

MP01B0000048

MITREPRODUCT

Un caso de negocio para el estudio del Software de Fuente Abierta

Julio de 2.001

Carolyn A. Kenwood

Patrocinado por: Ejército de los EEUU
Nº de Depto.: W803

Nº de contrato: DAAB07-01-C-C201
Nº de Proyecto: 0700M520-AA

Los puntos de vista, opiniones y/o resultados contenidos en este informe son los de la corporación MITRE y no deberían ser tenidos en cuenta como una posición, política o decisión oficial de la Administración, a menos que sea así designado por otro documento.

Este documento ha sido aprobado para su distribución pública e ilimitada.

© 2.001 de la Corporación MITRE. Todos los derechos reservados.

MITRE

**Washington C3 Center
Bedford, Massachussets**

Traducido para **Hispalinux** por:

- *Rafael Gómez Lago* (rafael.gomez@hispalinux.es): *Coordinación, supervisión y maquetación, Resumen Ejecutivo, Historia de Unix y Linux, Modelo de Análisis del Caso de Negocio, Análisis de los factores estratégicos, Análisis de Viabilidad y Anexos.*
- *Roberto Plà Aragonés* (roberto.pla@hispalinux.es): *Estudio del entorno.*

Traducción publicada originalmente el: 01 de octubre de 2002

Última actualización de la traducción: 26 de noviembre de 2002

Localización de esta traducción en la web en <http://www.hispalinux.es/informes>.

Aprobación departamental en MITRE:

Francis M. Dello Russo, W803
Responsable del departamento

Aprobación de proyecto en MITRE:

Paul R. Garvey, W800
Jefe Científico

Nota preliminar del equipo de traducción

Durante el curso de la traducción hubo algún debate sobre la forma adecuada de traducir el término “open source”. Si bien es cierto que Hispalinux, asociación de la que parte la iniciativa de traducción, es una organización defensora en general del **software libre** y que la definición que se da, ya en el primer párrafo, de “fuente abierta” es más próxima al término “software libre” que la GNU recomienda utilizar preferentemente, no es menos cierto que la autora del informe utilizó el término “open source”. Dado el carácter altamente significativo del informe, hemos considerado que lo más importante en este caso es permanecer fieles a la letra de la traducción (para evitar cualquier acusación de tergiversar el contenido del informe) y hemos traducido en consecuencia. La opinión del equipo técnico, no obstante, es que el informe no cambia en nada su significado si donde en el texto dice “fuente abierta” o “software de fuente abierta” el lector entiende “software libre” y donde dice “OSS” el lector entiende “FS” (por “free software”).

Si quiere profundizar sobre la conveniencia de utilizar el término adecuado para referir los distintos tipos de software, recomendamos la lectura del artículo “” que se puede encontrar en el sitio de GNU en Internet <http://www.gnu.org/>

Resumen

Este informe ha sido preparado como parte del proyecto de desarrollo FY00 Investigación y Experimentación Orientada a Misiones (MOIE) titulado “Software de Fuente Abierta en los Sistemas Militares” de la Corporación MITRE. Este informe analiza un caso de negocio para el software de fuente abierta. Su intención es ayudar a los Jefes de Programa a evaluar si el software de fuente abierta y las metodologías de desarrollo asociadas son aplicables a sus programas tecnológicos. En el resumen ejecutivo, el artículo explica lo que es la fuente abierta, describe su importancia, compara la fuente abierta con los productos tradicionales comerciales de distribución generalizada (COTS), presenta el caso de negocio en el entorno militar, muestra la aplicabilidad de Linux al caso de negocio militar, analiza el uso de Linux, discute las anomalías y ofrece las consideraciones a tener en cuenta por los Jefes de Programa Militar. El informe también ofrece una historia de Unix y Linux, presenta un modelo del caso de negocio y analiza el caso de negocio de Linux para un entorno comercial.

PALABRAS CLAVE: Linux, Fuente Abierta, Software de Fuente Abierta, OSS, Software

Agradecimientos

El autor quiere reconocer y agradecer a las personas que han contribuido a este documento. En particular, gracias a Terry Bollinger, Francis Dello Russo, David Emery, Paul Garvey, Robert Giallombardo, Michael Macpherson, Frank McPherson, James Moore y Audrey Taub por sus opiniones y valiosas revisiones. Me gustaría agradecer a Rosemarie Mauriello por su ayuda en la producción de este documento. También gracias a Janice Ballo y Theresa Dillon por los materiales que aportaron gracias a sus extensos esfuerzos de búsqueda.

Tabla de contenido

Sección	Página
<i>Resumen Ejecutivo</i>	<i>xi</i>
¿Qué es la Fuente Abierta?	xi
Importancia de la Fuente Abierta	xiii
Beneficios y riesgos del software de fuente abierta frente al COTS tradicional	xiii
El caso de negocio en el entorno militar	xix
Aplicabilidad de Linux al caso de negocio en el entorno militar	xix
Uso de Linux	xxi
Discusión	xxii
Consideraciones para Jefes de Programa Militar	xxiii
Premio Federal de Linux	xxiv
Conclusión	xxv
<i>Historia de Unix y Linux</i>	<i>1</i>
<i>Modelo de análisis del caso de negocio</i>	<i>3</i>
<i>Análisis de las posibilidades comerciales de Linux</i>	<i>5</i>
3.1 Estudio del entorno	5
3.1.1 Fortalezas	6
3.1.1.1 Experiencia masiva en su programación	6
3.1.1.2 I+D Cubierta por trabajo de voluntarios	6
3.1.1.3 Estructura de liderazgo aceptada	6
3.1.1.4 Rápido ritmo de liberación de versiones	7
3.1.1.5 Desarrollo y depuración en paralelo	7
3.1.1.6 Madurez del código	7
3.1.1.7 Cultura de la compartición	8
3.1.1.8 Accesibilidad a largo plazo	8
3.1.2 Debilidades	8
3.1.2.1 Falta de “propiedad”	8
3.1.2.2 Difícil de iniciar	9
3.1.2.3 Menos amigable para el usuario	9
3.1.3 Oportunidades	11
3.1.3.1 Conectividad con Internet	11
3.1.3.2 Múltiples distribuidores	12
3.1.3.3 Estructura de soporte competitiva	15
3.1.3.4 Afluencia de nuevas empresas	16
3.1.4 Amenazas	17
3.1.4.1 Riesgo de fragmentación	17

Sección	Página
3.1.4.2 Falta de aplicaciones compatibles _____	17
3.1.4.3 Necesita un control de versiones _____	18
3.1.5 Otras _____	18
3.1.5.1 Importancia para muchos _____	18
3.1.5.2 Personal formado _____	18
3.1.5.3 Competición _____	18
3.2 Análisis de los factores estratégicos _____	19
3.2.1 Viabilidad en el mercado _____	19
3.2.2 Segmentos del mercado _____	26
3.2.2.1 Mercado de servidores _____	29
3.2.2.2 Mercado de sobremesa _____	33
3.2.2.3 Dispositivos empotrados _____	35
3.3 Análisis de viabilidad de la oportunidad de negocio _____	37
3.3.1 Costes directos _____	41
3.3.1.1 Software y hardware _____	41
3.3.1.1.1 Software _____	41
3.3.1.1.2 Hardware _____	41
3.3.1.2 Soporte _____	41
3.3.1.2.1 Soporte interno _____	41
3.3.1.2.2 Soporte externo _____	41
3.3.1.3 Gastos de personal _____	41
3.3.1.3.1 Gestión de proyectos _____	42
3.3.1.3.2 Ingeniería/Desarrollo de sistemas _____	42
3.3.1.3.3 Administración de sistemas _____	42
3.3.1.3.4 Otros gastos de administración _____	42
3.3.1.3.5 Formación _____	42
3.3.1.4 Desinstalación y eliminación de software _____	42
3.3.2 Costes indirectos _____	42
3.3.2.1 Costes de soporte _____	43
3.3.2.1.1 Soporte entre pares _____	43
3.3.2.1.2 Aprendizaje informal _____	43
3.3.2.1.3 Formación oficial _____	43
3.3.2.1.4 Desarrollo de aplicaciones _____	43
3.3.2.1.5 El factor Futz _____	43
3.3.2.2 Tiempo de inactividad por caída de los sistemas _____	43
3.3.3 Beneficios y riesgos _____	44
3.3.3.1 Capacidad de adaptación _____	45
3.3.3.2 Disponibilidad/Fiabilidad _____	45
3.3.3.3 Interoperabilidad _____	45
3.3.3.4 Scalability _____	46
3.3.3.5 Flexibilidad del diseño _____	46
3.3.3.6 Tiempo de vida _____	46
3.3.3.7 Rendimiento _____	46
3.3.3.8 Calidad de servicio y soporte _____	46
3.3.3.9 Seguridad _____	47

Sección	Página
3.3.3.10 Nivel de dificultad o facilidad de gestión _____	47
3.3.3.11 Riesgo de fragmentación _____	48
3.3.3.12 Disponibilidad de aplicaciones _____	48
<i>Referencias</i> _____	51
<i>Glosario de acrónimos</i> _____	54
<i>Lista de distribución</i> _____	56

Lista de figuras

Figura	Página
Ilustración ES- 1 El OSS ofrece diferentes opciones de mantenimiento y soporte	xviii
Ilustración ES- 2 Beneficios para el usuario de Linux en el ámbito comercial y militar	xx
Ilustración ES- 3 Éxito a nivel mundial de Linux en el mercado.....	xxi
Ilustración ES- 4 Cuota del mercado de SO de servidor y de cliente en 1.998 y 1.999.....	xxii
Ilustración 1 Marco de análisis de caso de negocio aplicado a los productos y procesos de fuente abierta	4
Ilustración 2. Elementos clave del análisis DAFO.....	5
Ilustración 3. Fragmento de código fuente de Linux	10
Ilustración 4. Ejemplo de pantalla de Linux	11
Ilustración 5. Motivación del interés por Linux	19
Ilustración 6. Satisfacción con Linux	20
Ilustración 7. Éxito mundial de Linux en el mercado.....	22
Ilustración 8. Porcentaje de empresas que usan el sistema operativo Linux.	23
Ilustración 9. Porcentaje de sistemas operativos que son o serán Linux.....	24
Ilustración 10. ¿Cuánto tiempo han estado las empresas utilizando Linux?	25
Ilustración 11. Nuevos envíos a nivel de Linux (Cliente y Servidor)	26
Ilustración 12. Sitios con servidores linux organizados por sector de actividad, 1.999.....	27
Ilustración 13. Uso de Linux (encuesta de Datapro)	28
Ilustración 14. Uso de Linux en aplicaciones basadas en servidor (encuesta de Information Week).....	29
Ilustración 15. Cuotas de mercado de SO de servidor en 1.998.	30
Ilustración 16. Cuotas de mercado de SO de servidor en 1.999.	30
Ilustración 17. Distribución de servidores Linux y gasto de los clientes, 1.998-2.003.	31
Ilustración 18. Distribución del gasto en servidores según la carga de trabajo por sistema operativo	32
Ilustración 19. Cuotas de mercado de los sistemas operativos de cliente en 1.998.....	34
Ilustración 20. Cuotas de mercado de los sistemas operativos de cliente en 1.999.....	34
Ilustración 21. Debilidades más significativas de Linux.....	48
Ilustración 22. Puntuación otorgada por los usuarios de Linux de EEUU en calidad de servicio según el sistema operativo.	49

Lista de tablas

Tabla	Página
Tabla ES- 1 Taxonomía de elementos de coste del OSS	xvi
Tabla ES- 2 Taxonomía de beneficios y riesgos del OSS	xvii
Tabla 1. Distribuciones comunes de Linux ordenadas por distribuidor	12
Tabla 2. Distribuidores de Linux especializados en sectores específicos	14
Tabla 3. Recursos gratuitos de apoyo para Linux por distribuidor	16
Tabla 4. Conjuntos de aplicaciones de escritorio para Linux por fabricante y por producto.	33
Tabla 5. Taxonomía de elementos de coste para OSS y Linux	40
Tabla 6. Taxonomía de beneficios y riesgos del OSS	44
Tabla 7. Comparativa de sistemas operativos.....	50

Resumen Ejecutivo

¿Qué es la Fuente Abierta?

Fuente abierta, por definición, implica que el código fuente está disponible. El software de fuente abierta (OSS) es software cuyo código fuente está disponible y que puede ser utilizado, copiado y distribuido con o sin modificaciones y que puede ser ofrecido con o sin precio. Si el usuario final realiza alteraciones al software, puede decidir mantener los cambios para su uso privado o devolver el software modificado a la comunidad de forma que puedan potencialmente ser incluidos en versiones futuras.¹ Una licencia de fuente abierta es certificada por la Iniciativa por la Fuente Abierta (OSI), una asociación corporativa sin ánimo de lucro dedicada al desarrollo y la educación con la misión de proteger y defender la marca registrada “Open Source” y avanzar en la promoción del OSS. La comunidad de la fuente abierta está formada por individuos o grupos de individuos que contribuyen a un producto o tecnología de fuente abierta particular. El proceso de la fuente abierta se refiere a la aproximación al desarrollo y mantenimiento de productos y tecnologías de fuente abierta, incluidos software, ordenadores, dispositivos, formatos técnicos y lenguajes de ordenador.

Aunque el OSS se ha convertido en un tema candente en la prensa hace poco, en realidad existe desde los años 60 y hasta la fecha ha registrado considerables éxitos. Ejemplos de productos populares de fuente abierta son Emacs, el conjunto de herramientas de GNU, Apache, Sendmail y Linux. El desarrollo de Perl es un ejemplo de proceso de fuente abierta.

Emacs fue uno de los primeros productos de fuente abierta. Es un editor de textos es ampliamente usado para el desarrollo de software. Como herramienta de software, muchos desarrolladores (incluidos contratistas del Ministerio de Defensa) utilizan Emacs para desarrollar sus aplicaciones (no libres).² El éxito de Emacs condujo al programa GNU. GNU son las siglas de “GNU No es Unix.” El proyecto GNU consiste en un núcleo de sistema operativo y las herramientas Unix asociadas. Las herramientas de GNU han sido traducidas a una amplia variedad de plataformas, que incluye a Windows NT. También son utilizadas ampliamente por desarrolladores de software para producir tanto software abierto como software propietario.³

El servidor web Apache es un servidor web libremente disponible bajo una licencia de fuente abierta. Los desarrolladores de Apache forman un comité de voto y los votos de este comité deciden el futuro del proyecto. La Fundación Apache para el Software ofrece soporte organizativo, legal y financiero para los proyectos de Apache. Los servidores

¹ Hay varios modelos de licencia para el Software Abierto. Algunos requieren que todos los cambios que se hagan al código fuente sean distribuidos de forma libre junto con el producto modificado. Otras licencias permiten a una organización hacer cambios y mantenerlos para uso privado.

² Para más información sobre Emacs, véase <http://www.gnu.org/software/emacs/emacs.html>.

³ Para más información sobre GNU, visite el servidor web del Proyecto en <http://www.gnu.org/>.

Apache son conocidos por su funcionalidad y su fiabilidad. Forman la infraestructura troncal del funcionamiento de Internet. Hoy día, Apache tiene más del 60 por ciento del mercado de servidores y continúa creciendo.⁴

Sendmail es una plataforma para la transferencia de correo de una máquina a otra. El Consorcio Sendmail, una organización sin ánimo de lucro, sigue con el programa de código abierto y mantiene un sitio web de recurso. Se estima que Sendmail transporta cerca del 90 por ciento del tráfico de correo electrónico actual.⁵

Linux es un sistema operativo (SO) de fuente abierta semejante a Unix. El núcleo es mantenido por la comunidad Linux, liderada por Linux Torvalds, el creador de Linux.⁶ Torvalds ha nombrado delegados que son responsables de la gestión de ciertas áreas del proyecto y, a su vez, esos delegados tienen un equipo de coordinadores. Linux tiene múltiples usos: puede ser utilizado como SO para servidor, ordenadores de sobremesa o en entornos empotrados. Hay más de diez millones de usuarios de Linux en todo el mundo. Según una encuesta de InformationWeek, Linux supone cerca del 4 por ciento de todos los sistemas operativos, y se espera que ese número crezca al 15 por ciento en dos años.⁷ Linux es el entorno operativo de servidores que más rápido está creciendo, con un incremento desde el 16 por ciento del mercado en 1.998 al 25 por ciento en 1.999.⁸ También se espera que Linux juegue un papel significativo en el mercado de los sistemas empotrados⁹ (un dispositivo empotrado es una pieza de hardware basada en microprocesador, normalmente una sola placa de circuito electrónico, que ha sido construida para ejecutar una aplicación de software específica. El término *empotrado* hace referencia al hecho de que estos dispositivos se utilizaban originalmente como bloques constitutivos de sistemas mayores.)

Así como Emacs, el conjunto de herramientas de GNU, Apache, Sendmail y Linux son ejemplos de productos de fuente abierta, el Lenguaje Práctico de Extracción e Informes (Perl) es un ejemplo de proceso de fuente abierta. Perl es un lenguaje de programación de ordenadores y de administración de sistemas que es ampliamente utilizado en toda la Internet. Es el lenguaje interpretado estándar para todos los servidores web Apache y se utiliza normalmente en Unix. Perl es gestionado sobre una base rotatoria de entre los diez a veinte programadores más activos. Cada uno se turna en la gestión de las diferentes partes del proyecto. Actualmente hay un número estimado de un millón de usuarios de Perl.¹⁰

⁴ Serie de seminarios electrónicos sobre Linux de Tim O'Reilly, 1.999. Para más información sobre Apache véase la Fundación Apache para el Software en <http://www.apache.org/>.

⁵ Tim O'Reilly y Ether Dyson, "Mente abierta, Fuente Abierta." Para más información sobre Sendmail véase <http://www.sendmail.org/>.

⁶ La Página Oficial de Linus Torvalds puede encontrarse en <http://www.cs.helsinki.fi/u/torvalds/>.

⁷ Aaron Ricadela, "Linux vive," InformationWeek, 24 de enero de 2.000.

⁸ "El futuro de Linux," Cnet 2.000, cita datos de IDC, no se da fecha.

⁹ Para más información sobre Linux, visite la Página Oficial de Linux en <http://www.linux.org>, Linux Internacional en <http://www.li.org/>, y la Página de Recursos Linux de MITRE en <http://w030nt.mitre.org/users/terry/pub/linux>

¹⁰ Para más información sobre Perl, visite <http://www.perl.com/pub>.

Importancia de la Fuente Abierta

El proceso de desarrollo de fuente abierta es claramente diferente del modelo tradicional comercial de distribución generalizada (COTS). Eric Raymond asemeja el modelo corporativo o tradicional COTS, donde una corporación produce y vende software propietario, a una catedral y el modelo de fuente abierta a un bazar.¹¹ En el modelo corporativo, ciertos individuos o pequeños grupos de individuos desarrollan software tranquila y reverencialmente de forma aislada, sin liberar una versión beta antes de que se considere preparada. En contraste, el modelo de fuente abierta se basa en una red de programadores “voluntarios”, con diferentes estilos y agendas, que desarrollan y depuran el código en paralelo. De las modificaciones enviadas, el líder delegado elige las que aceptará y las que no. Si el líder piensa que la modificación beneficiará a muchos usuarios, elegirá el mejor código de todos los que le lleguen y lo incorporará a las actualizaciones del OSS. El software se libera cuanto antes y con frecuencia.

Beneficios y riesgos del software de fuente abierta frente al COTS tradicional

Debido a los diferentes modelos de desarrollo, los Jefes de Programa pueden conseguir muchos beneficios al utilizar OSS en lugar del tradicional software COTS. Productos populares de fuente abierta tienen acceso a muchísimos expertos y técnicos, y esto hace que el software consiga un nivel de eficiencia alto, utilizando menos líneas de código que sus contrapartidas COTS. El rápido ritmo de liberación de versiones en el OSS distribuye las correcciones y los parches rápidamente, potencialmente un orden de magnitud más rápido que los del software comercial. El OSS es relativamente fácil de gestionar porque a menudo incorpora elementos como administración centralizada y gestión remota. Dado que el código fuente está disponible de forma pública, los Jefes de Programa pueden personalizar el código para adaptarlo a sus necesidades específicas y controlar mejor los recursos de sus sistemas. Más aún, los Jefes de Programa puede reutilizar el código escrito por otros para tareas o propósitos similares. Esto permite a los Jefes de Programa concentrarse en el desarrollo de las características que son únicas a su tarea actual, en lugar de malgastar su esfuerzo en volver a pensar y rescribir código que ya ha sido desarrollado por otros. La reutilización de código reduce el tiempo de desarrollo y permite resultados predecibles. Con acceso al código fuente, la vida útil de los sistemas de OSS y sus actualizaciones pueden extenderse de manera indefinida. En contraste, la vida útil de los sistemas tradicionales COTS y sus actualizaciones no puede extenderse si el fabricante no comparte su código y o bien deja el negocio, o eleva los precios de forma prohibitiva, o reduce la calidad del software de forma prohibitiva. El modelo de fuente abierta construye estándares abiertos y consigue un alto grado de interoperabilidad. Mientras que el software tradicional COTS típicamente depende del soporte monopolístico de una compañía que tiene “todas las cartas en la mano” (es decir que acceso al código) para una pieza de software, el código fuente públicamente disponible del OSS permite que muchos fabricantes estudien la plataforma y den soporte a la misma. Dado que los fabricantes de OSS compiten entre ellos para dar soporte, la calidad del mismo aumenta y

¹¹ Eric Raymond, “La Catedral y el Bazar,” O’Reilly Asociados, 1.999.

los costes para el usuario final disminuyen. La fuente abierta puede crear soporte al menos mientras exista la demanda, aunque uno de los proveedores de soporte deje el negocio. Para los propósitos de adquisición en la Administración Pública, el OSS tiene más potencial al ser un “chollo” como fuente alternativa que mejora el soporte del software COTS.

El OSS puede ser una solución viable a largo plazo con beneficios significativos, pero hay cuestiones y riesgos para los Jefes de Programa. A menudo, si el proyecto de código abierto es demasiado pequeño o no atrae el interés de desarrolladores con habilidad suficiente el resultado es un código pobre; así, los Jefes de Programa deberían asegurarse de que la comunidad de OSS es extensa, tiene talento y está bien organizada para ofrecer una alternativa viable al COTS. Al tener una competencia técnica alta, los desarrolladores habilidosos tienden a enfocarse más en el usuario técnico a expensas del usuario no técnico. Por ello, el OSS tiende a tener una interfaz gráfica de usuario (IGU) relativamente pobre y menos compatibilidad entre aplicaciones, haciéndolo más difícil de utilizar y menos práctico, en particular, para las aplicaciones de escritorio (aunque algunos productos de OSS están mejorando mucho en esta área). El control de versiones puede ser una cuestión si el sistema OSS necesita de integración y desarrollo. Mientras las nuevas versiones del OSS se liberan, el Jefe de Programa necesita asegurarse de que las versiones que han de integrarse son compatibles, asegurar que los desarrolladores están trabajando con la versión adecuada y llevar un registro de los cambios realizados al software. Sin una estructura corporativa formal, el OSS se enfrenta a un riesgo de fragmentación del código base, o de código divergente, que se refleja cuando evolucionan las versiones múltiples e inconsistentes. Esto puede ocurrir cuando los desarrolladores intentan crear alternativas para que su código realice un papel más significativo que el que tenía en el producto básico. A veces, la fragmentación ocurre por razones legítimas (por ejemplo, si el encargado de mantener el código está realizando un trabajo pobre) y a veces por razones erróneas (por ejemplo, por un conflicto personal entre los desarrolladores principales).

El código del núcleo de Linux todavía no se ha dividido en líneas divergentes, y esto puede atribuirse a la estructura aceptada del liderazgo, a la pertenencia al grupo abierta y al potencial de la contribución a largo plazo, al sistema de la Licencia General Pública de GNU (GPL) que elimina las motivaciones económicas para la fragmentación, y la consiguiente amenaza de un grupo fragmentado de desarrolladores. El noventa y nueve por ciento del código distribuido como Linux es el mismo. La pequeña porción de fragmentación entre las diferentes distribuciones de Linux es positiva porque le permite dar servicio a los diferentes segmentos. Los usuarios se benefician de ello eligiendo la distribución de Linux que mejor se adapta a sus necesidades. Finalmente, existe el riesgo de que las compañías que desarrollan estrategias competitivas se centren específicamente en el OSS.

Cuando se comparan los costes y beneficios económicos a largo plazo del uso y mantenimiento de la fuente abierta frente al COTS tradicional, el vencedor cambia según el uso específico y las circunstancias. Típicamente, el código abierto es mejor en la comparación en muchos casos de instalaciones de servidor y sistemas empotrados que puedan requerir de alguna personalización, pero no es mejor que el COTS para las aplicaciones típicas del escritorio. De hecho, parte de las fuentes en la literatura sobre el tema generalizan diciendo que los productos de fuente abierta no son peores que los de fuente cerrada, pero nuestros resultados indican que la escala con la que se mide el valor

comparado del código abierto frente al cerrado puede estar fuertemente dirigido en una dirección u otra dependiendo de los requisitos específicos y el entorno operativo del software.

Una decisión entre el OSS y el COTS tradicional está basada en tres factores: (1) costes – tanto directos (por ejemplo, precio del software) como indirectos (por ejemplo, tiempo perdido por el usuario por la inactividad del sistema); (2) beneficios (por ejemplo, rendimiento); y, (3) otros criterios más intangibles (por ejemplo, calidad del soporte entre compañeros). Los costes directos son entendidos por la mayoría y tradicionalmente comprenden la mayor parte de los costes totales del ciclo de vida de un sistema. Sin embargo, los costes indirectos y los beneficios tanto operativos como de rendimiento (por ejemplo, escalabilidad, fiabilidad y funcionalidad) juegan un papel económico influyente en el mercado actual del software más maduro. Otros criterios más intangibles, son difíciles de cuantificar, aunque también tienen un impacto en la efectividad tanto del software de fuente abierta como el de fuente cerrada. Dado que los costes indirectos y los beneficios operativos y de rendimiento juegan un papel mucho mayor en el OSS cuando se comparan con los productos COTS tradicionales, el proceso de decisión sobre inversión de TI y orientada a la misión para la elección de OSS no puede basarse el modelo de coste del ciclo de vida tradicional y otras herramientas de software COTS.

