

# Análisis del Software Libre dentro de la Empresa

Carlos M. Cámara

Enero 2006

<i>Software Libre</i>	1
-----------------------	---

## Índice

<b>1. ¿Qué es el Software Libre?</b>	<b>2</b>
1.1. Qué no es software libre	3
<b>2. Protagonistas</b>	<b>3</b>
<b>3. Software Libre vs Software privativo</b>	<b>4</b>
3.1. Cuota de Mercado	5
3.1.1. Servidores Web	5
3.1.2. Servidores DNS	5
3.1.3. Sistemas Operativos	6
3.2. Estabilidad	8
3.3. Rendimiento	10
3.4. Escalabilidad	11
3.5. Seguridad	12
3.6. Coste Total de la Propiedad	14
3.6.1. Coste inicial del software	15
3.6.2. Coste del hardware	16
3.6.3. Downtime	16
3.6.4. Costes de personal	17
<b>4. Modelo de Software Libre</b>	<b>17</b>
<b>5. Tendencia actual</b>	<b>19</b>
<b>6. Debilidades del Software Libre</b>	<b>20</b>

## 1. ¿Qué es el Software Libre?

La definición de Software Libre que aporta la Wikipedia[1] es:

“Software libre es el software que, una vez obtenido, puede ser usado, copiado, estudiado, modificado y redistribuido libremente.”

Pero el concepto de Software Libre va un poco más allá y abarca a todo un movimiento filosófico creado alrededor de los programas informáticos y sus formas de desarrollo. Así, el Software Libre es una nueva forma de entender el desarrollo de programas informáticos que intenta aportar la máxima libertad posible a los desarrolladores para que así se beneficie el producto final. A pesar de lo novedoso del término y del auge que ha tomado en los últimos años, el software libre, en palabras de Richard Stallman, no hace más que rescatar la forma de trabajo de los programadores informáticos antes de la década de los 80, cuando el software no era considerado un producto, sino como un añadido que ofrecían los fabricantes de computadoras.

Debido a la ambigüedad de la denominación inglesa de este tipo de software: free software, en los primeros años del software libre hubo mucha confusión sobre el significado del término, ya que se entendía free en el sentido de gratis y no en el de libertad. Esto dio lugar a múltiples formas de entender el software libre que se distinguieron en múltiples ramas. Aquí destacamos las 3 formas principales de entender el Software Libre:

**Software gratuito:** Se caracterizan por tener un coste cero para los usuarios finales.

**Free Software:** Este tipo de programas se distribuyen junto con su código fuente y cumplen con las 3 libertades básicas de la [Free Software Foundation \(FSF\)](#).

**Open Source Software:** Este software también distribuye el código fuente pero se acoge a las normas de la [Open Source Initiative \(OSI\)](#).

En principio cuando se habla de software libre se habla indistintamente tanto de Free software como de Open Source software ya que las diferencias entre estas dos iniciativas son más filosóficas que prácticas. No obstante cabe reseñar algunos aspectos importantes de cara al rendimiento empresarial de este tipo de software: Según el texto *Why “Free Software” is better than “Open Source”* [2] que aparece en la web del proyecto GNU, el free software de Richard Stallman tiene su interés primario centrado en los aspectos éticos del software y en su forma de desarrollo, y bajo ningún concepto se puede mezclar con software privativo (aunque se suministre el código fuente) para conseguir mejores prestaciones, en todo caso se implementarán nuevas soluciones

para imitar o mejorar las aplicaciones necesarias. El Open Source software en cambio mantiene su mirada fija en obtener un software de calidad que sea capaz de competir a nivel empresarial con cualquier software no libre y no es extraño encontrar aplicaciones mixtas que incluyen software open source y software privativo.

## 1.1. Qué no es software libre

Llegados a este punto merece la pena desmentir algunas características que se le atribuyen erróneamente al software libre:

- El Software Libre no implica que el software sea gratuito, aunque es una característica bastante común a este tipo de programas, en ningún caso es un requisito indispensable y cualquier desarrollador puede crear software libre y cobrar por él.
- Que un programa incluya su código fuente no significa que sea Software Libre, ni siquiera es software Open Source según la definición[3] dada por la OSI.
- Es posible combinar en un mismo paquetes de software privativo y de software libre.
- El Software Libre no es sólo GNU/Linux y no es necesario usar GNU/Linux para poder usar Software Libre, existe Software Libre que funciona bajo múltiples arquitecturas.

Nosotros a lo largo de este documento no haremos distinciones dentro del software libre y trataremos a ambas formas de entenderlo de la misma manera, ya que nos centraremos en el rendimiento de las aplicaciones y los costes que significan para una empresa.

## 2. Protagonistas

En la última década la cantidad de software libre existente en el mercado se ha disparado a medida que los programadores se han ido encontrando más cómodos con los términos de distribución del software y a medida que las grandes empresas de software han ido adoptando este tipo de licencias entre sus productos. Así, tan sólo en sourceforge[4] a finales del mes de diciembre del 2005 existían alrededor de 109.059 proyectos de software libre. No obstante dentro del software libre hay algunos programas y sistemas que han conseguido que su nombre se asocie a este tipo de software ya sea por su antigüedad o por su rápida aceptación entre los usuarios. Algunas de estas aplicaciones son:

**Apache** es un servidor web y de conexiones seguras SSL y posiblemente sea la aplicación de software libre que mejor acogida ha tenido en el mundo empresarial (ver sección 3.1.1).

**GNU/Linux** es un sistema operativo libre tipo UNIX que hace uso de las herramientas desarrolladas por el proyecto GNU y un núcleo diseñado por Linus Torvalds. Por abuso del lenguaje se le suele conocer como Linux y es la terminología que usaremos

**FreeBSD** es otro sistema operativo libre tipo UNIX pero al contrario que Linux que está basado en Minix, FreeBSD se basa en los sistemas BSD de la Universidad de Berkeley.

**Firefox** es un navegador web que está ganando adeptos muy rápidamente dentro de la comunidad internauta (ver sección 3.1)

**OpenOffice.org** es una suite de ofimática basada en la suite ofimática de Sun Microsystems Star Office. A mediados del año 2000 la compañía liberó el código de Star Office y así nació el proyecto.

**MySQL** es una base de datos relacional muy usada dentro del mundo de los portales de internet debido sobretodo a su fácil integración con el lenguaje dinámico PHP.

Estos programas son cada día más comunes dentro del mundo de las computadoras. Tanto es así que en ocasiones se ha llegado a identificar a algunos paquetes de software como los únicos miembros de la comunidad del software libre (ver sección 1.1).

### 3. Software Libre vs Software privativo

A lo largo de estos años se ha ido creando alrededor de los distintos tipos de software una reputación sobre su calidad y estabilidad entre los usuarios de computadores. Esta fama en muchos casos inmerecida se ha basado siempre en la experiencia personal de cada usuario y en muchos casos no ha tenido en cuenta factores tan importantes como el estado del hardware o el uso que se le estaba dando a los computadores. Afortunadamente son muchas las empresas que desde mediados de los 90 se dedican a auditar los distintos paquetes de software que existen en el mercado dejándonos constancia de datos concretos sobre los que poder asentar nuestras opiniones. Así, temas clave para la elección de un tipo de software como son la seguridad, sus costes reales, la facilidad de su uso, no quedan sometidos a la opinión de un grupo de usuarios. Aquí mostramos algunos de los informes más destacados y actuales del software para intentar establecer una opinión sobre la calidad de los distintos tipos de software.

