



Software Abierto:
Factibilizando la neutralidad tecnológica para Chile

Enero 2005

Documento elaborado en
Fundación País Digital

Con la participación de los siguientes colaboradores
(en orden alfabético)

Hernán Astudillo⁵,
Constanza Capdevila,
Alberto Cerda⁵,
Paolo Colonnello¹,
Alejandro Fuentes²,
Jens Hardings²,
Eduardo Kaftanski³,
Alejandro Mery⁴,
Domingo Mery⁴,
Carlos Muñoz³,
Marcela Olivares,
José M. Piquer⁵,
Darío Riquelme

De las siguientes empresas o instituciones

¹Blue Company
²Centro de Software Libre
³Linux Center
⁴Pontificia Universidad Católica
⁵Universidad de Chile

PRESENTACION

Entre las discusiones más relevantes, de los últimos años, en la industria del software están los temas de la propiedad intelectual, los derechos comerciales y la dependencia de sistemas de software específicos.

En la medida que las Tecnologías de Información y Comunicaciones, TIC's, son parte integral en los modelos de operación de las organizaciones, los presupuestos que se destinan para estas áreas son cada vez más relevantes. De la misma manera, la dependencia de la organización está también muy ligada al destino de los sistemas que se hayan implantado en la institución, para facilitar o cambiar su modelo de operación.

El tema del costo pasa a ser relevante cuando se trata de software tipo *commodity*, incluyendo sistemas operativos y de uso general, tipo aplicaciones de oficinas. Estos sistemas son de uso masivo, lo que tradicionalmente significan grandes sumas de inversión, especialmente para entidades de gobierno.

Como la industria de software ha evolucionado a una regida por estándares para asegurar la interoperabilidad, hoy es posible construir sistemas operativos, de oficina y muchos otros, que cumpliendo con las normas establecidas, pueden satisfacer un requerimiento, a valores notablemente menores que lo que hoy ofrecen los actores dominantes. Se está así modificando una industria que por ser muy agresiva en términos de concentración, ha acumulado, también, ineficiencias en su modelo de negocios, que se traducen en altos precios para los consumidores.

Si bien, el Software Libre (*Free Software*) tuvo su origen en la premisa de que todos tienen derecho a usar el software una vez creado, existen muchas discusiones filosóficas acerca del concepto de libertad o gratuidad asociada al nombre de Software Libre. No es nuestra intención entrar en las discusiones si debe ser Software Libre o Software de Código Abierto, que para algunos expertos en estos temas puedan tener un gran valor. Para efectos del presente documento usaremos el término *Software Abierto* para referenciar al software que se entrega sin costo de licencia, con derecho a copiarlo y, con código fuente que puede ser modificado

Para lograr alcanzar un mercado más competitivo, nos interesa que exista en Chile una sólida oferta de Software Abierto que, en la práctica, permita que tanto sistemas de infraestructura como sistemas operativos, puedan ser usados en forma masiva. Los costos de licencias se cambian ahora por inversión en servicios de soporte, significando en la mayoría de los casos una inversión notablemente menor.

A pesar de que Chile se ha declarado como un país “tecnológicamente neutro”, esta neutralidad sin los debidos incentivos, no es más que una declaración de intenciones. El software comercial licenciado está muy posicionado con una infraestructura madura de soporte y canales de comercialización existentes. Lo que han hecho los múltiples países, estados o provincias que están fomentando el uso del Software Abierto, es entregar los incentivos para permitir un mercado

competitivo. No se trata de generar leyes para obligar al uso del Software Abierto, tendencia que impulsan algunas naciones, pero si -al menos- obligar a que sean consideradas las distintas alternativas tecnológicas en una licitación gubernamental.

El Software Abierto es hoy una realidad en varios puntos del planeta. Lugares como el Estado de California, una de las diez principales economías del mundo -impulsada entre otros por el desarrollo de Silicon Valley-, ya declararon su preferencia por Software Abierto, con ahorros que superan los varios billones de dólares. En Chile una discusión formal al respecto esta pendiente.

La Fundación País Digital, con el objeto de aportar al análisis serio del tema a nivel nacional, ha invitado a diversos actores a participar en una mesa de discusión, que entre otros resultados, genera el presente documento. “Software Abierto: Factibilizando la Neutralidad Tecnológica para Chile” contiene los principales conceptos tecnológicos y legales, necesarios para esta estratégica discusión.

Chile tiene la obligación de optimizar sus recursos y lograr la mayor independencia tecnológica posible. La discusión que permita promover una oferta sólida de Software Abierto para lograr un mercado mucho más competitivo no puede esperar.

Fernando Prieto Domínguez
Vicepresidente
Fundación País Digital

INDICE

1.- Introducción.....	6
<i>El creciente impacto de las TICs.....</i>	6
<i>La Irrupción del Software Abierto</i>	7
<i>Neutralidad tecnológica y Software Abierto.....</i>	7
<i>La Agenda Digital 2004</i>	8
2.- Algunos Antecedentes Históricos.....	10
<i>Software y Hardware.....</i>	10
<i>La Historia de Microsoft</i>	11
<i>La Historia de Linux.....</i>	11
3.- Efectos Económicos.....	13
<i>Impactos Económicos del Software Abierto.....</i>	13
<i>Implicancias para nuestro país</i>	15
<i>Estrategias de Negocios basadas en Software Abierto.</i>	15
4.- Seguridad.....	18
<i>Los efectos del software y su incidencia en la seguridad</i>	18
<i>Disponibilidad de código</i>	18
<i>Análisis teórico</i>	18
<i>Datos cuantitativos</i>	20
<i>Virus, Gusanos, Troyanos y demás código malicioso</i>	20
5.- Legalidad.....	21
<i>Licenciamiento de Software Abierto</i>	21
<i>Tipos de Licenciamiento</i>	22
<i>Conclusiones</i>	23
6.- Interoperabilidad.....	24
<i>Los estándares de facto</i>	24
<i>Software y estándares abiertos</i>	24
<i>Interoperabilidad de aplicaciones</i>	25
7.- Bases de una Estrategia para Chile.....	26
<i>Desafíos Para La Adopción Del Software Abierto</i>	26
<i>Entendiendo el tema y sus implicancias</i>	27
<i>Algunas Medidas de Corto Plazo</i>	28
<i>Estrategias de Largo Plazo</i>	28

ANEXOS.....	29
Anexo 1.- Definiciones Básicas.....	30
Anexo 2.- Mitos y Realidades.....	33
Anexo 3.- Preguntas Frecuentes.....	40
Anexo 4.- Casos de Éxito.....	44
Anexo 5.- Algunas Empresas Chilenas en Software Abierto.....	48
REFERENCIAS.....	49

1.- INTRODUCCION

El creciente impacto de las TI

Nadie se atrevería a discutir acerca de la importancia que hoy tienen las llamadas “tecnologías de la información” (TI). Si hace un par de décadas estas dos palabras eran conocidas por una pequeña proporción de los habitantes, hoy están presentes en todos los rincones del planeta y “deambulan” por los más variados ámbitos de la vida.