Para entender cómo los costes indirectos deberían incorporarse al análisis, los Jefes de Programa deben entender como estos costes inciden en sus programas. Dado que el salario y otros costes laborales asociados con un empleado son costes directos, sólo los costes laborales que son “malgastados” y podrían utilizarse de forma más productiva deberían incluirse como costes indirectos. En otras palabras, aunque no hay un coste directo adicional para la organización, no se ha recibido del empleado el rendimiento suficiente debido a las ineficiencias del proceso o del sistema. En una organización con ánimo de lucro se esperaría que la mejora de la productividad incrementase los beneficios. Por ejemplo, el tiempo malgastado podría utilizarse en la captación de más negocio. En una organización como el Ministerio de Defensa (DoD), el concepto de captar más negocio e incrementar los beneficios no se aplica, y los costes de esta pérdida de productividad podrían verse como una justificación para recortes estructurales forzosos. Si, por ejemplo, una organización migra a una nueva solución y experimenta una mejora de la productividad, la organización podría realizar el mismo trabajo con menos personas. Por esta razón, los empleados contemplan de forma negativa la recolección de datos necesaria para estudiar estas métricas. A menos que se establezca una relación directa de causa y efecto, puede que sea mejor considerar algunas de estas influencias como costes relativos antes que como costes absolutos en el soporte a los análisis de inversión en TI.

Los Jefes de Programa necesitan una taxonomía completa de los costes, beneficios y otros criterios más intangibles que aparecen durante el ciclo de vida para contabilizar los costes ocultos y los beneficios que, de otro modo, podrían pasar por alto. Con esta taxonomía, los Jefes de Programa pueden tomar decisiones de compra de software siendo completamente conscientes de sus implicaciones económicas, en el rendimiento y para la los objetivos de la misión. La siguiente tabla representa una taxonomía de los elementos de coste del OSS que ha sido desarrollada durante la investigación.

Tabla ES- 1 Taxonomía de elementos de coste del OSS¹²

Costes directos	
Software y Hardware	
Software	<ul style="list-style-type: none"> Precio de compra Actualizaciones y adiciones Tarifa de la licencia/propiedad intelectual
Hardware	<ul style="list-style-type: none"> Precio de compra Actualizaciones y adiciones
Coste del soporte	
Interno	<ul style="list-style-type: none"> Instalación y configuración Mantenimiento Resolución de problemas Herramientas de soporte (por ejemplo libros y publicaciones)
Externo	<ul style="list-style-type: none"> Instalación y configuración Mantenimiento Resolución de problemas
Costes de personal	
Gestión de proyectos	
Ingeniería de sistemas y desarrollo	
Administración de sistemas	<ul style="list-style-type: none"> Gestión de proveedores
Otro tipo de administración	<ul style="list-style-type: none"> Compras Otros
Entrenamiento	
Desinstalación y enajenación	
Costes indirectos	
Costes de soporte	<ul style="list-style-type: none"> Soporte entre compañeros Aprendizaje no reglado Entrenamiento formal Desarrollo de aplicaciones Factor difuso
Tiempo de inactividad por caída del sistema	

¹² El GartnerGroup incluye un factor difuso como coste indirecto. GartnerGroup describe este término como el gasto laboral en que se incurre cuando el usuario final explota los activos informáticos corporativos para uso personal durante las horas de trabajo productivo.

Además a la taxonomía de costes durante el ciclo de vida, los Jefes de Programa también necesitan un taxonomía de los beneficios y riesgos que existen, junto a una escala de ejemplo que permita medir los costes, los beneficios y los restantes criterios más intangibles del OSS y del software COTS. Nuestra investigación realizó una taxonomía de beneficios y riesgos del OSS y un ejemplo de escala de medición, que son presentados a continuación en la Tabla ES-2.

Tabla ES- 2 Taxonomía de beneficios y riesgos del OSS

Atributos cualitativos	
	Capacidades de personalización
	Disponibilidad/fiabilidad
	Interoperabilidad
	Escalabilidad
	Flexibilidad del diseño
	Vida útil
	Rendimiento
	Calidad del servicio y del soporte
	Seguridad
	Nivel de dificultad/facilidad de gestión
	Riesgo de fragmentación
	Disponibilidad de aplicaciones

Ejemplo de escala de valoración	
Muy fuerte	++
Fuerte	+
Neutro	0
Débil	=
Muy débil	= =

La taxonomía anterior comprende una lista de atributos cualitativos. Para cada atributo, los Jefes de Programa debería comparar la fortaleza o debilidad relativa de la alternativa OSS frente a los productos tradicionales COTS. Una fortaleza relativa indicaría un beneficio, una debilidad relativa indicaría un riesgo. También se muestra arriba un ejemplo de escala de evaluación que compara el valor relativo del OSS frente al software tradicional COTS. Este ejemplo presenta cinco valoraciones – muy fuerte, fuerte, neutro, débil y muy débil. Puesto que las valoraciones son diferentes dependiendo del entorno y del uso específico del software, los Jefes de Programa deben personalizar sus valoraciones de acuerdo a las circunstancias particulares.

Comparado con los productos COTS tradicionales, el OSS ofrece más opciones a los Jefes de Programa para el soporte a lo largo del ciclo de vida. La carga del mantenimiento del OSS puede ser similar a la opción pura COTS (“compra”), puede ser semejante al código a medida (“construcción”) o estar entre una de las dos. El OSS no modificado puede

ser considerado como similar al COTS puro. El software “COTS modificable,” o el OSS que se basa en modificaciones a corto plazo que se incorporan en cada actualización del OSS, se aprovecha de los beneficios tanto del COTS puro como del código a medida. El siguiente diagrama ilustra este espectro y señala las diferencias entre los anteriores escenarios.

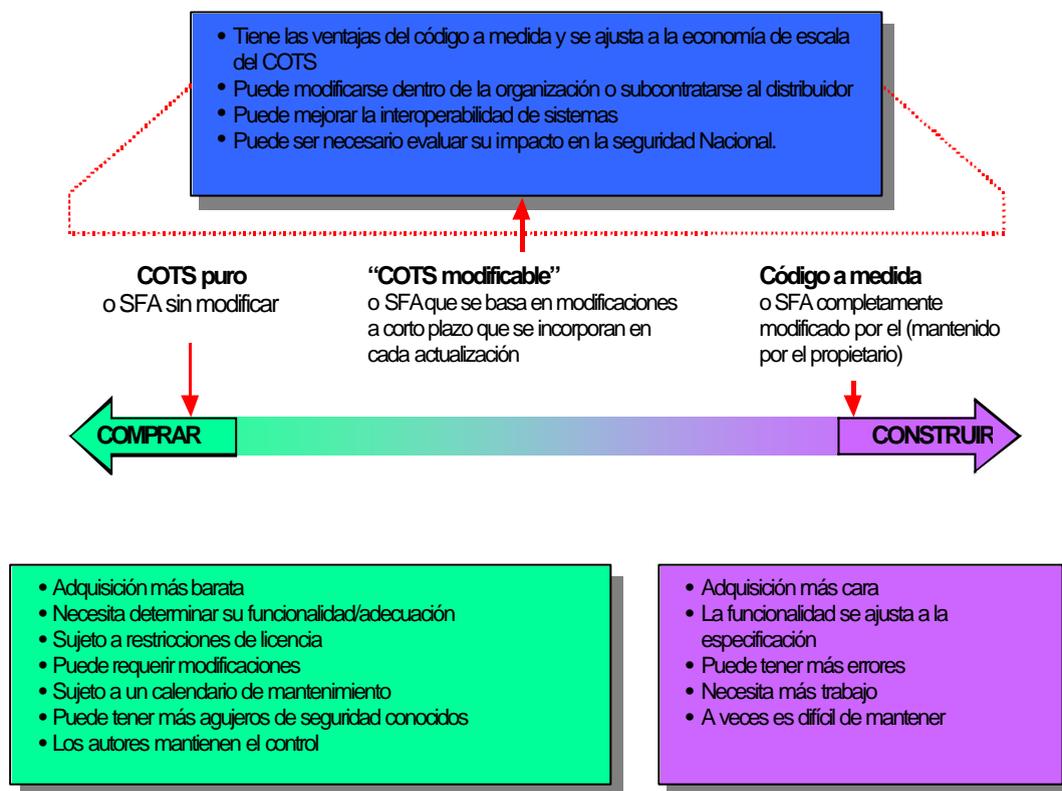


Ilustración ES- 1 El OSS ofrece diferentes opciones de mantenimiento y soporte

Los Jefes de Programa deberían evaluar las ventajas y desventajas relativas del software COTS puro, el “COTS modificable” y los modelos de mantenimiento y personalización del código para las circunstancias específicas de su caso. El COTS puro tiene la ventaja de que es más barato de adquirir. Sin embargo, los Jefes de Programa necesitan contrastar la funcionalidad del software y la adecuación a sus necesidades específicas. El software puede necesitar de modificaciones, y los Jefes de Programa están sujetos a las restricciones de la licencia y a la agenda programada para el mantenimiento. El software COTS puro puede tener más agujeros de seguridad conocidos, y el control está mantenido por los autores del software. El software “COTS modificable” tiene la ventaja de que el cliente puede adaptarlo a la vez que se ajusta a las economías de escala que se obtienen en los productos COTS. El software puede ser modificado en el seno de la organización o puede hacerlo un fabricante. La interoperabilidad de los sistemas puede mejorar con el software “COTS modificable.” Puede que sea necesario evaluar el impacto en la seguridad nacional. El código a medida es más caro de adquirir, puede haber más errores en la funcionalidad según las especificaciones, requiere más trabajo y, en ocasiones, es difícil de mantener.

La fuente abierta beneficiará a la administración al mejorar la interoperabilidad, facilitando el acceso a los datos a largo plazo y la capacidad de incorporar nueva tecnología. La interoperabilidad mejora porque la fuente abierta facilita que el mismo código, documentación y formatos de datos sean usados en todos los componentes del sistema. (Sin embargo, el riesgo de exposición del código debería ser evaluado; si la seguridad de un sistema de fuente abierta se compromete, la interoperabilidad también podría resultar comprometida.) El acceso a los datos a largo plazo da al usuario acceso completo a sus propios sistemas. Es posible subcontratar el trabajo de desarrollo de mantenimiento a proveedores de soporte, que tienen la misma información que el proveedor original. La fuente abierta permite a la administración adoptar nuevas tecnologías más fácilmente porque reduce el coste y el riesgo del cambio. Los proyectos de fuente abierta tienden a ser evolutivos y afectan menos a las operaciones.

El caso de negocio en el entorno militar

El ejército tiene necesidades de software diferentes a los del sector comercial debido a lo singular de su misión y entorno. Los atributos del software que son más importantes para el sector comercial incluyen la posibilidad de elegir las aplicaciones, la facilidad de uso, el servicio y el soporte, el precio, la fiabilidad y el rendimiento. Los atributos del software operacionalmente significativos para el caso del ejército incluyen la fiabilidad, la capacidad de soporte a largo plazo, la seguridad y la escalabilidad. Otros atributos de la mayor importancia programática para el ejército incluyen el coste o precio, la disponibilidad o la existencia de múltiples fuentes de distribución y la popularidad o reputación de la marca comercial.

Así como los sectores comercial y de la administración se preocupan del precio y la fiabilidad, ciertos clientes comerciales tienen generalmente requisitos menos estrictos en cuanto a la seguridad, disponibilidad y capacidad de soporte a largo plazo. Sin embargo, estas características se están volviendo más importantes en el sector privado. Las compañías que se dedican al comercio electrónico deben alcanzar altas cotas de seguridad para proteger la información personal y las transacciones financieras. La disponibilidad del software a través de múltiples fuentes incrementa la competencia, lo que resulta en una mayor calidad a precios bajos. La capacidad de soporte a largo plazo es importante para los negocios que necesitan acceder a datos heredados. Si un producto o proceso comercial, como la fuente abierta, se considera adecuado y ofrece la funcionalidad requerida, el ejército puede aprovecharse de ello para lograr ahorros de coste importantes. Hay otros beneficios potenciales de la equiparación con productos o procesos comerciales, como un tiempo de entrega menor, una mejora en la calidad y la fiabilidad, una reducción en los riesgos asociados al desarrollo y un sistema de soporte ya implantado.

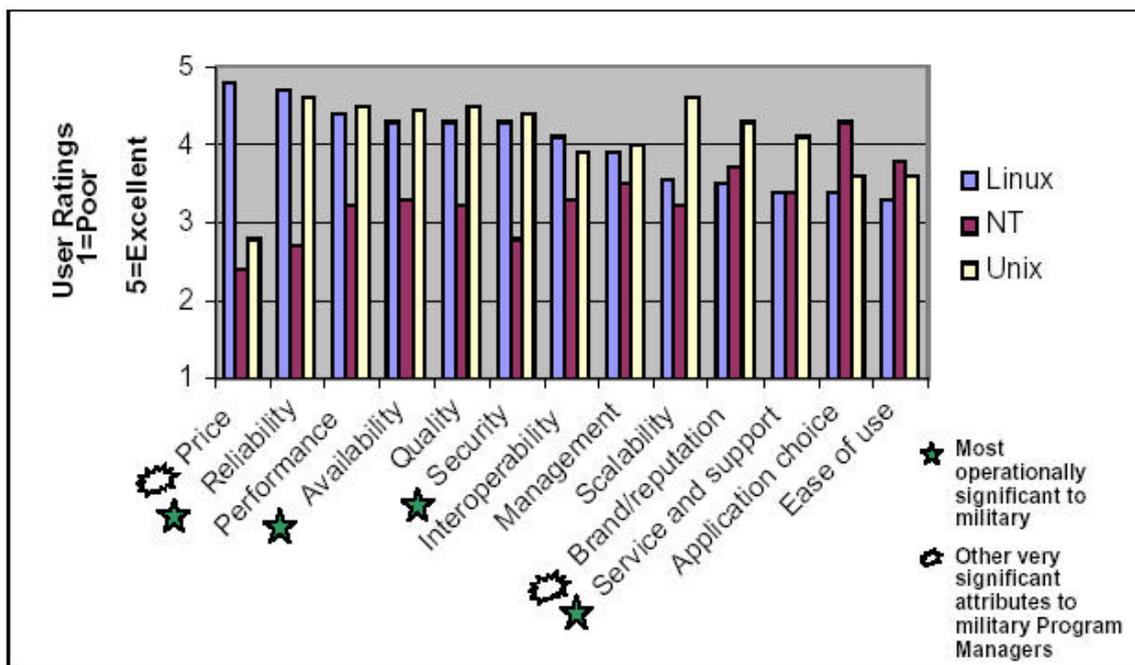
Aplicabilidad de Linux al caso de negocio en el entorno militar

Linux ha atraído a un gran grupo de desarrolladores altamente cualificados y “con los suficientes ojos, todos los errores son superficiales.”¹³ Más de 120.000 programadores contribuyen a Linux, en una acción voluntaria que suma un trabajo que vale unos dos mil

¹³ Eric Raymond, “La catedral y el bazaar,” O’Reilly Asociados, 1.999.

millones de dólares.¹⁴ Los proveedores de productos COTS tradicionales no podrían haberse permitido esta cantidad masiva de experiencia técnica. Como resultado del proceso de fuente abierta, el software producido es altamente fiable y estable. Esta ventaja comparativa, junto con su precio percibido como bajo, hace que Linux atraiga a una enorme base de usuarios en todo el mundo.

La siguiente gráfica compara la calificación que los usuarios dan a Linux, NT y Unix.¹⁵ Mientras que Linux se utiliza por el precio bajo percibido y fiabilidad, NT es preferido por la cantidad de aplicaciones entre las que se puede elegir y su facilidad de uso. Los usuarios seleccionan Unix por su rendimiento, disponibilidad, calidad, seguridad, facilidad de gestión, escalabilidad, reputación de la marca y su servicio y soporte.



Fuente: valoración de distintos SO realizada por usuarios de Linux de EEUU de Michelle Bailley, Vernon Turner, Jean Bozman y Janet Waxman, “Servidores Linux: Qué hay de exageración, que hay de realidad,” IDC, marzo de 2.000.

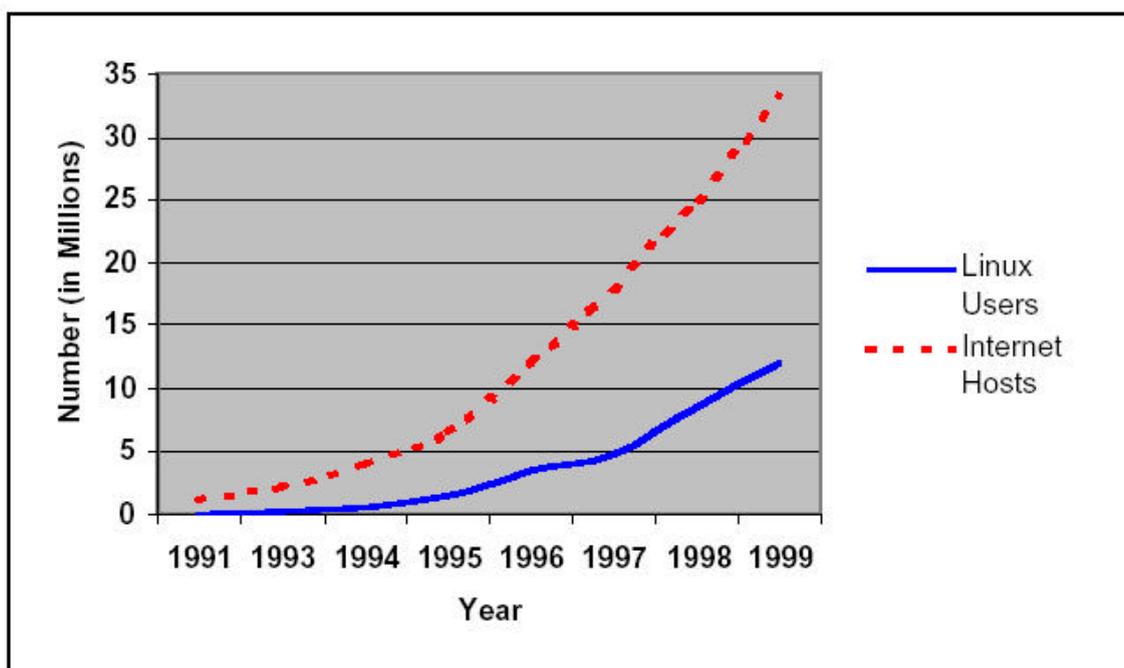
Ilustración ES- 2 Beneficios para el usuario de Linux en el ámbito comercial y militar

¹⁴ Dan Orzech, “Linux y la saga del software de fuente abierta,” Datamation, Febrero de 1.999 y Dan Kaminsky, “Competencias nucleares: porqué la fuente abierta es el paradigma económico óptimo para el software,” 2 de marzo de 1.999.

¹⁵ Calificación de los usuarios estadounidenses por SO de servidor de Michele Bailey, Vernon Turner, Jean Bozman y Janet Waxman, “Servidores Linux: Qué hay de exageración, que hay de realidad,” IDC, marzo de 2.000.

Uso de Linux

El número de usuarios de Linux en todo el mundo ha crecido desde un usuario (Linus Torvalds) en 1.991 a un número estimado de 12 millones de usuarios en 1.999. La siguiente gráfica muestra la evolución del número de usuarios de Linux en todo el mundo contra el número de ordenadores conectados a Internet a nivel mundial. Según la Internet va creciendo, el número de equipos de desarrollo de fuente abierta y su productividad crecen y atraen a más usuarios.¹⁶



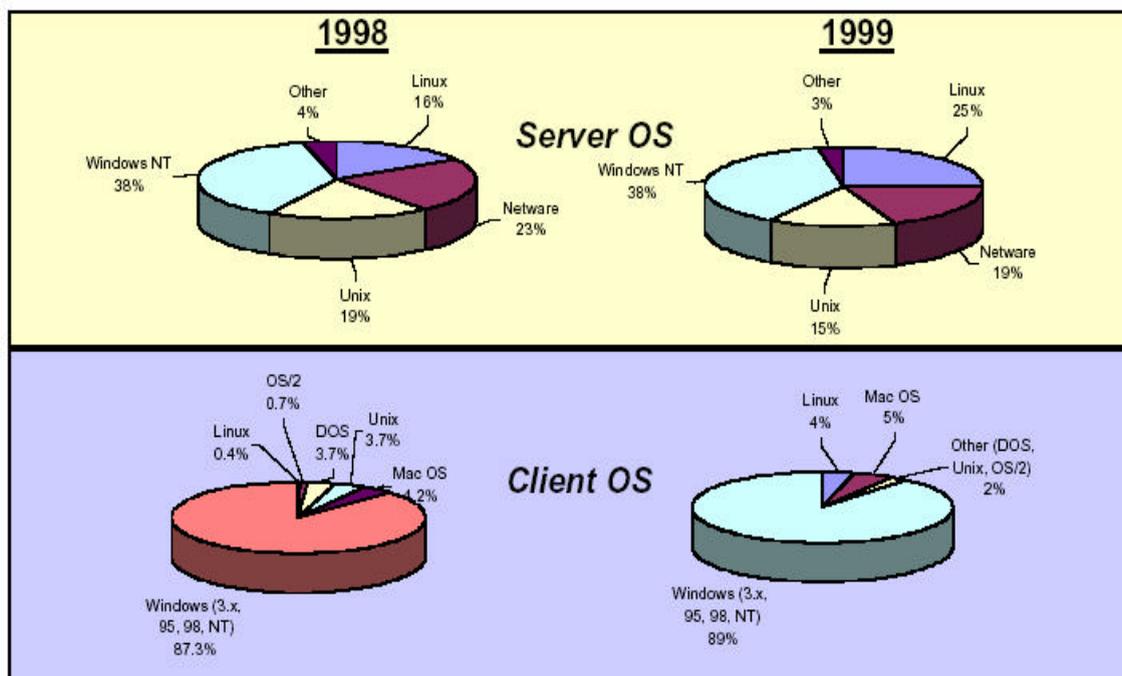
Fuente: Estimaciones sobre Linux del GartnerGroup, IDC, y del desarrollo del mercado de Red Hat. Estimaciones sobre Internet de Bruce L. Egan de 1.996. Los datos son estimaciones al final del año.

Ilustración ES- 3 Éxito a nivel mundial de Linux en el mercado

Se espera que la mayoría de las instalaciones de Linux sean servidores. Se necesitan inversiones significativas en áreas como la facilidad de uso y la configuración para que Linux alcance el éxito en los escritorios.

¹⁶ Las estimaciones sobre Linux vienen del GartnerGroup, IDC, y del desarrollo del mercado de Red Hat. Las estimaciones sobre Internet vienen de un desarrollo de Bruce L. Egan realizado en 1.996. Los datos son estimaciones al final del año.

La siguiente gráfica de tarta muestra la cuota de mercado de Linux para el mercado de SO de servidor y de cliente en 1.998 y 1.999.¹⁷



Fuente: “El futuro de Linux,” Cnet, 2.000 cita a IDC.

Ilustración ES- 4 Cuota del mercado de SO de servidor y de cliente en 1.998 y 1.999

Aunque las instalaciones de Linux se están extendiendo, el dato no es tan significativo. Entre 1.998 y 1.999, la cuota de mercado de SO servidor Linux creció de un 16 a un 25 por ciento y la cuota de mercado de SO cliente Linux creció desde un 0.4 por ciento al 4 por ciento. Parece que la mayor parte de su crecimiento vino de usuarios Unix que cambiaron a Linux.

Discusión

Aunque el proceso de desarrollo de fuente abierta ofrece muchos beneficios sobre el COTS tradicional, Microsoft Windows continua dominando el mercado. Hay varias razones para esto. Primero, Microsoft ha invertido una cantidad muy importante en publicitar Windows a los desarrolladores. Segundo, NT es una plataforma muy amplia que permite que servidores de diferentes distribuidores trabajen en NT. De hecho hay más de 100 distribuidores de servidores NT.¹⁸ Tercero, los usuarios con frecuencia eligen Windows por

¹⁷ “El futuro de Linux,” Cnet, 2.000 cita a IDC, no se da la fecha.

¹⁸ Hohmann Deate, Gartnet Group, conversación telefónica en diciembre de 2.000.

la gran cantidad de aplicaciones compatibles para elegir y su facilidad de uso. Hay una afinidad entre los entornos de escritorio y de servidor cuando se usan los productos de Microsoft. Cuarto, Windows NT ha tenido históricamente un coste de entrada muy inferior comparado con Unix. Los costes de hardware y software son menores cuando se utiliza NT porque el sistema se ejecuta en componentes de bajo coste y circuitería y dispositivos de almacenamiento estándar. Por estas razones, los ejecutivos de TI perciben a Windows como una alternativa menos arriesgada. Los analistas de la industria añaden que “nadie ha sido nunca despedido por comparar Microsoft.”¹⁹

A pesar de estas observaciones favorables a Microsoft, el GartnerGroup ha concluido que no se puede concluir si NT o Unix ofrecen el soporte a largo plazo menos caro. Por el contrario, la opción menos cara depende de la aplicación concreta, el entorno y la base actual de preparación técnica de la organización.²⁰ También debería notarse que Windows no escala también como Unix, y esto vuelve las tornas sobre los costes relativos totales de Windows frente a Unix. NT no es tan potente como Unix y, según las pruebas de GartnerGroup, NT sólo puede soportar hasta 1000 usuarios concurrentes.²¹ Las organizaciones más pequeñas que crezcan deben en consecuencia añadir más equipos para dar soporte a una base de usuarios mayor. En algunos casos, se pueden requerir cinco veces más equipos NT para obtener el mismo rendimiento que el que se obtiene con un equipo Unix. Las organizaciones que no planifican su crecimiento a menudo eligen Windows por su bajo coste inicial, mientras que las organizaciones que planean tener un crecimiento fuerte desde el principio pueden elegir Unix. Por eso, la mejor opción entre Windows y Unix depende del número de usuarios a que el sistema debe dar soporte. Cuando el número de usuarios crezca por encima de los 1000, Unix se vuelve la plataforma más efectiva o la mejor plataforma a seleccionar.

Desde el reciente resurgir del uso de ordenadores en línea que ha ayudado a impulsar la maduración de Linux, se han dado pocas migraciones a Linux. Algunos usuarios de Unix se han pasado a Linux, un SO parecido a Unix. Además, algunas empresas recién creadas con poco capital eligen Linux porque se puede ejecutar con soltura en ordenadores más antiguos. Si más Jefes de Programa compararan el OSS frente al software COTS tradicional para su caso de negocio específico, es probable que hubiese muchos más usuarios de OSS hoy día.