### 3.1. Cuota de Mercado

La cuota de mercado de un producto es un coeficiente que consiste en la relación del número de ventas que una empresa ha realizado del mismo entre el número total de ventas que hay disponibles en el mercado. En nuestro contexto debido a que en muchos casos no se realiza la venta del producto consideraremos la relación entre el número de unidades de software que se está usando entre el número de software total usado. Para ver con mayor el estado del software libre hemos dividido los estudios en base al software que comparaban.

#### 3.1.1. Servidores Web

Si examinamos los datos facilitados por netcraft [5] del software utilizado en los servidores web podemos obtener algunos datos interesantes:

*Apache es el servidor web dominante*

Analizando los datos de servidores activos , vemos que con un 69,36 % de cuota, los servidores Apache dominan el mercado frente al tímido 24,31 % de su competidor más inmediato que son la familia de servidores web de Microsoft.

Si además analizamos los datos históricos podemos ver que prácticamente desde su creación los servidores Apache se situaron en una posición dominante dentro del mercado.

Uno de los principales motivos de esta subida lo podemos encontrar en la gran estabilidad que ofrecen estos servidores. Si nos fijamos en la lista de servidores más confiables en el mes de diciembre facilitada por netcraft, en la figura [?].

De estos 10 servidores, tan sólo 2 utilizan servidores de microsoft, y el resto utilizan servidores Apache bajo sistemas tipo Unix, ya sean libres o como en el caso de [www.above.net](http://www.above.net) comerciales.

#### 3.1.2. Servidores DNS

Dentro del mercado de los servidores DNS la aplicación dominante dentro del software libre es BIND. Así la encuesta [12] consultó a 646,524 servidores de DNS de internet y encontró que más del 70 % de los servidores usaban BIND (ver figura 2).

Vemos que la amplia mayoría de servidores confían en el software libre para realizar su cometido.

Rank	Performance graph	Company site	OS	Outage hh:mm:ss	Failed Req% ▼	DNS	Connect	First byte	Total
1	www.hostway.com	www.hostway.com	Linux	0:00:00	0.01	0.229	0.082	0.173	0.342
2	www.nyi.net	New York Internet	FreeBSD	0:00:00	0.03	0.204	0.071	0.144	0.354
3	www.above.net	www.above.net	unknown	0:00:00	0.04	0.084	0.058	0.121	0.184
4	www.inetu.net	INetU	FreeBSD	0:00:00	0.04	0.061	0.089	0.191	0.522
5	www.demon.net	Demon Internet	Solaris 9/10	0:00:00	0.04	0.085	0.102	0.207	0.493
6	myhosting.com	myhosting.com	Windows Server 2003	0:00:00	0.05	0.046	0.050	0.109	0.358
7	www.ipowerweb.net	www.ipowerweb.net	FreeBSD	0:00:00	0.05	0.414	0.068	0.138	0.346
8	www.hostingzoom.com	hostingzoom.com	Linux	0:00:00	0.05	0.034	0.081	0.166	0.506
9	www.interland.com	Interland	Windows 2000	0:00:00	0.05	0.673	0.083	0.210	0.493
10	www.1and1.com	www.1and1.com	Linux	0:00:00	0.06	0.141	0.049	0.100	0.100

Figura 1: Top 10 de los servidores con menos fallos

### 3.1.3. Sistemas Operativos

En el mundo de los Sistemas Operativos (SO) es quizá donde más destaca el software libre, no ya por las innumerables distribuciones de su software estrella GNU/Linux, sino por la enorme variedad de sistemas tipo Unix que han ido surgiendo a lo largo del tiempo, tales como FreeBSD, Minix, GNU/Hurd...

A pesar de la enorme variedad de SO libres existentes, los sistemas propietarios son mayores en número en los computadores del mundo. Así los datos de acceso al buscador google el pasado enero del 2004 dejan clara la supremacía de los sistemas operativos propietarios frente a los sistemas libres (ver figura ??)[43].

No obstante, si nos fijamos en otro tipo de aplicaciones más críticas que la navegación web nos encontramos con otros datos mucho más interesantes. En concreto cuando estamos hablando de aplicaciones que necesitan de una alta estabilidad, se igualan mucho las estadísticas. Así, según el informe *Linux Outlook 2005* [7] en el que se le preguntó a 493 empresas tecnológicas qué tipo sistemas operativos utilizarían en las distintas aplicaciones que necesitaban los resultados pueden verse en la figura 3

Podemos ver la gran diferencia que existe dentro del uso de Software Libre según las aplicaciones que se requieran. Así, en sus servidores, el porcentaje de uso que tendría cada SO se equipara, mientras que en las aplicaciones consideradas "de escritorio", los datos se decantan claramente por los sistemas propietarios.

No obstante las expectativas dentro del mercado son muy halagüeñas para el software libre

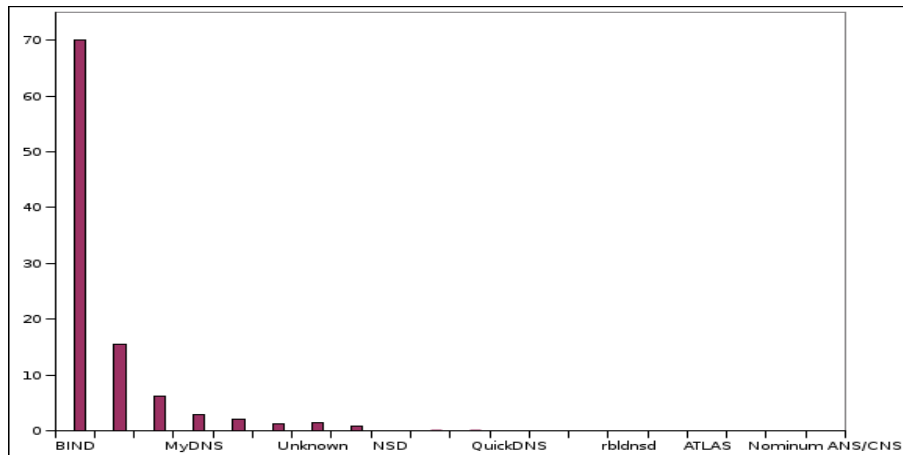


Figura 2: Porcentaje de servidores DNS en internet

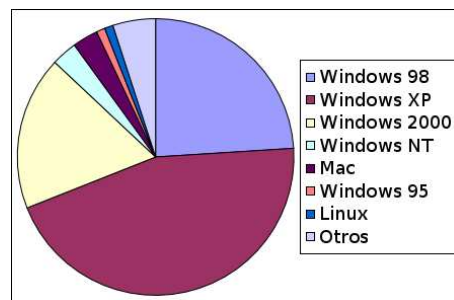


Figura 3: Porcentaje de Sistemas Operativos registrados por Google

ya que son muchas las empresas que están migrando sus SO a modelos libres, un buen ejemplo de ello son las empresas de Forrester Research Inc. que según el artículo *Linux Inc.* [11] de Bussines Magazine, el 52% de las empresas de este grupo están actualmente migrando sus servidores Windows a Linux.

