de este artículo, David Salant, es de la misma consultora NERA, y ha realizado el estudio por encargo de la misma. Aunque el estudio intenta justificar la necesidad de “no intervención” del gobierno en la telefonía móvil.

las capacidades auto-reguladoras de las “fuerzas del mercado” y de los beneficios de las decisiones liberalizadoras de infraestructuras tan básicas.

Respecto a la situación de la industria informática, en la página 12 se afirma que no existe una concentración importante en la industria del software y que los índices de concentración de otras industrias es superior a la informática. No solamente se contradice con la sentencia del juicio en USA²³ (*mantenimiento ilegal de monopolio, actos y prácticas excluyentes*), sino que en la página 73 del informe se afirma que en el año 2000 *Microsoft ha distribuido el 93 % de los sistemas operativos clientes mono-usuario*²⁴.

La falacia consiste en tomar y comparar datos estadísticos muy genéricos y de dos fuentes distintas. En primer lugar usan datos de índices de concentración del Censo de Estados Unidos de 1997 (nota al pie 44). Usan lo que llaman los cuatro dígitos SIC, en particular el 5112, que corresponde a la “Publicación y Reproducción de Software”. La oficina del Censo incluye en esta rama industrial a todas las empresas que se dedican a la producción, distribución, documentación, asistencia de instalación y servicios. Es decir, están incluidas todas aquellas empresas, grandes, pequeñas y hasta unipersonales, cuyas actividades estén relacionadas a cualquier rama de la informática, desde la programación, servicios de consultoría y mantenimiento de aplicaciones y copia de datos hasta la publicación de manuales o documentación de programas. En pocas palabras, si los *top-mantas* fueses legales, seguramente estarían contabilizados en este capítulo.

Primero toma los datos del Censo de Estados Unidos —las del código 5112—, pero la oficina del censo norteamericana no proporciona índices de concentración de la industria informática, el menos las correspondientes al censo de 1997. Para poder comparar el nivel de concentración de la industria del software con otras industrias toman los datos otro informe de características muy disímiles, el informe de IDC (www.idc.com) que está elaborado con técnicas y objetivos distintos a los del censo.

El documento de IDC referenciado (#25569) no pudo ser encontrado en el web de IDC, sin embargo se encontró uno muy similar y que posiblemente sea la versión actualizada del informe [16]²⁵. Las notas de Evans-Reddy y el resumen de [16] disponible en el web indican que la muestra para obtener los índices HHI²⁶ del informe de IDC incluyen sólo a las cien principales empresas productores de software de características y mercados muy diferentes.

Según el informe de Evans y Reddy, el informe de IDC indica que el HHI de las firmas de “paquetes de software” es igual a 244. Luego hace una comparación de los índices

²³Y la percepción y sentido común de los usuarios.

²⁴Los que conocemos generalmente como ordenadores personales, hogareños o de escritorio. Aunque comete un error en denominarlos “mono-usuarios”. Técnicamente el Windows NT, 2000, XP y 2003 son ordenadores multi-usuario.

²⁵No hemos podido leer dicho informe debido a su elevado coste —4.500 dólares—.

²⁶El índice HHI —*Herfindahl-Hirschman Index*— se calcula de la siguiente manera. Si hay tres empresas, cada una de ellas con una penetración porcentual de mercado A, B y C respectivamente, donde $A = 40$, $B = 35$, $C = 25$, el índice HHI será igual a:

$$HHI = 40^2 + 35^2 + 25^2 = 1600 + 1225 + 625 = 3,450$$

Si se consideran 10 empresas, cada una de ellas con una participación de mercado del 10 %, el HHI será:

$$HHI = 10^2 + 10^2 + \dots + 10^2 = 10 * 10^2 = 1,000$$

Si ahora se quiere calcular el HHI para una industria de 100 compañías, cada una con un 1 % del mercado:

$$HHI = 100 * 1^2 = 100$$

Se observa que el índice será más alto mientras mayor sea la concentración de mercado en menos empresas (el máximo es 10.000).

de IDC con los datos de la Oficina del Censo de Estados Unidos²⁷ y que comprende a otras industrias mucho más específicas, como los automóviles (2.506) o los cereales... ¡de desayuno! (2.446).

Como resultado de esa comparación de fuentes y metodologías distintas, deducen que la industria informática tiene unos índices de concentración muy bajos por lo que no hay deficiencias en el mercado (¿y por lo tanto Microsoft no es un monopolio?).

Pero en la página 73 indican que Microsoft distribuyó en el año 2000 el 93% de sistemas operativos de escritorio, una penetración de mercado inexistente en ningún sector importante de la industria. Haciendo cálculos rápidos (93^2) observamos que el *HHI en este sector es como mínimo 8.469*.