Consideraciones para Jefes de Programa Militar

El OSS ofrece más opciones que el software COTS tradicional para la capacidad de soporte a lo largo del ciclo de vida, particularmente para los sistemas de larga vida. Puede ser utilizado en forma de COTS puro, “COTS modificable,” o código a medida. Los requisitos de los Jefes de Programa para los sistemas operativos difieren considerablemente dependiendo de su entorno particular y requisitos de la misión.

¹⁹ Martín J. Garvy, “El coste oculto de NT,” InformationWeek, julio de 1.998.

²⁰ Deate Hohmann, GartnerGroup, conversación telefónica, diciembre de 2.000.

²¹ Deate Hohmann, GartnerGroup, conversación telefónica, diciembre de 2.000.

Los Jefes de Programa de Comando y Control (C2) se rigen por la operación. Para esos jefes, el coste de fallo es muy alto. La fiabilidad y el rendimiento son esenciales. Los Jefes de Programa C2 utilizan software COTS tradicional a menos que el sistema requiera más personalización y las actualizaciones del sistema tiendan a ser frecuentes. Los Jefes de Programa C2 deberían considerar utilizar Linux porque ofrece el mayor nivel de fiabilidad junto a un buen rendimiento. NT es el más débil para ambas de estas métricas.

Los Jefes de Programa de Sistemas de Información (SI) se rigen por los costes, la calidad del soporte y la cantidad de aplicaciones que pueden elegir. Los sistemas se reemplazan generalmente entre cada cinco a siete años. Si la selección de aplicaciones es importante, los Jefes de Programa SI deberían considerar NT. No obstante, los Jefes de Programa pueden encontrar que tienen más opciones de servicio y soporte con Unix y con Linux. Un intento con la opción de “COTS modificable” con Linux podría ofrecer características adicionales muy valiosas sin la carga añadida de mantenimiento asociada a éstas.

Los Jefes de Programa de Sistemas Empotrados y de Armamento se rigen por la portabilidad, la dureza y los duros requisitos de tiempo real. Las actualizaciones de los sistemas son empresas normalmente costosas. Los Jefes de Programa de Sistemas Empotrados y de Armamento probablemente encontrarán Linux muy atractivo. La flexibilidad de su diseño permite que el núcleo sea recortado para eliminar las características innecesarias o expandido para incluir características adicionales. Linux se puede migrar a muchas unidades centrales de proceso (CPU) y plataformas de hardware. El software puede reconfigurarse dinámicamente si necesidad de rearrancar el sistema. Linux puede aislar fallos y procesos. Los procesos pueden cargar y descargar módulos del núcleo, controladores de dispositivos y módulos personalizados basándose en los recursos disponibles y las necesidades dinámicas de la aplicación. Las aplicaciones también son modulares y tienen interfaces bien definidas. Además, las capacidades de tiempo real están disponibles para la extensión del núcleo de Linux RTLinux.

Premio Federal de Linux

La Corporación MITRE recibió recientemente una Distinción de Liderazgo de una organización sin ánimo de lucro, el Forum Potomac, por mostrar que el OSS puede ofrecer ventajas sustanciales sobre el software comercial, particularmente cuando los requisitos clave son la fiabilidad y el soporte a largo plazo. El galardón fue recientemente presentado conjuntamente a MITRE y al Gabinete del Ministro de Defensa en la primera conferencia del Grupo Federal de Usuarios de Linux en Crystal City, Virginia. MITRE ganó el premio por investigar la tecnología y las motivaciones económicas del OSS en su proyecto de desarrollo, “Código fuente abierto para sistemas militares.” De acuerdo a Mark Norton, Funcionario de la Secretaría del Ministro de Defensa, “este estudio de MITRE es el primer estudio sobre Linux y otro tipo de OSS que trata tanto las ventajas técnicas como y un caso de negocio en el que se utiliza código fuente abierto en el Ministerio de Defensa.” El equipo de desarrollo de MITRE incluía miembros del departamento técnico Frank McPherson, David Emery, Terry Bollinger y Carolyn Kenwood. El trabajo de MITRE incluía una demostración del uso de Linux en sistemas empotrados como el tanque Abrams y el aseguramiento de información en los Centros de Operaciones Tácticas del Ejército. MITRE también analizó el caso de uso federal del OSS para ayudar a los Jefes de Programa

a evaluar la adecuación de éste a sus problemas tecnológicos. Para más información sobre este premio, visite el sitio web http://www.mitre.org/news/articles_00/linux12_5_00.shtml.

Conclusión

El OSS es una solución viable a largo plazo que merece una consideración cuidadosa debido al potencial que tiene en cuanto a ventajas de coste, fiabilidad y soporte. Sin embargo, estos beneficios potenciales deben ser adecuadamente contrastados con la serie de riesgos que están asociados a las aproximaciones y a los productos de OSS. La mejor opción al comparar OSS con software COTS tradicional varía según los requisitos específicos y el entorno operativo del software. El OSS es a menudo una buena opción para productos relevantes y que interesan a una gran comunidad en la que hay desarrolladores altamente cualificados. Típicamente es mejor para servidores e implantaciones de sistemas empotrados que puedan necesitar de alguna personalización, pero no se sitúa mejor que el COTS tradicional para las aplicaciones típicas del escritorio. Cuando se tome una decisión sobre si utilizar OSS o software COTS tradicional se recomienda que los Jefes de Programa sigan los cinco pasos que se presenta a continuación.

1. **Estudie la comunidad de desarrolladores de OSS de soporte (por ejemplo Linux, Apache).** Busque comunidades grandes, con talento y bien organizadas.
2. **Examine el mercado.** ¿Hay una demanda fuerte y creciente de un producto específico de OSS? ¿Hasta que punto los distribuidores y proveedores de servicio han ido en la oferta comercial para ofrecer servicios complementarios y el soporte que la comunidad no ofrezca?
3. **Realice un análisis específico de riesgos y beneficios.** El esfuerzo de MITRE ha desarrollado una taxonomía de beneficios y riesgos del OSS (vea la Tabla ES-2) que puede usarse para comparar productos candidatos de OSS en lo referente a sus objetivos específicos económicos, de rendimiento y necesarios para su misión.
4. **Compare los costes a largo plazo.** Utilice la taxonomía de elementos de coste de OSS desarrollada por MITRE (vea la tabla ES-1) para comparar los costes a largo plazo asociados con el uso y el mantenimiento de OSS frente al COTS tradicional relativos a sus objetivos específicos.
5. **Elija una estrategia.** Siguiendo los anteriores cuatro pasos obtendrá suficiente información de detalle para elegir la combinación más eficaz de OSS, COTS tradicional y desarrollo a medida para alcanzar sus objetivos.

En conclusión, los métodos y productos de fuente abierta bien merecen ser considerados seriamente en un amplio rango de aplicaciones de la Administración, particularmente si se aplican con cuidado y con una sólida comprensión de los riesgos que entrañan. El OSS alienta al desarrollo significativo de software y a la reutilización de código, puede ofrecer beneficios económicos importantes y tiene el potencial de ahorrar costes directos e indirectos especialmente importantes en sistemas militares que necesiten de grandes despliegues de productos de software costosos.

Sección 1

Historia de Unix y Linux

Muchos de los esfuerzos cooperativos de desarrollo en los 70 se enfocaron en construir un sistema operativo que pudiera correr en múltiples plataformas informáticas. El sistema operativo Unix emergió como el más exitoso de estos esfuerzos. El proceso de compartición de código rápidamente se aceleró con el surgimiento de Usenet, una red informática que en 1979 comenzó a interconectar a la comunidad de programadores Unix. Hasta este punto, los esfuerzos cooperativos para el desarrollo de software eran informales y no intentaron definir cuestiones como los derechos de propiedad o las restricciones de uso. Este esquema informal se hizo problemático al principio de los 80 cuando AT&T reclamó los derechos de propiedad intelectual que tenían que ver con Unix.

En 1986 los desarrolladores intentaron construir una versión libre del sistema operativo Unix. Este proyecto, llamado GNU, permitía a los programadores individuales, con independencia del interés individual o comercial, contribuir al esfuerzo de desarrollo. GNU son las iniciales de “GNU No es Unix.” Al final, no se cobraba a los usuarios por el sistema operativo.

La Licencia Pública General de GNU,²² también conocido como acuerdo de donación de copia, incluye los siguientes puntos clave:²³

- El software licenciado bajo la Licencia Pública General de GNU puede ser copiado y distribuido bajo esta misma licencia.
- Los productos obtenidos y distribuidos bajo esta licencia pueden venderse.
- Los usuarios pueden alterar el código fuente, pero si lo distribuyen o publican el trabajo resultante, deben hacer disponible el código bajo los mismos términos de licencia.
- La tecnología auxiliar que pueda desarrollarse, siempre y cuando no incluya productos cuyo código esté licenciado bajo la Licencia Pública General de GNU, no necesita ser licenciada bajo los términos de la Licencia Pública General de GNU.

La licencia GNU completa se encuentra en <http://www.gnu.org/copyleft/gpl.html>.

El núcleo de Linux salió de un proyecto educativo en la Universidad de Helsinki en 1991. Linus Torvalds, un joven estudiante, creó el sistema operativo Linux y dio a los hackers su código para que pudieran contribuir a su desarrollo. Muchos programadores analizaron su código y escribieron mejoras que Linus incorporó a Linux. Linux creció y se convirtió en un potente y avanzado sistema operativo multiusuario.

La adecuación a un estándar abierto siempre fue un objetivo importante de Torvalds. Linux apunta hacia el Interfaz de Sistema Operativo Portable (POSIX), un interfaz de programación de aplicaciones (API) estándar y habitualmente utilizado por los sistemas Unix y los sistemas de tipo Unix, pero no es conforme a todas las especificaciones POSIX

²² El acrónimo GNU son las siglas de GNU No es Unix.

²³ Al Gillen y Dan Kusnetzky, “Linux de un vistazo: comprensión del modelo de mercado de Linux,” IDC, febrero de 2.000.

que están incluidas en los principales sistemas operativos Unix. El uso de POSIX hace más fácil escribir código fuente que pueda ser compilado en sistemas POSIX diferentes. Da a los desarrolladores Linux una API compartida bien definida de forma que no tengan que controlar la mayoría de los cambios en el núcleo siempre y cuando sigan POSIX. El uso de POSIX permitió a Linus y a los desarrolladores iniciales de Linux el uso de programas libres escritos para el Proyecto GNU, el sistema operativo BSD y muchos otros programas libres basados en POSIX. Sin embargo, Linux no fue evaluado por la Fundación del Software Abierto, la propietaria de la marca Unix y, por eso, no es considerada una implementación de Unix.

Sección 2

Modelo de análisis del caso de negocio

El modelo de análisis del caso de negocio fue aplicado a los productos y procesos Linux, y el marco adaptado fue utilizado para analizar la viabilidad de los productos software y las metodologías de diseño de Linux para los Jefes de Programa gubernamentales.

Primero, se analizamos el entorno. El entorno externo fue examinado — incluyendo clientes, competidores, obstáculos en la entrada, sustitutos, suministradores y distribuidores. Las oportunidades y las amenazas fueron identificadas a partir del análisis externo. También la comunidad interna de Linux fue contrastada — incluyendo la experiencia, la competencia técnica, la gestión, la salud financiera, la cultura, la estructura organizativa, y los productos y servicios. Las fortalezas y las debilidades fueron reconocidas a partir del análisis interno.

En un segundo paso analizamos los factores estratégicos. La investigación interna y externa se integró para definir el análisis de las fortalezas, las debilidades, las oportunidades y las amenazas (DAFO). La experiencia interna se comparó con los competidores para identificar las competencias distintivas. Para analizar éstas, el equipo de desarrollo examinó la capacidad que tenían las ventajas competitivas de realizar una necesidad no cubierta por el mercado, y la estrategia para competir en precio, calidad y/o personalización. Además, se evaluaron nuevos productos y servicios con relación a los existentes. La demanda del mercado, tanto actual como potencial, fue contrastada. Se hicieron preguntas como: ¿Cuántos clientes están invirtiendo actualmente? ¿Qué porcentaje está dispuesto a comprar los nuevos productos y/o servicios? ¿Cuánto están dispuestos a gastar? El equipo analizó la posición de Linux en el mercado, identificando las oportunidades a corto y largo plazo y señalando los riesgos.

Finalmente, el equipo MOIE estudió la viabilidad de Linux basada en la información obtenida en los pasos 1 (análisis del entorno) y 2 (análisis de los factores estratégicos).

La siguiente figura ilustra el marco de análisis del caso de negocio. Lo que incluye los tres pasos antedichos así como las numerosas entradas al proceso.

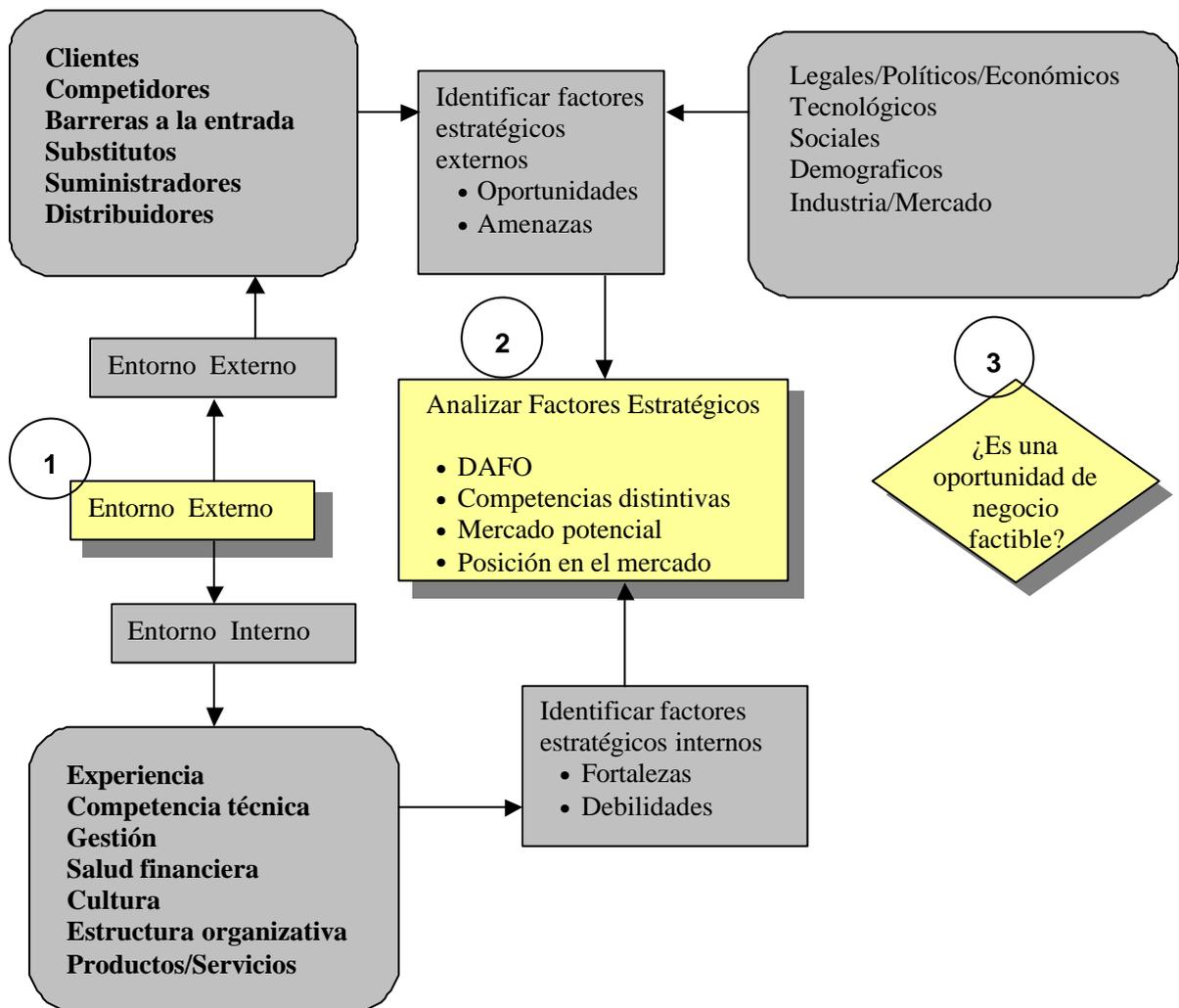


Ilustración 1 Marco de análisis de caso de negocio aplicado a los productos y procesos de fuente abierta

Sección 3

Análisis de las posibilidades comerciales de Linux

El modelo de análisis del aspecto comercial fue utilizado para evaluar la viabilidad comercial de Linux. Los resultados indican que existen oportunidades de negocio asociadas a Linux.

3.1 Estudio del entorno

La figura siguiente resume los elementos clave del análisis DAFO (Debilidades, Amenazas, Fortalezas y Oportunidades) Bajo el encabezado "Entorno Interior", los colores verde y rojo indican las fortalezas y las debilidades respectivamente. Bajo el encabezado "Ambiente exterior", el verde especifica oportunidades y el rojo las amenazas. El color amarillo indica en ambos una circunstancia que puede ser fortaleza y debilidad u oportunidad y amenaza.

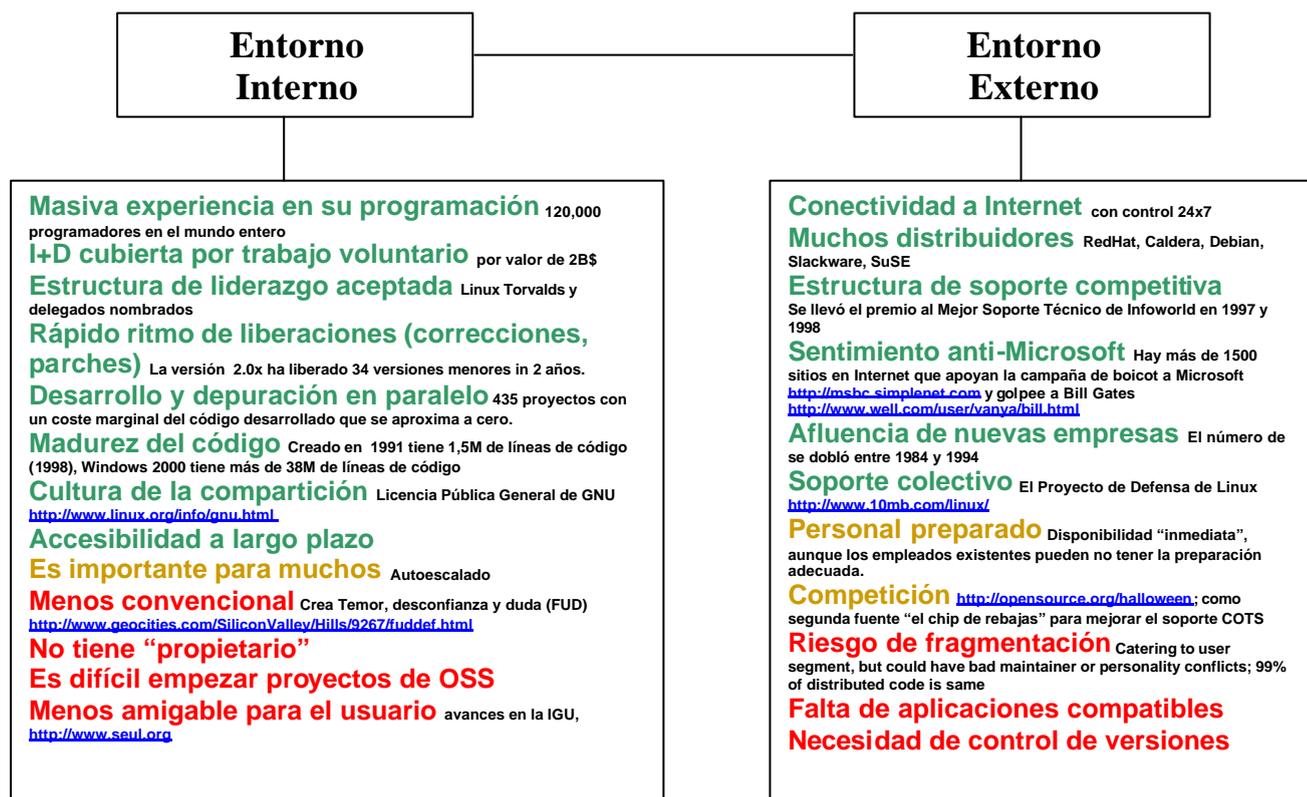


Ilustración 2. Elementos clave del análisis DAFO

3.1.1 Fortalezas

3.1.1.1 Experiencia masiva en su programación

Linux tiene un equipo numeroso de programadores expertos, por encima de 120.000 programadores de diversos países y hay veinticuatro desarrolladores en Linux repartidos por el mundo.²⁴ Mucha gente fuera de EE.UU. apoya Linux como un medio de reducir la dominación técnica americana. El desarrollo de software de código abierto es un fenómeno auto regulado; cuanto más interesante es un proyecto, más programadores querrán unirse a él. Se estima que sólo el cinco o diez por ciento del código actual del núcleo de Linux ha sido escrito por Linus Torvalds.²⁵

3.1.1.2 I+D Cubierta por trabajo de voluntarios

El esfuerzo de Investigación y Desarrollo está cubierto con trabajo voluntario valorado aproximadamente en unos dos mil millones de dólares.²⁶ Las compañías que desarrollan sus propios sistemas operativos gastan aproximadamente entre 80 y 100 millones por año para mantenerse en el mercado.²⁷ Los programadores que contribuyen a crear Linux lo hacen como afición o por satisfacción personal, normalmente al margen de sus responsabilidades profesionales. Sin embargo, la aparición de distribuciones comerciales, está empezando a cambiar esto un poco; los distribuidores de Linux contratan frecuentemente, programadores a jornada completa para mejorar el código y contribuir al crecimiento del mercado de Linux. Los desarrolladores están motivados para contribuir con su tiempo y sin reembolso monetario. A veces corrigen un error o personalizan un programa en su propio beneficio (y, por consiguiente, para el beneficio de otros también). Otros contribuyen al desarrollo de código libre para satisfacer su ego y obtener una reputación entre otros hackers. Como en las culturas del regalo, "el nivel social no se determinado por aquello que controlas sino por aquello que regalas".²⁸ Otras recompensas tardías e inesperadas, como ofertas de trabajo, participación en las compañías comerciales basadas en la fuente abierta, o acceso al mercado de opciones de capital fueron recibidas por Linus Torvalds y otros programadores de fuente abierta.

3.1.1.3 Estructura de liderazgo aceptada

Linux tiene una estructura de liderazgo aceptada, similar a la encontrada en una organización corporativa. La comunidad de Linux está presidida por Linus Torvalds, un gerente muy respetado, que ha adquirido el estatus de celebridad de forma que sus

²⁴ Dan Orzech, "Linux y la saga del Software de Fuente Abierta", Datamation, febrero 1.999.

²⁵ Dan Kaminsky, "Competencias nucleares: Porqué la Fuente Abierta es el mejor paradigma económico para el software", 02/03/1.999.

²⁶ Dan Kaminsky, "Competencias nucleares: Porqué la Fuente Abierta es el mejor paradigma económico para el software", 02/03/1.999.

²⁷ Dan Kusnetsky, IDC y Greg Weiss, DH Brown & Asociados, Serie de seminarios electrónicos sobre Linux, 1.999.

²⁸ Raymond, Eric, "Contruyendo la noosfera" (no figura fecha).

decisiones son consideradas finales. Torvalds tiene nombrados delegados que son responsables de la gestión de ciertas áreas del proyecto y a su vez, estos delegados tienen un equipo de coordinadores. Sin embargo, esta estructura de liderazgo sólo afecta al kernel de Linux; no se extiende a áreas como la Interfaz Gráfica de Usuario (IGU), las utilidades del sistema y los servicios o las bibliotecas del sistema.

3.1.1.4 Rápido ritmo de liberación de versiones

Linux presenta nuevas versiones y actualizaciones corrigiendo fallos rápidamente, proporcionalmente más rápido que el software comercial. Por ejemplo, de la versión 2.0x se han publicado 34 versiones en dos años.²⁹

3.1.1.5 Desarrollo y depuración en paralelo

Los proyectos de fuente abierta utilizan muchos equipos pequeños de individuos que trabajan de forma independiente para resolver problemas específicos. Debido a que los programadores de fuente abierta trabajan desde el principio del voluntariado, el proceso de desarrollo paralelo no supone un coste prohibitivo como ocurriría en el sector privado. La fuente abierta fomenta la creatividad, ya que a los programadores no se les establecen limitaciones a su trabajo. El proceso de desarrollo paralelo hace posible que se estén desarrollando 435 proyectos para Linux de forma concurrente.³⁰ Como los desarrolladores no perciben remuneración por sus contribuciones a los productos de fuente abierta, el costo marginal de desarrollo se aproxima a cero. La depuración paralela y esfuerzos de desarrollo le permiten al coordinador del proyecto escoger la implementación potencialmente mejor de las muchas opciones ofrecidas.

La depuración paralela, según Eric Raymond, mejora la eficacia casi linealmente con el número de individuos que trabajan en el proyecto. Aunque pequeños, los costes de dirección existen en la depuración de la fuente abierta. Es típico de la depuración paralela solucionar errores más rápido que los procesos tradicionales. Por ejemplo, la comunidad de Linux desarrolló una solución para la vulnerabilidad "TearDrop IP" menos de 24 horas después de su primera aparición en el web. Las organizaciones no dependen del horario de un proveedor comercial para solucionar un error que impide trabajar pero pueden en cambio, optar por arreglar ellos mismos el problema.

3.1.1.6 Madurez del código

El código de Linux se creó en 1.991, y en 1.998 ya comprendía 1,5 millones de líneas de código. Windows 2.000 tiene unos 38 millones de líneas de código. Frederick Brooks

²⁹ Vinod Valloppillil y Josh Cohen, Microsoft, "Análisis competitivo del SO Linux," Halloween II, 11 de agosto de 1.998.

³⁰ Dan Orzech, "Linux y la saga del Software de Fuente Abierta", Datamation, Febrero 1.999.

afirma que “la complejidad crece con el cuadrado del número de líneas de código” por ello Windows 2.000 es muchísimo más complejo que Linux.³¹

3.1.1.7 Cultura de la compartición

La licencia GPL (GNU General Public License) ha fomentado una cultura de compartir que se extiende a través de la comunidad. La fuente abierta elimina la pérdida económica asociada a la duplicidad de trabajos. Aproximadamente el 75 por ciento de todo el código escrito para una tarea específica por una sola organización nunca se usa para otro propósito.³² Muchos desarrollos en ingeniería de software son solución de una amplia gama de problemas en diferentes campos. Este desperdicio económico disminuye la productividad de Norteamérica.