### Otras aplicaciones

Pero el software libre no se restringe a estas aplicaciones que hemos expuesto anteriormente, sino que también abarca el mundo de los lenguajes de programación, navegadores web o las bases de datos. Así algunos datos importantes dentro de los mercados son:

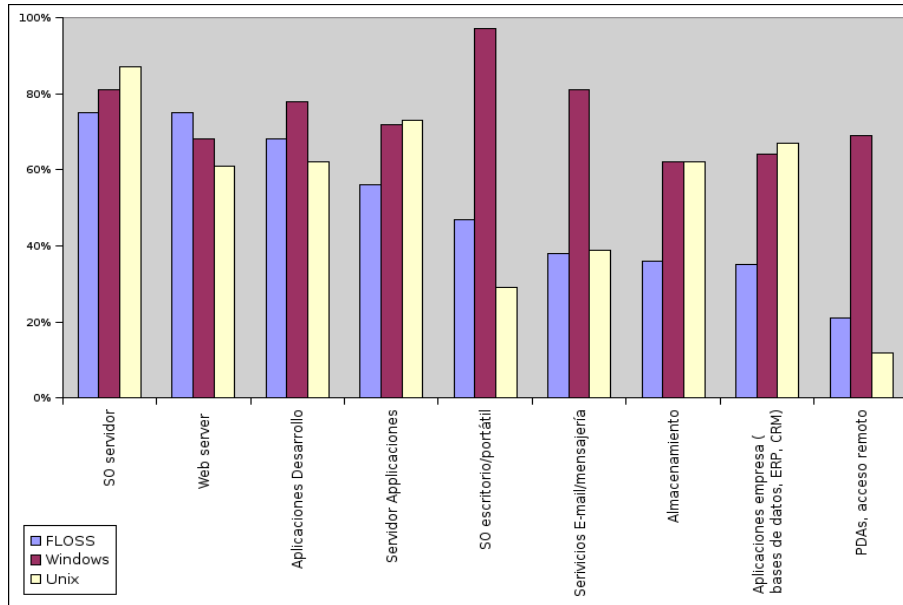


Figura 4: Datos arrojados por la encuesta de Information Week

Según una encuesta de Evans Data publicada en enero [13] las bases de datos de MySQL habían aumentado en un 30 % su cuota de mercado mientras que las de su inmediato competidor (Microsoft Windows) tan sólo habían crecido un 6 %. A estos datos hay que unir la encuesta de Evans Data [16] en la que un 64 % de los desarrolladores y administradores de bases de datos afirmaban que utilizaban soluciones libres en su trabajo.

Aunque domina el mercado desde hace tiempo con más de un 90 % de usuarios, según la revista PC-World [14], el navegador web de Microsoft Internet Explorer comenzó a perder su cartera de usuarios en favor de la solución Libre Mozilla Firefox. Y no es el único estudio que se inclina por esta conclusión, ya que estudios como [15] confirman esta tendencia.

### 3.2. Estabilidad

La estabilidad es una de las razones por las que históricamente las empresas más se han decantado por el uso de los sistemas libres y paradójicamente es uno de los mayores miedos con los que se enfrentan los ejecutivos a la hora de decidirse por este tipo de soluciones. Pero ¿Qué hay de verdad en todas las leyendas urbanas que circulan por internet sobre la estabilidad del Software Libre?

Las primeras comparativas de estabilidad que se hicieron entre SO libres y propietarios datan de 1995, cuando empezaba a despuntar Linux como SO competitivo. Así en el artículo *emphFuzz Revisited: A Re-examination of the Reliability of UNIX Utilities and Services* [17] se comparaban los distintos sistemas operativos que existían en el año 1995 con los resultados de la gráfica 4.

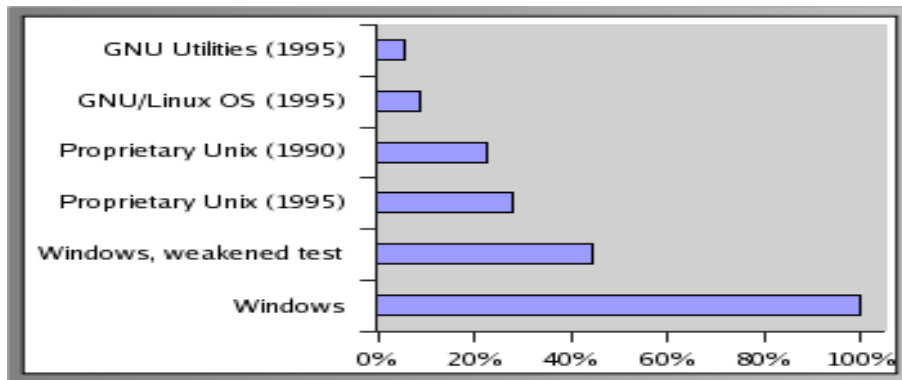


Figura 5: Primeros pasos “estables” del Software Libre

La barra de datos denominada “Windows, weakened test” se corresponde con un cambio que tuvieron que realizar en el test aplicado a los sistemas Windows para que no fallaran el 100 % de las pruebas, tal y como se ve en una fila posterior. No obstante en el año 2000 volvieron a realizar el test teniendo en cuenta tan sólo a los sistemas Windows NT y Windows 2000. Los resultados mejoraron considerablemente en lo que a Windows NT se refiere, no obstante, los resultados aportados por Windows 2000 (sustituto del NT) empeoraban con respecto al producto anterior[18]. Ver figura 5

No obstante estos datos demuestran una mejora considerable en el software propietario con respecto a sus anteriores versiones.

Posteriormente en el año 1999, *ZDNet* realizó un estudio exhaustivo de 10 meses sobre varias distribuciones Linux y un sistema Windows NT [19]. Los resultados fueron claramente favorables a las distribuciones libres ya que mientras que el sistema NT tuvo una caída cada 6 meses con un tiempo de recuperación de 30 minutos, los sistemas Linux no fallaron ni una sola vez.

Todo esto nos da una idea de por qué históricamente se ha considerado siempre que el Software Libre es más robusto que el propietario. Si seguimos ahondando en los diferentes estudios, nos encontramos con que en el año 2003 la empresa de análisis de código Reasoning hizo un estudio sobre los defectos en la implementación del protocolo TCP/IP en 3 SO propietarios, 2

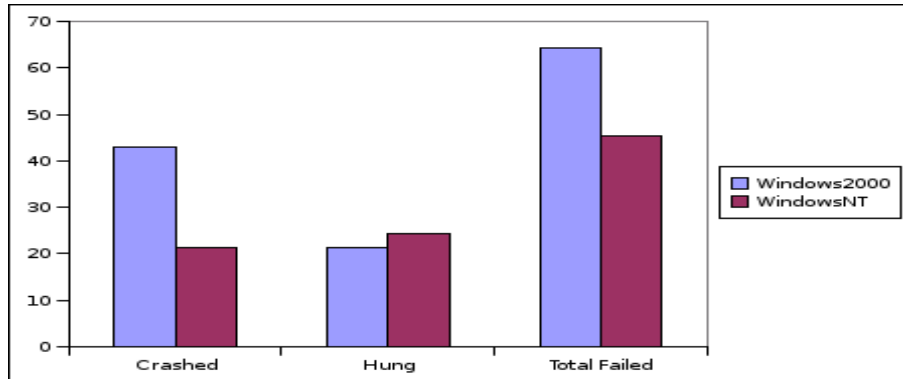


Figura 6: Comparativa entre Windows NT y su sucesor Windows 2000

SO empotrados y un SO Linux [20]. A pesar de que se esperaba una mayor estabilidad en los sistemas operativos empotrados, los resultados fueron muy favorables al sistema Linux:

SO	fallos/KSLOC
Linux	0,1
Comerciales	0,6 - 0,7
Empotrados	0,1 - 0,3

Ya en el 2004 Coverity, otra empresa de análisis de código, publicó un estudio sobre los fallos encontrados en el kernel de Linux [21]. El total de fallos encontrados fue de 985 en un total de 5,7 millones de líneas de código, mientras que según un estudio de la universidad de Carnegie Mellon, en un programa como el núcleo de Linux debería haber unos 5000 fallos. La compañía Microsoft no cedió su código fuente a Coverity para que pudiera realizar su estudio.