La otra aplicación importante de Microsoft —que le genera una parte importante de sus ingresos— es la *suite* ofimática Microsoft Office. Este conjunto de aplicaciones, que incluye el procesador de texto Word, la hoja de cálculo Excel y el programa para presentaciones PowerPoint, están presentes en los ordenadores Windows, ya sean personales o de la administración, de tal forma que los formatos propietarios de los ficheros de esos programas se usan para intercambio de datos, hasta en los ficheros que la administración pone a disposición de los usuarios. Estudios independientes [14, 15] revelan que la *suite* Office de Microsoft tiene entre un 94 y 95 por ciento del mercado, los que significa un índice *HHI de por lo menos 8.836*.

Según el mismo informe (nota al pie número 46), si el índice es superior a 2.000 suele ser una preocupación para el Departamento de Justicia y la Comisión Federal de Comercio de Estados Unidos.

Por los elevados índices que acabamos de mostrar, queda claramente reflejado que *las áreas de “sistemas operativos de escritorio” y de “suites ofimáticas” están altamente concentradas, y ambos mercados monopolizados por la misma empresa.*

Los datos comentados demuestran que la situación del mercado de la informática no puede considerarse en ningún caso como “normal”, y seguramente existen problemas de monopolios y actividades anti-competitivas.

3.3. La calidad del software aumentó y los precios bajaron

Los autores del informe NERA recurren repetidamente (páginas 16, 17, 18, ...) al argumento de que los precios —en valores nominales y “ajustados a la calidad”— han decrecido continuamente. Por un lado explican:

... la calidad del software ha crecido enormemente en los últimos años.

Esa afirmación, desde el punto de vista de la ingeniería del software es como mínimo discutible. El concepto “calidad” está interpretado en el informe como “el número de características o funcionalidades” del software, no como una medida cualitativa de los componentes ni como medida cuantitativa de la disminución de errores o disfunciones de los programas.

Por otro lado no toman en cuenta que la elaboración de “precios ajustados a la calidad” es muy compleja con el software y aún más con los sistemas operativos y *suites* ofimáticas. No se pueden separar claramente en componentes y características para comparar sus precios individuales, como se hace en la “metodología o índices hedónicos” [20, 19].

Los argumentos de precios ajustados y las referencias citadas corresponden a argumentos presentados por Microsoft en el juicio [13, 5]. Sin embargo se cuidan de no citar las audiencias, pruebas y documentación presentadas por la Federación de Consumidores de

²⁷Cabe recordar que la Oficina del Censo sólo publica los índices de concentración para empresas de manufacturación, no existen datos para la industria del software.

Precios de sistemas operativos		
	OEM	Precio en tiendas
1981	\$ 40	ND
1990	\$ 19	\$ 138
1998	\$ 50	\$ 179

Cuadro 1: Precios de sistemas operativos (Fuente: CFA)

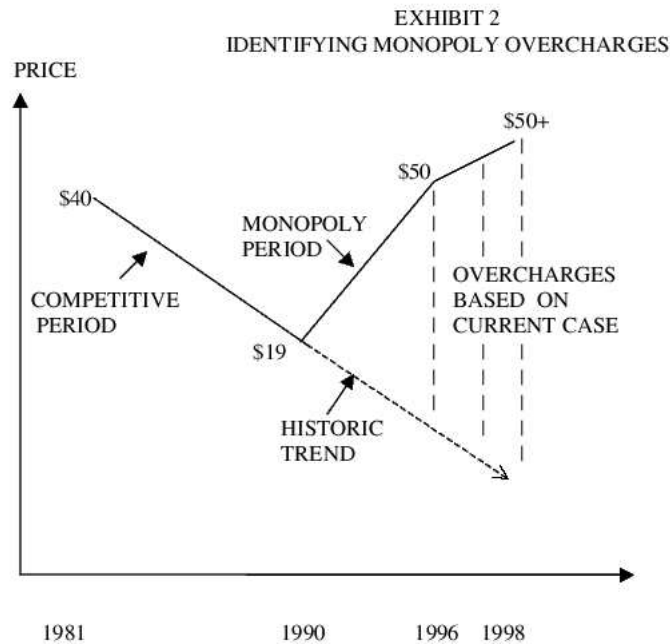


Figura 3: Sobrepuestos debidos al monopolio (Fuente: CFA)

América (CFA) [18]. Por ejemplo en la prueba número 1 (Tabla 1) muestran los precios comparativos de los sistemas operativos pre-instalados y el precio de venta en las tiendas. Esta tabla pone en evidencia que el *monopolio de Microsoft no ha favorecido la reducción de precios de sistemas operativos*.