Si existe un terreno donde las TI juegan un rol clave, ese es el de la competitividad y productividad. Los países que han logrado dar el tan ansiado salto, y mejorar sus estándares, lo han hecho fundamentalmente, porque han tomado conciencia de la vital importancia de las TI, y han optado por invertir fuertemente en ellas.

Al igual que en un círculo virtuoso, hemos sido testigos de cómo, mientras mayor inversión se realiza en este espectro, más importantes e imprescindibles se vuelven las tecnologías de la información.

Uno de los principales resultados que ha ocasionado esta apuesta masiva por la tecnología en los países que se han involucrado en esta “aventura” es que los costos que supone la adopción de las TI han disminuido considerablemente.

Por eso, es clave que aquellas naciones que hoy están jugando en las ligas mayores, como Chile - que en los últimos años ha firmado acuerdos de libre comercio con países y bloques de la talla de Estados Unidos, Corea del Sur y la Unión Europea-, sean capaces de ponerse a la altura de las nuevas exigencias. No es casualidad, que destinos como Tailandia, California o Munich, entre otros, han analizado acuciosamente el tema y han optado por declararse tecnológicamente neutrales.

La experiencia y los casos de éxito demuestran que la forma de reducir los costos de la tecnología no es infringiendo la ley ni amparando la “piratería”. Muy por el contrario, la “jugada maestra” debe apuntar a que, como el hardware es cada vez más barato, el software debe seguir esa misma dirección.

Si Chile quiere realmente masificar la tecnología hacia toda la población, el camino más viable es contar con más alternativas de software, lo que debiera significar reducciones de costo de estos productos. No se trata de abanderizarse por uno u otro proveedor, ni enfrentar el tema desde una perspectiva política, sino de aprovechar las oportunidades que nos presenta el libre mercado, con el único objetivo final de lograr beneficios para toda la comunidad.

Desde hace varios años, y en todo el mundo, se ha debatido sobre las ventajas que tiene el uso de Software Abierto en vez de Software Propietario. Se esgrimen razones técnicas, filosóficas y económicas, pero ¿por qué se sigue usando Software Propietario?, ¿es el Software Abierto realmente una opción para reemplazar a los ya posicionados Sistemas de Software Propietario?, ¿se puede usar en un Sistema Propietario herramientas abiertas?, ¿Pueden coexistir e interactuar ambos sistemas? y, más importante aún, ¿está Chile preparado para usarlo masivamente?

La irrupción del Software Abierto

Aún en la actualidad cuando se habla de “Internet”, lo primero que se viene a la mente es el World Wide Web y la publicitada burbuja de las .com. Sin embargo, Internet representa una revolución mucho más amplia y compleja, que está produciendo profundos cambios en la forma en que se comunican los seres humanos y en la forma en que se entienden los conceptos de tiempo y distancia.

El Software Abierto es sin duda, uno de los primeros resultados concretos y tangibles de la Internet. A pesar de que sus orígenes se remontan a bastantes años atrás, este es un modelo de desarrollo de software que sólo es posible en la medida de que se cuenta con la posibilidad de comunicarse y compartir información.

Si partimos de la base de esta natural asociación entre Software Abierto e Internet, se puede determinar con bastante claridad, las áreas donde este tipo de tecnologías constituyen alternativas reales y otras donde su aplicación pueda ser más compleja o menos adecuada.

En particular, en las aplicaciones basadas y orientadas a Internet, el Software Abierto a todas luces, una alternativa viable y madura. Ejemplos de esto son el servidor web Apache, que en la actualidad posee más del 67% del mercado¹, o que el 75% de los todos dominios del mundo utilicen Software Abierto para sus servicios de resolución de nombres (DNS)².

Por otra parte, existen otras áreas donde, si bien se han logrado grandes avances, sin duda existe mucho camino por recorrer. A pesar de esto, al momento de evaluar soluciones tecnológicas, se hace fundamental realizar un análisis detallado de las necesidades y posibilidad reales con que se cuenta. Sin duda, cuando existen ciertas limitaciones de presupuesto, el Software Abierto puede constituir una excelente alternativa, al poder liberar recursos destinados a licencias y orientarlos a servicios que tienden a mejorar la operación y experiencia del usuario.

El presente documento tiene por objetivo entregar una mirada general del Software Abierto, con énfasis en las áreas que han sido públicamente indicadas como de interés nacional. Esta presentación y análisis pretende centrarse en hechos más que en especulaciones o declaraciones de “buenas intenciones” y busca incorporar la mayor cantidad de fuentes, como forma de dar una visión completa del “estado del arte” del Software Abierto y dar una clara visión de la factibilidad de incorporar este tipo de tecnologías en nuestro país.

Neutralidad tecnológica y Software Abierto

Las máximas autoridades relacionadas con TIC's a nivel público, han definido la “neutralidad tecnológica” como el gran principio que rige y determina la forma en que se evalúan y toman las decisiones de TI en el Estado.

Sin embargo, para que efectivamente exista una neutralidad tecnológica -entendida como que la regulación no determine la preferencia de una tecnología sobre otra- resulta fundamental contar

¹ http://news.netcraft.com/archives/web_server_survey.html

² <http://mydns.bboy.net/survey/>

con un mercado competitivo, donde exista una amplia oferta de productos y servicios, de forma de poder realizar una evaluación considerando la mayor cantidad de alternativas posibles.

En esa línea, hoy en día en Chile, si bien existe una cierta oferta de productos y servicios basados en Software Abierto, no tiene ninguna comparación en tamaño y número de empresas, con las que se basan en sistemas propietarios.

Este hecho, deja en evidencia la urgente necesidad de incorporar el Software Abierto a la discusión tecnológica nacional, de forma de hacer realmente efectivo el principio de “neutralidad tecnológica”. Esto puesto, que en el mundo globalizado en que vivimos, no nos podemos permitir como país correr el riesgo de no utilizar tecnologías que en ciertos ámbitos ya son ampliamente utilizadas a nivel mundial y posean una conveniente relación costo/beneficio.