En el 2005 un nuevo análisis de Coverity sobre el núcleo 2.6.12 de Linux [22], demuestra que a pesar de que se han incrementado considerablemente las líneas de código con respecto a la versión 2.6.9, la tasa de fallos dentro del núcleo había descendido un 2,2%.

### 3.3. Rendimiento

Si la estabilidad es uno de los pilares del software dentro de la empresa, el rendimiento sin duda es otro de ellos. No obstante a pesar de su importancia es realmente difícil medirlo de una forma general que indique qué solución es mejor de forma global ya que es totalmente dependiente del entorno en el que se realicen los estudios. Aquí intentaremos poner algunos de

los casos más relevantes para tener una visión general de las diferencias entre los SO libres y propietarios.

En su número de noviembre del 2001 la revista PC Magazine realizó varios tests en servidores de ficheros equipados con Windows 2000 y un sistema Linux con Samba (Samba es una implementación libre del protocolo CIFS de Microsoft)[24]. Los resultados se pueden ver en las figuras 6 y 7

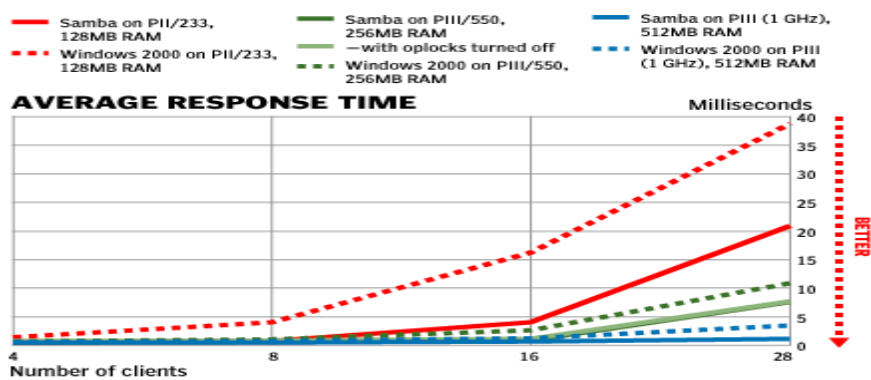


Figura 7: Tiempo de salida

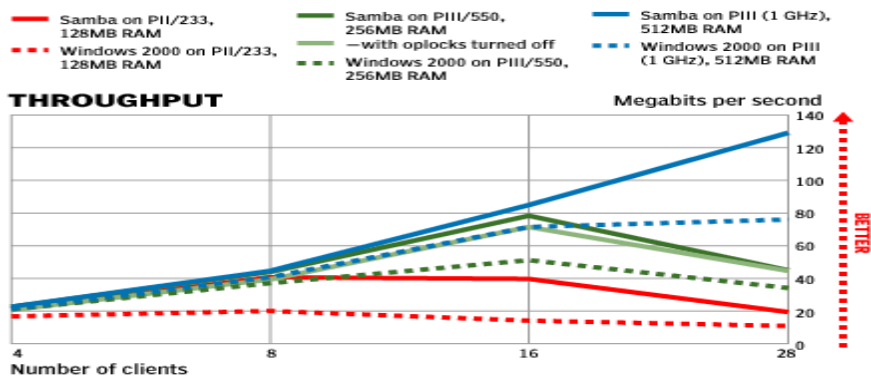


Figura 8: Flujo de datos

Vemos que los resultados se decantan por los sistemas Linux con Samba consiguiendo una mayor velocidad de respuesta en todos los casos y un mayor servicio de datos.

Un año más tarde en octubre del 2003 la revista IT Week volvió a poner a prueba a Samba contra un sistema con Windows Server 2003 [25]. La prueba estaba motivada por la inseguridad de los clientes de Microsoft ante el abandono por parte de la compañía del soporte para Windows NT. Los resultados de la prueba no obstante no podían ser más descorazonadores para la empresa de Redmon, ya que Samba 3 era 2.5 veces más rápida que un equipo Windows 2003 en las mismas condiciones. Según el informe uno de las mayores ventajas de Samba era su enorme escalabilidad, mientras que los servidores Windows 2003, aunque funcionan muy bien al principio, conforme se le van añadiendo usuarios pierden rendimiento estrepitosamente

No obstante no todo han sido hitos del Software Libre y también hay informes que demuestran sus flaquezas en el mundo empresarial. Por ejemplo *Linux and NT as Web Server on the Test Bed* [26] donde se realizó una comparativa entre Windows NT Server 4.0 y un sistema Linux con un núcleo 2.2.9. En él se encontró que los servidores Linux ofrecen peor rendimiento sirviendo páginas estáticas, en especial a conexiones rápidas, que los servidores NT. No obstante los servidores Linux mostraron un mejor rendimiento en el manejo de las páginas dinámicas que el servidor NT.

Finalmente en su artículo *Is It Time for Linux* [27] la empresa Network Computing encontró que no había apenas diferencias de velocidad entre un servidor de ficheros Samba en una distribución Red Hat Linux y un servidor Windows NT. Así el sistema Linux tan sólo podía superar el rendimiento del NT en la escritura de ficheros y afirmaban que configurando adecuadamente cada uno de los dos sistemas podía conseguirse superar al otro ligeramente.

### 3.4. Escalabilidad

Dentro de una gran empresa uno de los aspectos más importantes para conseguir ahorrar dinero en equipos y mejorar el rendimiento es la escalabilidad. En muchas ocasiones poder disponer de máquinas que intercambien información a gran escala y a gran velocidad puede suponer el ahorro de dinero y tiempo. Es por esto que es un punto muy a tener en cuenta de cara a la elección de software y es en este punto donde el Software Libre juega su mejor baza ya que el 78 % de los 500 supercomputadores que existen en el mundo utilizan Software Libre [28]. Si a esto añadimos la más que demostrada potencia del modelo de Software Libre para desarrollar grandes aplicaciones [?], en cuestiones de escalabilidad el software libre no tiene rival.

### 3.5. Seguridad

Sin duda la seguridad es uno de los puntos cardinales a la hora de elegir el software a utilizar dentro de una empresa. Hoy día la amenaza de los crackers es cada vez más fuerte y es preciso

disponer de programas que sean lo suficientemente seguros como para no facilitar los ataques de los malhechores.

Normalmente las compañías no ofrecen información sobre el asalto a sus sistemas por la desconfianza que pueda aparecer entre sus consolidados y posibles clientes, no obstante existen muchas empresas aseguradoras especializadas en asegurar la información y las pérdidas ocasionadas por los asaltos de crackers. Estas empresas exigen que se les faciliten los datos de asaltos al sistema antes de asegurar cualquier compañía y según el artículo [30], en el año 2001 la empresa aseguradora Wurzler Underwriting Managers cobraba entre un 5-15 % más a sus clientes que usaran el sistema Windows NT ya que según su vicepresidente Walter Kopf: “Hemos encontrado que la posibilidad de pérdidas es mayor usando el sistema NT. Donde las perdidas son mayores debemos incrementar la cuota.”