La CFA analizó la evolución de los precios antes y después del monopolio de Microsoft y demuestra que sí existieron sobrepuestos favorecidos por el monopolio de Microsoft (Figura 3).

En otro informe de Evans y Reddy [5] se hace un estudio para justificar el precio de Windows como “competitivo”. Entre los parámetros se indica que el hardware cuesta unos 2.000 dólares, mientras que el Windows menos de 100. Esas proporciones, al menos actualmente, no se mantienen. Por ejemplo el Windows XP Professional (OEM) cuesta en una tienda 169 euro, el “MS Office XP Professional”, 399 euro. Por otro lado, un PC de muy altas prestaciones, con pantalla TFT incluida se ofrece a menos de 1000 euro (todos con IVA incluido) casi en cualquier tienda informática.

3.3.1. Los juegos de consolas también cuentan

El informe también menciona precios de productos que no tienen relación con los sistemas operativos y aplicaciones de escritorio, por ejemplo el de los videojuegos de consolas.

En la página 8 mencionan precios de aplicaciones y utilidades informáticas, pero incluyen en la comparativa a un juego de la consola Sony Play Station 2 (*Final Fantasy X*). Dicha inclusión indica un desconocimiento profundo de las diferencias de metodologías, diseño, desarrollo, grupos de trabajo y tecnologías entre desarrollar una “aplicación informática” y juegos comerciales. Algunas de las características diferenciadoras del desarrollo de juegos son:

- Ciclo de vida: los juegos tienen un ciclo de vida muy corto, hay que comercializarlo en muy poco tiempo. Mientras que el desarrollo puede durar varios años, la comercialización se realiza en un período muy corto de tiempo, no superior a 2 años.
- Presupuesto: aunque el ciclo de vida es relativamente muy corto, los presupuestos de desarrollo son enormes, similares a la producción de una película de cine, e involucran grupos interdisciplinarios y muy especializados.
- Arte, argumento, gráficos: el desarrollo de aplicaciones informáticas corrientes consiste básicamente en la especificación de los algoritmos mediante un lenguaje de alto nivel. En los juegos sólo una pequeña parte del trabajo (y del presupuesto) se dedica a las actividades de programación, el resto es escribir la historia, diseñar los árboles de navegación, desarrollar la parte gráfica, con un *componente artístico considerable* (y costoso en tiempo y dinero). Gran parte de ese desarrollo artístico debe ser realizado desde cero para posibles secuelas o nuevas versiones de los juegos.
- Sustitución del código: mientras que en las aplicaciones informáticas el código es acumulativo, los juegos suelen ser en gran parte sustitutos. Ya sea porque toda la parte artística, gráfica, argumental es nueva, o porque las tecnologías de juegos avanzan muy rápidamente.

Aún con todas las diferencias descritas, en la página 40 el informe vuelve a insistir en comparar la industria del software y los sistemas operativos con la de los juegos —y de paso insistir en las supuestas deficiencias del software con licencia GPL—:

No tenemos conocimiento de ningún juego de calidad comercial desarrollado bajo la licencia GPL.

A pesar de las diferencias y que quizás no cumplan sus “requerimientos de calidad”, podríamos citar Xpilot, GNU Go, GNU Chess, el paquete GCompris, Tuxracer, Armage-tron, Nethack y Freeciv como ejemplos de que también existen juegos con licencia GPL —algunos de ellos muy populares en la comunidad—.

3.4. La licencia GPL es mala para los negocios

En la página 10 se menciona —por primera vez— que la licencia GPL puede tener un coexistencia difícil con el software propietario. Aunque en algunos casos puede ser verdad —por ejemplo si Microsoft desea modificar e integrar módulos GPL *completos* en sus propios programas propietarios— pero no es la regla para el 95 % de las empresas que desarrollan software pero que no lo comercializan de la misma forma que lo hace Microsoft²⁸.

²⁸Se estima que un 95 % de los programas desarrollados son para uso internos (*in-house*) de las empresas.

El Linux tiene licencia GPL, pero no existe ningún problema legal para desarrollar, compilar, probar, ejecutar y comercializar aplicaciones comerciales propietarias que se ejecuten sobre Linux. De hecho, Microsoft podría adaptar y comercializar, si desearan, todas sus aplicaciones para Linux gracias a dos razones fundamentales:

1. La interacción entre las aplicaciones y el núcleo del sistema operativo se realiza a través de una interfaz perfectamente definida (“llamadas de sistemas”) y las aplicaciones que la usan no son consideradas parte ni trabajo derivado del núcleo del sistema operativo.
2. La licencia LGPL [12]. Esta licencia permite que se compilen e incluyan funciones (“edición de enlaces”) de librerías bajo licencia LGPL en programas propietarios y/o con licencias incompatibles con la GPL sin que haya conflictos. La LGPL fue diseñada y definida específicamente para evitar estos problemas. Dicha licencia suele ser usada en todas las librerías fundamentales del sistema GNU/Linux, por ejemplo en las `libc` (biblioteca de funciones elementales e imprescindibles de un sistema Unix) o en las librerías gráficas GNU/Gnome.