Esto, principalmente, porque la utilización de tecnologías de mayor costo, puede implicar una amenaza a la competitividad de nuestras empresas, frente a países que si utilicen estas tecnologías más económicas e igualmente productivas y, una fuerte limitación a la necesaria masificación de las TI en el país.

La Agenda Digital 2004

Partiendo de la base de las medidas e iniciativas enunciadas en la propia Agenda, queda de manifiesto de que existen varias áreas donde el Software Abierto puede constituir una alternativa conveniente, que debe ser al menos considerada y evaluada adecuadamente.

En lo referente a “Masificación del Acceso”, sin duda existen un importante número de aplicaciones de Software Abierto, que podrían utilizarse. Por ejemplo, en lo relacionado a servidores orientados a prestar servicios a Internet, se complementa con una serie de programas que permiten la construcción de aplicaciones en variados lenguajes (como PHP, Perl, Java, etc), e inclusive está disponible bajo Software Abierto.

Por otra parte, servicios relacionados con comunicaciones, como servidores de correo electrónico y servidores Proxy³, poseen aplicaciones ampliamente usadas a nivel mundial y con un alto nivel de madurez. A su vez, en lo referente a seguridad, existen variadas aplicaciones de seguridad para diversos usos, que van desde la protección de redes -conocidas como firewalls-, hasta la detección de intrusos en la red (IDS/IPS), pasando por herramientas para combatir el SPAM.

Sin duda, la seguridad constituye un elemento clave al momento de evaluar soluciones, considerando el importante aumento que han tenido los virus computacionales y el SPAM a través de Internet. Por lo tanto, en la actualidad resulta prácticamente imprescindible contar con soluciones adecuadas de seguridad, que sin embargo, en el caso de las soluciones propietarias poseen un alto costo.

Desde el punto de vista de los usuarios finales de Internet, existe una amplia disponibilidad de aplicaciones bajo Software Abierto para satisfacer las distintas necesidades de los usuarios. Dentro de esto se incluyen navegadores de Internet, lectores de correo electrónico, aplicaciones de

³ Orientados a hacer más eficiente y controlada la conexión a Internet

mensajería instantánea, chat, entre otros.

En el contexto de los usuarios finales, se debe destacar los altos niveles de seguridad y estabilidad de los sistemas operativos de Software Abierto, como por ejemplo Linux. Esto tiene importantes implicancias prácticas como el hecho de no necesitar utilizar programas antivirus -que tienen un costo considerable y consumen altos niveles de los recursos del PC- y que hoy en día constituyen aplicaciones sin las cuales se hace difícil operar en los sistemas operativos propietarios.

En lo relacionado con “Educación y Capacitación”, el Software Abierto podría constituir una excelente alternativa al menos en dos áreas: por una parte, existe una gran cantidad de programas y aplicaciones orientados a la infraestructura tecnológica necesaria, por ejemplo, para la conexión a Internet, navegación, utilización de correos electrónicos, entre otros.

Por otra parte, existe una interesante oferta de aplicaciones de uso educativo disponibles bajo Software Abierto y orientadas a operar vía Internet. Este tipo de aplicaciones, complementadas con otras tecnologías ampliamente utilizadas en Internet, como son los blogs o wikis, generan un ambiente propio para la comunicación y cooperación a través de la red, que sin lugar a duda, será una habilidad clave para el futuro.

Respecto al “Estado en línea”, existen variadas áreas donde el Software Abierto puede constituir una alternativa. Un elemento que podría resultar clave, lo constituye la generación de políticas que fomenten la utilización de modelos de licenciamiento similares a los del Software Abierto, de forma de que sea posible acelerar el proceso de incorporación de tecnologías y reducir los costos, por la vía de la reutilización de aplicaciones ya desarrolladas para otros estamentos del gobierno o la definición de ciertos “frameworks” sobre los cuales se construyan un número importante de aplicaciones.

Este modelo de trabajo basado en la cooperación, trae consigo otros beneficios, como la disminución de los costos de mantención de las aplicaciones en el tiempo, al compartir los gastos asociados entre varias instituciones. Por otra parte, se facilita la integración de los sistemas, puesto que utilizan estructuras internas y subsistemas similares.

Finalmente no hay que dejar fuera las oportunidades de negocios que surgen a partir del Software Abierto. Sin duda, si deseamos tener una industria de TIC’s competitiva a nivel mundial, el Software Abierto puede constituir un interesante esquema de trabajo, puesto que posee elementos estructurales claves, como fomentar la cooperación, estar orientado a Internet, privilegiar los estándares abiertos, y por sobre todo, adaptado a los constantes y frecuentes cambios del mundo de hoy.

2.- ALGUNOS ANTECEDENTES HISTORICOS

Software y Hardware

Software, Hardware y Computación no son términos originales del mundo de la Electrónica, sino que existen desde tiempos ancestrales. Básicamente todo instrumento es hardware y los procedimientos e instrucciones para usarlo, software. Por ejemplo, podríamos decir que un violín es hardware y la partitura su software.

Mirándolo desde este punto de vista, todo software tiene autor o autores, a los cuales se reconoce por ello, si es que son conocidos. Algunos son patrimonio de la humanidad, así como el conocimiento científico, y otros están restringidos debido a deseos de sus inventores o creadores. Y es sobre el conocimiento público que se construye nuevo conocimiento gracias a lo cual la sociedad va creciendo y perfeccionándose. En este documento se usa hardware y software dentro del contexto de la electrónica.

En un principio, el software era desarrollado por quienes diseñaban y construían el hardware, por lo cuál el software era distribuido gratuitamente con él. A medida que aumentaba la cantidad de computadores en el mundo, el software era de dominio público y se beneficiaba del trabajo conjunto entre las empresas de hardware y las universidades.

Los comienzos del fenómeno que cambiaría el negocio del desarrollo del software se remontan a 1965, cuando los Laboratorios Bell de AT&T financian el desarrollo de un sistema operativo experimental denominado Multics. Pero después de cuatro años de financiamiento AT&T retira los fondos por los problemas de rendimiento del sistema operativo.

Este equipo de ingenieros se dedica a desarrollar un juego, y para abaratar los altos costos de arrendar un supercomputador de la época, crean un sistema operativo bautizado como Unics⁴- en burla del fallido proyecto de AT&T- para así utilizar equipos más baratos. Este hito marca el nacimiento de UNIX.