Según un proyecto conjunto del proyecto HoneyNet, la compañía AOL y otros participantes [31], en el año 2004 un sistema Linux sin parchear duraba 3 meses en internet sin ser atacado, mientras que un sistema Windows en las mismas condiciones tan sólo duraba unas horas o minutos antes de ser comprometido. Este estudio además es consistente con otros estudios como *Time To Live On The Network* [32] en donde no obstante también se pone de manifiesto la gran seguridad demostrada por los sistemas Windows XP en los que se había instalado un cortafuegos o se les había mantenido debidamente actualizados con el Service Pack 2.

Pero no sólo son importantes los ataques y vulnerabilidades de los sistemas, un punto muy a tener en cuenta a la hora de considerar un determinado producto es cómo de rápido responde su vendedor a los fallos de seguridad reportados. En esto los sistemas operativos libres vuelven a tener ventaja según el artículo *eWeek Labs: Open Source Quicker at Fixing Flaws* [33] tanto el proyecto Apache como el proyecto KDE (no completamente libre) consiguieron una respuesta record en sendos fallos de seguridad muy importantes, mientras que la compañía de software privativo Microsoft no solucionó sus fallos de seguridad hasta que publicó su Service Pack 1 para Windows XP. En el artículo se reconoce también que a pesar de estos casos puntuales las empresas de software privativo suelen responder de forma rápida a sus fallos de seguridad, aunque afirman que los proyectos de Software Libre suelen actuar aun más rápido.

Aunque no esté relacionado directamente con la calidad del producto, también hay que tener en cuenta la probabilidad que tienen ciertos sistemas propietarios a ser atacados frente a otros. En concreto, los sistemas operativos Windows de Microsoft son atacados un 17 % más que el conjunto de todos los sistemas operativos tipo Unix (Libres o no) según el artículo *RSA: Security in 2002 worse than 2001, exec says* [34](Ver figura 8).

Otro dato a considerar es el de la gravedad de los fallos de seguridad encontrados. Según un estudio sobre las vulnerabilidades más recientes de los sistemas Windows y Red Hat Linux, realizado por Nicholas Petreley [35], el 50 % las vulnerabilidades de los sistemas Windows son

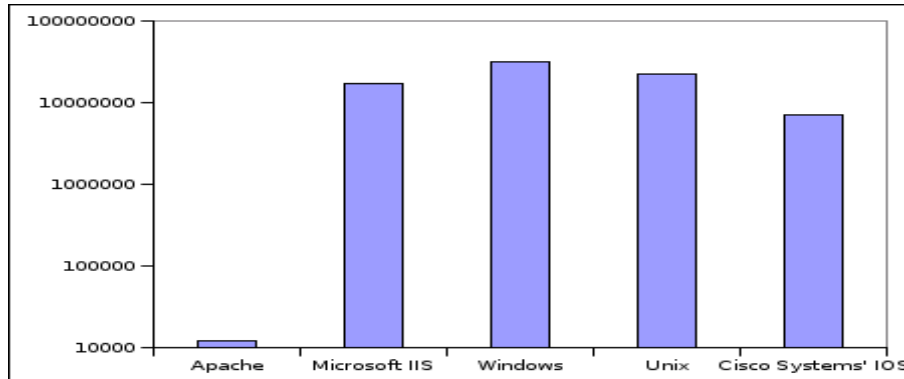


Figura 9: Número de ataques a distintos sistemas en el 2001

críticas frente al 10 % de las del sistema Linux.

Desde que en 1972 Robert Thomas Morris consiguiera infectar una máquina IBM 360, los virus informáticos se han convertido en un riesgo de seguridad muy importante para las empresas. Los datos económicos referentes a las pérdidas que una infección por virus puede causar si no es controlada son escalofriantes. Así, tan sólo el virus LoveLetter causó un perjuicio económico directo de unos 960 millones de dólares, mientras que en perjuicio indirecto debido a la productividad que no pudo llevarse a cabo la cifra asciende hasta los 7,7 billones de dólares [36]. No es por tanto desdeñable la idea de aspirar a un sistema que sea cuanto más inmune mejor a los ataques. Si echamos un vistazo a las cifras aproximadas de virus para los SO más importantes en la figura 9, podemos comprobar cómo la mayoría de los ataques están originados a sistemas windows mientras que el resto de SO mantienen unas cifras sorprendentemente bajas en relación con él.

De entre los sistemas menos atacados cabe destacar la baja incidencia que tienen los sistemas comerciales de Unix comerciales con la ventaja añadida que ninguno de estos virus ha conseguido extenderse más allá de los confines de la máquina en la que estaban y de hecho su creación fue meramente académica.

### 3.6. Coste Total de la Propiedad

Los usuarios más noveles dentro del mundo del Software Libre tienden a justificar las bondades económicas que les proporciona este tipo de software basándose en la idea de que no existe un coste por tener las licencias de los programas. No obstante, de cara a un estudio serio y

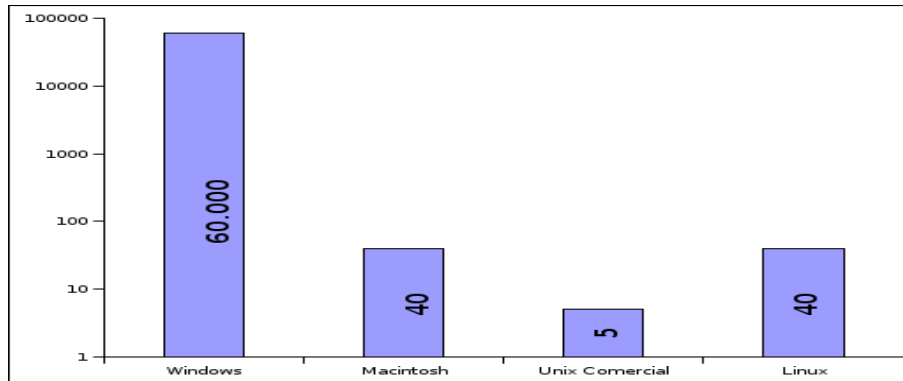


Figura 10: Datos estimados de virus para los distintos SO

detallado de los costes asociados con los programas, no podemos quedarnos simplemente en este dato para decantarnos en favor de un tipo de software u de otro. Hay que ir más allá y utilizar herramientas financieras que nos ayuden a ver si a largo plazo la inversión nos favorecerá o no. Existen multitud de análisis usados por las empresas tales como Balance Scorecard, Information Economics, Portfolio Management, Real Options Evaluation, Total Cost of Ownership, etc.. Todos estos métodos se ven muy afectados por el entorno en que se realicen y todas las empresas distribuidoras de software tienen algún estudio donde se ven claramente beneficiadas y su producto es más barato que el de la competencia. Aquí nos limitaremos a apuntar algunos datos que se utilizan en el análisis del Coste Total de la Propiedad (TCO) para hacernos una ligera idea comparativa.

### 3.6.1. Coste inicial del software

Si vemos la diferencia de precios entre sistemas de software libre y privativo:

	Microsoft Windows 2000	Red Hat Linux	Novell SUSE Linux
SO	\$1510	\$156	\$59.95
Servidor Email	\$1300	incluído	incluído
Servidor RDBMS	\$2100	incluído	incluído
Desarrollo C++	\$500	incluído	incluído

Podemos ver que los productos de Software Libre tienen un coste de adquisición más bajo que los de software privativo, debido sobretodo a la ausencia de coste por licencias. Además tienen la ventaja añadida de que no nos restringen el número de clientes que podemos tener en nuestros servidores. Si bien es cierto que estos dos paquetes de software poseen muchas características

técnicas que difieren, nos sirve para hacernos una idea del coste de dos productos parecidos que intentan cumplir la misma función.