También hay empresas que publican y comercializan sus programas con licencias duales, GPL —si se usan para programas compatibles con dicha licencia— u otra licencia distinta —generalmente comercial y similares a las licencias comerciales de productos propietarios— si. Tal es el caso de las librerías gráficas QT de Trolltech o de bases de datos MySQL.

En la página 39 se afirma:

El “código abierto” ha sido incapaz de desarrollar software amigable para el mercado de masas.

Ya hay estudios que indican claramente que la facilidad de uso de los escritorios Gnome y KDE son igual de simples (o igual de complejos) de usar que las últimas versiones del sistema Windows [27, 28, 29].

En la misma página 39 afirman que la única *suite* informática de código abierto exitosa es “probablemente” OpenOffice, y a continuación que “es la base de un producto propietario, el StarOffice”. No sólo se contradicen en que el código abierto es “perjudicial para los negocios” dando un ejemplo contrario sino que además no mencionan otra *suite* informática que está avanzando rápidamente, KOffice.

También se han olvidado de mencionar otros productos exitosos de “mercados de masa”, tales como el Evolution, Mozilla, Konqueror, Kmail, proyecto Kolab²⁹, etc. Es notable además que los componentes de visualización Html (Khtml) y ejecución de Javascript (Kjs) del KDE, ambos con licencia GPL, han sido seleccionados para el desarrollo del navegador web de Apple, Safari[30]. Es otro ejemplo de una empresa importante de desarrollo de software que se beneficia del software libre, con licencia GPL, y sin que existan problemas legales.

El Linux y software GPL en general han servido de plataforma para otros proyectos innovadores muy importantes, como el primer grabador doméstico de vídeo digital, TiVo (www.tivo.com). Ante los requerimientos tan complejos de interacción con vídeo digital y tiempo real —grabación y reproducción simultánea— han seleccionado al Linux y herramientas GNU para su plataforma.

²⁹Resultado del *Kroupware Project*, un proyecto comercial realizado para el *Bundesamt für Sicherheit in der Informationstechnik* (Agencia Federal Alemana de Seguridad de Tecnologías de la Información). <http://kolab.kde.org>

Las intenciones de NERA de transmitir información parcial quedan nuevamente desveladas en la página 32 y la nota al pie 101. En ambas explica que Corel había dejado de dar soporte a Linux y que la última versión de su producto estrella —Wordperfect— en Windows es más reciente que la de Linux. Pero no informan que en Octubre de 2001 Microsoft hace una “alianza estratégica” con Corel [31], con una inversión estimada en 150 millones de dólares³⁰. Es la propia empresa Microsoft que ha forzado a Corel a dejar de desarrollar sus productos para Linux. Seguramente una estrategia más para mantener su monopolio en el área de las *suites* ofimáticas.

3.5. El “código abierto” no es innovador, aún menos la GPL

En diversas partes del informe se menciona la falta de innovación en el “código abierto”, especialmente del software libre con licencia GPL. La primera de este tipo de afirmaciones —en la página 20— marca una clara diferencia entre la época anterior al ordenador personal (desintegrada, programas caros o menos versátiles) versus la era del PC, donde el software propietario ha florecido y con él han bajado los precios y aumentado la flexibilidad.

Pero no mencionan que el PC no es producto de la “innovación” de Microsoft, ni siquiera de IBM, sino que fue concebido por un grupo de entusiastas³¹ —miembros de lo que en ese tiempo era el “código abierto”— que, aprovechando la potencia del recién creado microprocesador, querían hacer llegar los ordenadores a cada hogar [21, 22]. El grupo más importante de esos entusiastas se hacía llamar *Homebrew Computer Club* y estaba formado entre otros por Steve Wozniak, uno de los fundadores de Apple.

Durante ese periodo “pre PC” se han definido las características principales de los sistemas operativos, las mismas que ha usado Microsoft más de 10 años después para implementar el Windows 95 y Windows NT. A principios de los 90 toda la tecnología de Internet ya estaba definida y desarrollada —tal como lo reconoce el mismo informe NERA— con software libre: BIND, Sendmail, FTP (usado por Microsoft en sus productos), Telnet (también usado por Microsoft).