Para 1980, la Oficina de Investigación Avanzada de Proyectos para la Defensa de Estados Unidos (DARPA) adjudicó a Berkeley⁵ el desarrollo de un sistema de comunicaciones, conocido como TCP/IP, para la red que uniría a los principales centros de investigación del país. Con lo que financia la cuarta versión de BSD Unix, base de la Internet actual.

Al igual SUN y Silicon Graphics, en 1984, AT&T decide comercializar Unix, trabajo realizado en conjunto con Berkeley, exigiendo el pago de derivados del desarrollo en los Laboratorios Bell. Esto genera gran competencia en el mercado del desarrollo de software, por lo que para diferenciarse las empresas buscan contratar a los mejores programadores y a los obligaban firmar contrato de exclusividad y confidencialidad conllevando al termino del desarrollo colaborativo.

Para ese entonces, Richard Stallman del Instituto Tecnológico de Massachussets (MIT), se negó a firmar un contrato que a su juicio “atentaba contra el derecho de los programadores para compartir lo que aprenden” y funda la *Free Software Foundation* (FSF) con el objeto de evitar lo ocurrido

⁴ Sistema operativo simple, modular y capaz de correr en múltiples plataformas.

⁵ En 1977, la Universidad de Berkeley crea un versión de Unix llamado **Berkeley Software Distribution** (BSD Unix).

con el Unix de AT&T, creando una licencia cuya restricción más importante es que no puede ser restringida, conocida como **General Public Licence (GPL)**. El trasfondo de esta acción fue crear un sistema que jamás pudiese ser *cerrado* con licencias comerciales y que gracias al trabajo de comunidades de todo el mundo reescribiesen, bajo GPL, programas que ya existían, agregando también funcionalidades, reduciendo la complejidad y aumentando significativamente su calidad, gracias a la discusión e intercambio abierto de ideas.

Desde 1986, GNU⁶ trabajó en el *kernel* TRIX, desarrollado por la universidad Carnegie-Mellon (CMU) y el *microkernel* Mach, en su búsqueda por un sistema operativo totalmente libre. Fue así como para 1990, GNU se comprometió con su *kernel* **Hurd**, que debido a su revolucionario diseño estaba lejos de poder ser usado realmente a corto plazo.

La Historia de Microsoft

Microsoft también tiene sus comienzos en estos momentos de la historia. En 1975, un joven programador llamado William (Bill) Gates convence a su amigo Paul Allen de fundar una empresa llamada Microsoft con el objetivo de desarrollar un intérprete de BASIC para el primer computador personal, Altair 8800. Para llevar a cabo dicho desarrollo, le compraron los derechos a la empresa fabricante del hardware (MITS). Dentro de los clientes de BASIC estaban Apple, Commodore y Tandy, entre otros fabricantes de computadores personales. Para 1977, Gates y Allen lanzan Fortran y en un corto plazo, los intérpretes de BASIC para los procesadores Intel (8080).

Por su trabajo, IBM los contrata en 1980 para desarrollar el sistema operativo para su IBM-PC que saldría a la venta el año siguiente. Al verse enfrentado a tan corto plazo para un nuevo desarrollo, Microsoft le compra a Tim Paterson su QDOS (Quick and Dirty Operative System), por sólo US\$50.000, y luego lo revende sin exclusividad a IBM bajo el nombre de MS-DOS. Para 1984, ya habían vendido el sistema operativo a otros 200 fabricantes de computadores personales.

Durante este tiempo, Microsoft se dedicó a elaborar su planilla de cálculo para DOS llamada Multiplan y programas de oficina para Macintosh (Word, Excel y Works).

En 1985, Microsoft lanza Windows, una interfaz gráfica para DOS, y para 1987 lanza la segunda versión de éste. Pero es en 1990 donde Windows 3.0 vio la luz, ya que venía preinstalado en la mayor parte de los computadores personales coronándose como la empresa de Software más importante del mundo, lugar que ostenta hasta hoy.

La Historia de Linux

A pesar del éxito comercial de UNIX y de su aceptación como sistema operativo, su código fuente no podía ser utilizado en las salas de clases de las universidades, de modo que el desarrollo de sistemas operativos volvía a ser una ciencia restringida a un reducido grupo de empresas y personas.

⁶ GNU: acrónimo para “GNU’s Not Unix”; comunidad de programadores del FSF

Ante esta situación, el profesor Andrew Tanenbaum, de la Universidad de Vrije, en Amsterdam, decidió imitar a Ken Thompson cuando escribió el código de UNIX basándose en MULTICS, e inspirándose en UNIX desarrolló un nuevo sistema operativo mucho más reducido, al que llamó MINIX (de Mini-UNIX) cuyo código era completamente libre.

En 1990, Linus Torvals, un estudiante de 23 años de la Universidad de Helsinki, en Finlandia, comenzó a desarrollar, como hobby, un proyecto basado en el MINIX de Andrew Tanenbaum. Quería usar sobre un computador con procesador 386 (Intel 80386) un sistema operativo tipo UNIX que ofreciese más capacidades que el limitado MINIX, el cual solo se utilizaba para enseñar una cierta filosofía de diseño.

Después de *jugar* un poco, empezó a tomar en serio la idea de hacer un "MINIX mejor que MINIX". Su resultado, la primera versión 0.01 no tenía soporte para disquetera, y era de carácter privado. Tenía muchos errores pero *funcionaba*.

Para octubre de 1991, anunció la primera versión "oficial" de LINUX (0.02), la que era capaz de ejecutar los programas básicos de GNU. A partir de ahí, este sistema empezó a crecer y de todas partes del mundo, le llegaban cartas interesándose por la idea con lo que comenzaron a desarrollarse proyectos destinados a incrementar la potencia de la plataforma. Varios componentes de LINUX fueron *inspirados* por otro sistema UNIX de libre distribución llamado FreeBSD, desarrollado en la Universidad de Berkeley.

Las comunidades rápidamente empezaron a desarrollar sistemas operativos (distribuciones) GNU/Linux, es decir que integran el kernel Linux y herramientas GNU, de las cuales hoy existen miles. Hay para todas las necesidades y para todos los gustos.

Actualmente, ningún experto ni académico pondría en duda la calidad, estabilidad y robustez del kernel Linux, y muchas empresas grandes que mantenían sus propios sistemas UNIX han migrado ya a Linux. Entre ellas se destaca IBM donde AIX fue reemplazado totalmente por Linux y AT&T (dueño original de UNIX) que migró toda su infraestructura a Linux.