Pero los costes iniciales de los productos no deben afectar a nuestra decisión de qué tipo de software elegir ya que los procesos de amortización contable hacen que se eliminen por completo este tipo de gastos, sí debemos tener en cuenta que las licencias del software privativo debe ser renovada cada cierto tiempo según los términos del contrato de compra, mientras que el software libre tiene licencia ilimitada.

Cabe destacar también que los SO libres aquí comparados pueden obtenerse también a precios más bajos o incluso descargarse gratuitamente de la red (en algunos casos con menos aplicaciones), aunque con esto perdemos el derecho al soporte técnico ofrecido por la empresa.

### 3.6.2. Coste del hardware

Otro de los aspectos importantes en el cálculo del TCO es el cálculo de los costes asociados al Hardware que tenemos que usar. Cuanto mayor antigüedad tenga el hardware más probabilidades de fallo tiene pero más rentable es para la empresa debido a que su gasto se ha amortizado completamente en los diferentes ejercicios contables. Así, si encontramos un software que sea capaz de aprovechar al máximo y durante mucho tiempo los equipos conseguiremos un ahorro económico importante. En este aspecto el Software Libre es mucho más generoso que el Software Privativo, en la siguiente tabla podemos ver los distintos medios de hardware mínimos requeridos por algunos SO en la arquitectura i386:

SO	Procesador	RAM	Disco duro
Windows XP Prof.	300 MHz	64 MB	1,5 GB
Windows 2003 Server	133 MHz	128 MB	1,25 GB
Open Suse Linux	100 MHz	128 MB	500 MB
Linux Debian Sarge	100 MHz	32 MB	150 MB

Si a esto añadimos la ya mencionada escalabilidad(sección 3.4), podemos obtener grandes resultados a muy bajo coste con productos de software libre. Así por ejemplo en el artículo *The Do-It-Yourself Supercomputer* [41], varios científicos explican cómo conseguir un supercomputador que alcanza picos de 1,2 Gigaflops usando software libre y equipos antiguos.

### 3.6.3. Downtime

El downtime es uno de los factores más complicados de calcular dentro del TCO debido a lo imprevisible de sus causas. No obstante resaltaremos algunos de los factores que más afectan a esta variable.

**Virus:** Como ya vimos en ??, los virus pueden afectar de una manera muy grave al rendimiento de una empresa y hoy por hoy son responsables de gran parte del downtime sufrido por las empresas. Por ejemplo, según la empresa de seguridad Mi2G Consulting, en el año 2002 se perdieron alrededor de veintidos mil millones de dólares en la economía mundial por culpa de los virus [?]. La baja incidencia de éstos en el software libre hace que el downtime por culpa de los virus sea prácticamente nulo.

**Bloqueos del sistema:** Este factor está muy relacionado con la estabilidad del software que estemos utilizando y puede ocasionar graves pérdidas en el downtime. Así empresas como Heinz Tröber han cambiado sus sistemas privativos por software libre ya que representaban unas pérdidas semanales de 25 horas de trabajo. Desde que en el 2001 cambiaron sus sistemas de escritorio por Linux no han registrado downtime alguno [42].

#### 3.6.4. Costes de personal

A pesar de que los costes de personal técnico es bastante parecido entre los productos de Software Libre y de Software privativo, es necesario tener en consideración que es más fácil para un usuario no técnico tratar con un entorno conocido que con un escritorio que no haya visto nunca. En este aspecto debido a la supremacía demostrada en los entornos privados por los sistemas privativos (figura ??), estos parten con una ventaja clara de cara al aprendizaje del personal. Así en la mayoría de los casos el coste de personal (no técnico) disminuye debido a que no existe la necesidad de impartir cursos de reciclaje informático ni un periodo de adaptación a un software que no están acostumbrados.

## 4. Modelo de Software Libre

Después de haber visto detenidamente las características del Software Libre, una de las cosas que más llaman la atención es la calidad del código que se programa [37] a pesar de que no parece que haya una estructura bien definida en el desarrollo de los programas.

Uno de los primeros en analizar el proceso de diseño y realización de un proyecto de Software Libre fue el fundador de la iniciativa Open Source Eric S. Raymond en su artículo *La catedral y el Bazar* [38]. En este artículo Eric analizaba el proceso de desarrollo de un proyecto de software libre concreto, la aplicación de correo popclient. Este artículo causó una revolución en el mundo del software [39] porque ponía al descubierto las claves que hacían de la aparente anarquía del software libre un modelo de desarrollo viable y en muchos casos mejor que el modelo del software

privativo. Nosotros no entraremos en detalle en el artículo, pero sí consideraremos algunos puntos que resalta el artículo:

- *Todo buen trabajo de software comienza a partir de las necesidades del programador.*

Posiblemente esta es una de las razones por las que el Software Libre produce programas bastante más estables que los del software privativo. Los programas surgen para cubrir las necesidades de sus programadores y por tanto se afinan tanto más como lo preciso que sea el programador.

- *Tratar a los usuarios como colaboradores es la forma más apropiada de mejorar el código, y la más efectiva de depurarlo.*

*Dada una base suficiente de desarrolladores asistentes y beta-testers, casi cualquier problema puede ser caracterizado rápidamente, y su solución ser obvia al menos para alguien.*

Quizá sea ésta una de las características que hacen tan fiable al Software Libre, ya que al ser los usuarios una parte activa del proyecto colaboran mucho más en la búsqueda e informe de errores y aportan sugerencias muy interesantes a los programas que más tarde se incorporan a los mismos en la mayoría de los casos. No en vano esta característica de desarrollo ha sido adoptada por las empresas de software privativo para poder ofrecer a sus usuarios una mejor atención. En un lenguaje más coloquial y parafraseando a Eric Raymond en lo que él llama la "Ley de Linus" (en honor a Linus Torvalds, creador del kernel de Linux): "si se tienen las miradas suficientes, todos los bugs saltarán a la vista".

- *Frecuentemente, las soluciones más innovadoras y espectaculares provienen de comprender que la concepción del problema era errónea.*

Sin duda este punto implica una forma de concebir el producto de una forma totalmente opuesta al modelo del software propietario. Al no existir plazos de entrega del producto ni obligación de actualizaciones, los desarrolladores pueden permitirse el lujo de tirar líneas y líneas de código cuya concepción inicial fue errónea y que sólo conseguirían acabar empeorando las sucesivas versiones del software.

- *Un sistema de seguridad es tan seguro como secreto. Cuidese de los secretos a medias.*

Uno de los principales argumentos de seguridad de muchas empresas de software se basa en la oscuridad de su código. Es decir, si no se tiene acceso al código fuente, no es posible crackear el software. Aunque en principio la idea es bastante plausible, pierde fuerza en base a los resultados que hemos visto en las anteriores secciones. Un software oscuro no sólo acaba siendo crackeado tarde o temprano, sino que normalmente acaba siendo crackeado por personas mal intencionadas que pretendían colarse a través de un agujero de

seguridad. Si nuestro código es abierto cualquiera podrá ver los errores que contenga y eso precisamente hará que se informe más rápidamente de los fallos y se puedan corregir con mayor celeridad.