También el desarrollo de la “Interfaz Gráfica de Usuario” es previa a la época de los PCs. Sus orígenes son el sistema y lenguaje de programación Smalltalk, desarrollado en Xerox Palo Alto. La primera empresa que comercializó exitosamente ordenador personales con interfaz gráfica fue Apple, fundada por uno de los miembros del *Homebrew Computer Club*.

Cuando Microsoft lanza su primer sistema gráfico como producto comercial exitoso (Windows 3.11), el sistema de Apple ya llevaba varios años en el mercado, también existían implementaciones de ese paradigma de interacción con el usuario en todos los sistemas Unix, principalmente basados en el estándar X-Window.

En otras palabras, la intervención de Microsoft —a pesar de su poderío y dominio del mercado— no ha definido las características más innovadoras de la industria, ni de los sistemas operativos, ni de las tecnologías de Internet, ni de la interfaz hombre-máquina, y ni siquiera de las *suites* ofimáticas (cuyos orígenes comerciales se sitúan en el Wordstar, dBaseII, Lotus 1-2-3).

En la página 38 se afirma que el software más conocido con licencia GPL es Linux pero que es sólo usado mayoritariamente en servidores, pero no en “clientes”³², a pesar que hay estudios que indican que Linux ya tiene una porción del mercado de ordenadores

³⁰Y bajo presión por amenazas de demandas de Microsoft a Corel por violación de patentes.

³¹*Hackers*

³²Por sistemas de escritorio.

“clientes” superior a Apple³³. El sistema Linux y las herramientas GNU/GPL se ejecutan eficientemente en una gran diversidad de plataformas —por lo menos 11 distintas, desde grandes servidores y ordenadores de escritorio a sistemas empujados y móviles, como TiVo, Zaurus, servidores inalámbricos de Linksys/Cisco— a diferencia de los productos de Microsoft, que sólo funcionan en la arquitectura Intel³⁴.

A pesar de las razones recién explicadas —compatibilidad y flexibilidad— dice el informe en la página 48 que el Linux no es innovador, sino “imitador”. Pero no se han percatado de un fenómeno inesperado generado a partir del desarrollo de Linux: *una revolución en la ingeniería del software*, objeto de estudios de informáticos y economistas [23, 24, 25, 26].

Hace sólo diez años muy poca gente —incluso los expertos en sistemas operativos— creía que se podría desarrollar algo tan complejo como un sistema operativo con gente desconocida y sólo comunicándose entre ellos a través de Internet³⁵, con una coordinación y dirección relativamente baja³⁶, sin un diseño previo y sin que haya retribución monetaria directa a los programadores.

En la página 46 del informe se afirma:

Samba es sólo un intento de clonar las funcionalidades de Windows.

¡Por supuesto! El Samba³⁷ es un software para permitir la integración de sistemas Unix con los métodos y protocolos propietarios de Microsoft. De nada serviría que se desarrollase un sistema de integración que no implementase las funcionalidades de una de las plataformas.

A continuación (página 46) dicen:

MySQL es sólo una implementación de una base de datos con características estándares (SQL).

Otra vez ¡por supuesto!, esta es una de las características fundamentales, y una de las ventajas y necesidades del código abierto: la compatibilidad con estándares es necesaria y beneficiosa. La empresa que desarrolla MySQL nunca ha querido competir en características no estándares con bases de datos comerciales como Oracle o DB2 de IBM, sólo ser una base de datos muy eficiente, de bajo coste y mantenimiento. Objetivos claramente conseguidos, al ser una de las bases de datos más usadas en sitios web dinámicos.

En la página 47 hay una frase que resume en pocas palabras las contradicciones del informe:

Hasta que Internet se comercializó, había pocas razones para que las empresas intentaran escribir software propietario para ella.

Internet se considera una de las revoluciones más importantes desde la imprenta, y como lo manifiestan en el informe NERA, las empresas de desarrollo de software propietario comercial *no han participado en el desarrollo de la misma hasta que estuvo preparada para generar beneficios monetarios a las empresas*.

Entonces ¿quién ha innovado?.

Una parte considerable del desarrollo que hizo posible Internet —desde los protocolos, pasando por el servidores de nombres (BIND), correo electrónico (Sendmail), hasta servidores web (Cern, httpd, Apache) y navegadores web (Mosaic, Amaya)— fue desarrollado

³³Las estimaciones varían entre el 2 y 5 por ciento del mercado.

³⁴O *Wintel*.

³⁵La lista de correo *Linux Kernel Mailing List* o LKML. <http://lkml.org>

³⁶Comparado a los estándares tradiciones de desarrollo de software.

³⁷<http://samba.org>

en un entorno de “código abierto” con participación mayoritaria de la comunidad científica y de entusiastas de las emergentes tecnologías de telecomunicaciones.