3.1.1.8 Accesibilidad a largo plazo

A diferencia de las compañías privadas, al movimiento de la fuente abierta no se le puede echar del negocio a corto plazo. Mientras exista un interés suficiente por parte de la comunidad de desarrolladores, el ciclo de vida del producto de fuente abierta continuará. Además, como el código es público, el usuario no depende de una organización para mantener y desarrollar el software. El usuario siempre tiene la opción de mantener por sí mismo el código y prolongar indefinidamente la vida del producto.

3.1.2 Debilidades

3.1.2.1 Falta de “propiedad”

Los usuarios quieren responsabilidad. Una compañía privada es más tangible que la “comunidad de la fuente abierta” y puede posicionarse ante los clientes actuales y potenciales como un atento empleado. Microsoft, por ejemplo, utiliza las tácticas FUD (temor, incertidumbre y duda). Se trata de una técnica de ventas usada por las compañías con una posición dominante en el mercado (p.e., Microsoft, IBM) y que son incapaces de responder con hechos evidentes a un producto de la competencia que es mejor y más barato. Esta técnica genera dudas y rumores entre los usuarios potenciales, enviándoles un mensaje parecido a “Eh, podría ser arriesgado seguir ese camino, únase a nosotros y pertenezca a la mayoría. Nuestra nueva versión, que saldrá muy pronto, será sin duda mejor que eso.”³³

Para reducir esta debilidad de Ausencia de “Propietario”, Eric Raymond propone crear documentación de más alta calidad que promueva la confianza de los clientes en los autores y distribuidores. A diferencia de la fuente abierta, los fabricantes de software propietario pueden garantizar la compatibilidad con versiones anteriores y poseen una personalidad jurídica susceptible de ser demandada si no se cumplen las promesas. Existe también preocupación por la ausencia de una dirección estratégica al frente de los proyectos

³¹ Walker White: “Observaciones, consideraciones y direcciones”. Oracle 5 de agosto de 2.000 citado por Federick Brooks en “El mítico hombre/mes”.

³² Mitch Stoltz: “La promoción de Software de Fuente Abierta en la Administración”, Informe de NetAction 1.999.

³³ Eric Raymond: “¿Qué es el FUD?” <http://www.opensource.org/>, no consta fecha.

de fuente abierta, aunque los innovadores trabajan continuamente para agregar nuevas funcionalidades, mejor diseño y eficacia. Como resumió Martin J. Garvey en InformationWeek: “nadie nunca fué despedido por comprar Microsoft.”³⁴

3.1.2.2 Difícil de iniciar

Para que un proyecto de fuente abierta sea viable, debe poder reunir un grupo suficientemente grande de programadores expertos interesados en concentrarse en el problema. Una de las reglas de Eric Raymond es que “todo buen trabajo de software empieza provocando en un programador un desafío personal.”³⁵ El proyecto de fuente abierta debe ser relevante e interesante para un grupo numeroso de diseñadores. Un proyecto más grande, recibe una cantidad mayor de líneas de código y correcciones de errores. Los desarrolladores deben compartir un objetivo común que está clara y bien definido, así como de la misión de organización. Linux tuvo éxito venciendo esta debilidad potencial porque la comunidad de Linux tenía (más de 25) años de experiencias compartidas llevando adelante otras formas de Unix; ya había adoptado unos puntos de referencia comunes a través de Unix.

3.1.2.3 Menos amigable para el usuario

Aunque Linux está trabajando mejorar su amigabilidad para con el usuario, su interfaz gráfica de usuario (IGU), es el pariente pobre de Microsoft y otros productos de software. Linux se desarrolló para los programadores, en lugar de para usuarios profanos. La figura siguiente muestra un trozo de código de Linux, parte del código de fuente del núcleo que se ocupa de la bifurcación de procesos (una operación básica de cualquier núcleo tipo Unix).³⁶

³⁴ Martin J. Garvey: “El coste oculto de NT”, InformationWeek, julio 1.998.

³⁵ Eric Raymond: “La catedral y el bazaar”, O’Reilly Asociados, 1.999.

³⁶ Frank McPherson, Corporación MITRE, 2.000.

```

/*
 * For SMP, we need to re-test the user struct counter
 * after having acquired the spinlock. This allows us to do
 * the common case (not freeing anything) without having
 * any locking.
 */
#ifdef __SMP__
    #define uid_hash_free(up) (!atomic_read(&(up)->count))
#else
    #define uid_hash_free(up) (1)
#endif
void free_uid(struct task_struct *p)
{
    struct user_struct *up = p->user;
    if (up) {
        p->user = NULL;
        if (atomic_dec_and_test(&up->count)) {
            spin_lock(&uidhash_lock);
            if (uid_hash_free(up)) {
                uid_hash_remove(up);
                kmem_cache_free(uid_cache, up);
            }
            spin_unlock(&uidhash_lock);
        }
    }
}

```

Fuente: Frank McPherson, Corporación MITRE, 2.000.

Ilustración 3. Fragmento de código fuente de Linux

Al contrario que el núcleo, mantenido por Linus Torvalds, la IGU de Linux no ha sido mantenida de ninguna manera en especial así que es un árbol con muchas ramas diferentes. El software no tiene una IGU consistente y homogénea, por lo que los usuarios deben adaptarse a las diferencias. Las iniciativas GNOME, KDE y CDE están trabajando para mejorar la IGU. KDE integra navegador, terminal y paquete de ofimática para el escritorio de Unix. Pueden encontrarse capturas de pantalla de KDE en <http://www.kde.org/ksscreenshots.html>. A continuación se muestra una imagen ejemplo de la pantalla de la IGU de Linux.³⁷

³⁷ <http://www.gnome.org/screenshots/index.html>, sin indicación de fecha.



Fuente: <http://www.gnome.org/screenshots/index.html>

Ilustración 4. Ejemplo de pantalla de Linux

Existen aplicaciones y componentes de conectividad para facilitar la instalación de servidores. Por ejemplo, Samba es un paquete servidor de ficheros de fuente abierta que permite a un servidor Linux proporcionar servicios de compartición de impresoras y ficheros a clientes de escritorio Microsoft Windows.

3.1.3 Oportunidades

3.1.3.1 Conectividad con Internet

El número y productividad de equipos de desarrollo de fuente abierta se expande con Internet. La tecnología de Internet permite el desarrollo y soporte técnico de Linux 24 horas al día, 7 días por semana alrededor de todo el mundo. El crecimiento de Internet continuará extendiendo Linux y otros proyectos de fuente abierta haciéndolos accesibles a un número

mayor de personas. Las tecnologías de colaboración, como las listas del correo electrónico, los grupos de noticias y las páginas web han mantenido el crecimiento de la fuente abierta.

3.1.3.2 Múltiples distribuidores

Hay unas 204 distribuciones de Linux en el mercado.³⁸ Los distribuidores ofrecen Linux empaquetado con herramientas integradas. Actúan como un intermediario entre el acelerado proceso de desarrollo y los clientes de que no se preocupan día a día de los cambios del núcleo, eliminando la molestia de descargar de Internet el sistema operativo. Los distribuidores también están proporcionando ventas, apoyo, e integración, dando énfasis a los servicios en lugar de al propio producto de software. Estos servicios complementarios frecuentemente no son prestados por la comunidad de desarrollo. Como los distribuidores de Linux compiten entre si, realizan un aumento de la evolución del sistema operativo, agregando prestaciones, mejorando su empaquetamiento e información de uso. La comunidad de fuente abierta proporciona un continuado “servicio” evolucionando con el usuario para identificar nuevas necesidades, en lugar de ser un “producto” que permanece estático y que en el futuro requerirá ser reemplazado.

Mientras tanto es probable que se produzca una estabilización del mercado en los próximos dos años, las versiones regionales y versiones específicas del software continuarán existiendo para satisfacer las demandas del mercado la demanda. La tabla siguiente lista algunas de las distribuciones de Linux más comunes.

Tabla 1. Distribuciones comunes de Linux ordenadas por distribuidor

Distribuidor	Nombre de la distribución	URL
Caldera Systems	OpenLinux	www.caldera.com
CoolLogic	Coollinux	www.coollogic.com
Corel	Corel LINUX	www.corel.com
Debian Project	Debian	www.debian.org
LinuxPPC	LinuxPPC	www.linuxppc.com
MandrakeSoft	Linux-Mandrake	www.linux-mandrake.com/en/
CLE Project	CLE	cle.Linux.org.tw/CLE/e_index.shtml
Red Hat	Red Hat Linux	www.redhat.com
Slackware Linux	Slackware Linux	www.slackware.com
Stampede GNU/Linux	Stampede GNU/Linux	www.stampede.org
SuSE	SuSE Linux	www.suse.com
TurboLinux	TurboLinux	www.turbolinux.com

³⁸ “Conociendo Linux”, Colorado Business, julio de 2.000.

Red Hat, fundada en 1.994, distribuye la versión más popular de Linux y preferida por el 68,7% de los usuarios de Linux en Estados Unidos, según IDC.³⁹ La compañía proporciona soluciones de fuente abierta para servidores de Internet a través de una amplia gama de aplicaciones para dispositivos empujados, clusters y servidores web. La compañía aporta sus innovaciones en software de forma gratuita para la comunidad bajo la Licencia GPL. Red Hat ha firmado acuerdos con los mayores fabricantes de PC y servidores, incluidos Compaq, Dell, Gateway, IBM, Hewlett-Packard y Silicon Graphics. Red Hat tiene un sistema gráfico que proporciona ayuda en línea durante el proceso de instalación. Ha simplificado la tarea de instalar utilidades de terceros en el sistema con su Gestor de Paquetes Red Hat (Red Hat Package Manager, RPM) con un formato propio de paquetes. Red Hat también ofrece entrenamiento y un programa de certificación. Red Hat y el SO Linux Red Hat han recibido muchos premios incluyendo: su inclusión en la lista de Red Herring de las 100 mejores compañías del sector electrónico, así como en la lista de las 100 mejores compañías en crecimiento y en la lista de las 10 compañías a destacar en el mundo de la Red, mejor compra para la Administración según Federal Computer Week's, premio Jolt al desarrollo de software y producto del año por tres años consecutivos de InfoWorld. La compañía ha anunciado su salida a bolsa con una oferta de acciones por valor de 96,6 millones de dólares en junio de 1.999.

Como Red Hat, el OpenLinux de Caldera ofrece la gestión de paquetes y sistema gráfico de instalación. Tiene amplios programas de entrenamiento tanto a distancia como a través del Centro Autorizado de Formación Linux, diseñado para preparar a sus alumnos para el Certificado del Instituto Profesional de Linux.

Debian, una organización sin ánimo de lucro, construyó la distribución de GNU/Linux a través del trabajo de voluntarios. Esta distribución es la mejor opción de software libre. La distribución se pone al día frecuentemente y es apoyada por un grupo bien organizado de partidarios. El formato de los paquetes es similar a RPM. Aunque la instalación inicial del sistema inicial no es tan amigable como la de Red Hat, un administrador de sistemas con experiencia lo encontrará más eficaz. Aunque la distribución de Debian no está disponible a través de la organización Debian, está disponible a través de otros distribuidores, como Corel.

Slackware era una de las primeras distribuciones que surgieron y continúa siendo muy utilizada. Sin embargo, su instalación es menos intuitiva que la de otras distribuciones. Slackware confía en los archivos comprimidos y no apoya el popular formato RPM.

SuSE es la distribución más popular entre los usuarios de Linux europeos. A menudo es recomendado para usuarios entre intermedios y avanzados que buscan una distribución segura en su instalación por defecto.

CoolLogic es un importante desarrollador de sistemas operativos empujados.

Coollinux es un sistema operativo de tiempo real diseñado para aplicaciones en Internet. Se puede reducir el tamaño del núcleo para cumplir las restricciones de memoria y requisitos funcionales en los sistemas empujados. Coollinux puede llegar a ocupar 355 Kilobytes.

³⁹ RedHat, <http://www.redhat.com/>, no consta fecha.

LinuxPPC, MandrakeSoft, CLE Project, Stampede GNU/Linux, y TurboLinux también son proveedores de distribuciones comunes de Linux.

Además de los distribuidores anteriores, muchos vendedores en sectores específicos del mercado venden distribuciones de Linux o versiones que no incluyen royalties para el creador de la distribución original. Estas compañías son negocios más pequeños dirigidos a un sector demográfico específico del mercado. La tabla siguiente incluye una lista de estos vendedores especializados.⁴⁰

Tabla 2. Distribuidores de Linux especializados en sectores específicos

Vendor	URL	Vendor	URL
Admail Japan	ms.shi.nu	Linuxbutiken.com	www.linuxbutiken.com
Alcove	www.alcove.fr/guide-linux/logiciels/debian.html	LinuxLand International	www.linuxland.de
Alejandro Sierra	debian ipt.com.mx	LinuxPPC	www.linuxppc.com
Armed.net	www.armed.net	LinuxStore	www.linuxstore.com.br
BeNeLinux	home1.freegates.be/blinux	Livraria Tempo Real	linux.temporeal.com.br
Buchhandlung Lehmanns	www.lcb.de	LordSutch.com	www.lordsutch.com/ods
CD House di CA Barwood	www.sbf.it/cdnet	LSL Australia	www.lsl.com.au
Cheap Bytes	www.cheapbytes.com	Macmillan USA	www.mcp.com
Ciberdroide Informatica	www.ciberdroide.com	MandrakeSoft	www.linux-mandrake.com/en
Coyote Linux	www2.vortech.net/coyte/coyte.htm	MNIS	www.mnis.fr
Data-Portal	www.data-portal.com	Mr O's Linux Emporium	www.ouin.com
DLX Linux	www.wu-wien.ac.at/user/h93/h9301726/dlx.html	Net North West	www.netnw.com
DOS Linux	www.tux.org/pub/people/kent-roboitt/index.html	NetArt	www.netart.com.pl
Dr Floppy Computers	www.drffloppy.co.nz	nettstore	www.nettstore.de
Dragon Linux	www.dragonlinux.nu	NoMad Linux	www.nomadlinux.com/nomad.html
Dutch Debian Distribution Initiative	panic.et.tudelft.nl/debian/od	Open Source project	www.webnix.com/CLEa_index.shtml
easy Information	www.eit.de/c/index.html	Peanut Linux	www.metalab.unc.edu/peanut
edv-multimedia service	www.net-operations.de	Penguin Computing	www.penguincomputing.com
endani Star System	www.endani.co.uk	Philip Charles	www.copyleft.co.nz
Esware Linux	www.esware.com	Philip Hands Computing	www.hands.com/debiangold.html
Euritec Linux	www.etsi.upm.es/~euritec/indexuk.html	Prosa Debian Linux	www.prosa.it/prosa-debian
Everything Linux	everythinglinux.com.au	Rebel.com	www.rebel.com
Frank CDROM	www.frank-cdrom.co.at	Rock Linux	www.rocklinux.org/index.html
Greenbush Technologies	www.greenbush.com/linuxod	rrbs Linux-Software, alpha-Systems and Network Solutions	www.rrboehm.de/english/index.html
German Unix User Group	www.guug.de/linux.html	Schlittermann	debian.schlittermann.de
Hal91 Linux	home.sol.no/~okolaas/hal91.html	SGI Linux	files.c3.hu/cgi-linux
Hypercore Software Design	www.hypercore.co.jp	Slackware Linux	www.slackware.com
IKARIOS	www.ikarios.com	Softcopy Systems	www.softcopy.onca
Indelible Blue	www.indelible-blue.com	SUFTEX	cypla.com/pc
Infomagic	www.infomagic.com	softlinux.com.br	www.softlinux.com.br
Investigación y Desarrollo Agora	www.id-agora.com	SOT Finnish Software Engineering	www.sot.com
iplabs	www.iplabs.ru	Step Computers & Communications	www.step.gr
IntraLinux	www.intralinux.com	Steve McIntyre	www.chiark.greenland.org.uk/~stevem/DebianCD
ixsoft Softwareentwicklung und-vertrieb Bernd Hentig	www.ixsoft.de	Stivell	www.stivell.com
Libra Computer Systems	www.libra.net.com	Storm Linux	www.stormix.com
Lineo	www.lineo.com	Terra Soft Solutions	www.yellowdoglinux.com
Linpus Technologies	www.linpus.com.tw	TCMS/IT	www.tcms.net/itb
Linux Canada	www.linuxcanada.com	Trees.UK	tree.uk.com
Linux CDs	www.tienpiek.net	Trinux	www.trinux.org
Linux Central	linuxcentral.com	Tyse.Net	www.tyse.net
Linux Emporium	www.linuxemporium.co.uk	Virtuale Web	www.virtuale.com.br
Linux Mall	www.LinuxMall.com	Walnut Creek Linux	www.cdrom.com
Linux Press	www.linuxpress.com	WinLinux2000	www.winlinux.com
Linux Shop	linux.qb.net	Yggdrasil Computing	www.yggdrasil.com
Linux Systems Labs	www.lsl.com		

Fuente: Al Gillen y Dan Kusnetzky, "Linux de un vistazo: comprensión del modelo de mercado de Linux," IDC, febrero de 2.000.

⁴⁰ Al Gillen y Dan Kusnetzky, "Linux de un vistazo: comprensión del modelo de mercado de Linux," IDC, febrero de 2.000.

Algunas compañías convencionales están intentando aprovechar la ascensión de Linux creando versiones de sus productos para Linux bajo un acuerdo de licencia. Tales vendedores (y los productos) incluyen: Oracle (Oracle 8i y las aplicaciones Oracle), Hewlett-Packard (OpenMail), IBM (DB2, VisualAge, MQ Series, TX Series), Lotus (Domino), Tivoli (TME-10), Transarc (AFS y DFS), Computer Associates (Unicenter TNG), Sybase (Adaptive Server), Informix (Dynamic Server), SAP (fragmentos de R3), y WebTrends (WebTrends Enterprise Reporting Server).⁴¹ Netscape y Sun también han hecho versiones Linux de sus productos.

IBM se promueve como una alternativa competitiva a Solaris. Iniciando una estrategia de mercado en apoyo de Linux, IBM ha desarrollado lazos íntimos con la comunidad de la fuente abierta y ha mostrado una buena disposición para invertir en iniciativas de fuente abierta y distribuidores de Linux. La compañía ha afirmado que Linux mejorará y estará maduro para su uso en la empresa hacia el 2.005.⁴²

3.1.3.3 Estructura de soporte competitiva

En el software propietario el soporte técnico es el monopolio de una compañía que lo proporciona, pero se "guarda cartas en la manga" (p.e. el acceso al código) para una aplicación de software determinada. Esto da a los usuarios la opción de aguantarse con el servicio que los autores originales proporcionan o cambiar a otro software diferente. Dado que el costo de cambiar de software puede ser prohibitivo, los usuarios se ven obligados a aceptar el servicio del monopolio. En contraste, el código públicamente disponible de Linux y otros productos de fuente abierta permite a muchos profesionales conocer la plataforma y proporcionar el apoyo técnico. Debido a que diferentes empresas compiten entre sí para proporcionar el servicio, la calidad de este mientras los costes del usuario final disminuyen. La fuente abierta puede tener asistencia técnica durante tanto tiempo como dure la demanda y aun cuando la empresa que lo proporciona deje el negocio. También, ocurre que la estructura de apoyo se auto dimensiona, ya que contra más usuarios utilizan fuente abierta, más son los usuarios que aprenden y pueden a su vez proporcionar apoyo a otros. El tradicional compañerismo y apoyo entre usuarios de productos de fuente abierta asegura que existirá una base extensa de personal de mantenimiento familiarizado con el paquete.

El soporte para Linux está disponible en Internet tanto de forma gratuita como a través de compañías que proporcionan consultoría y servicio técnico. En la tabla siguiente se incluyen ejemplos de recursos gratuitos por proveedor.⁴³

⁴¹ Al Gillen y Dan Kusnetzky, "Linux de un vistazo: comprensión del modelo de mercado de Linux," IDC, febrero de 2.000.

⁴² G. Weiss: "El impacto en la competencia del anuncio de IBM sobre Linux", GartnerGroup, 2 de agosto de 2.000

⁴³ Al Gillen y Dan Kusnetzky, "Linux de un vistazo: comprensión del modelo de mercado de Linux," IDC, febrero de 2.000.

Tabla 3. Recursos gratuitos de apoyo para Linux por distribuidor

Vendor	URL
LinuxHelp Online	www.linuxhelp.org
Linux Documentation Project	www.linuxdoc.org
Linux Support Services	free.linux-support.net
Red Hat Support Links	www.redhat.com/support/docs/tips/urls/urls.html
News groups (multiple)	comp.os.linux
News groups (multiple)	alt.os.linux

Fuente: Al Gillen y Dan Kusnetzky, "Linux de un vistazo: comprensión del modelo de mercado de Linux," IDC, febrero de 2.000.

Red Hat, Caldera, SuSE, TurboLinux, y Linuxcare ofrecen servicio técnico a nivel internacional, aunque algunos son más fuertes en unas regiones que otros. Estas empresas, sin embargo, pueden ofrecer preferentemente apoyo a su propia distribución de Linux.

Los proveedores de hardware, como Compaq, Dell, Hewlett-Packard, IBM, y SGI, ofrecen asistencia primaria y secundaria para Linux en sus equipos de sobremesa y servidores. Además, en algunos casos, tienen un acuerdo de asistencia con uno o más distribuidores de Linux para proporcionar apoyo terciario. Mission Critical Linux, www.missioncriticallinux.com, ha surgido de Compaq para proporcionar apoyo a cualquier cliente. VA Linux proporciona sistemas, hardware, mantenimiento y servicios para ofrecer una ayuda completa a sus clientes con el software de Linux y está particularmente especializada en el comercio electrónico.

3.1.3.4 Afluencia de nuevas empresas

El número de nuevas empresas en el sector se ha doblado entre 1.984 y 1.994. Estas nuevas empresas representan una oportunidad para Linux porque usualmente no pueden permitirse el lujo de mantener un grupo de desarrolladores numeroso en nómina. Los costes directos derivados de usar Linux son frecuentemente menores que los de Microsoft.

3.1.3.5 Soporte colectivo

Linux representa la mejor oportunidad para Unix de derrotar a Microsoft. El sentimiento Anti-Microsoft prevalece, de forma especial en la comunidad de fuente abierta. La fuente abierta, y Linux en el particular, son considerados a menudo como el héroe alternativo (underground). Linux ha sido aclamado como "el asesino de Windows."⁴⁴ Unos 1500 sitios web, incluido el de la campaña de boicot a Microsoft <http://msbc.simplenet.com>, contienen anuncios sobre el movimiento anti-Microsoft. Distribuidores de Unix se agrupan en torno a Linux para aumentar la cuota de mercado de sistemas Unix. Según Forrester Research, Linux continúa aumentando su impulso así como agregando distribuidores, productos y proveedores de servicio técnico.⁴⁵

El Vicepresidente de IBM para Tecnologías de Internet nombrado Linux como uno de los tres mayores hitos que él ha visto en la historia de IBM (los otros dos antes de Linux

⁴⁴ "El futuro de Linux." CNet, 2.000.

⁴⁵ Peter Jordan: "Picando trocitos de UNIX", VARBusiness, 14 de enero de 2.000.

son los PC en 1.989 y el TCP/IP en 1.991); todos ellos fueron rechazados inicialmente por la industria. El apoyo de los mayores vendedores de hardware, incluido IBM, Hewlett-Packard, los Silicon Graphics, Compaq y Dell, ha supuesto un enorme espaldarazo para Linux. Distribuidores y vendedores han lanzado campañas de ventas con notable éxito. Red Hat en particular, ha logrado el reconocimiento de su marca de fábrica a nivel mundial. Hay también nuevo fenómeno respecto a Linux: los profesionales instalan Linux para mantenerse al tanto de las últimas tecnologías emergentes.

3.1.4 Amenazas

3.1.4.1 Riesgo de fragmentación

El riesgo de fragmentación del código base, o código ahorquillado, ocurre cuando múltiples versiones inconsistentes del código base del proyecto evolucionan de forma divergente. Esto puede ocurrir cuando diferentes desarrolladores prueban a crear versiones alternativas en las cuales su código juega un papel más significativa que en el producto base. A veces la fragmentación ocurre por razones de peso, como que el coordinador del proyecto esté desarrollando deficientemente su papel, y a veces ocurre por mal ambiente, como un conflicto de personalidad entre desarrolladores. Las aplicaciones de Unix comerciales (p.e. SCO, Solaris, IRIX, HP-UX) son ejemplos de código fragmentado así como en el software de fuente abierta encontramos las versiones de Unix BSD (p.e. Open BSD, NetBSD, BSDI). Sin embargo, el código del núcleo de Linux se ha fragmentado todavía, y esto puede ser debido a su aceptada estructura de liderazgo, ser abierto a la admisión de nuevos miembros y su potencial de contribución a largo plazo, el licenciamiento GPL que elimina las motivaciones económicas para la fragmentación y la amenaza subsiguiente de fragmentación del equipo de diseñadores. El noventa y nueve por ciento del código distribuido de Linux es el mismo. La pequeña cantidad de código fragmentado en las diferentes distribuciones de Linux es un hecho positivo porque les permite satisfacer las necesidades de segmentos diferentes del mercado. Los usuarios se benefician disponiendo de una distribución Linux que reúna características más acordes a sus necesidades.

El proyecto de Base Estándar Linux (LSB) está trabajando para estandarizar un subconjunto de Linux común a todas las distribuciones. Este mínimo esfuerzo de estandarización están intentando hayar el equilibrio entre hacer el desarrollo de Linux más rígido y la posibilidad de fragmentación en varias versiones totalmente incompatibles. Se espera que el estándar esté disponible en el tercer trimestre del 2.000. Se puede encontrar más información sobre el LSB en <http://www.linuxbase.org>

3.1.4.2 Falta de aplicaciones compatibles

El número de aplicaciones escrito a Linux está creciendo de forma desproporcionada comparada con otros sistemas operativos importantes. Debido a que las aplicaciones son importantes para las compañías, muchas suelen seleccionar primero las aplicaciones que reúnen sus requisitos y luego el sistema operativo que mejor apoye esas aplicaciones. Contra más popular sea la aplicación, más usuarios entrenados se podrán encontrar para esa aplicación. O bien los usuarios de Linux necesitan ser convencidos que este dispone de aplicaciones comparables a otros sistemas operativos (por ejemplo, la suite StarOffice puede ser una alternativa a Microsoft Office) o las aplicaciones requeridas por los usuarios

deben portarse a Linux. Corel ha publicado WordPerfect para Linux. Sun Microsystems, la suite StarOffice para Linux.