Uno de los aspectos que no tuvo en cuenta Eric Raymond debido a lo joven que era en aquel momento era el de *comunidad de soporte técnico* y es que gracias sobretodo al aumento de las velocidades de conexión a internet y a la enorme difusión de las tecnologías de PHP y MySQL<sup>1</sup> los foros de internet se han convertido en el mejor servicio técnico que cualquier empresa puede desear. Tanto es así que muchas de las firmas que comercian con software libre ofrecen en sus paquetes gratuitos soporte a través de foros web o wikis<sup>2</sup>. Estos foros y wikis se sustentan de comunidades de usuarios que aportan sus conocimientos para solucionar los problemas que van surgiendo a cada individuo creando auténticos manuales repletos de información.

Este aspecto tan novedoso de soporte técnico puede convertirse en un enorme aliado para las empresas si los gastos destinados a la adquisición del software y su soporte no son muy elevados. No obstante son un arma de doble filo ya que dependen de que exista una comunidad que los sustente y si esta no existe son poco menos que inútiles direcciones web.

## 5. Tendencia actual

Actualmente la mayoría de las empresas de software privativo están dando un giro en sus políticas de negocio y están adoptando paulatinamente el Software Libre. Un ejemplo claro está en la compañía Sun que afirma en su página web que

*El Software Libre es el modelo ideal de desarrollo y negocio para la actual era de participación masivamente conectada ¿Por qué? Porque ofrece libertad a cada usuario para usar y desarrollar programas motivando una colaboración genuina e intuitiva.*

La empresa Oracle también es otro gigante que decidió adoptar el software libre dentro de su gran abanico de productos y está muy involucrada dentro de la comunidad.

Pero sin duda la apuesta más fuerte por el Software Libre ha llegado de la mano de la compañía IBM como medio para eliminar su dependencia de las compañías de software privativo y reafirmar su dominio en el mercado [45]. Así IBM además de ofrecer sus productos con distribuciones de Software Libre como Red Hat o SUSE Linux, también ha aportado grandes contribuciones al mundo del Software Libre con la liberación de Eclipse y otros productos.

---

<sup>1</sup>La mayoría de las veces en soluciones LAMP (Linux+Apache+MySQL+PHP)

<sup>2</sup>Un wiki es una página en internet que cualquiera puede modificar. El ejemplo más claro es la [Wikipedia](#)

Hasta ahora la única empresa que se había mantenido al margen en una posición casi de ataque hacia el software libre era Microsoft. No obstante la compañía lleva un tiempo mostrando su cara más amable para la comunidad libre y ha liberado alguno de sus productos como muestra de buena voluntad [40].

Así, parece que todas las empresas de software poco a poco empiezan a comprender las ventajas del modelo de desarrollo de código abierto y se unen a las iniciativas ya existentes o crean sus propias iniciativas como parte de su inversión en I+D.

## 6. Debilidades del Software Libre

A pesar de las bonanzas del software libre demostradas a lo largo de este informe, todavía quedan muchos aspectos del mismo que hacen que se frene su adopción en la empresa. Según Sergio Montoro, consejero delegado de Knowgate en su ponencia *Hipergate: Software Libre en la Empresa* [44] algunos de los aspectos son:

**Curva de aprendizaje:** Como ya hemos resaltado en la sección 3.6.4, el software privativo suele ser el más usado en los entornos de usuario y esto hace que el cambio conlleve unos costes en cursos de reciclaje que no muchas empresas están dispuestas a afrontar.

**Inercia:** La inercia es la causa de que muchas empresas no cambien de software si funciona razonablemente bien.

**Escasez de aplicaciones verticales:** Como ya hemos dicho revisando el artículo de Eric Raymond en la sección 4, el germen de un programa es la necesidad de su programador y en este contexto hay pocos programadores que hayan requerido de aplicaciones verticales por lo que la variedad es mucho menor y más limitada en prestaciones que la del software privativo.

## Referencias

- [1] *Software Libre* [Documento en Línea]. Wikipedia, La enciclopedia libre.  
Disponible en internet:  
[http://es.wikipedia.org/wiki/Software\\_Libre](http://es.wikipedia.org/wiki/Software_Libre)
- [2] Richard Stallman, *Why “Free Software” is better than “Open Source”* [Documento en Línea]. Free Software Foundation.  
Disponible en internet:  
<http://www.gnu.org/philosophy/=open-source-or-free.html>
- [3] Bruce Perens, *The Open Source Definition* [Documento en Línea]. Open Source Initiative.  
Disponible en internet:  
<http://www.opensource.org/docs/definition.php>
- [4] Datos Ofrecidos por Sourceforge.net  
Disponible en internet:  
<http://sourceforge.net/>
- [5] *November 2005 Web Server Survey* [Documento en Línea]. Netcraft.com.  
Disponible en internet:  
[http://news.netcraft.com/archives/2005/11/07/november\\_2005\\_web\\_server\\_survey.html](http://news.netcraft.com/archives/2005/11/07/november_2005_web_server_survey.html)
- [6] *Hosting Providers sites ordered by failures* [Documento en Línea]. Netcraft.com.  
Disponible en internet:  
[http://uptime.netcraft.com/perf/reports/performance/Hosters?tn=november\\_2005](http://uptime.netcraft.com/perf/reports/performance/Hosters?tn=november_2005)
- [7] *Linux Outlook 2005* [Documento en Línea]. Information Week.  
Disponible en internet:  
<http://i.cmpnet.com/infoweek/1057/IWKLinuxOutlook-2005.pdf>
- [8] *Great Gadget Smack-Down Round 4* [Documento en Línea]. Linux/Windows devices.com.  
Disponible en internet:  
<http://linuxdevices.com/articles/AT6743418602.html>
- [9] *Sparks fly as Magneto morphs into Windows Mobile 5.0* [Documento en Línea] Windows for Devices.  
Disponible en internet:  
<http://www.windowsfordevices.com/news/NS2937141620.html>

- [10] *Industry jumps on Windows Mobile 5.0 bandwagon* [Documento en Línea] Windows for Devices.  
Disponible en internet:  
<http://www.windowsfordevices.com/news/NS6226829827.html>
- [11] Steve Hamm, *Linux Inc.* [Documento en Línea] Business Week Magazine.  
Disponible en internet:  
[http://www.businessweek.com/magazine/content/05\\_05/b3918001\\_mz001.htm](http://www.businessweek.com/magazine/content/05_05/b3918001_mz001.htm)
- [12] Don Moore, *DNS Server Survey* [Documento en Línea] DNS Survey.  
Disponible en internet:  
<http://mydns.bboy.net/survey/>
- [13] Martin Lamonica, *Open-source databases gaining favor* [Documento en Línea] ZDNet.  
Disponible en internet:  
[http://news.zdnet.com/2100-3513\\_22-5134836.html?tag=tu.swblog.6566](http://news.zdnet.com/2100-3513_22-5134836.html?tag=tu.swblog.6566)
- [14] Robert McMillan, *Study shows Microsoft has lost a percentage of market share to open source browser.* [Documento en Línea] PC World Magazine.  
Disponible en internet:  
<http://www.pcworld.com/news/article/0,aid,116848,00.asp>
- [15] Thomas Claburn, *Firefox Eats More Microsoft Market Share* [Documento en Línea] Information Week.  
Disponible en internet:  
<http://www.informationweek.com/story/showArticle.jhtml?articleID=159902316>
- [16] Vance McCarthy, *Evans: IT Using More Open Source Databases* [Documento en Línea] Open Enterprise Trends.com.  
Disponible en internet:  
[http://oetrends.com/news.php?action=view\\_record&idnum=392](http://oetrends.com/news.php?action=view_record&idnum=392)
- [17] Barton P. Miller, *Fuzz Revisited: A Re-examination of the Reliability of UNIX Utilities and Services* [Documento en Línea]. Computer Sciences Department - University of Wisconsin  
Disponible en internet:  
[ftp://ftp.cs.wisc.edu/paradyn/technical\\_papers/fuzz-revisited.pdf](ftp://ftp.cs.wisc.edu/paradyn/technical_papers/fuzz-revisited.pdf)
- [18] Justin E. Forrester, Barton P. Miller, *An Empirical Study of the Robustness of Windows NT Applications Using Random Testing* [Documento en Línea]. Computer Sciences Department - University of Wisconsin

Disponible en internet:

<http://www.cs.wisc.edu/~bart/fuzz/fuzz-nt.html>

- [19] Steven J. Vaughan-Nichols, *Can You Trust This Penguin?* [Documento en Línea]. ZDNet Sm@rtPartner

Disponible en internet:

<http://web.archive.org/web/20010606035231/http://www.zdnet.com/sp/stories/issue/0,4537,238728>

- [20] *Reasoning Releases New Study Showing Open Source Model Produces High Quality Software* [Documento en Línea]. Business Wire.