En la página 41, en la incompleta y reducida sección “Proyectos de código abierto en desarrollo”, el informe dice textualmente:

Parece muy claro que los usuarios finales no son la principal audiencia de los desarrolladores.

A pesar de miles de grupos de usuarios de soporte, las listas de correos para soporte y comentarios de usuarios³⁸ y el contacto directo de programadores con sus usuarios, hasta podríamos otorgarles el beneficio de la duda. Pero esas funciones las cumplen los grupos de evaluación de usabilidad [27, 28] departamentos de soporte de las empresas dedicadas al desarrollo, distribución o soporte (IBM, HP, SuSE, RedHat, Mandrake, Ximian/Novell, Sun...). ¿O es que los programadores de Microsoft tienen un contacto más directo con los usuarios finales?.

3.6. El gobierno no debe financiar a proyectos software GPL

El informe critica la intervención de la “mayoría de los gobiernos” a través de “financiación sustancial” (página 2, 75) a proyectos de software con licencia GPL. No definen el significado cuantitativo de “sustancial” ni “mayoría”. Tampoco citan ni explican las fuentes de donde han extraído esos datos.

Dada la juventud relativa de la licencia GPL y de la formalización del movimiento de “código abierto”, lo más probable es que la inmensa mayoría de financiación en proyectos de software haya estado dedicada al desarrollo de software propietario.

El informe cita en la misma página (75) el proyecto Beowulf desarrollado en la NASA como ejemplo de software desarrollado bajo GPL y por lo tanto “inservible” para las empresas. Pero Beowulf *no es un programa, es un concepto*, factible de ser implementado sobre prácticamente cualquier plataforma. La más popular es con librerías de intercambios de mensaje MPI (*Message Passing Interface*) o PVM (*Parallel Virtual Machine*), ambas disponibles para plataformas Windows y Unix, y bajo licencias BSD.

En la página 76 (nota al pie 234) también cita al proyecto de Hans Reiser, el ReiserFS, un sistema de ficheros novedoso[32] desarrollado con financiación parcial de DARPA. Reiser libera todo el código bajo licencia GPL, y esa es la razón principal de las críticas del informe. La razón fundamental es que *Microsoft no puede apropiarse de dicho código para modificarlo y desarrollarlo como producto propietario*³⁹ *sin respetar los estándares y formatos originales*⁴⁰.

El deseo de Microsoft, expresado perfectamente en el informe, es que los gobiernos sólo financien a proyectos de código abierto cuyos resultados se liberen bajo licencia BSD o de “dominio público”. Así puedan ser utilizados y modificados *sin restricciones* por empresas que no han participado en el trabajo.

La solicitud sería coherente si además solicitase que el desarrollo de software propietario financiado por las administraciones siguiesen la misma regla. Pero no lo hacen, es la expresión más elaborada de *lo tuyo es mío, lo mío es mío*.

³⁸Una rápida visita a algunos de los sitios web de la mayoría de proyectos de software libre servirá para convencer. Todas tienen enlaces a listas de correo, sistemas de gestión de *bugs*, documentación de programación y de uso.

³⁹Aunque si son cuidadosos en cómo lo hacen, hasta podrían integrarlo en sus sistemas operativos, como hizo TiVo, ATI o NVidia con sus módulos propietarios.

⁴⁰Ya existen precedentes con otros programas bajo licencia BSD, como la pila TCP/IP y Kerberos, que son parte fundamental de los sistemas de red y de autenticación de Microsoft.

Pero lo más interesante de las “evidencias” y conclusiones del informe son:

- *Los gobiernos sólo deben tomar decisiones técnicas y económicas, no políticas* (página 60).
- La sugerencia de que *los gobiernos deberían tomar decisiones basadas en principios de máxima ganancia*⁴¹, *como las empresas* (página 67).
- *La legislación a favor del código abierto implica que hay algo más que un simple debate técnico* (página 70).

En otras palabras, los gobiernos democráticos, con representantes políticos elegidos que deben dirigir a estados independientes y soberanos, no pueden ni deben debatir y tomar decisiones políticas. Sólo técnicas y económicas, aunque sean temas que afectan a tecnologías e infraestructuras básicas y transversales como la informática.

4. Conclusiones

Como se ha demostrado, el informe NERA no es un informe independiente. Está realizado por encargo de Microsoft para servir de apoyo a sus actividades de *lobbying* para evitar leyes más favorables a desarrollos de software no propietarios. Los autores del informe hace años que trabajan estrechamente con Microsoft. Posiblemente gran parte de la facturación de NERA provenga de Microsoft.