3.- EFECTOS ECONÓMICOS

Si bien los proveedores afectados por el creciente uso del Software Abierto esgrimen que esta tendencia tiende a destruir el valor de la propiedad intelectual, la verdad es que estamos frente a un cambio del modelo económico del software, que perfecciona el mercado eliminando distorsiones mantenidas por la enorme predominancia de estos proveedores. Este cambio es similar al que vivieron hace un par de décadas los entonces actores predominantes de la industria informática, proveedores de grandes equipos centrales propietarios, frente a la irrupción del sistema operativo Unix que hizo bajar drásticamente los valores de los equipos centrales, prácticamente eliminando las barreras de salida para migrar las aplicaciones de un proveedor a otro.

El Software Abierto, por variadas razones, representa una nueva mirada en la forma en que se entienden el negocio de las TIC. Esto, porque además de establecer un régimen de amplios privilegios para los usuarios de este tipo de soluciones (mediante el licenciamiento Open Source), posee un proceso de desarrollo de software completamente distinto (conocido como "bazaar") a los esquemas tradicionales de la industria (también conocidos como "cathedral").

Por esta razón, es muy probable de que el Software Abierto no deje a nadie indiferente. Para algunos constituirá una nueva alternativa al momento de buscar soluciones tecnológicas o crear nuevos negocios que utilicen TIC's. Para otros determinará que surjan nuevos actores que compitan con las soluciones ya establecidas o con empresas operando bajo el modelo del software propietario.

Las empresas que se han iniciado en el Software Abierto, han utilizado algunas estrategias relacionadas con las características inherentes al Software Abierto, las que conllevan efectos económicos en las soluciones tecnológicas, que describimos a continuación.

Impactos Económicos del Software Abierto

Junto con incorporar un cambio relevante al modelo económico del software, en lo específico el Software Abierto impacta diversos aspectos relacionados con la implementación de soluciones tecnológicas. A continuación se entregan algunos ejemplos:

a) Hardware, Sistemas Operativos y Herramientas.

El Software Abierto es capaz de operar en una amplia variedad de plataformas de hardware, ampliando la gama de opciones al momento de determinar la arquitectura de la solución. Por otra parte, en muchas ocasiones el Software Abierto permite operar en hardware más antiguo, debido a menores requerimientos de recursos.

Los Sistemas Operativos propietarios usualmente cuestan cientos o miles de dólares. Linux, el más popular sistema operativo de Software Abierto, puede obtenerse, duplicarse e instalarse sin pagar ningún tipo de licencia.

Cuando las grandes organizaciones desarrollan herramientas propias, con el pasar del tiempo, resulta compleja su actualización y soporte y deben tomarse medidas para resguardar su

código fuente. Con Software Abierto, es probable que ya existan herramientas con funcionalidades muy cercanas a lo requerido, con su debida evolución y soporte

b) Actualizaciones y Licenciamiento.

En el Software Abierto, las actualizaciones en los sistemas se realizan en los términos de dicho usuario, en función de los plazos y tiempos que este estime conveniente. Prácticamente no existe el peligro o la presión de perder soporte por parte del proveedor, y por encontrarse disponible el código fuente, siempre existen opciones para continuar con el soporte.

Desde el punto de vista de licenciamiento, las grandes corporaciones, a pesar de negociar con cierta fuerza, gastan significativas sumas de dinero tan solo en licencias de infraestructura como sistemas operativos o sistemas de oficinas. Al utilizar Software Abierto se elimina este problema. Existen sin embargo, servicios relacionados por los cuales se puede pagar, como soporte, capacitación y mantención.

c) Capacitación y Soporte.

La Capacitación es similar al caso del Software Propietario e incluso en la actualidad existe una limitada capacidad de gente preparada en soluciones de Software Abierto. Sin embargo, la utilización creciente de plataformas y estándares abiertos implica que debería existir en el tiempo una mayor disponibilidad de recursos preparados para capacitar a usuarios en general. Los equipos internos de una institución por su parte, podrán capacitarse para construir a partir de su código fuente.

Cualquier empresa calificada puede proveer soporte sobre Software Abierto ya que el código de los programas está disponible para ser analizado. En el caso del software cerrado, sólo la empresa creadora del software puede realizar soporte ya que solo ella conoce el código de los programas. Esto hace que el soporte sea un punto de riesgo para el usuario ya que el proveedor al liberar nuevas versiones, puede decidir dejar de entregar soporte a versiones anteriores, obligando a la actualización, con los consiguientes costos económicos. Además, el hecho de que no exista un significativo número de proveedores que compitan entre sí, limita la competencia en calidad y costo del servicio.

d) Costo de *Downtime*.

En el mundo del software, la disponibilidad de servicio no está garantizada. En general se negocian contratos para minimizar la posibilidad de que ocurra una falla grave. En el ambiente del software propietario generalmente se está a merced de un único proveedor. Si el proveedor no puede hacer los cambios en el tiempo que es requerido por el cliente, no existen alternativas de solución. En el caso del Software Abierto existe, además de servicios contratados o equipos internos, una amplia comunidad que puede ser eficiente resolviendo dichos problemas, que también puede tener acceso al código fuente.

Implicancias para nuestro país

Llevado al contexto de nuestro país, resulta altamente probable lograr importantes ahorros mediante la utilización de Software Abierto. Esto, por una parte, a que en muchos casos se requiere de un conjunto de aplicaciones acotado (navegador web, correo electrónico, herramientas de productividad y poco más). En la actualidad existen una serie de soluciones de Software Abierto para estas necesidades que compiten seriamente con las soluciones propietarias, en seguridad, usabilidad, estabilidad y desempeño, como es el caso de Mozilla Firefox e Internet Explorer.

Por otra parte, un estudio de Gartner de 2003 (1), señala que las empresas que utilicen Linux como sistema operativo, obtendrán al menos un ahorro de US\$80 dólares en la adquisición por concepto de hardware y un promedio de US\$74 dólares por usuario y por año, por la utilización de OpenOffice en vez de Microsoft Office.

A partir de este cálculo, que es bastante conservador puesto que no considera otros elementos de los costos de una solución tecnológica, que se verían impactados por la utilización de Software Abierto, ya da un lineamiento claro de que el Software Abierto es una alternativa real y que los ahorros que se pueden obtener son concretos y significativos.