3.1.4.3 Necesita un control de versiones

El Control de Versiones puede convertirse en un problema si el sistema requiere integración y desarrollo. El desarrollador debe asegurarse que las versiones a integrar son compatibles.

3.1.5 Otras

3.1.5.1 Importancia para muchos

Esto puede ser considerado una fuerza o una debilidad. La comunidad de Linux crece por su propio impulso. De la misma forma en que aumentan los usuarios interesados en Linux, aumenta la comunidad de Linux. Con una comunidad de Linux más grande aumenta la excelencia técnica. Este atributo es un factor de fortaleza cuando hay un nivel alto de interés, pero es una debilidad cuando el nivel de interés decrece.

3.1.5.2 Personal formado

Esto puede ser considerado una oportunidad y una amenaza. Es una oportunidad porque muchos recién graduados de la universidad son expertos en Linux, después de haberlo usado como una herramienta de aprendizaje en la universidad. Los Investigadores también usan normalmente Linux debido a su amplia disponibilidad. Sin embargo, es una debilidad cuando la plantilla actual de una organización no conoce Linux.

3.1.5.3 Competición

La competición puede ser una oportunidad y una amenaza. Es una oportunidad durante el el proceso de adquisición porque Linux es potencialmente otra vía para mejorar a buen precio el apoyo de las tecnologías comerciales de uso público (COTS). Sin embargo, es una amenaza porque existe el riesgo de compañías que han desarrollado estrategias enfocadas específicamente contra Linux. El éxito de Linux lo ha convertido en un competidor en el mercado. Microsoft ha desarrollado un análisis competitivo formal y una estrategia contra Linux y el movimiento de la fuente abierta. Esto ha quedado demostrado en el memorando interno de Microsoft sobre estrategia referente a la comunidad de Linux y conocido como el documento Halloween. El movimiento de software de fuente abierta tuvo acceso al documento Halloween y como consecuencia lo publicó en el URL siguiente: <http://opensource.org/halloween>

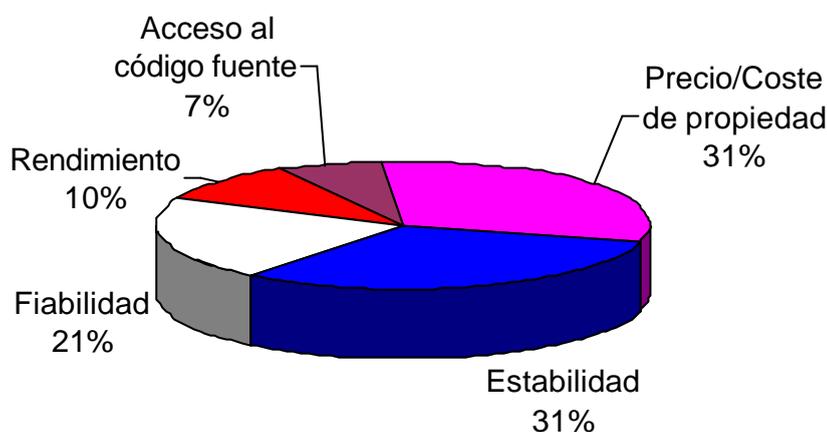
Existen varios competidores de Linux en el mercado. Microsoft Windows NT y el recientemente presentado Microsoft Windows 2.000 ocupan una elevada porción del mercado. DOS es más pequeño y puede usarse para sistemas empotrados. LynuxWorks, anteriormente conocido como Lince, ofrece un sistema propietario tipo Unix con elevadas capacidades para aplicaciones en tiempo real. LynuxWorks está haciendo su software compatible con Linux. BeOS es un sistema propietario con una actitud amistosa hacia el movimiento de software de fuente abierta. Aunque su objetivo es el mercado multimedia, BeOS está ganando notoriedad y es más fácil encontrar personas a las que les gusta. Sun Solaris es considerado a menudo la plataforma de hardware de Unix típica de los servidores

Internet. Sin embargo, Sun ha recibido alguna crítica recientemente de la comunidad de software de fuente abierta por publicar bajo licencia código fuente público y para su propio provecho en las características de compatibilidad que permiten a las aplicaciones de Linux correr en máquinas Solaris.

3.2 Análisis de los factores estratégicos

3.2.1 Viabilidad en el mercado

La mayor parte de los potenciales usuarios se han interesado en Linux por su precio o coste de propiedad y estabilidad. Un reciente sondeo de Datapro encontró que el 31% de los encuestados eligió Linux por estos atributos. Otros encuestados citaron la confianza (21%), nivel de actuación (10%) y acceso al código fuente (7%) como motivaciones por su interés por Linux. Estos resultados se muestran en el siguiente gráfico.⁴⁶

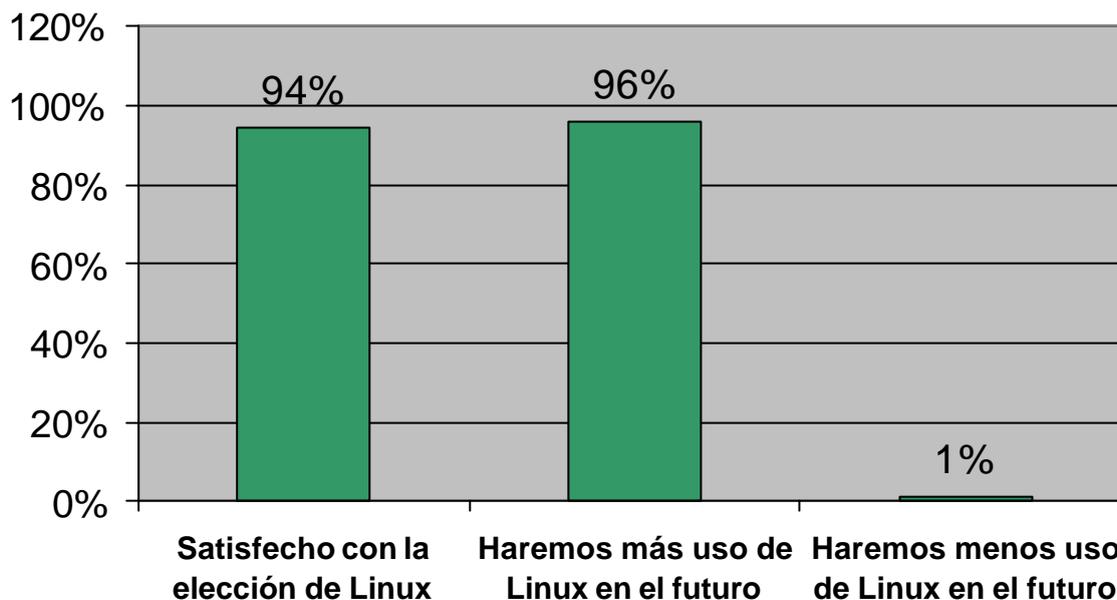


Fuente: Datapro, febrero de 1.999. Hay que tener en cuenta que la naturaleza de las respuesta a través de cuestionarios emplazados en una página web, hace de esta una muestra autoseleccionada; por lo tanto no representan un muestra científica. Un gran número de encuestados pertenecen a pequeñas organizaciones o a los sectores educativo, industrial y de desarrollo de software.

Ilustración 5. Motivación del interés por Linux

⁴⁶ Datapro, febrero de 1.999. Hay que tener en cuenta que la naturaleza de las respuesta a través de cuestionarios emplazados en una página web, hace de esta una muestra autoseleccionada; por lo tanto no representan un muestra científica. Un gran número de encuestados pertenecen a pequeñas organizaciones o a los sectores educativo, industrial y de desarrollo de software.

En este mismo sondeo, a través de su página web, Datapro valoró la satisfacción de los usuarios de Linux. 94% de los encuestados consideran que Linux es la elección correcta. Cuando el usuario es preguntado sobre si incrementará o reducirá su uso de Linux en el futuro, 96% dijo que lo incrementará y sólo 1% que lo reducirá. Estos resultados se muestran en el siguiente gráfico.⁴⁷



Fuente: Datapro, febrero de 1.999. Hay que tener en cuenta que la naturaleza de las respuesta a través de cuestionarios emplazados en una página web, hace de esta una muestra autoseleccionada; por lo tanto no representan un muestra científica. Un gran número de encuestados pertenecen a pequeñas organizaciones o a los sectores educativo, industrial y de desarrollo de software.

Ilustración 6. Satisfacción con Linux

⁴⁷ Datapro, febrero 1.999. Hay que tener en cuenta que la naturaleza de las respuestas a través de cuestionarios emplazados en una página web, hace de esta una muestra auto-seleccionada; por lo tanto, los resultados no representan una muestra científica. Un gran número de encuestados pertenecen a pequeñas organizaciones o a los sectores educativo, industrial y de desarrollo de software.

Las empresas que son reticentes a convertir a Linux sus sistemas, citan compatibilidad con sistemas ya existentes como su preocupación principal, seguida por falta de rápido desarrollo de herramientas y apoyo técnico.⁴⁸

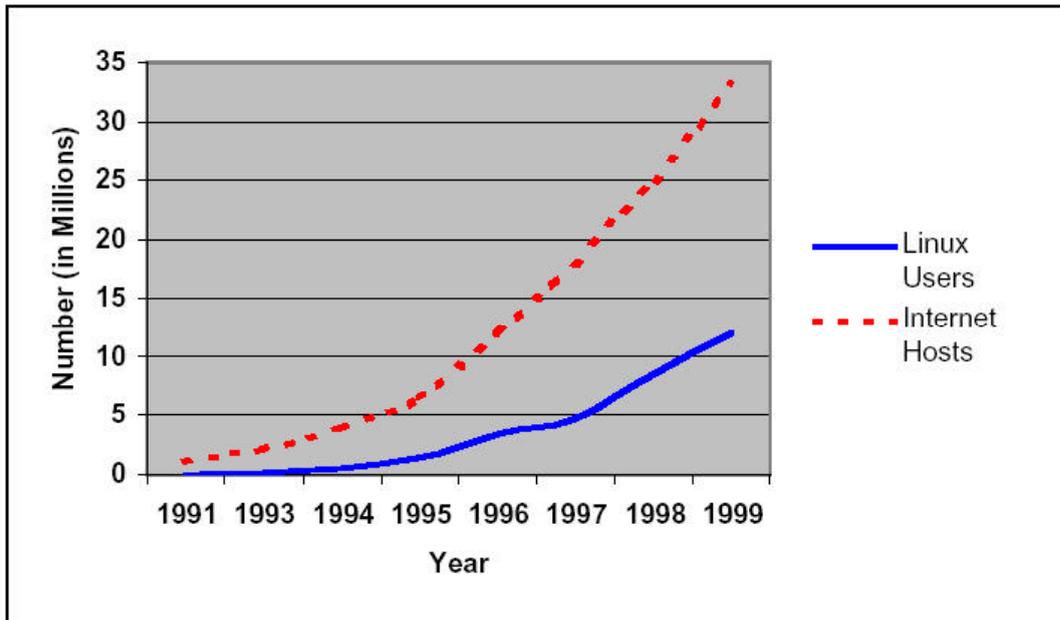
La cuota de mercado exacta de Linux es difícil de calcular debido a que: hay instalaciones desde anónimos sitios FTP, los programas comerciales Linux pueden ser utilizados para su instalación en múltiples máquinas, hay una gran probabilidad de doble contabilidad ya que salen nuevas versiones con frecuencia, y no hay distribuciones separadas cliente y servidor. Es más, estimaciones históricas del mercado de Linux pueden diferir dependiendo de la metodología usada para obtenerlas. Está generalmente asumido que hay entre 4 y 27 millones de usuarios Linux.⁴⁹

El equipo de investigación comparó el número de usuarios Linux con el número de servidores de Internet en los últimos nueve años. El equipo hizo su propia estimación de los usuarios de Linux basándose en diversas fuentes, incluyendo GartnerGroup, IDC y Red Hat. Estas estimaciones muestran que el número de usuarios de Linux, a nivel mundial, creció desde 1 (Linus Torvalds) en 1.991 hasta unos 12 millones en 1.999. El siguiente cuadro compara, a nivel mundial, el número de usuarios Linux con el número de servidores de Internet. El gráfico muestra que mientras Internet se expande, el número de usuarios de Linux también. Esto puede ser atribuido al efecto de Internet en el proceso de desarrollo del código abierto; al expandirse Internet, el número y productividad de los equipos de desarrollo de código abierto aumenta.⁵⁰

⁴⁸ Paul Harmon, "Arquitectura y Linux," Consorcio Cutter, 9 de febrero de 2.000.

⁴⁹ Eric Raymond, <http://www.opensource.org>, no consta fecha.

⁵⁰ Estimaciones de Linux obtenidas a partir de los estudios de mercado de GartnerGroup, IDC y Red Hat. Estimaciones de Internet basadas en la investigación de Bruce L. Egan, 1.996. Datos basados en estimaciones a fin de año.

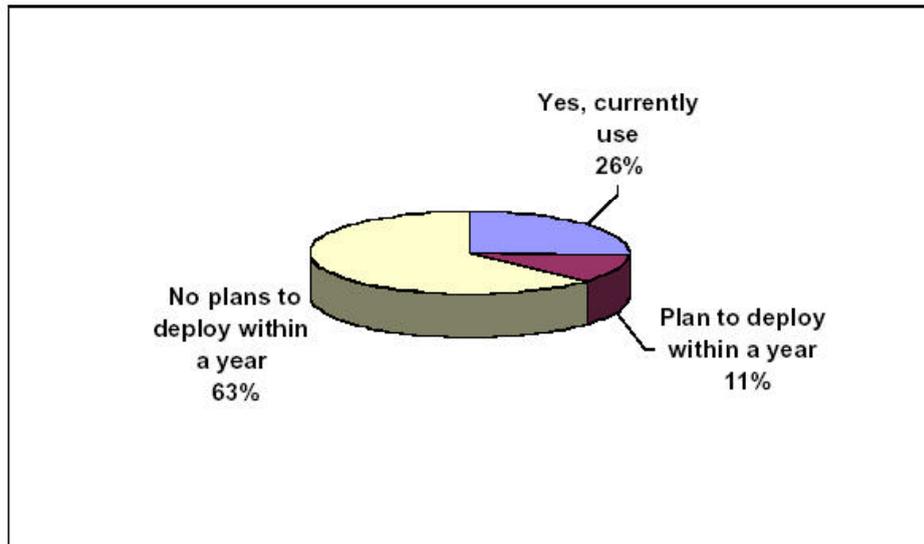


Fuente: Estimaciones de Linux obtenidas a partir de los estudios de mercado de GartnerGroup, IDC y Red Hat. Estimaciones de Internet basadas en la investigación de Bruce L. Egan, 1.996. Datos basados en estimaciones a fin de año.

Ilustración 7. Éxito mundial de Linux en el mercado

Alrededor de un 26% de las empresas actualmente usan el sistema operativo Linux, otro 11% planean instalarlo en un año, y el 60% no tienen previsto instalarlo en el plazo de un año. Estos resultados fueron obtenidos de un sondeo de InformationWeek sobre 300 directivos de TI, y los resultados se muestran en el gráfico que sigue.⁵¹

⁵¹ Aaron Ricadela, "Linux vive," InformationWeek, 24 de enero de 2.000.



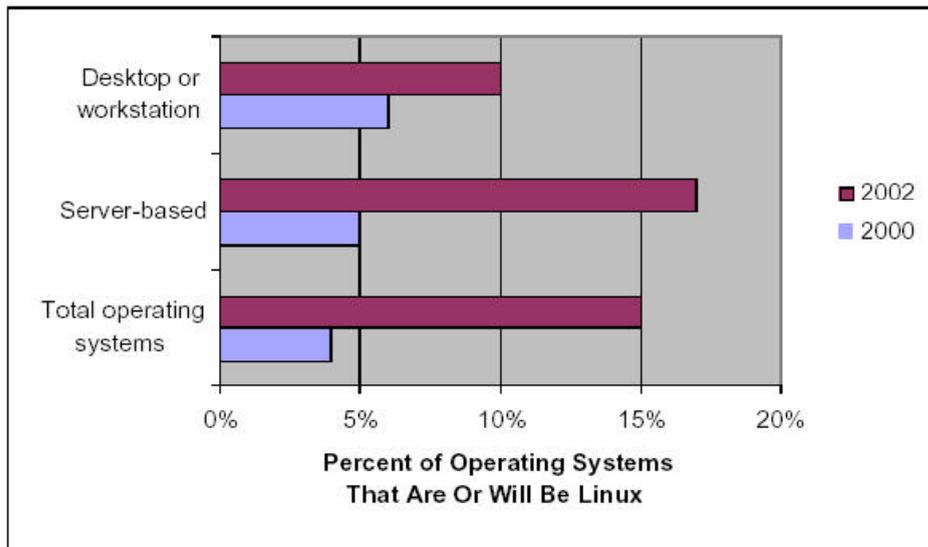
Fuente: Aaron Ricadela, "Linux vive," InformationWeek, 24 de enero de 2.000.

Ilustración 8. Porcentaje de empresas que usan el sistema operativo Linux.

Dos tercios de las empresas que usan la plataforma linux, llevan haciéndolo desde hace un año o menos, de acuerdo con el sondeo de InformationWeek.⁵² Aunque las instalaciones de Linux se están ampliando, no son profundas. Los encuestados dicen que sólo el 4% del total de sus entornos de sistema operativo consiste en Linux. Ese número se espera que se incremente en un 15% para 2.002, pero continuará siendo inferior que Microsoft windows. El siguiente gráfico compara el porcentaje de sistemas operativos Linux en el año 2.000 con aquellos que se esperan para el 2.002.⁵³

⁵² Aaron Ricadela, "Linux vive," InformationWeek, 24 de enero de 2.000.

⁵³ Aaron Ricadela, "Linux vive," InformationWeek, 24 de enero de 2.000.

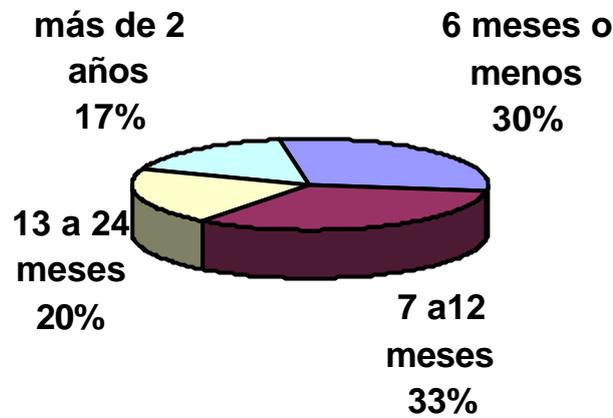


Fuente: Aaron Ricadela, "Linux vive," InformationWeek, 24 de enero de 2.000.

Ilustración 9. Porcentaje de sistemas operativos que son o serán Linux

De aquellas empresas con Linux, la mayoría lo usan sólo recientemente. Alrededor de un 63% han estado usando Linux desde hace un año o menos. El siguiente gráfico muestra por cuanto tiempo han estado utilizando Linux las empresas.⁵⁴

⁵⁴ Aaron Ricadela, "Linux vive," InformationWeek, 24 de enero de 2.000.

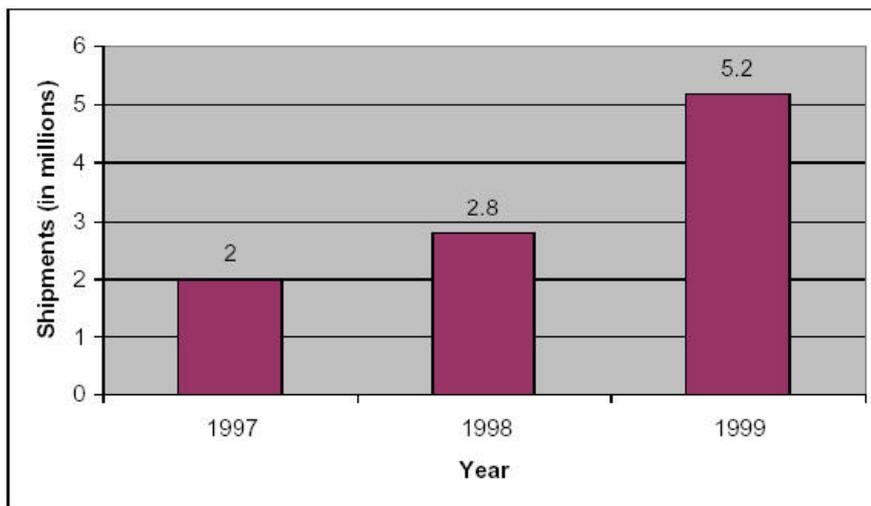


Fuente: Aaron Ricadela, "Linux vive," InformationWeek, 24 de enero de 2.000.

Ilustración 10. ¿Cuánto tiempo han estado las empresas utilizando Linux?

A nivel mundial, nuevos envíos para aplicaciones cliente-servidor han estado aumentando a un tipo creciente durante los pasados años. IDC determinó que los envíos se incrementaron en un 40% en 1.998 (hasta 2.8 millones) comparados con el año anterior. En 1.999, los envíos se incrementaron en casi el 86% (a 5.2 millones). El siguiente cuadro muestra el creciente número de nuevos envíos Linux para aplicaciones cliente-servidor, a nivel mundial.⁵⁵

⁵⁵ IDC, 2.000.



Fuente: IDC, 2.000.

Ilustración 11. Nuevos envíos a nivel de Linux (Cliente y Servidor)

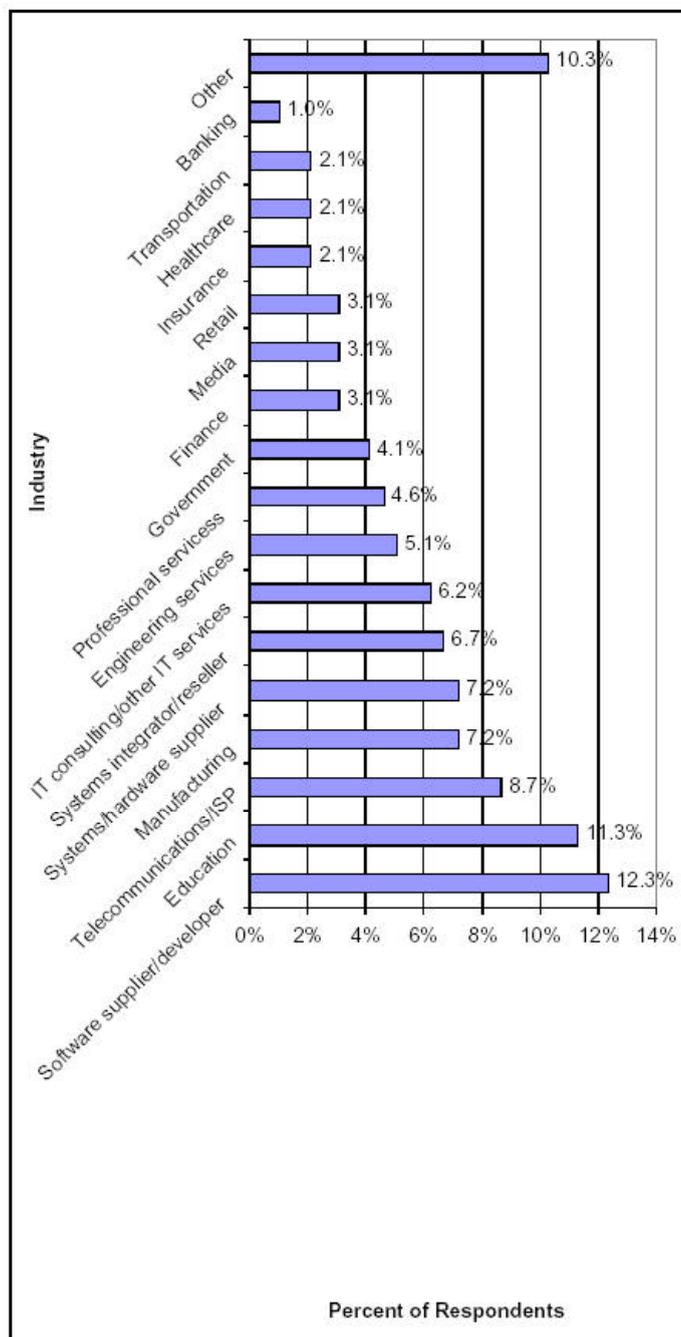
3.2.2 Segmentos del mercado

Dadas las raíces de Linux en organizaciones dedicadas a la investigación y la educación, la mayor concentración de usuarios de Linux se encuentra en la industria de la TI y los ordenadores (especialmente ISP y desarrolladores de software) e instituciones educativas. Los ISP son una base nuclear de usuarios de Linux. Más del 26 por ciento de los ISP utilizan Linux, según una encuesta de Infoheads.⁵⁶ Los ISP deben soportar márgenes de beneficio muy pequeños, y la gratuidad de Linux y el amplio soporte para hardware son atractivos. Si algo se estropea, los ISP deben solucionar el problema en seguida; los mayores ISP tienen personal con experiencia para depurar problemas con el código o instalar rápidamente los parches disponibles. Es más, los administradores de sistemas de los ISP normalmente tienen un sólidos conocimientos de Unix. Las capacidades de gestión remota, la fiabilidad y la escalabilidad son también características clave para los ISP.