El informe, por supuesto, es favorable a Microsoft, pero está lleno de falsedades y opiniones presentadas como si fuesen hechos estudiados e irrefutables. Todo ello trufado con un lenguaje innecesariamente complicado y técnico para confundir al lector y dar la impresión de haber sido desarrollado y redactado por expertos documentados, a conciencia, y siguiendo criterios científicos.

A lo largo de la exposición no han ahorrado esfuerzos en demonizar al “código abierto” en general y al “software libre” en particular mediante falsedades y críticas extremas a la licencia GPL —creada por la *Free Software Foundation* con el simple objetivo de asegurar que las empresas que modifiquen, vendan o distribuyan programas desarrollados por la comunidad devuelvan de alguna forma el esfuerzo realizado por ella—.

El informe primero niega la evidencia que existan monopolios en la industria informática.

Luego justifica la existencia del mismo en aras de la innovación e interés de los usuarios, que adjudican casi exclusivamente al software propietario. Intenta justificar esta supuesta exclusividad en la innovación mediante una más que controvertida extrapolación de la evolución del número de patentes con un incremento de la innovación. A pesar de los informes contrarios de especialistas y de los cambios sustanciales en los criterios de evaluación de patentes de hace poco más de una década.

A continuación critica que los políticos tomen decisiones políticas y exige que se atengan a decisiones técnicas, como si dirigir un país fuese igual que dirigir un departamento de informática de una pequeña empresa sin ningún tipo de planificación a medio plazo.

Finalmente, y ante la previsión de que los estamentos políticos finalmente decidan ayudar y financiar proyectos de código abierto, exige que las licencias del software generado no sea GPL, sino preferiblemente BSD o de “dominio público”. Por la sencilla razón de que de esta forma las empresas desarrolladoras no sólo serán propietarias de patentes que impedirán desarrollar aplicaciones en software libre, sino que si se desarrolla algún proyecto

⁴¹¿Es un *lapsus*?

novedoso en código abierto lo podrán utilizar y comercializar sin la obligación de dar nada a cambio al grupo que lo ha desarrollado. Ni a la sociedad que lo ha financiado.

La “producción de iguales”⁴² nos presenta un fenómeno fascinante que podría permitirnos canalizar importantes reservas de esfuerzos creativos infra utilizados. Es de vital importancia que no lo ignoremos en los debates políticos, o mejor aún, desplacemos sus beneficios a economías que sí lo valoren, y aseguremos las condiciones institucionales que necesita para florecer.

Yochai Benkler [24]

Referencias

- [1] David Evans, Bernard J. Reddy. “Government Preferences for Promoting Open-Source Software: A Solution in Search of A Problem”. (May 24, 2002). http://papers.ssrn.com/sol3/papers.cfm?abstract_id=313202, <http://www.idei.asso.fr/Commun/Conferences/Internet/OSS2002/Papiers/Evans.pdf>
- [2] David Evans, Bernard J. Reddy. “Government Preferences for Promoting Open-Source Software: A Solution in Search of A Problem”. *9 Mich. Telecomm. Tech. L. Rev* 313 (2003) <http://www.mttl.org/volnine/evans.pdf>
- [3] Jonathan Krim. The Quiet War Over Open-Source. August 21. The Washington Post. <http://www.washingtonpost.com/ac2/wp-dyn/A23422-2003Aug20>
- [4] David Evans. “Sorry, Wrong Model”. *Washington Post*, 6 de Mayo de 2000. <http://www.washingtonpost.com/ac2/wp-dyn?pagename=article&node=&contentId=A19887-2000May6>
- [5] Bernard J. Reddy, David S. Evans, and Albert L. Nichols. “Why Does Microsoft Charge so Little for Windows?”. http://www.neramicrosoft.com/NeraDocuments/Analyses/why_does_micro.html
- [6] Dennis Kucinich. “Lights Out on Deregulation”. <http://www.lessig.org/blog/archives/001424.shtml>
- [7] Neil Gandal, David Salant. “Standardization versus Coverage in Wireless Telephone Networks”. *Leonard Waverman London Business School and NERA Economic Consulting*. <http://arxiv.org/html/cs.CY/0109055>
- [8] James Bessen, Robert M. Hunt. “An Empirical Look at Software Patents”. Research on Innovation and MIT - Federal Reserve Bank of Philadelphia. Julio 2003. <http://swpat.ffii.org/papers/bessenhunt03/>
- [9] Carl Shapiro. “Navigating the Patent Thicket: Cross Licenses, Patent Pools, and Standard Setting”. March 2001. <http://haas.berkeley.edu/~shapiro/thicket.pdf>
- [10] Alan Greenspan. “Market Economies and Rule of Law. Remarks by Chairman Alan Greenspan”. *2003 Financial Markets Conference of the Federal Reserve Bank of Atlanta, Sea Island, Georgia*. April 2003. <http://www.federalreserve.gov/BoardDocs/speeches/2003/20030404/default.htm>

⁴²Se refiere al método de producción en el software de código abierto.