Si lo vemos en la práctica, una licencia de Microsoft Windows XP Home Edition cuesta hoy US\$200 y XP Professional US\$280 (amazon) y una de Office XP US\$579. A esto podríamos agregarle los costos de antivirus además de aplicaciones de productividad (gráficos como Photoshop, diseño técnico o de multimedia). En Software Abierto podemos encontrar soluciones bastante competentes para todas estas necesidades sin costo por concepto de licenciamientos. Un computador personal se puede conseguir desde aproximadamente US\$400 con una clara tendencia a la baja en el tiempo. Así, el valor en software puede llegar a representar casi el doble del valor del equipo.

Por este mismo motivo existen diversas estrategias comerciales fomentadas por los integradores de sistemas o por los mismos fabricantes de hardware que, en alianza con los proveedores de software, empaquetan la solución, maximizando su retorno a través de comisiones cruzadas. Crecientemente dado el costo del software, existe a nivel de personas y Pymes un aumento de la “piratería”. Si bien esta “solución” permite ahorrar costos, es ilegal y deja a quienes la usan en situación de riesgo. Por los tratados de libre comercio y por la mayor tecnología usada por los proveedores de software para mantención y control de versiones, la piratería será controlada e forma cada vez más eficiente.

Las empresas e instituciones grandes por su parte, están obligadas a usar formalmente - y por lo tanto a pagar- el software, aunque en condiciones más favorables que las personas y Pymes dados sus volúmenes.

Estrategias de Negocios basadas en Software Abierto

En la actualidad existen numerosas empresas que han logrado generar negocios exitosos en torno a soluciones de Software Abierto. Como se mencionó al inicio de este capítulo, el Software Abierto agrega una dosis de competitividad que puede resultar clave para la industria TI y de las empresas

en general, facilitando la adopción masiva de tecnologías de información.

Las principales estrategias que han adoptado las empresas que hoy se consideran "exitosas" y basan su negocio en Software Abierto son las siguientes:

a) Coexistencia de Software Abierto y Propietario

Consiste en la combinación de software propietario con Software Abierto. Este último reemplazando a productos de software que se ha transformado en un "commodity", como sistemas operativos o sistemas de oficinas. Existen ejemplos exitosos, como Oracle y SAP donde se han complementado soluciones de Software Abierto con software propietario o servicios, de forma de generar una oferta que mantiene atractivos márgenes, pero con un costo menor producto de la utilización de Software Abierto en el *stack* de la solución.

b) Licenciamiento Dual:

En este caso está disponible un software bajo una licencia de Software Abierto y a su vez, se ofrece su opción de licencia, bajo un modelo propietario. Con esto es posible adaptarse a las distintas necesidades de los clientes, que en ciertos casos, deciden pagar las licencias, a cambio de no tener que dejar disponibles las mejoras, puesto que pueden ser claves para su negocio.

c) Servicios de Consultoría:

En la actualidad, aproximadamente un 30% de los costos de una solución tecnológica están relacionados con licencias de software, mientras que un 70% se destina a implementación. Por lo tanto, realizar consultorías de implementación de soluciones utilizando Software Abierto, puede ser una alternativa de negocio interesante, puesto que los costos de implementación son equivalentes. La solución completa tiende a tener un costo menor debido a la disminución del costo de las licencias de software.

d) Servicios de Soporte

Por no incurrir en costos de licencias, las soluciones tecnológicas basadas en Software Abierto tienen un costo inicial menor. Sin embargo, en el tiempo, existe la conveniencia de contratar servicios de soporte y mantención para asegurar la continuidad y vida útil de la solución. Empresas como Redhat y SuSE son ejemplos de este modelo de negocio exitosos donde se han obtenido utilidades por servicios de suscripción.

e) Apoyo Corporativo.

Importantes proveedores tecnológicos están patrocinando soluciones de Software Abierto con el objetivo de incentivarlos como estándares. Esto puede causar un efecto de "comotidización" en ciertos mercados, como lo que ha hecho IBM con Linux. Si bien esta estrategia no representa un beneficio directo, apunta a posicionar al proveedor en el tema y desgastar a competidores.

f) Software como Servicio y como Plataforma de Negocios.

Desde hace un tiempo existe una tendencia dentro del mercado del software de proveerlo en modalidad de "arrendamiento" más que venderlo. Dentro de este modelo, la utilización de Software Abierto, como base para la prestación de dichos servicios, constituye una tremenda oportunidad de negocio. Es así, como empresas como Salesforce.com y otras, más que instalar su solución en el cliente, le entregan por un cargo mensual, el servicio que el cliente necesita. Por otra parte, compañías como Google y Amazon, han hecho fortunas en modelos de negocio que utilizan como plataforma una amplia gama de soluciones y herramientas de Software Abierto.

g) Sistemas "*Embedded*" o Incrustados.

El software está crecientemente presente en múltiples dispositivos y aparatos electrónicos. Existen hoy en este campo, muchas soluciones que se están desarrollando en plataformas abiertas. En la actualidad, productos de consumo sumamente exitosos en el mundo de la entretenimiento, como *TiVo Box*, utilizan este sistema operativo.

Resulta importante este tipo de ejemplos, para mostrar que el Software Abierto no erosiona el mercado de la TIC's, sino que lo modifica e incluso amplía las oportunidades de negocio para las empresas que están en el rubro. Entrar ya en esta creciente tendencia, puede abrir nuevas e interesantes oportunidades para la alicaída industria TIC nacional.

4.- SEGURIDAD

En estos días, el tema de la seguridad es de importancia creciente en muchos ámbitos. En informática no es la excepción. En todo sistema computacional, es un elemento clave garantizar que los datos son seguros, no alterables ni con posibilidades de ser copiados sin las debidas autorizaciones. La información almacenada digitalmente puede llegar a tener altísimos valores incluyendo desde listados de clientes hasta transacciones financieras de alto monto. Existen en computación sofisticados mecanismos de resguardo y auditorías para prevenir este tipo de delitos.

Adicional a lo anterior, en los últimos años han proliferado los ataques masivos provocados por virus computacionales. Estos ataques son difíciles de prevenir, de naturaleza sorpresiva. Los proveedores de software propietario están gastando enormes sumas de dinero para tratar de mantener limitados los eventuales daños masivos provocados por este tipo de situaciones.