A menudo los usuarios con competencia técnica prefieren Linux, y a las organizaciones dedicadas a la investigación y a la educación les interesa el bajo coste de adquisición de Linux y la facilidad que ofrece para apañar el mismo sin la necesidad de negociar ni de acuerdos contractuales con el propietario del código fuente. Sin embargo, los servidores Linux también se encuentran en un amplio rango de industrias y en compañías de todos los tamaños. Los usuarios de Linux probablemente seguirán creciendo según el software gane más atractivo comercial. La siguiente gráfica muestra la alta

⁵⁶ "Uso de Linux en los negocios," <http://www.linuxbusiness.com/en/case.html>, no consta fecha.

proporción del número de usuarios de servidores Linux en los sectores dedicados a la investigación y a la educación.⁵⁷

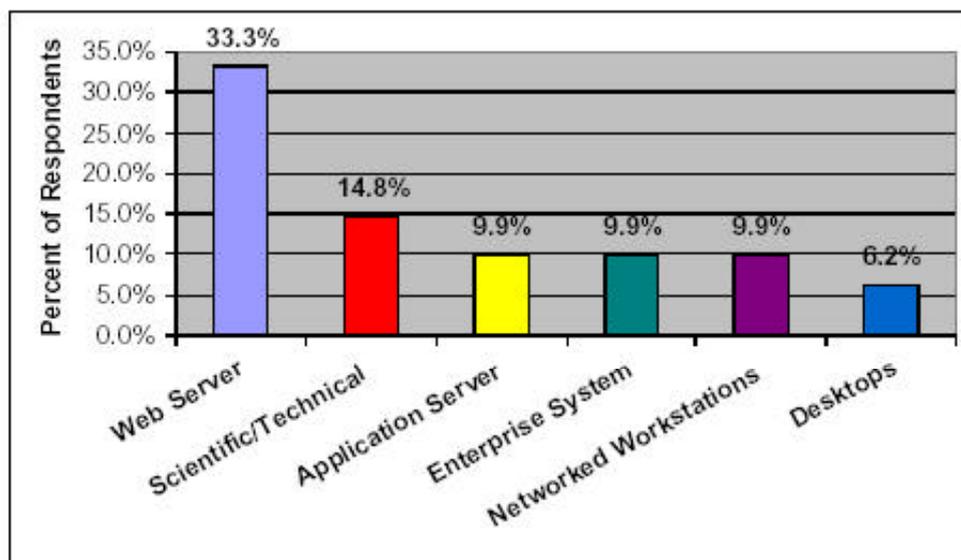


Fuente: IDC, 2.000.

Ilustración 12. Sitios con servidores linux organizados por sector de actividad, 1.999

⁵⁷ IDC, 2.000.

Según una reciente encuesta de Datapro, Linux se utiliza mayormente como servidor web (33.3 por ciento) y para aplicaciones científicas y técnicas (14.8 por ciento). Se utiliza menos como servidor de aplicaciones (9.9 por ciento), sistemas de empresa (9.9 por ciento) y estaciones de trabajo en red (9.9 por ciento) y aún menos para aplicaciones de escritorio (6.2 por ciento). La siguiente figura ilustra estas contestaciones.⁵⁸

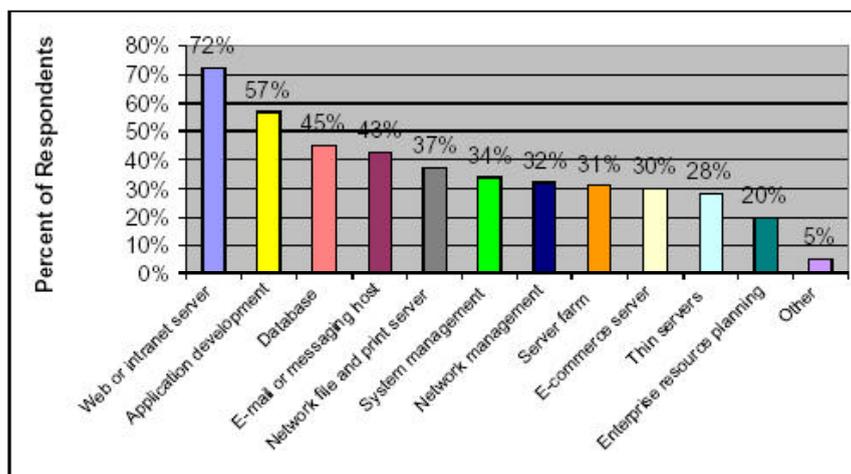


Fuente: Datapro, febrero 1.999. Los datos fueron recabados mediante un estudio de opinión de usuarios emplazado en una página web en la página de inicio de Datapro. La naturaleza de las respuesta a través de cuestionarios emplazados en una página web, hace de esta una muestra auto-seleccionada. Por lo tanto, los resultados no representan una muestra científica. Un gran número de encuestados pertenecen a pequeñas organizaciones o a los sectores educativo, industrial y de desarrollo de software. En la encuesta, un total de 1841 individuos dijeron que sus organizaciones utilizaban Linux.

Ilustración 13. Uso de Linux (encuesta de Datapro)

⁵⁸ Datapro, febrero 1.999. Los datos fueron recabados mediante un estudio de opinión de usuarios emplazado en una página web en la página de inicio de Datapro. La naturaleza de las respuesta a través de cuestionarios emplazados en una página web, hace de esta una muestra auto-seleccionada. Por lo tanto, los resultados no representan una muestra científica. Un gran número de encuestados pertenecen a pequeñas organizaciones o a los sectores educativo, industrial y de desarrollo de software. En la encuesta, un total de 1841 individuos dijeron que sus organizaciones utilizaban Linux.

Una encuesta realizada por InformationWeek a 300 responsables de TI concluyó de manera similar que de las aplicaciones basadas en servidor, Linux es utilizado principalmente para servidor web o en la Intranet. Esta encuesta permitía respuestas múltiples, y los resultados se muestran en la siguiente figura.⁵⁹



Fuente: Aaron Ricadela, "Linux vive," InformationWeek, 24 de enero de 2.000.

Ilustración 14. Uso de Linux en aplicaciones basadas en servidor (encuesta de Information Week)

3.2.2.1 Mercado de servidores

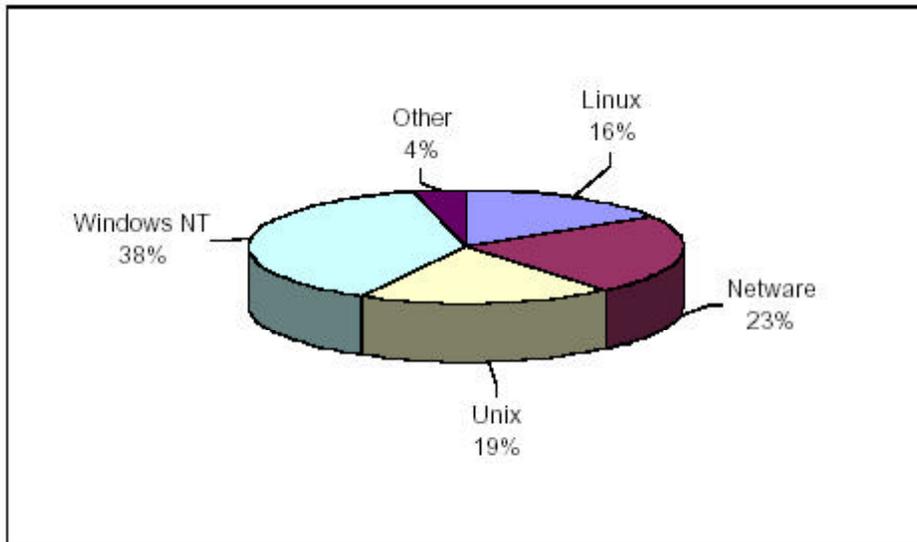
Se espera que la mayoría de las instalaciones de Linux sea en el entorno de los servidores. El mercado de los servidores es particularmente el fuerte de Linux por dos razones. Primero, el mercado de los servidores, especialmente de gama alta, está ya familiarizado con Unix y se siente a gusto con él. Segundo, los administradores de los servidores de gama alta tienden también a ser desarrolladores y se sienten a gusto con las cuestiones técnicas. Linux puede utilizarse para un rango bastante amplio de propósitos: servidores web, FTP, intermediarios, de correo o DNS; cortafuegos; o encaminadores TCP/IP. Un único servidor también puede realizar todas las anteriores funciones.

Según un informe desarrollado por IDC, Linux fue el entorno de operaciones para servidor de mayor crecimiento en 1.998, con un incremento de más del 190 por ciento ese año y capturando más del 15.8 por ciento de beneficios derivados de distribuciones en el segmento de sistemas operativos de servidor valorado en 4.4 millones.⁶⁰ Linux ha incrementado su cuota de mercado en el segmento de SO de servidor desde el 16 por ciento en 1.998 hasta el 25 por ciento en 1.999. Desde 1.998, Unix y NetWare han perdido terreno con respecto a Linux. La cuota de mercado de Windows NT todavía no ha sufrido impacto por el crecimiento de Linux. Los siguientes diagramas muestran las cuotas de mercado de

⁵⁹ Aaron Ricadela, "Linux vive," InformationWeek, 24 de enero de 2.000.

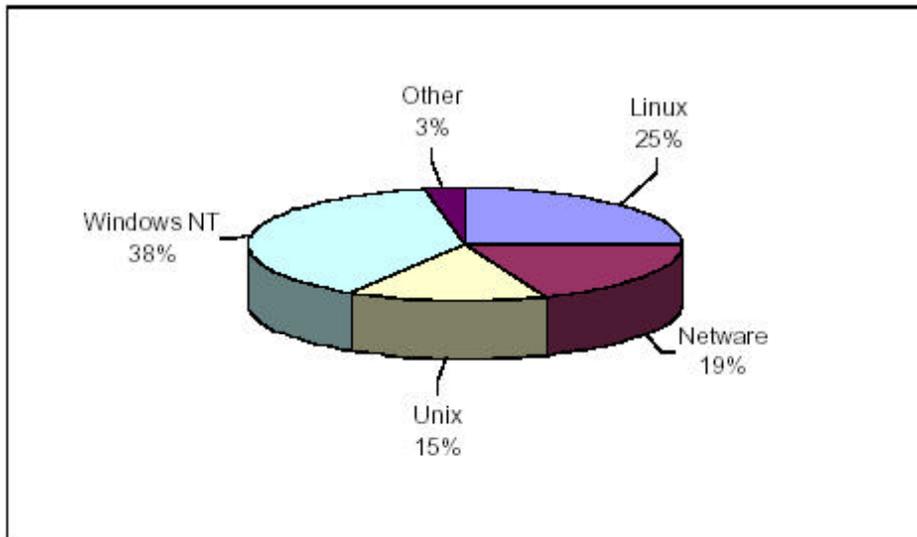
⁶⁰ Red Hat, <http://www.redhat.com>, no consta fecha.

los SO de servidor para Linux, NetWare, Unix, Windows NT y otros sistemas tanto en 1.998 como en 1.999.⁶¹



Fuente: "El futuro de Linux," CNet, 2.000 en el que se cita un informe de IDC

Ilustración 15. Cuotas de mercado de SO de servidor en 1.998.

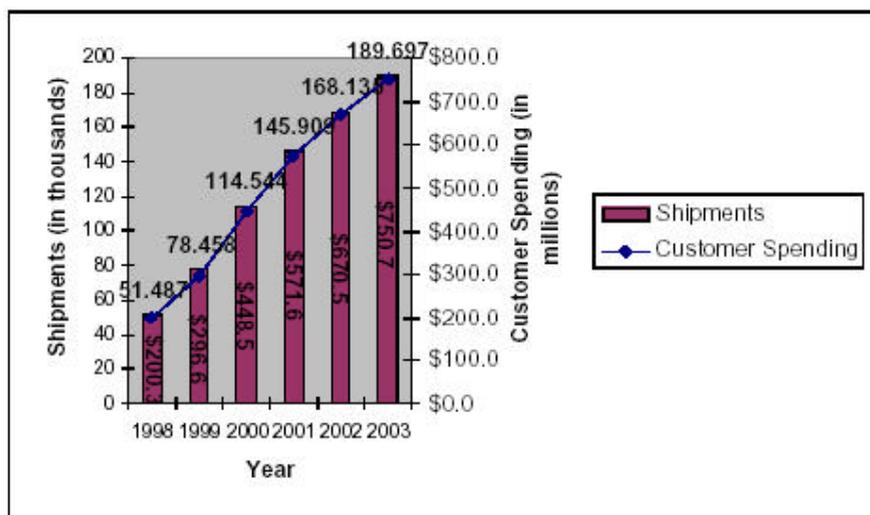


Fuente: "El futuro de Linux," CNet, 2.000 en el que se cita un informe de IDC

Ilustración 16. Cuotas de mercado de SO de servidor en 1.999.

⁶¹ "El futuro de Linux," Cnet, 2.000 en el que se cita un informe de IDC.

Según las estimaciones de IDC, los clientes de EEUU gastaron cerca de 300 millones de dólares en servidores Linux en 1.999. El beneficio iguala a la distribución de más de 75000 servidores que se vendió en EEUU en 1.999. En estas cifras se incluyen los servidores instalados por los fabricantes y los redistribuidores y los sistemas de construcción propia. IDC calcula que los beneficios y distribuciones representan sólo algo más de 5 por ciento del mercado total disponible para la distribución de servidores en el nivel de entrada (aquellos servidores que cuestan menos de 100.000 dólares). La siguiente gráfica muestra el número de distribuciones tanto desde una perspectiva histórica como de lo que se espera en el futuro en cuanto al gasto de los clientes de EEUU desde 1.998 hasta el 2.003.⁶²



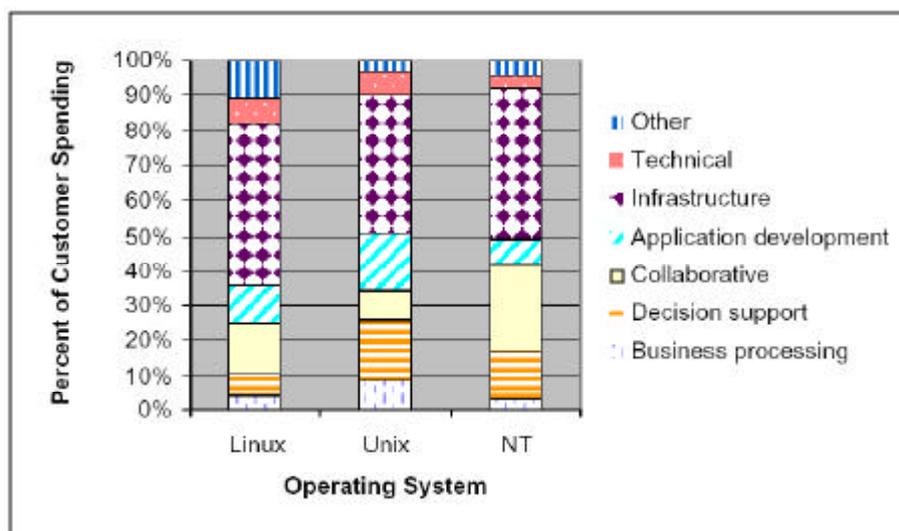
Fuente: Michelle Bailey, Vernon Turner, Jean Bozman y Janet Waxman, “Servidores Linux: qué hay de exageración, qué de realidad” IDC, marzo 2.000.

Ilustración 17. Distribución de servidores Linux y gasto de los clientes, 1.998-2.003.

En su reciente encuesta, IDC definió una topología de carga con cargas mutuamente exclusivas. Todas las cargas de un servidor totalizaban el 100 por ciento de la funcionalidad total disponible del servidor. La topología fue definida de la siguiente manera: proceso de negocio (planificación de desarrollo empresarial, procesamiento transaccional en línea y aplicaciones batch); el sistema de soporte a la decisión (almacén de datos, suministros de datos, análisis de datos y minería de datos); informática colaborativa (utilización del correo-e y del trabajo en grupo); desarrollo de aplicaciones; infraestructura (archivo/impresión, red, gestión de sistemas y sistemas tipo proxy/caché o sistemas de seguridad); técnicos (proceso en tiempo real, proceso intensivo numérico y aplicaciones científicas); y otros (publicación/información/educación, video/audio bajo demanda y aplicaciones de uso personal). Los resultados de la encuesta mostraron que los servidores Linux se compran principalmente para dar soporte a las aplicaciones de infraestructura. Más del 40 por ciento de todo el gasto en servidores Linux se utilizan para aplicaciones de archivo e impresión, funciones de red, sistemas de gestión y servicios proxy, cachés y de

⁶² Michelle Bailey, Vernon Turner, Jean Bozman y Janet Waxman, “Servidores Linux: qué hay de exageración, qué de realidad” IDC, marzo 2.000.

seguridad. Otros usos importantes de Linux incluyen las aplicaciones de colaboración como el correo (14.2 por ciento) y desarrollo de aplicaciones (10.9 por ciento). El diagrama siguiente compara la distribución del gasto según la carga de trabajo de los servidores en EEUU para los sistemas operativos Linux, Unix y NT.⁶³



Fuente: IDC, 2.000.

Ilustración 18. Distribución del gasto en servidores según la carga de trabajo por sistema operativo

El GartnerGroup está de acuerdo en que Linux está muy enfocado en algunos nichos, en la actualidad con más del 80 por ciento de las instalaciones dedicadas a aplicaciones relacionadas con la infraestructura web, aunque está ganando cada vez más aceptación como sistema empresarial.⁶⁴ NetCraft dice que el 29 por ciento de todos los servidores web funcionan sobre Linux, haciendo de Linux el sistema operativo más popular entre los sitios web públicos.⁶⁵

Agrupar microordenadores ofrece la potencia de un supercomputador a bajo precio. Se pueden crear ya sea mediante la agregación de ordenadores dedicados (como Beowulf o Titanic) o de redes existentes durante las horas de baja carga de trabajo.

Los servidores de aplicaciones de propósito específico son aquellos que son diseñados a priori para cubrir las necesidades de aplicaciones de un tipo específico, como ventas por Internet o publicación en la web, y se espera que su importancia crezca. Para los usuarios con objetivos específicos, estas plataformas ofrecen un rendimiento avanzado y excepcional. En el futuro, los servidores de aplicaciones de propósito específico se

⁶³ IDC, 2.000.

⁶⁴ G. Weiss, "Buenas prácticas en la adopción de Linux: un programa de 10 puntos," GartnerGroup, 8 de febrero de 2.000.

⁶⁵ Red Hat, <http://www.redhat.com>, no consta fecha.

utilizarán en nuevos tipos de aplicaciones, que incluyen las especializadas en el comercio electrónico, agrupación de contenidos y personalización.

3.2.2.2 Mercado de sobremesa

Para que Linux tenga éxito en el mercado de la sobremesa, es necesario realizar una inversión significativa en aplicaciones software, facilidad de uso y facilidad de configuración. La mayoría de los usuarios finales no elijen sus sistemas operativos. Típicamente, los usuarios compran sus aplicaciones software basándose en sus necesidades informáticas y elijen entonces un ordenador que pueda ejecutar ese software. La mayoría del software se desarrolla para PCs o Macs, y la mayoría ejecutan Windows o el sistema operativo de Mac. Actualmente, no hay mucho software disponible para Linux. La comunidad de Linux está trabajando para aumentar la cantidad de software compatible con Linux. El Proyecto Wine es un esfuerzo para implantar las interfaces de programación de aplicaciones (API) de Microsoft Windows 95 y NT como código abierto y posibilitar su uso en Linux. Algunas compañías como Corel están trabajando para hacer que aplicaciones como CorelDraw y WordPerfect para Windows puedan funcionar en Linux.

Otra barrera para el éxito de Linux en escritorio es que no es tan amigable para el usuario como Windows. Se están intentando algunas iniciativas para mejorar la amigabilidad para el usuario de Linux. Algunas compañías como Corel y Caldera están mejorando la facilidad de instalación de sus distribuciones Linux. Sin embargo a veces es necesario escribir comandos y la mayoría de los usuarios prefieren el estilo de apuntar y pulsar de los sistemas operativos Windows y Mac. Los entornos de escritorio GNOME, KDE y CDE vienen incluidos con las aplicaciones básicas de productividad. Hewlett-Packard y Sun Microsystems han anunciado que empezarán a utilizar el escritorio de GNOME como su interfaz de escritorio por defecto para Unix. Corel y StarOffice ofrecen conjuntos de productos completos de oficina con cierto grado de compatibilidad de ficheros con el Office de Microsoft. También son necesarios un paquete de gestión de ventanas para el escritorio dominante y más aplicaciones. La siguiente tabla lista los conjuntos de aplicaciones de escritorio disponibles para Linux por fabricante y por producto.⁶⁶

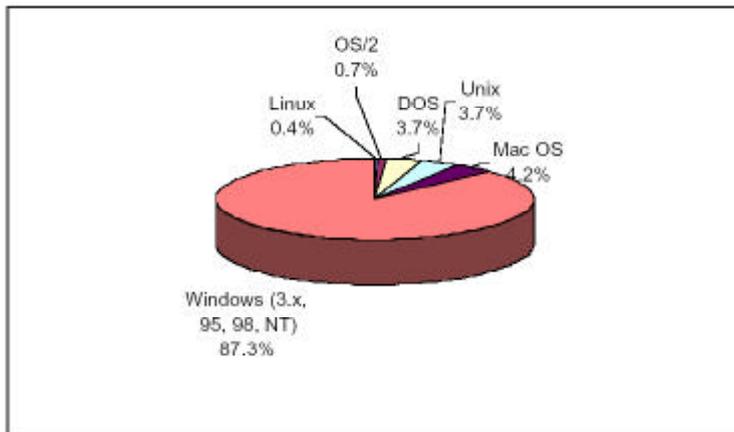
Tabla 4. Conjuntos de aplicaciones de escritorio para Linux por fabricante y por producto.

Fabricante	Producto	URL del fabricante y del producto
Applixware	Applixware	www.applix.com/applixware
Corel	WordPerfect	www.corel.com
Sun Microsystems	StarOffice	www.sun.com/staroffice
SuSE	Office Suite	www.suse.com

Fuente: Al Gillen y Dan Kusnetzky, "Panorama general de Linux: comprender el modelo de mercado de Linux," IDC, febrero de 2.000.

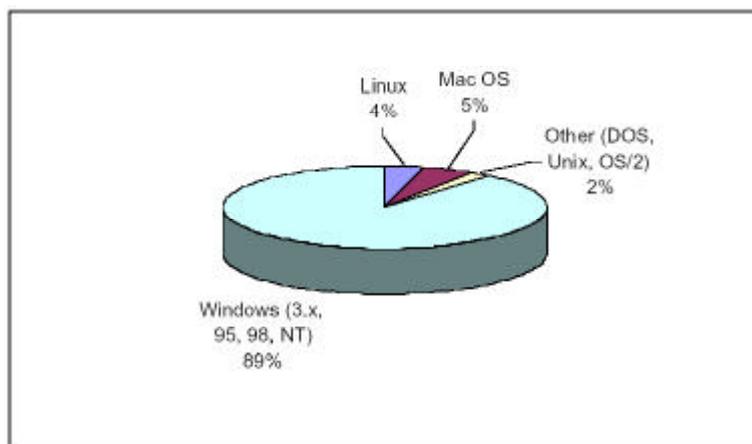
⁶⁶ Al Gillen y Dan Kusnetzky, "Panorama general de Linux: comprender el modelo de mercado de Linux," IDC, febrero de 2.000.

Los siguientes diagramas de tarta ilustran las cuotas de mercado de sistema operativo de cliente en 1.998 y 1.999.⁶⁷



Fuente: "El futuro de Linux," Cnet, 2.000, en el que se cita un informe de IDC.

Ilustración 19. Cuotas de mercado de los sistemas operativos de cliente en 1.998.



Fuente: "El futuro de Linux," Cnet, 2.000, en el que se cita un informe de IDC.

Ilustración 20. Cuotas de mercado de los sistemas operativos de cliente en 1.999.

⁶⁷ "El futuro de Linux," Cnet, 2.000, en el que se cita un informe de IDC.

3.2.2.3 Dispositivos empotrados

Un dispositivo empotrado es una pieza de material informático basada en un microprocesador, normalmente emplazada en una única tarjeta de circuito impreso, que ha sido construida para ejecutar una aplicación software específica. El término *empotrado* hace referencia al hecho de que estos dispositivos se utilizaban originalmente como bloques constitutivos de sistemas mayores. Un servidor ligero es un ordenador que contiene sólo el hardware y software necesario para dar soporte a una función concreta y que los usuarios pueden compartir en una red, como por ejemplo acceso a archivos que están en un dispositivo de almacenamiento, acceso a unidades de CD-ROM, impresión o acceso a Internet. El sistema operativo Linux ofrece muchas ventajas para los servidores empotrados y ligeros. Se puede portar a muchas unidades centrales de proceso (CPUs) y plataformas hardware, es estable y escalable para un amplio rango de capacidades y es fácil de usar para los desarrolladores. El software Linux puede ser reconfigurado dinámicamente sin necesidad de rearrancar el sistema. Puede aislar fallos y procesos. Los procesos pueden cargar y descargar módulos del núcleo, manejadores de dispositivos y módulos personalizados basándose en los recursos disponibles y en las necesidades dinámicas de cada aplicación. Las aplicaciones también son modulares y tienen interfaces bien definidas.

Los dispositivos empotrados a veces tienen un requisito de tiempo-real. El tiempo-real es un término relativo, y para algunos sistemas, una respuesta dada en un rango entre cinco y cincuenta milisegundos es suficiente. Estos sistemas de tiempo-real pueden responder a sucesos específicos o bien procesar un flujo de información constante.⁶⁸ Existe una extensión de Linux para el tiempo-real, llamada RTLinux. Se puede encontrar más información sobre RTLinux en www.rtlinux.org.

Existe un gran mercado en expansión para los dispositivos empotrados, que incluye consolas de información y dispositivos móviles. Las aplicaciones comerciales incluyen terminales de punto de venta, discotecas digitales, estereos para los coches, sistemas de tarjeta de crédito para los surtidores de gasolina, equipamiento médico, sistemas accesorios del televisor, asistentes personales digitales (PDAs), lavadoras y cierres de las habitaciones en hoteles. El mercado de los ordenadores empotrados absorbe más del 95 por ciento de todos los microchips que se venden cada año.⁶⁹ Según el GartnerGroup se espera que el mercado de las herramientas para desarrollo de software empotrado crezca a una tasa anual equivalente al 25 por ciento o más a corto plazo.⁷⁰ Se espera que los ordenadores pequeños, ligeros y baratos que utilizan sistemas operativos empotrados ocupe el hueco en el que actualmente se ubican los usuarios que no se encuentran conectados a la Internet, pero que están interesados en acceder a ella. Según Júpiter Communications, el mercado para los usuarios de consolas para Internet puede ser mayor que la base total actual de usuarios de PC. Se ha proyectado que la penetración de consolas de Internet en los hogares alcance los 37.3 millones en el 2.002.⁷¹

⁶⁸ IT-Director.com, <http://www.it-director.com/ts/linux/embedded.html>, no consta fecha.

⁶⁹ Consorcio de Linux Empotrado, <http://www.embedded-linux.org/pressroom.php3#1>, no consta fecha.

⁷⁰ GartnerGroup, 1.998.

⁷¹ Jupiter Communications, "Consolas para Internet Appliances," citado por Be, Inc., <http://www.be.com>, no consta fecha.