Disponible en internet:

[http://www.businesswire.com/cgi-bin/f\\_headline.cgi?bw.021103/230420300](http://www.businesswire.com/cgi-bin/f_headline.cgi?bw.021103/230420300)

- [21] Robert Lemos, *Security research suggests Linux has fewer flaws* [Documento en Línea]. Business Wire.

Disponible en internet:

[http://news.com.com/Security+research+suggests+Linux+has+fewer+flaws/2100-1002\\_3-5489804.htm](http://news.com.com/Security+research+suggests+Linux+has+fewer+flaws/2100-1002_3-5489804.htm)

- [22] *Fewer Defects Found in Latest Linux Kernel* [Documento en Línea]. Information Week.

Disponible en internet:

<http://informationweek.com/story/showArticle.jhtml?articleID=167100724>

- [23] *Desktop Linux wins plaudits for stability* [Documento en Línea]. ZDNet.com.

Disponible en internet:

<http://news.zdnet.co.uk/software/linuxunix/0,39020390,39190950,00.htm>

- [24] Oliver Kaven, *Performance Tests: File Server Throughput and Response Times* [Documento en Línea]. PCMag.com.

Disponible en internet:

<http://www.pcmag.com/article2/0,1895,16227,00.asp>

- [25] Roger Howorth, *Samba beats Windows* [Documento en Línea]. IT Week.

Disponible en internet:

<http://www.itweek.co.uk/itweek/news/2085206/samba-beats-windows>

- [26] Jürgen Schmidt, *Linux and NT as Web Server on the Test Bed* [Documento en Línea]. c't.

Disponible en internet:

<http://www.heise.de/ct/english/99/13/186-1/>

- [27] Greg Shipley, *Is It Time for Linux* [Documento en Línea]. Network Computing.  
Disponible en internet:  
<http://www.networkcomputing.com/1011/1011f1.html>
- [28] *26th TOP500 list* [Documento en Línea]. Top500.org.  
Disponible en internet:  
[http://www.top500.org/lists/2005/11/TOP500\\_Nov2005\\_Highlights.pdf](http://www.top500.org/lists/2005/11/TOP500_Nov2005_Highlights.pdf)
- [29] David A. Wheeler, *Counting Source Lines of Code (SLOC)* [Documento en Línea]  
Disponible en internet:  
<http://www.dwheeler.com/sloc/>
- [30] Erich Luening , *Windows users pay for hacker insurance* [Documento en Línea] Cnet -  
News.com  
Disponible en internet:  
<http://news.com.com/2100-1001-258392.html>
- [31] *Know Your Enemy - Trend Analysis* [Documento en Línea] The HoneyNet Project  
Disponible en internet:  
<http://www.honeynet.org/papers/trends/life-linux.pdf>
- [32] *Time To Live On The Network* [Documento en Línea] Avantgarde  
Disponible en internet:  
<http://www.avantgarde.com/xxxxttltn.pdf>
- [33] Jim Rapoza, *eWeek Labs: Open Source Quicker at Fixing Flaws* [Documento en Línea]  
Eweek.com  
Disponible en internet:  
<http://www.eweek.com/article2/0,3959,562226,00.asp>
- [34] *RSA: Security in 2002 worse than 2001, exec says* [Documento en Línea] CNN  
Disponible en internet:  
<http://archives.cnn.com/2002/TECH/internet/02/25/2002.security.idg/index.html>
- [35] Nicholas Petreley, *Security Report: Windows vs Linux* [Documento en Línea] The Register.co.uk  
Disponible en internet:  
[http://www.theregister.co.uk/2004/10/22/security\\_report\\_windows\\_vs\\_linux.pdf](http://www.theregister.co.uk/2004/10/22/security_report_windows_vs_linux.pdf)

- [36] *Analysis of the Impact of Open Source Software* [Documento en Línea] The UK Gov Talk Disponible en internet:  
[http://www.govtalk.gov.uk/policydocs/policydocs\\_document.asp?docnum=781](http://www.govtalk.gov.uk/policydocs/policydocs_document.asp?docnum=781)
- [37] Ioannis Samoladas, Ioannis Stamelos, Lefteris Angelis, Apostolos Oikonomou *Open source software development should strive for even greater code maintainability* [Documento en Línea] ACM Queue Disponible en internet:  
<http://www.acmqueue.com/modules.php?name=Content&pa=showpage&pid=240>
- [38] Eric S. Raymond, *La Catedral y el Bazar* [Documento en Línea] Wikisource, biblioteca online Disponible en internet:  
[http://es.wikisource.org/wiki/La\\_Catedral\\_y\\_el\\_Bazar](http://es.wikisource.org/wiki/La_Catedral_y_el_Bazar)
- [39] J.T.S. Moore, *Revolution OS* [Documental] ASIN: B0000A9GLO
- [40] Todd Bishop, *Microsoft open to open source* [Documento en Línea] Disponible en internet:  
[http://seattlepi.nwsourc.com/business/179256\\_msftopen25.html](http://seattlepi.nwsourc.com/business/179256_msftopen25.html)
- [41] William W. Hargrove, Forrest M. Hoffman, Thomas Sterling, *The Do-It-Yourself Supercomputer* [Documento en Línea] Disponible en internet:  
<http://www.sciamdigital.com>
- [42] Ingrid Marson, *Desktop Linux wins plaudits for stability* [Documento en Línea] ZDNet UK. Disponible en internet:  
<http://news.zdnet.co.uk/software/linuxunix/0,39020390,39190950,00.htm>
- [43] *Google Zeitgeist - Search patterns, trends, and surprises according to Google* Google Inc. Disponible en internet:  
<http://www.google.com/press/zeitgeist/zeitgeist-jan04.html>
- [44] Sergio Montoro - Consejero Delegado de Knowgate *hipergate: Software Libre en la Empresa* [Libro de ponencias] I Conferencia Internacional de Software Libre. Disponible en Línea:  
<http://www.opensourceworldconference.com/malaga04/files/Proceedings-Ponencias.zip>
- [45] David Berlind *Open source: IBM's deadly weapon* [Documento en Línea] ZDNet Disponible en Línea:  
<http://techupdate.zdnet.com/techupdate/stories/main/0,14179,2860394,00.html>

- [46] David A. Wheeler, *Why Open Source Software / Free Software (OSS/FS, FLOSS, or FOSS)? Look at the Numbers!* [Documento en Línea]

Disponible en internet:

[http://www.dwheeler.com/oss\\_fs\\_why.html](http://www.dwheeler.com/oss_fs_why.html)