Los defectos del software y su incidencia en la seguridad

En el diseño de un sistema se define cómo deben interactuar todas las partes para obtener el resultado deseado. En sistemas complejos, es sumamente difícil no cometer errores al momento de construirlos. Por ello, se supone que todo sistema computacional tiene un número de errores, los cuales se intentará enmendar a futuro. Esto implica que un sistema debe mantenerse actualizado para disminuir al máximo el potencial de que algún error afecte la seguridad del sistema y genere un potencial problema de esta naturaleza.

En algunas ocasiones, un error en una aplicación puede llevar a un atacante a tener acceso a elementos que por configuración no le corresponden. Por ejemplo, modificación de la configuración del equipo, acceso a leer o escribir datos para los cuales no tiene autorización, entre otros.

Disponibilidad de código

Hay dos razones por las que se puede pensar que la disponibilidad de código fuente puede afectar negativamente la seguridad de un sistema. Por un lado, se podría pensar que la disponibilidad de código fuente hace más fácil encontrar vulnerabilidades que permitan aprovecharse del sistema, y por otro, al modificar el código se podría introducir de manera intencional o involuntaria nuevas vulnerabilidades que se difundirán ampliamente entre todos los usuarios de estos sistemas. Sin embargo, en la práctica esos supuestos no son valederos.

Análisis teórico

De diversas investigaciones se desprende que si hay una diferencia con respecto a la resolución de errores, el Software Abierto en conjunto con la metodología de desarrollo tipo bazar es la que se ve como más segura frente a otros desarrollos. Al tener código fuente disponible y el permiso para modificarlo y compilarlo, es más fácil encontrar errores que si se tiene que hacer el mismo trabajo sin acceso al código fuente, ya que en los desarrollos de tipo bazar se conjugan los siguientes

elementos:

- Existe un alto número de desarrolladores que revisan el código en comparación con desarrollos centralizados. Esto permite que se cumpla el principio de Linus: *Given enough eyeballs, all bugs are shallow*, es decir, si existen suficientes personas viendo el código, para alguno va a ser obvia la solución a un error dado.
- Según el principio de Kerkhoffs, se debe asumir que el atacante conoce a fondo el funcionamiento del sistema. Por lo tanto, cualquier sistema que base su seguridad en esconder la forma en la que opera, será igualmente vulnerable. Este principio es la base de la criptografía moderna y se puede aplicar a sistemas generales. Averiguar el funcionamiento de un sistema puede dificultarse, con efectos colaterales, pero no se puede confiar en que no se puede llegar a conocer. Es importante notar que la poca legibilidad del código compilado no es intencional, sino una consecuencia de un proceso de traducción, por lo cual jamás fue diseñado para evitar conocer el funcionamiento de un sistema.
- Dado que es posible encontrar con mayor facilidad los errores, existen más oportunidades de arreglarlos a tiempo y así disminuir la tasa de falla.
- En un proyecto de Software Abierto existen múltiples revisiones por parte de personal responsable, por lo que no se aceptan contribuciones que sean poco entendibles o no se adhieran a las prácticas de programación acordadas. Esto hace que el nivel del código que se encuentra en Software Abierto sea en general de muy alta calidad, y es virtualmente imposible que alguien agregue código sin que pase por exhaustivas revisiones por parte de pares.

Esconder el código fuente puede jugar a favor y en contra. Podría resultar más difícil encontrar errores en el programa siendo menos probable que se puedan explotar esos errores para abusar del sistema. Por otra parte, al esconder el código fuente, resultará también más difícil corregir eventuales errores, ya que es mayor la dificultad de encontrarlos.

Por otro lado, en muchas aplicaciones propietarias existen los llamados "huevos de pascua" o *easter-eggs*. Estos *easter-eggs* corresponden a partes de un programa que no tienen ninguna relación con el objetivo de ese programa, como por ejemplo un juego de laberintos dentro de una planilla de cálculos. Estos *easter-eggs* pueden estar incluidos dentro de programas durante varios años antes que sean de conocimiento público, lo cual es evidencia de la dificultad de revisar el correcto funcionamiento de un programa sin contar con su código fuente. En el caso de Software Abierto, código que no corresponde a la aplicación puede ser detectado con mucha más facilidad, y más importante aún, pueden ser eliminados sin problemas. Lo mismo aplica para funcionalidades que malevosamente actúan de manera diferente a la que se da a entender al usuario.

En el caso de software desarrollado a medida, los usuarios internos pueden tener acceso al código fuente no así terceros externos, por lo cual no hay suficiente revisión del código. Si se considera además que entre el 30% y 50% de los ataques informáticos son causados por empleados contra sus empleadores, el riesgo de un ataque no necesariamente disminuye al evitar acceso al código fuente a desarrolladores externos.

Datos cuantitativos

De acuerdo a estudios hechos por la empresa Reasoning⁷, Inc. sobre el kernel TCP/IP de Linux y la base de datos MySQL⁸, el código de los proyectos de Software Abierto no tienen nada que envidiarle a proyectos propietarios en sus respectivas áreas.

En el estudio hecho al *stack* TCP/IP, protocolo básico en Internet, se compararon los *stacks* TCP/IP de cinco sistemas propietarios, cuatro de los cuales tienen una madurez mayor a diez años. El resultado fue que en Linux la densidad de errores encontrados fue cinco veces menor que en el promedio de los proyectos propietarios.

De manera similar, en MySQL la densidad de errores encontrados fue seis veces menor a la densidad de errores encontrados en desarrollos propietarios similares. Además, en el caso de MySQL fue posible verificar un descenso en la densidad de errores a medida que el producto ha ido madurando, mientras que en los sistemas propietarios analizados no se pudo verificar dicha relación.

Evidentemente estos resultados no son extrapolables a cualquier desarrollo de Software Abierto, pero sí deja en evidencia que es posible que proyectos de Software Abierto superen en seguridad a sus pares propietarios.

Virus, Gusanos, Troyanos y demás código malicioso

Teóricamente, es posible que existan virus y otros tipos de programas destructivos que funcionan en Linux. Sin embargo, hasta ahora los únicos virus que se conocen para Linux son aquellos desarrollados como forma de demostrar esa posibilidad, y no existen aún casos donde se haya logrado un contagio masivo de sistemas. Eso se puede deber a que todavía existe menor cantidad y heterogeneidad de los sistemas, con lo cual se limita el número de posibles blancos a atacar.

Pero por otro lado, la seguridad de Linux, basada en los sistemas Unix permite contener este tipo de ataques de mejor manera y dejar el problema confinado, por ejemplo al usuario que lo gatillo, y no a todo el sistema.