Hay varias opciones de software para sistemas empotrados, incluyendo DOS, Microsoft Windows y Linux. Aunque hay más de 100 millones de usuarios en el mundo,⁷² DOS tiene limitaciones muy conocidas para los sistemas empotrados. Microsoft Windows carece de capacidad para tiempo real. En los mercados del escritorio y del servidor, aparecen nuevos dispositivos cada tres meses o menos, y encontrar los manejadores de los dispositivos para los productos COTS puede ser difícil.⁷³ Para dar soporte al hardware elegido por sus clientes, los fabricantes de sistemas operativos tradicionales de tiempo-real normalmente suelen bien cargar hasta 30.000 dólares en tarifas para consultoría y mantener el código del manejador o utilizar manejadores para Linux de la comunidad de la fuente abierta.

Un subconjunto del núcleo de Linux, específicamente el *Embeddable Linux Kernel Subset* (ELKS), puede funcionar en dispositivos empotrados, y el sistema operativo Linux es de tiempo-real. Dado que Linux es escalable y flexible, puede manejar un amplio rango de funciones del sistema empotrado. El usuario puede elegir sólo aquellos componentes de Linux que son necesarios para la aplicación. Linux se hará apropiado para el tiempo-real y más determinista en su rendimiento mediante ciertas estrategias de sustitución del núcleo y por mejoras en el núcleo estándar. El código de los manejadores de dispositivos para Linux aparece de forma regular junto con o antes de la llegada al mercado de los nuevos manejadores.⁷⁴

En los sistemas empotrados los márgenes comerciales son muy bajos. La gratuidad de Linux ayuda a este mercado. Utilizar código abierto para los sistemas empotrados evita las cuotas de licencia de los fabricantes de código cerrado, lo que implica ahorros de coste enormes en la producción de un volumen elevado de sistemas empotrados. Los desarrolladores de sistemas empotrados pueden comprimir Linux y estrujarlo para que se ajuste a las necesidades de sus aplicaciones específicas. Los desarrolladores de todo el mundo pueden mejorar de forma cooperativa el software y arreglar los errores en tiempo-real.

Linux no fue originalmente diseñado para los sistemas empotrados, pero ha sido adoptado por ellos. Linux fue diseñado desde el principio de forma no integrada, como componentes, de forma que es fácil construir cajas que no tengan monitor, teclado, etc. Los desarrolladores de sistemas empotrados tienen la seguridad de que su sistema se puede cambiar y reparar inmediatamente dado que el código de Linux está siendo constantemente mejorado.

Hay un riesgo de fragmentación en el mercado de Linux para sistemas empotrados. Existen más de 100 sistemas operativos comerciales de tiempo-real (RTOSs) porque hay una gran variedad de dispositivos empotrados (desde teléfonos celulares a frigoríficos).⁷⁵

⁷² "PC DOS," IBM, <http://www-3.ibm.com/software/os/dos/>, no consta fecha.

⁷³ Jim Ready y Bill Weinberg, "Posicionando Linux para aplicaciones en sistemas empotrados," LinuxDevices.com, no consta fecha.

⁷⁴ Jim Ready y Bill Weinberg, "Posicionando Linux para aplicaciones en sistemas empotrados," LinuxDevices.com, no consta fecha.

⁷⁵ John Matthews "¿Conseguirá Cygnus salvar al Linux para sistemas Empotrados de la fragmentación?" *Linux Today*, 30 de septiembre de 1.999, wysiwyg://35/http://linuxtoday.com/stories/10601.html.

Para adaptar Linux a estos dispositivos y utilizar la menor cantidad de hardware posible, se extraen diferentes partes del código básico común de GNU/Linux y se añaden diferentes extensiones específicas a los dispositivos que optimicen el rendimiento. En un intento por evitar la fragmentación, Cygnus Solutions ha desarrollado el interfaz de programación de aplicaciones EL/IX, un “nivel de compatibilidad”, o nivel de abstracción del hardware, para que el Linux estándar pueda manejar diferentes dispositivos.

El consorcio de Linux para sistemas empuotrados (ELC) es una asociación de comercio sin ánimo de lucro independiente de los fabricantes y que se dedica a promover y hacer avanzar a Linux como sistema operativo en la comunidad de los sistemas empuotrados. Se pueden encontrar más información sobre ELC en www.embedded-linux.org.

Los principales actores en el mercado de los sistemas empuotrados, como Motorola e Intel, están dando soporte al uso de Linux. La lista de fabricantes que ofrecen Linux para sistemas empuotrados incluye a Lineo/Caldera (Emdedix), Monta Vista (Hard Hat Linux), Cygnus (eCos) y Be (BeOS, Stinger). LynuxWorks ofrece productos y servicios para desarrollar software empuotrado en el sistema operativo Linux. Planea ofrecer programas especializados de formación y el primer programa de soporte a largo plazo de la industria para sistemas empuotrados. LynuxWorks abraza el modelo de fuente abierta de Linux para ofrecer a sus clientes una solución optimizada para las aplicaciones empuotradas. Se puede encontrar más información sobre Linux empuotrado en www.linuxdevices.com.

3.3 Análisis de viabilidad de la oportunidad de negocio

Cuando se comparan los costes y beneficios a largo plazo de la utilización del software de fuente abierta y el mantenimiento de los COTS tradicionales, el vencedor varía según el uso específico y las circunstancias. Los costes y beneficios están influidos por el entorno de la plataforma, las necesidades operativas y los objetivos de la misión. En sus entornos específicos y con sus parámetros específicos, los usuarios deberían los programas más efectivos, aquellos que optimizan los beneficios y minimizan los costes. La literatura cita varios datos cuantitativos y opiniones cualitativas sobre el coste total de propiedad y los beneficios del OSS cuando se comparan con las alternativas tradicionales de COTS.

A continuación se dan algunas opiniones:

- El GartnerGroup estima que el modelo de negocio de la fuente abierta en la industria del Unix dará un resultado de un ahorro del 20 por ciento en el presupuesto de IT para servidores comparado con el modelo de licencias tradicional a partir de 2.001.⁷⁶
- El GartnerGroup espera que la ventaja de Linux debida al coste total de propiedad desaparezca. Los fabricantes de plataformas Unix como IBM y Sun ya ofrecen sus sistemas operativos Unix virtualmente gratis.⁷⁷

⁷⁶ G. Weiss, “Cómo afectará el movimiento de la fuente abierta a los usuarios,” GartnerGroup, 26 de enero de 1.999.

⁷⁷ G. Weiss, “Predicción del GartnerGroup para los sistemas operativos de servidor,” GartnerGroup, 26 de marzo de 2.000.

- Según Computerworld, el coste total de utilizar Linux es aproximadamente el mismo que con NT, Unix o cualquier otra cosa.⁷⁸
- Una compañía gestionada de forma independiente con una rotación de personal alta puede experimentar que el coste de propiedad de un ordenador es casi un 50 por ciento más alto que el de las empresas equivalentes más estables debido al coste adicional de formación y reconfiguración de sistemas.⁷⁹
- Jack Bryar concluye que la discusión sobre el coste total de propiedad de Linux está sin resolver. Ahora alguien de la comunidad Linux tiene que dar la cara y resolverla.⁸⁰
- En varias pruebas, Linux se mostró más o menos equivalente o un 15 por ciento mejor, según la revista alemana de informática técnica c't. En general, Linux gana a NT, pero no en todas las circunstancias (en una prueba, NT fue claramente superior), y definitivamente no hubo ninguna superioridad apabullante de ningún sistema.⁸¹
- El informe de Bloor concluye que el ganador general es Linux, aunque la diferencia con NT es a menudo pequeña. Linux destaca como servidor de ficheros, impresoras, web y de correo, y también en trabajos mixtos. En el entorno de servidor de bases de datos hay poca o ninguna diferencia entre Linux y NT. Windows NT es mejor como servidor de aplicaciones debido a la gran cantidad de software disponible para esta plataforma.
- Según la encuesta de rendimiento de Mindcraft, patrocinada por Microsoft, Windows NT es hasta 2.7 veces más rápido que Linux.⁸² Sin embargo, Mindcraft ha sido criticada por haber optimizado NT para obtener el máximo rendimiento, y no haberlo hecho con Linux.⁸³
- El servidor de ficheros de Windows opera casi un 200 por ciento más eficientemente con un sistema operativo de fuente abierta de lo que lo hace con el sistema operativo NT de Microsoft.⁸⁴
- El GartnerGroup, KPMG y Forrester Research coinciden en afirmar que el hardware y el software en conjunto suponen menos de un 20 por ciento del coste total de propiedad para la informática corporativa.⁸⁵
- Un informe de Compaq indica que en menos del 4 por ciento de las compras de TI se realiza un análisis ni siquiera preliminar del coste total de propiedad.⁸⁶

⁷⁸ Frank Hayes, "El secreto de Linux," Computerworld, marzo de 1.999.

⁷⁹ Jack Bryar, "¿Cuánto cuesta lo gratuito?" The Andover News Network, 15 de marzo de 2.000.

⁸⁰ Jack Bryar, "¿Cuánto cuesta lo gratuito?" The Andover News Network, 15 de marzo de 2.000.

⁸¹ Jürgen Schmidt, "Doble juego," c't (Revista alemana de informática técnica), 1.999.

⁸² Frans Godden, "¿Cómo se comportan Linux y Windows NT en la vida real?" GNet, enero de 2.000.

⁸³ Bruce Weiner, Mindcraft, "Test abierto: Windows NT Server 4.0 y Linux," Mindcraft, 30 de junio de 1.999.

⁸⁴ Mayor C. Justin Seiferth, jefe delegado de la División de Operaciones Globales de Tráfico Aéreo de las Fuerzas Aereas de EEUU, "La adoption de licencias abiertas," Periódico del COTS, noviembre/diciembre de 1.999.

⁸⁵ Jack Bryar, "¿Cuánto cuesta lo gratuito?" The Andover News Network, 15 de marzo de 2.000.

Normalmente la fuente abierta es mejor en las comparaciones en muchas circunstancias cuando se trata de implantaciones de servidores o sistemas empotrados que puedan necesitar de alguna personalización, pero no es mejor que el COTS en las aplicaciones típicas de escritorio. En realidad, algunas fuentes generalizan diciendo que los productos de fuente abierta no son peores que los de fuente cerrada, pero nuestros hallazgos indican que la escala con la que se mide el valor derivado del software de fuente abierta frente al de fuente cerrada puede favorecer en una u otra dirección dependiendo de los requisitos específicos y el entorno operativo del software.

Una decisión entre software abierto y software cerrado se basa en tres factores: (1) el coste – tanto directo (por ejemplo, su precio) como indirecto (por ejemplo, el tiempo de inactividad de los usuarios); (2) beneficios (por ejemplo, en rendimiento); y (3) otros criterios más intangibles (por ejemplo la calidad del soporte entre usuarios). Los costes directos son ampliamente entendidos y tradicionalmente comprenden la mayor parte los costes del ciclo de vida total de un sistema. Sin embargo, los costes indirectos y los beneficios de rendimiento y operativos (por ejemplo, la escalabilidad, la fiabilidad y la funcionalidad) juegan un papel económico más influyente en el mercado más maduro del software actual. Otros criterios más intangibles son difíciles de cuantificar, pero también afectan a la efectividad del software de fuente abierta y cerrada. Por ello, no se puede seguir utilizando como base los modelos del ciclo de vida tradicional y otras herramientas de software COTS cuando se trata de realizar decisiones para optimizar la inversión en entornos orientados a la misión.

El GartnerGroup, el principal fabricante de herramientas de estimación de coste de propiedad de TI, cuestiona en sus publicaciones el hecho de que sus herramientas de estimación del coste total de propiedad no están bien adaptadas para evaluar los costes empresariales del desarrollo y mantenimiento de aplicaciones de software. Además GartnerGroup admite que sus herramientas no se pueden personalizar para reflejar alternativas de software. El estudio FY99 el MOIE del ejército *Integración de los métodos de Coste Total de Propiedad con las estrategias de inversión* confirmaron que es necesario personalizar cualquier estructura de descomposición del trabajo al entorno específico y las iniciativas propuestas por el cliente. Más aún, las herramientas de estimación de costes de propiedad de TI no consideran las diferencias en los beneficios que existen entre las diferentes alternativas de software en cuestión. Por eso, los Jefes de Programa necesitan una taxonomía completa de los costes, los beneficios y otros criterios más intangibles para contabilizar los costes ocultos y los beneficios que podrían pasar desapercibidos de otra manera. Con esta taxonomía, los Jefes de Programa pueden tomar decisiones para la adquisición de software siendo completamente conscientes de las implicaciones económicas y de rendimiento en la misión. La siguiente tabla representa la Taxonomía de Elementos de Coste desarrollada por MITRE para el OSS y Linux.

⁸⁶ Jack Bryar, “¿Cuánto cuesta lo gratuito?” The Andover News Network, 15 de marzo de 2.000.

Tabla 5. Taxonomía de elementos de coste para OSS y Linux⁸⁷

Costes directos	
Software y Hardware	
Software	<ul style="list-style-type: none"> Precio de compra Actualizaciones y adiciones Tarifa de la licencia/propiedad intelectual
Hardware	<ul style="list-style-type: none"> Precio de compra Actualizaciones y adiciones
Coste del soporte	
Interno	<ul style="list-style-type: none"> Instalación y configuración Mantenimiento Resolución de problemas Herramientas de soporte (por ejemplo libros y publicaciones)
Externo	<ul style="list-style-type: none"> Instalación y configuración Mantenimimiento Resolución de problemas
Costes de personal	
Gestión de proyectos	
Ingeniería de sistemas y desarrollo	
Administración de sistemas	<ul style="list-style-type: none"> Gestión de proveedores
Otro tipo de administración	<ul style="list-style-type: none"> Compras Otros
Entrenamiento	
Desinstalación y enajenación	
Costes indirectos	
Costes de soporte	<ul style="list-style-type: none"> Soporte entre compañeros Aprendizaje no reglado Entrenamiento formal Desarrollo de aplicaciones Factor difuso
Tiempo de inactividad por caída del sistema	

⁸⁷ El GartnerGroup incluye un factor difuso como coste indirecto. GartnerGroup describe este término como el gasto laboral en que se incurre cuando el usuario final explota los activos informáticos corporativos para uso personal durante las horas de trabajo productivo.

3.3.1 Costes directos

3.3.1.1 Software y hardware

3.3.1.1.1 Software

Linux puede ser descargado gratuitamente de la Internet o comprado a un fabricante por un precio nominal de unos 60 dólares. No hay cuota de licencia. El coste de Microsoft Windows NT es de unos 600 a 800 dólares para cinco usuarios y 35 dólares cada usuario adicional. Las características adicionales de Microsoft, como telnet, servidor news, un servidor DNS mejor y cuotas de disco pueden suponer unos 3.800 dólares; estas características se incluyen en Linux sin coste adicional. El coste de Unix va desde los 1.000 a los 5.000 dólares, o 15.000 dólares para un número ilimitado de usuarios.⁸⁸

3.3.1.1.2 Hardware

Dado que Linux tiene relativamente pocas líneas de código y es altamente modular, puede funcionar en ordenadores menos potentes que otros sistemas operativos. Los usuarios normalmente ejecutan Linux en ordenadores antiguos y por lo demás inútiles (por ejemplo PCs 486) y no pagan por el hardware. Para aquellas compañías que han utilizado Unix, normalmente es más barato cambiar a Linux que a NT.

3.3.1.2 Soporte

3.3.1.2.1 Soporte interno

El coste laboral anual para soporte de ayuda para el escritorio es de unos 30.000 dólares.⁸⁹ Con código de fuente abierta es posible que los problemas sean fijados internamente en la organización del usuario. Con código cerrado, software de marca, los problemas deben ser arreglados por el suministrador o fabricante externo.

3.3.1.2.2 Soporte externo

El soporte externo para Linux cuesta entre 60 y 85 dólares por incidencia o 3.000 dólares para un grupo de 10 incidencias. También hay paquetes de soporte empresarial disponibles para Linux por unos 60.000 dólares al año. El coste de soporte de NT cuesta 200 dólares por incidencia o 1.700 dólares por un grupo de 10 incidencias.⁹⁰

3.3.1.3 Gastos de personal

De lo que se ha encontrado en la revisión de la literatura y las entrevistas con expertos se infiere que a menudo no hay una diferencia identificable entre los costes laborales anuales por empleado entre los OSS y COTS. Sin embargo, dependiendo del proyecto específico y la estructura de la organización, la cantidad de trabajo necesaria

⁸⁸ D.H. Brown y Asociados, "Linux: ¿Cuán bueno es?", 1.999; John Kirch, "Microsoft Windows NT Server 4.0 frente a Unix," 7 de agosto de 1.999; y Robert Lauriston, "La oficina de No-Microsoft," *computercurrents.com*, 23 de marzo de 1.999.

⁸⁹ Coste laboral basado en datos de Medzilla; Wageweb; y Chim-Net, no consta fecha.

⁹⁰ "¿Únirse a la revolución del software libre?" CIO, 19 de marzo de 1.999 y Robert Lauriston, "La oficina de No-Microsoft," *computercurrents.com*, 23 de marzo de 1.999.

podría diferir entre el OSS y el COTS. Nosotros recomendamos que el coste del personal de cada caso sea evaluado por separado.

3.3.1.3.1 Gestión de proyectos

El coste laboral anual de un jefe de proyecto puede ser aproximadamente unos 60.000 dólares.⁹¹

3.3.1.3.2 Ingeniería/Desarrollo de sistemas

El coste laboral de un ingeniero de sistemas está entre los 45.000 y los 90.000 dólares al año.⁹²

3.3.1.3.3 Administración de sistemas

El coste laboral anual de un administrador de sistemas está entre los 30.000 y los 65.000 dólares. El coste por hora está entre los 30 y los 100 dólares.⁹³

3.3.1.3.4 Otros gastos de administración

El coste laboral anual para otros servicios administrativos están entre los 21.000 dólares y los 45.000 dólares por persona.⁹⁴

3.3.1.3.5 Formación

Los cursos de formación cuestan unos 2.000 dólares para un curso de cuatro días y 2.500 dólares para un curso de cinco días. Para los administradores de sistemas, los profesionales de Unix suelen necesitar un curso; los profesionales que vienen desde otros sistemas operativos suelen necesitar cuatro cursos. Los desarrolladores suelen necesitar entre una y dos clases.⁹⁵

3.3.1.4 Desinstalación y eliminación de software

Es relativamente fácil desinstalar y enajenar el software de Linux. Sin embargo puede que haya costes de integración que tienen que ver con la compatibilización del nuevo software con el sistema.

3.3.2 Costes indirectos

Los costes indirectos incluyen las influencias “ocultas” y las interrelaciones causales que pueden ser difíciles de medir. Dado que son potencialmente significativas, es importante identificarlas y considerarlas. Estos costes indirectos pueden ser medidos en términos de pérdida de productividad atribuible al entorno informático. Mientras que el salario y otros costes laborales asociados a un empleado se reflejan bajo la categoría de costes directos, los costes indirectos representan trabajo que es “desperdiciado” y podría ser

⁹¹ Coste laboral basado en datos de Medzilla; Wageweb; y Chim-Net, no consta fecha.

⁹² Coste laboral basado en datos de Medzilla; Wageweb; y Chim-Net, no consta fecha.

⁹³ Coste laboral basado en datos de Jeff Covey, “Un nuevo plan de negocio para el software libre,” Freshmeat, 22 de enero de 2.000; Medzilla; Wageweb; y Chim-Net, no consta fecha.

⁹⁴ Coste laboral basado en datos de Wageweb, no consta fecha.

⁹⁵ Red Hat, <http://www.redhat.com>, no consta fecha.

usado de manera más productiva. En otras palabras, aunque no hay un coste adicional directo para la organización, no se recibe del empleado el resultado que sería de esperar debido a ineficiencias en el proceso o el sistema.

3.3.2.1 Costes de soporte

Los costes de soporte indirecto varían según el uso específico y el entorno del software. Estos costes pueden ser diferentes entre el OSS y el COTS tradicional. Por eso, para estudiar los costes del soporte indirecto hay que evaluar cada caso por separado.

3.3.2.1.1 Soporte entre pares

El coste indirecto debido al soporte entre compañeros incluye el gasto de trabajo de los usuarios finales que se ayudan unos a otros en lugar de obtener soporte técnico del personal de apoyo y ayuda. Un caso de estudio mostró que el soporte técnico que se llevaba de manera informal en el lugar de trabajo en lugar de mediante profesionales de TI ahorra dinero al departamento del TI, pero añadía hasta un 27 por ciento a los costes administrativos generales.⁹⁶

3.3.2.1.2 Aprendizaje informal

Este apartado incluye los gastos por el trabajo que realizan los usuarios finales en autoformación y autosoporte en lugar de la formación reglada y los programas de soporte.

3.3.2.1.3 Formación oficial

Este coste indirecto incluye todo el tiempo utilizado por los usuarios finales en formación para el uso de aplicaciones y sistemas informáticos. Los cursos de formación típicamente duran entre cuatro y cinco días. Este tiempo debería ser considerado un gasto, ya que requiere que el asistente se ausente de sus responsabilidades directas. El coste del transporte también podría incluirse, dependiendo de dónde se imparte el curso.

3.3.2.1.4 Desarrollo de aplicaciones

El desarrollo de aplicaciones incluye el coste del trabajo de los usuarios finales que realizan el desarrollo y personalización de aplicaciones no críticas y no directamente relacionadas con el negocio o la misión.

3.3.2.1.5 El factor Futz

El GartnerGroup incluye el factor difuso como coste indirecto. Este es el gasto laboral en que se incurre cuando el usuario final explota los activos informáticos corporativos para uso personal durante las horas de trabajo productivo.

3.3.2.2 Tiempo de inactividad por caída de los sistemas

El tiempo de inactividad por caída de los sistemas refleja las pérdidas de productividad debidas a la no disponibilidad del ordenador de sobremesa, los servidores, las aplicaciones u otras herramientas. Los costes se miden en términos de salarios perdidos. Puede calcularse sumando el porcentaje de horas de inactividad planificadas y no planificadas correspondiente al impacto en la productividad de los usuarios cuando se

⁹⁶ Jack Bryar, “¿Cuánto cuesta lo gratuito?” The Andover News Network, 15 de marzo de 2.000.

producen las caídas y reflejándolo como salario extra. Según la mayoría de los estudios, los usuarios de Windows experimentan un mayor tiempo de inactividad por caída de los sistemas que los usuarios de Linux. Puede encontrarse más información sobre los tiempos relativos de funcionamiento y caída en la sección 3.3.3.2.

3.3.3 Beneficios y riesgos

La siguiente tabla representa la taxonomía desarrollada por MITRE relativa a los beneficios y riesgos asociados al OSS y Linux, así como un ejemplo de escala de valoración.

Tabla 6. Taxonomía de beneficios y riesgos del OSS

<i>Atributos cualitativos</i>	
	Capacidades de personalización
	Disponibilidad/fiabilidad
	Interoperabilidad
	Escalabilidad
	Flexibilidad del diseño
	Vida útil
	Rendimiento
	Calidad del servicio y del soporte
	Seguridad
	Nivel de dificultad/facilidad de gestión
	Riesgo de fragmentación
	Disponibilidad de aplicaciones

Ejemplo de escala de valoración	
Muy fuerte	
Fuerte	
Neutro	
Débil	
Muy débil	

Los Jefes de Programa pueden utilizar esta taxonomía como base de comparación de costes, beneficios y otros criterios más intangibles entre el OSS y el software tradicional COTS. La anterior taxonomía comprende una lista de atributos cualitativos. Para cada atributo, los Jefes de Programa deberían comparar la fortaleza relativa o la debilidad del OSS frente a los productos tradicionales COTS. Una fortaleza relativa indicaría un beneficio, una debilidad relativa indicaría un riesgo. Arriba se muestra un ejemplo de escala de valoración que permite comparar los valores relativos del OSS frente a los tradicionales COTS. Este ejemplo presenta cinco niveles – muy fuerte, fuerte, neutro, débil y muy débil. Dado que estos niveles varían dependiendo del uso específico y del entorno del software, los Jefes de Programa deberían personalizar sus niveles según las

circunstancias particulares. A continuación se ofrece una discusión de los beneficios y riesgos de Linux.

3.3.3.1 Capacidad de adaptación

Dado que el código fuente está abierto, Linux puede ser adaptado para que se ajuste a las necesidades de una organización. Las organizaciones pueden contribuir con ideas útiles y expandir la funcionalidad existente para proporcionar una característica o sistema completamente nuevos. La aplicación estricta de Linux, la división en componentes del OS y la disponibilidad de las interioridades expuestas la convierten en la alternativa preferida para tareas a medida. A menudo se prefiere a Windows NT para servidores aislados y que deben realizar una única tarea, como DNS, servicio de ficheros, correo y web. La personalización se hará más importante con el tiempo según se incrementen el número de servidores y su dedicación a tareas específicas. La modularidad de Linux permite su uso en un amplio rango de sistemas, desde un supercomputador a un frigorífico.

3.3.3.2 Disponibilidad/Fiabilidad

Esta es la cantidad de tiempo que un sistema está activo y funcionando. Se trata de un objetivo prioritario de la comunidad de Linux y una de las grandes debilidades de Windows. Debido a que hay tantos programadores trabajando en la mejora del código de Linux, los errores se descubren con mayor facilidad y se arreglan, lo que mejora la estabilidad del software. Además, el núcleo de Linux utiliza una sistema de gestión de memoria virtual que comparte la memoria entre todos los programas activos. Da a cada programa un espacio de direcciones virtuales separado, con lo que se reduce el efecto de un programa sobre otro. Este sistema de gestión también evita que los programas sobrescriban áreas críticas de memoria (por ejemplo, las áreas donde se almacena el núcleo de Linux). Normalmente, un ordenador ha de ser rearrancado cuando se reconfigura Windows NT u ocurren problemas de carga de software; esto no suele ser necesario en Linux. Los estudios de rendimiento coinciden en señalar que Linux es más fiable que Windows. El estudio de rendimiento de Bloor Research midió la razón entre el tiempo de actividad y el tiempo de inactividad de Linux y Windows NT durante un año. En este tiempo, Linux falló una única vez debido a un fallo de hardware (problemas con el disco) y llevó cuatro horas arreglarlo. Windows NT falló 68 veces debido a problemas de hardware, memoria, gestión de ficheros y otros problemas varios, todo lo cual llevó 65 horas arreglarlo. Así, la disponibilidad de Linux fue del 99,95 por ciento y la de NT 99,26 por ciento. En un estudio de rendimiento similar, el Giga Information Group determinó que la disponibilidad de Unix era de un 99,8 por ciento mientras que la de Windows NT era de un 99,2 por ciento.⁹⁷

3.3.3.3 Interoperabilidad

Todo protocolo abierto existente ha sido incorporado a Linux. Samba permite a Linux simular un NT. Tanto Linux como Microsoft Windows NT se pueden utilizar con diversas plataformas de hardware.