- [11] *USPTO Software Patent Hearings*. <http://www.spi.org/hearing.htm>, http://www.jameshuggins.com/h/tek1/software_patent_hearings.htm
- [12] *GNU Lesser General Public License*. <http://www.gnu.org/copyleft/lesser.html>
- [13] *State of New York et al. Executive Summary. Civil Action 98-1233*. <http://www.dcd.uscourts.gov/98-1233summary.pdf>
- [14] James Maguire. "Corel Wins Market Share from Microsoft Office". *NewsFactor Network*. <http://www.newsfactor.com/perl/story/19693.html>
- [15] Nathan Taylor. "MS Office under siege". *Bulletin EdDesk*. Vol. 120 No. 43. <http://bulletin.ninemsn.com.au/bulletin/EdDesk.nsf/0/9B995147966985F2CA256C130008E1F7>
- [16] Karen D. Moser, Richard V. Heiman, Dennis Byron, Paul Mason, Stephen D. Hendrick. "Worldwide Software Market Forecast Summary, 2002-2006". *IDC Doc #27982*. <http://www.idc.com/getdoc.jhtml?containerId=27982>
- [17] The Transaction Processing Performance Council (TPC). *TPC-C Benchmark Results Version 3 Results*. http://www.tpc.org/tpcc/results/tpcc_v3_results.htm
- [18] Consumer Federation of America. *Microsoft Monopoly Pricing Costs Consumers Billions, Consumer Group Tells Joint Economic Committee*. <http://www.commondreams.org/pressreleases/june99/061599a.htm>
- [19] Hugo Hollanders, Huub Meijers. "Quality-Adjusted Prices and Software Investments: The Use of Hedonic Price Indexes". *NewKInd - New Indicators for the Knowledge Based Economy*. <http://www.researchineurope.org/newkind/documents/Quality-AdjustedPricesandSoftwareInvestments-2001-07-06.doc>
- [20] Arthur Fishman, Rafael Bob. "Product Innovations and Quality-Adjusted Prices". *Penn CARESS Working Papers from UCLA Department of Economics*. <http://econpapers.hhs.se/paper/clapenntw/814ac5493fac0655bff878eb6a40e12b.htm>
- [21] *History of Computing Industrial Era 1976*. <http://www.thocp.net/timeline/1976.htm>
- [22] Bob Lash. *Memoir of a Homebrew Computer Club Member*. <http://www.bambi.net/bob/homebrew.html>
- [23] Eric Raymond. *The Cathedral and the Bazaar*. <http://www.catb.org/~esr/writings/cathedral-bazaar/>
- [24] Yochai Benkler. *Coase's Penguin, or, Linux and The Nature of the Firm*. <http://www.benkler.org/CoasesPenguin.PDF>
- [25] Luyin Zhao, Sebastian Elbaum. "Quality assurance under the open source development model." *The Journal of Systems and Software* (2003) 65-75.
- [26] Ganesh Prasad. "Open Source-onomics: Examining some pseudo-economic arguments about Open Source." <http://linuxtoday.com/infrastructure/20010412006200PBZCY-->

- [27] *Linux Usability Test Report*. Relevantive AG. Summer 2003. http://www.relevantive.de/Linux_e.html. Resumen: http://www.linux-usability.de/download/summary_linux_usability.pdf
- [28] David M. Nichols and Michael B. Twidale. "Usability and Open Source Software". *Working Paper 10/02*, Department of Computer Science, University of Waikato, New Zealand. <http://www.cs.waikato.ac.nz/~daven/docs/oss-wp.html>
- [29] Suzanna Smith, Dave Engen, Andrea Mankoski, Nancy Frishberg, Nils Pedersen, Calum Benson. "GNOME Usability Study Report". *Sun GNOME Human Computer Interaction (HCI)*, Sun Microsystems, Inc. March, 2001. http://developer.gnome.org/projects/gup/ut1_report/report_main.html
- [30] *Apple Webcore*. Apple Developer Connection. <http://developer.apple.com/darwin/projects/webcore/>
- [31] Paul Thurrott. "Details of Corel/Microsoft deal revealed". October 13, 2001. *WinInfo*. <http://www.wininformant.com/Articles/Index.cfm?ArticleID=19509>
- [32] Ricardo Galli. "Journal File Systems in Linux". <http://bulmalug.net/body.phtml?nIdNoticia=1154>
- [33] Robert M. Hunt. "You Can Patent That? Are Patents on Computer Programs and Business Methods Good for the New Economy?". *Business Review*. Q1, 2001. Pag 5-15.