⁹⁷ Laura DiDio, citada por Derek Slater, "Factores decisivos – Sistemas Operativos," CIO Magazine, 1 de febrero de 2.000 y Frans Godden, "¿Cómo se comportan Linux y Windows NT en la vida real?" GNet, enero de 2.000.

3.3.3.4 Scalability

La escalabilidad es la capacidad del software para añadir usuarios al sistema manteniendo el rendimiento. Hay varios estudios de rendimiento que comparan la escalabilidad del sistema operativo. La revista alemana *c't* mostraba que Linux alcanzó una mayor escalabilidad. La revista *CIO* informó que Unix puede dar soporte a 2.500 usuarios concurrentes. Microsoft NT sólo soporta hasta 850 según la revista *CIO* o 1.000 según el *GartnerGroup*.⁹⁸ Frans Godden especificó que Linux tiene más espacio para escalar porque puede utilizarse tanto en servidores de gama alta como en máquinas pequeñas y ordenadores ultraportables, mientras que Microsoft debe ser cambiado por un sistema operativo de diferente tamaño. El informe de *Bloor* indicaba que el rendimiento escalado de Linux y NT era prácticamente igual en la gama baja. El informe de rendimiento de *Mindcraft*, patrocinado por Microsoft, mostraba que NT era más escalable que Linux.⁹⁹

3.3.3.5 Flexibilidad del diseño

El núcleo de Linux puede ser simplificado poco a poco para eliminar las características innecesarias o expandido para incluir características adicionales.

3.3.3.6 Tiempo de vida

El tiempo de vida de los sistemas licenciados con código abierto puede extenderse de forma indefinida dado que tanto el código fuente como la documentación están libremente disponibles. La información que requiere el acceso a largo plazo no quedará enterrada como ocurre con los formatos de marca no documentados, obsoletos y disfuncionales cada pocos años. Con el licenciamiento de código abierto, el usuario puede desarrollar el software dentro de la propia empresa o subcontratarlo al fabricante original o subcontratarlo a un proveedor de soporte que entre en el mercado con posterioridad.

3.3.3.7 Rendimiento

El rendimiento es la capacidad de uso de los recursos informáticos (por ejemplo, procesadores, memoria y discos) de forma eficiente. En un PC monoprocesador típico, Linux se comporta mejor que la familia de sistemas operativos Windows, porque los sistemas Windows tienden a necesitar bastante más procesador, memoria y recursos de disco que de otro modo podrían estar disponibles para que los programas de aplicación los usasen.

3.3.3.8 Calidad de servicio y soporte

Como se muestra en el análisis DAFO, la competitividad en la estructura de servicio y soporte del OSS permite que los distribuidores de Linux ofrezcan un soporte de mayor calidad a un precio menor que los fabricantes de COTS tradicionales.

⁹⁸ Deate Hohmann, *GartnerGroup*, conversación telefónica, diciembre de 2.000.

⁹⁹ Thomas Bittman, *GartnerGroup* Derek Slater, "Factores decisivos – Sistemas Operativos," *CIO Magazine*, 1 de febrero de 2.000 y Frans Godden, "¿Cómo se comportan Linux y Windows NT en la vida real?" *GNet*, enero de 2.000.

3.3.3.9 Seguridad

Para las auditorías de seguridad, los consultores prefieren Linux a Sun Solaris con un ratio de 50:1, según Network Associates.¹⁰⁰ Un grupo improvisado de trabajo compuesto por el DARPA, el GSA, el NIST y la NSA concluyó que el uso de OSS puede tener efectos tanto negativos como positivos en la seguridad de los sistemas federales.¹⁰¹

Los beneficios del OSS en materia de seguridad incluyen los siguientes:

- Una oportunidad para las partes interesadas de aplicar herramientas de análisis estadístico para detectar la presencia de código malicioso o características no documentadas. Las herramientas automáticas pueden utilizarse para reducir el esfuerzo que implica la búsqueda de vulnerabilidades.
- Los fallos y errores que se encuentran en el software se pueden reparar rápidamente mediante la creación y distribución de parches de software.
- Dado que el código fuente está ampliamente disponible, no sólo el desarrollador, sino que también las partes interesadas pueden aplicar métodos de prueba de “caja blanca”.
- La mayor disponibilidad del código fuente hace más probable que las consecuencias negativas o inesperadas de las modificaciones en los componentes sobre el resto del sistema sean descubiertas. Esto se debe al tamaño y diversidad de las partes interesadas que sean capaces de estudiar dichos impactos.
- Las organizaciones de usuarios individuales pueden modificar el código fuente para ajustarlo a sus propias necesidades específicas. Por ejemplo, el gobierno podría desarrollar versiones a medida del OSS incorporando características extra de seguridad a la versión estándar para producir una versión aprobada por el gobierno.

Sin embargo, el OSS a menudo carece de características clave de seguridad que son necesarias para proteger información y proceso crítico. Ya sea un sistema de código abierto o no, los sistemas operativos deficientemente configurados y gestionados son por lo general inseguros.

3.3.3.10 Nivel de dificultad o facilidad de gestión

Como se discute en el análisis DAFO, muchos encuentran que Linux es menos amigable para el usuario y, por eso, más difícil de utilizar que los productos tradicionales COTS. Sin embargo, comparado con Unix y Windows NT, Linux es el más fácil de gestionar dado que la gestión está más centralizada y permite características como la gestión remota, el soporte de cuotas de disco, la seguridad remota y el arranque sin disco; con Linux un administrador de redes no necesita estar en todos los sitios. NT es más difícil de gestionar y es particularmente malo con la gestión remota.

¹⁰⁰ Tim Clark, “Network Associates añade un producto para Linux,” CNET News, 8 de febrero de 1.999.

¹⁰¹ “Informe sobre el código de fuente abierta y la seguridad de los sistemas federales,” DARPA, GSA, NIST y NSA, preparado para el Coordinador Nacional de Seguridad, Protección de Infraestructuras y Contraterrorismo, junio de 1.999.

3.3.3.11 Riesgo de fragmentación

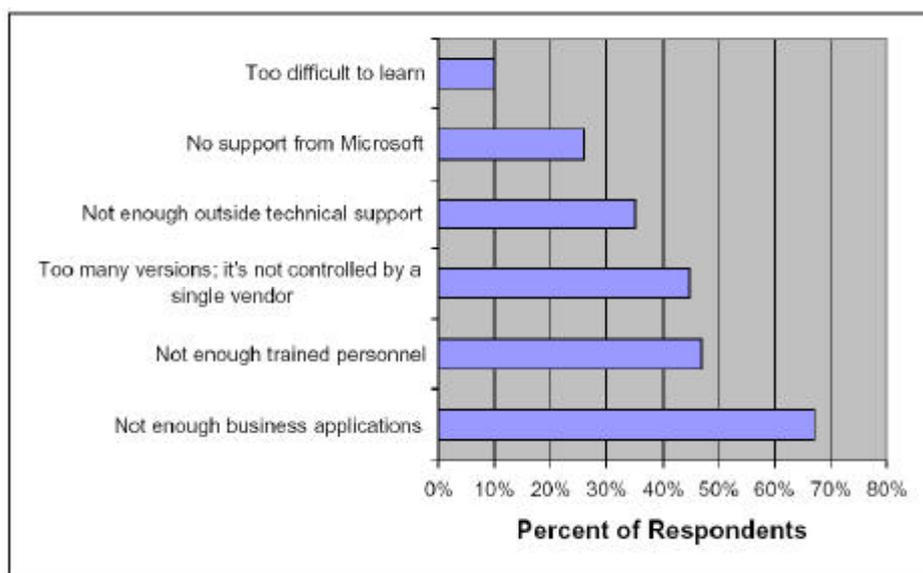
Como se muestra en el análisis DAFO, el código del núcleo de Linux todavía no se ha dividido. Sin embargo, los productos de OSS se enfrentan a un riesgo mayor de fragmentación de la base del código que el que se enfrentan los productos COTS tradicionales. Existen unas pocas incompatibilidades entre las muchas distribuciones únicas de Linux disponibles. Esta fragmentación pueden producir peleas internas entre los distribuidores, según se mueven para dominar el mercado. Sin embargo, la fragmentación permite a los diferentes distribuidores dirigirse a diferentes mercados.

Se puede usar Linux en un amplio rango de sistemas y la modularidad del software tiende por si misma a versiones diferentes.

3.3.3.12 Disponibilidad de aplicaciones

Como se describe en el análisis DAFO, el número de aplicaciones escritas para Linux va creciendo a una velocidad desproporcionada comparado con otros sistemas operativos de uso generalizado.

La siguiente gráfica muestra las debilidades más significativas de Linux según el informe de Information Week.¹⁰²



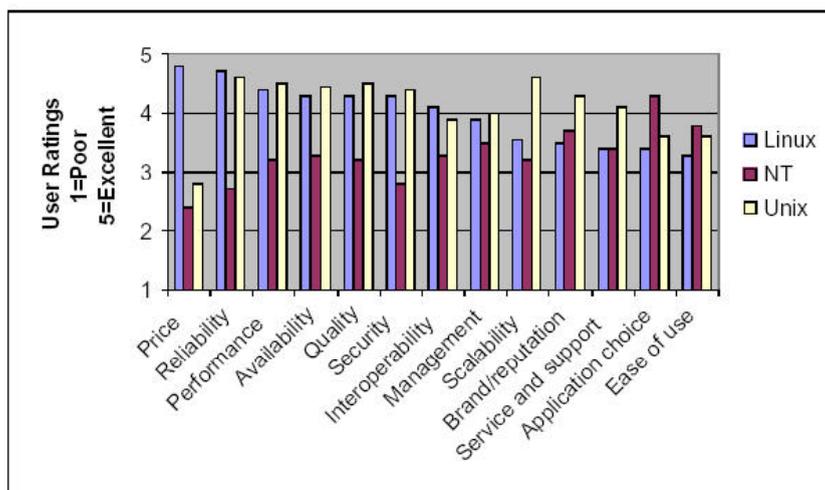
Fuente: Aaron Ricadela, "Linux vive," InformationWeek, 24 de enero de 2.000.

Ilustración 21. Debilidades más significativas de Linux

En la encuesta de IDC se pidió a los usuarios de Linux en EEUU que calificaran la calidad de Linux, NT y Unix como servidores sobre la base de diversas métricas de rendimiento. Estas métricas incluían el precio, la fiabilidad, el rendimiento, la disponibilidad, la calidad, la seguridad, la interoperabilidad, la gestión, la escalabilidad, la reputación de la marca, el servicio y el soporte, la selección de aplicaciones y la facilidad

¹⁰² Aaron Ricadela, "Linux vive," InformationWeek, 24 de enero de 2.000.

de uso. Los usuarios calificaron el precio de los servidores Linux mucho mejor que el precio de los servidores NT o Unix. Además, los usuarios calificaron mejor la calidad de los servidores Linux frente a NT en las características más importantes para un servidor: rendimiento, disponibilidad, calidad, seguridad, interoperabilidad y gestión. Unix se calificó de forma equivalente a Linux en estas importantes características para un servidor. Sin embargo los servidores Unix consiguieron mejor puntuación que los servidores Linux en cuestiones como la escalabilidad, la reputación de la marca y el servicio y el soporte. Estos ratios mayores de Unix se pueden atribuir a la mayor madurez de la plataforma. La siguiente gráfica ilustra la calidad de los niveles de rendimiento en servidores según el sistema operativo.¹⁰³



Fuente: Michelle Bailey, Vernon Turner, Jean Bozman y Janet Waxman, “Servidores Linux: qué hay de exageración, qué de realidad” IDC, marzo 2.000.

Ilustración 22. Puntuación otorgada por los usuarios de Linux de EEUU en calidad de servicio según el sistema operativo.

Muchos de estos hallazgos son consistentes con una reciente evaluación realizada por el GartnerGroup sobre los sistemas operativos Linux, NT, Unix para RISC y Windows 2.000. El GartnerGroup compara las puntuaciones de la funcionalidad actual (en 1.999 o 2.000) con las proyecciones para el año 2.003. La evaluación concluye que Linux está haciendo progresos técnicos, pero que seguirá enfrentándose a desafíos en el segmento superior según Unix y Windows 2.000 avancen. Para el 2.003, Linux habrá conseguido una puntuación aceptable para la mayoría de las categorías, pero pocas puntuaciones excelentes. Las mejores puntuaciones de Linux son por su estabilidad, la capacidad de clustering, su seguridad y su precio. NT obtiene puntuaciones excelentes por sus manejadores de dispositivo con tecnología conectar-y-utilizar y por el soporte de los distribuidores de servicio independientes y de los proveedores de valor añadido (ISV/VAR). Unix es alabado por su estabilidad, su escalabilidad para el multiproceso simétrico (SMP), la capacidad de clustering, la alta disponibilidad, el tamaño de los sistemas de gestión de bases de datos relacionales (RDBMS), el soporte técnico, el soporte de los ISV/VAR, la gestión del

¹⁰³ Michelle Bailey, Vernon Turner, Jean Bozman y Janet Waxman, “Servidores Linux: qué hay de exageración, qué de realidad” IDC, marzo 2.000.

sistema y su seguridad. Windows 2.000 es excelente en escalabilidad en SMP, manejadores de dispositivos conectar-y-utilizar, soporte técnico, soporte de los ISV/VAR y gestión del sistema. Estas puntuaciones se muestran en la siguiente tabla.¹⁰⁴

Tabla 7. Comparativa de sistemas operativos

	Linux 99/'03	NT 99	Unix 99/'03	Win 2000 00/'03
Stability	++/++	-	++/++	-/+
SMP scaling	-/+	+	++/++	+/++
Clustering	+/++	-	+/++	-/+
High availability	-/+	-	+/++	-/+
RDBMS size	-/+	+	+/++	+/+
Ease of use	-/-	+	-/-	+/+
Plug-and-play drivers	-/+	++	-/-	++/++
Technical support	+/+	+	++/++	+/++
ISV/VAR support	-/-	++	++/++	+/++
System management	-/+	+	+/++	+/++
Security	+/++	-	+/++	-/+
Pricing	++/+	+	-/+	+/+
<p style="text-align: center;"> ++ Excellent + Acceptable - Deficient </p>				

Fuente: G. Weiss, "Evaluación de SO actualizada: Linux frente a Unix y Windows 2.000," GartnerGroup, 25 de Julio de 2.000.

¹⁰⁴ G. Weiss, "Evaluación de SO actualizada: Linux frente a Unix y Windows 2.000," GartnerGroup, 25 de Julio de 2.000.

Referencias

Nota de traducción: Aunque en las notas a pie de página se han traducido todos los títulos de los documentos referidos y muchos de los nombres de las empresas que los publicaron, en esta sección se ha preferido dejar títulos y referencias en inglés. La motivación es clara: facilitar la localización de los documentos al lector en la búsqueda bibliográfica.

1. Bailey, Michelle, Vernon Turner, Jean Bozman y Janet Waxman, "Linux Servers: What's the Hype, and What's the Reality?" IDC, marzo de 2.000.
2. Be, www.be.com.
3. Bryar, Jack, "How Much Does Free Cost?" The Andover News Network, 15 de marzo de 2.000.
4. Caldera, www.caldera.com.
5. Clark, Tim, "Network Associates Adds Linux Product," CNET News, 8 de febrero de 1.999.
6. Chime-Net; Medzilla, 1.999; y Wageweb, 2.000.
7. CoolLogic, www.coollogic.com.
8. Corel, www.corel.com.
9. Covey, Jeff, "A New Business Plan for Free Software," Freshmeat, 22 de enero de 2.000.
10. Datapro, febrero de 1.999.
11. Debian, www.debian.org.
12. D. H. Brown Associates, "Linux: How Good Is It?" 1.999.
13. Embedded Linux Consortium, www.embedded-linux.org.
14. Epplin, Jerry, "Linux as an Embedded Operating System," octubre de 1.997, <http://www.embedded.com/97/fe39710.htm>.
15. "French Ministry Adopts Open-Source Culture, Linux," InfoWorld.com, 8 de febrero de 2.000, <http://www.infoworld.com/articles/ec/xml/00/02/08/000208eclinparis.xml>.
16. "The Future of Linux," CNet, 2.000.
17. "Getting to Know Linux," Colorado Business, Julio de 2.000.
18. Gillen, Al, y Dan Kusnetzky, "Linux Overview: Understanding the Linux Market Model," IDC, febrero de 2.000.
19. Gutfraind, Alexander, "Introductory into the World of Linux," The Linux World, <http://www.tht.net/~gutfrnd/linux/intro/linworld.htm>, 1.998.
20. Harmon, Paul, "Linux and Architecture," Cutter Consortium, 9 de febrero de 2.000.
21. Hohmann, Deate, GartnerGroup, conversación telefónica, diciembre de 2.000.
22. Hontañón, Ramón J., UUNET, "Building a Robust Linux Security Solution," Network Magazine, 2.000.

23. "Join the Freeware Revolution?" CIO, 19 de marzo de 1.999.
24. Jordan, Peter, "Nibbling Away at UNIX," VARBusiness, 14 de enero de 2.000.
25. Kaminsky, Dan, "Core Competencies: Why Open Source is the Optimum Economic Paradigm for Software," 2 de marzo de 1.999.
26. Kirch, John, "Microsoft Windows NT Server 4.0 Versus UNIX," 7 de agosto de 1.999.
27. Lauriston, Robert, "The Un-Microsoft Office," ComputerCurrents.com, 23 de marzo de 1.999.
28. Lerner, Josh y Jean Tirole, "The Simple Economics of Open Source," National Bureau of Economic Research, marzo de 2.000.
29. LinuxDevices, <http://www.linuxdevices.com>.
30. "Linux Is Biggest Focus Shift Since TCP/IP, Says IBM," Network News, 30 de agosto de 2.000.
31. "National Security Agency Selects Secure Computing to Provide Type Enforcement on Linux OS," 14 de enero de 2.000.
32. "Open Source Code and the Security of Federal Systems," Informe del Working Group, DARPA, GSA, NIST and NSA, preparado para el National Coordinator for Security, Infrastructure Protection and Counter-Terrorism, junio de 1.999.
33. O'Reilly, Tim y Ether Dyson, "Open Mind, Open Source."
34. O'Reilly, Tim, Linux eSeminar Series, 1.999.
35. Orzech, Dan, "Linux and the Saga of Open Source Software," Datamation, febrero de 1.999.
36. "PC DOS," IBM, <http://www-3.ibm.com/software/os/dos/>.
37. Quinlan, Daniel, "The Past and Future of Linux Standards," Linux Journal, junio de 1.999.
38. Raymond, Eric, "The Cathedral and the Bazaar," O'Reilly Associates, 1.999.
39. Raymond, Eric, <http://www.opensource.org>.
40. Ready, Jim y Bill Weinberg, "Leveraging Linux for Embedded Applications," LinuxDevices.com.
41. RedHat, <http://www.redhat.com>.
42. Ricadela, Aaron, "Linux Comes Alive," InformationWeek, 24 de enero de 2.000.
43. Schmidt, Jürgen, "Mixed Double," c't (German technical computer magazine), 1.999.
44. Secure Computing Corporation, "Reconsidering Assurance for Open Source Software," 30 de noviembre de 1.999.
45. Seiferth, C. Justin, Major US Air Force, Deputy Chief, Global Air Traffic Operations Division, "Opening the Military to Open Source," COTS Journal, noviembre/diciembre de 1.999.

46. Seiferth, C. Justin, Major, US Air Force, Deputy Chief, Global Air Traffic Operations Division, "Adoption of Open Licensing," COTS Journal, noviembre/diciembre de 1.999.
47. Seiferth, C. Justin, Major, US Air Force, "Open Source and the United States," Air Command and Staff College, Air University, 11 de junio de 1.999, http://ceu.fi.udc.es/GPUL/articulos/varios/US_DoD_and_OSS.txt.
48. Slackware, www.slackware.com.
49. Slater, Derek, "Deciding Factors - Operating Systems," CIO Magazine, 1 de febrero de 2.000.
50. Stoltz, Mitch, "The Case for Government Promotion of Open Source Software," un artículo de NetAction, 1.999.
51. SuSE, www.suse.com.
52. United States Air Force Scientific Advisory Board, "Ensuring Successful Implementation of Commercial Items in Air Force Systems," SAB-TR-99-03, abril de 2.000.
53. Valloppillil, Vinod, Microsoft, "Open Source Software," editado por Eric Raymond como Halloween I, 1.998.
54. Valloppillil, Vinod, y Josh Cohen, Microsoft, "Open Source Software," editado por Eric Raymond como Halloween II, 1.998.
55. Vaughn-Nichols, Steven J., "TripWire Delivers Open Source DDoS and Security Answer," Sm@rt Reseller, 1 de marzo de 2.000.
56. Weiner, Bruce, Mindcraft, "Open Benchmark: Windows NT Server 4.0 and Linux," Mindcraft, 30 de junio de 1.999.
57. Weiss, G., "The Competitive Impact of IBM's Linux Announcement," GartnerGroup, 8 de febrero de 2.000.
58. Weiss, G., "The GartnerGroup Server Operating System Forecast," GartnerGroup, 26 de marzo de 2.000.
59. Weiss, G., "How the Open Source Movement Will Affect Users," GartnerGroup, 26 de enero de 1.999.
60. Weiss, G., "Linux Adoption Best Practices: A 10-Point Program," GartnerGroup, 8 de febrero de 2.000.
61. Weiss, G., "Updated OS Evaluation: Linux vs. Unix and Windows 2.000," GartnerGroup, 25 de julio de 2.000.
62. White, Walker, "Observations, Considerations, and Directions," Oracle, 8 de mayo de 2.000 cita a Frederick Brooks en "The Mythical Man Month."
63. Williams, Tom, "Linux Catches the Embedded Wave," febrero de 2.000, <http://www.embeddedsystems.com>.

Glosario de acrónimos

Nota de traducción: En los documentos técnicos, la regla general suele ser no traducir los acrónimos salvo que su traducción sea común en la lengua (por ejemplo EEUU) o se utilice regularmente en la literatura especializada (TI, SO o DAFO). Nosotros hemos seguido esta costumbre, por lo que lo habitual es que la correspondencia entre las letras que componen el acrónimo y su significado quede rota. No obstante, y para evitar confusiones se indican los acrónimos no traducidos con una nota que indica “del inglés”.

API Interfaz de programación de aplicaciones (del inglés)

B Billón (el billón americano equivale a mil millones de unidades)

C2 Control y comando (del inglés)

COTS (Software) Comercial de distribución generalizada (del inglés)

CPU Unidad central de proceso (del inglés)

DAFO Debilidades, Amenazas, Fortalezas y Oportunidades

DNS Servicio/servidor de nombres de dominio (del inglés)

DOD Ministerio de Defensa (del inglés)

EEUU Estados Unidos de Norteamérica

ELC Consorcio para el Linux empotrado (del inglés)

ELKS Subconjunto empotrable del núcleo de Linux (del inglés)

FTP Protocolo de transferencia de ficheros (del inglés)

FUD Temor, desconfianza y duda (del inglés)

GNU Gnu No es Unix

GPL Licencia Pública General (del inglés)

I+D Investigación y desarrollo

IGU Interfaz gráfica de usuario (del inglés)

ISP Proveedor de servicios de Internet (del inglés)

ISV/VAR Distribuidor de servicios independientes/Proveedor de valor añadido (del inglés)

LSB Base estándar de Linux (del inglés)

M Millones

MOIE Investigación y Experimentación Orientada a Misiones (del inglés)

PC Ordenador personal (del inglés)

PDA Asistente digital personal (del inglés)

Perl Lenguaje práctico de extracción e informes (del inglés)

POSIX Interfaz para un sistema operativo portable (del inglés)

OSI Iniciativa para la fuente abierta (del inglés)

OSS Software de fuente abierta (del inglés)

RPM Gestor de paquetes de Red Hat (del inglés)

SGBDR Sistema de gestión de bases de datos relacionales

SI Sistema de información

SMP Multiproceso simétrico (del inglés)

SO Sistema operativo

SOTR Sistema Operativo de tiempo-real

TCP/IP Protocolo de control de transmisión/Protocolo Internet (del inglés)

TI Tecnologías de la información

Lista de distribución

Interna

D480

B. N. Obando

F063

D. C. Hite

G040

G. L. Hollis

J. P. Root

W010

S. D. Huffman

W030

C. C. Howell

W032

T. B. Bollinger

W110

J. W. Moore

W118

D. E. Emery

M. A. Macpherson

W800

R. C. Evans

P. R. Garvey

R. Haller

S. E. MacReynolds

W802

R. T. Arace

S. A. Castro

J. M. Deems

J. C. Ellenbogen

T. L. Hoffman-Boswell

M. A. Janiga

B. J. Jasper

P. A. Kelley

M. W. Kilgore

H. W. Loomis

J. E. Manring

W. B. Reading

T. J. Restivo

B. M. Rolfe

D. A. Smith

R. C. Tepel

J. R. Valaika

D. L. White

W803

D. A. Crawford

F. M. Dello Russo

W803 (Continúa)

R. J. Gialombardo
A. M. Goldberg
E. S. Goyette
C. A. Kenwood (10)
R. A. Moynihan
D. H. Plummer
W. F. Schaefer (2)
A. E. Taub
J. A. Vitkevich

Externa

Program Manager, Abrams Tank
System

Attn: Kevin Houser
6501 E. 11 Mile Rd.
Building 229
Warren, MI 48397-5000

R. C. Seay
Department Chief of Staff for

Programs

Attn: DAPR-FDT
700 Army Pentagon
Washington DC 20310-0700

Stan Levine, DOI
C/o R. C. Seay
Department Chief of Staff for

Programs

Attn: DAPR-FDT
700 Army Pentagon
Washington DC 20310-0700

Externa (Continúa)

Program Manager, Abrams Tank
System

Attn: LTC Robert Lovett
6501 E. 11 Mile Rd.
Building 229
Warren, MI 48397-5000

Barry Mareiro
ESC/ACW,
50 Griffiss Street (2M-104)
Hanscom AFB, MA 01731-1625

Frank McPherson
1415 Andover Ct.
Evans, GA 30809

Colonel (P) James Moran,
Commandant
Defense Systems Management
College

9820 Belvoir Rd., Suite G3
Ft. Belvoir, VA 22060-5565