Se puede concluir entonces, que la seguridad no es una debilidad inherente del Software Abierto, sino al contrario, por ahora representa una ventaja. Si bien no se garantiza que un Software Abierto va a ser más seguro que un software propietario, y que existen sistemas propietarios de gran calidad y probada seguridad, hay características inherentes al modelo de desarrollo que se traducen en ventajas de seguridad.

⁷ Reasoning Inc. (2003) <http://www.reasoning.com/downloads/opensource.html>

⁸ Reasoning Inc. (2003) <http://www.reasoning.com/downloads/mysql.html>

5. LEGALIDAD

Licenciamiento de Software Abierto

Un programa no puede ser usado ni distribuido por nadie salvo por el titular del derecho de autor, a menos que se cuente con un permiso para ello. Por lo general, dicho permiso se da a través de una licencia que es la única forma de tener derecho a usar el software. En la licencia se especifica qué derechos se le confieren al usuario del software, y qué deberes se le exigen a cambio.

“*Free software*” o “*software libre*” es aquél cuyo desarrollador franquea el acceso al código fuente, a fin de que el usuario pueda modificarlo con miras a adecuarlo a sus propias necesidades, vale decir, le confiere libertad para obrar a su respecto. En consecuencia, ello no implica que el software sea gratuito o se encuentre exento de costo alguno, sino tan solo que el usuario dispondrá de ciertas libertades esenciales para conservar el control respecto de él.⁹ Como se mencionó, la principal entidad que promueve el software libre es la Free Software Foundation (FSF), fundada por Richard Stallman en 1984, cuyo proyecto GNU gestiona diversas licencias de este tipo, entre las cuales destacan GNU Public License (GPL), la Lesser GNU Public License (LGPL) y la GNU Free Documentation License (GFDL).

Por su parte, “*open source*” o “*software de código abierto*”, que hemos denominado en este documento como “*Software Abierto*” es aquél cuyo desarrollador ha permitido acceder al código fuente, a efectos de permitir la adopción de soluciones técnicas apropiadas. La principal entidad promotora de este tipo de software es la Open Source Initiative (OSI) creada en 1998, que ha renegado el concepto software libre por estimar que el error a que conduce su empleo –dada la ambigüedad conceptual entre gratuidad y libertad– ha provocado una merma ante la comunidad informática. Entre las licencias que certifica la OSI está la Artistic License.

En la práctica, como podrá apreciarse, ambos suponen brindar acceso al código fuente, aun cuando para unos ello es imprescindible para garantizar nuestras libertades y para otros ello sea resultado de la solución técnica más apropiada. En los hechos, se oponen a la solución suministrada generalmente por los proveedores de “software propietario” que hacen entrega al usuario de los archivos ejecutables sin posibilidad de acceder, menos aún modificar, el código fuente.

Es habitual que las licencias contemplen cláusulas de exclusión de responsabilidad y denegación de garantía, las que, no obstante, admiten que el desarrollador y/o distribuidor final asuma responsabilidad y brinde garantías al usuario, con tal que ellas no afecten al desarrollador inicial o colaboradores. Excepcionalmente, algunas licencias prevén cláusulas de prórroga de jurisdicción, sumisión normativa, convenciones probatorias o exención de responsabilidad, algunas de ellas contrarias a nuestro derecho. Sin embargo, es habitual que las mismas licencias admitan la posibilidad de prescindir de tales cláusulas.

Junto a las condiciones comunes, imprescindibles para la propia configuración de un software como de código abierto, cada entidad puede sumar obligaciones diversas a los licenciarios; de hecho, actualmente existe sobre un centenar de prácticas de licenciamiento, siendo las más frecuentes de ellas: GPL, bajo cuyos términos se desarrolla sobre el 70% del Software Abierto;

⁹ Con afán de graficar sus efectos se sostiene que “*free as in a free speech, not as in free beer*”, dado el sentido ambivalente del vocablo free en la lengua inglesa.

LGPL, con un 10,8% de desarrollos libres licenciados bajo sus términos; y, un 7,1% de software se encuentra licenciado bajo los términos de la Berkeley System Distribution (BSD).¹⁰ A tales licencias, pueden adicionarse la MIT License (MIT), la Artistic License, la Sun Public License (SPL) y la Mozilla Public License (MPL).

Tipos de Licenciamiento

Según la mayor o menor libertad que las licencias de software de código abierto confieren a los usuarios para servirse del código fuente, se distinguen tres categorías de ellas: las permisivas, las persistentes y las intermedias.

a) Licencias permisivas

Se caracterizan porque admiten que los usuarios puedan servirse íntegramente del software brindando acceso al código fuente del mismo; sin embargo, no es necesario que el desarrollo efectuado a partir de tal software quede bajo la condición de código abierto; en otros términos, no es necesario que el software resultante conserve su condición inicial. En esta condición se encuentra la Berkeley System Distribution (BSD) y la MIT License (MIT), que, en lo sustancial, se ciñe a los términos de la primera.

Este tipo de licencias no ha alcanzado el arraigo de otras, fundamentalmente porque se ve en ellas un efecto perverso: al admitir que el resultado obtenido a partir del acceso al código fuente no quede bajo la condición de código abierto, están permitiendo el desarrollo de software propietario sobre la base del esfuerzo colectivo de las comunidades informáticas.

b) Licencias persistentes

Entre éstas se encuentran las administradas por la FSF de Richard Stallman, a saber, la GPL y la LGPL, ambas diseñadas para asegurar las libertades de: distribuir copias de software libre – y cobrar por tal servicio, si se desea–, recibir el código fuente o brindar acceso a él, si se lo quiere, modificar el software y usar partes de él en nuevos programas libres y, ser informado de las facultades que la licencia confiere al usuario del software. Ambas licencias precisan que su solo uso hace suponer la aceptación a las condiciones impuestas por ellas y califican toda infracción a sus términos como nula.

GPL y LGPL son licencias persistentes, pues no pueden modificarse sus términos en posteriores trabajos distribuidos. En otras palabras, la distribución del software, con o sin modificación, da lugar a una licencia automática en iguales condiciones. Salvo tratándose de aquellas partes del programa que no estén afectas a la licencia, cuando no se basan en el programa licenciado. Es el denominado efecto “viral”, también conocido como “*copyleft*”.

c) Licencias intermedias

Estas presentan efectos similares a los previstos respecto de las precedentes licencias, aún cuando ciertas peculiaridades les distancias de unas u otras y obligan a su revisión específica

¹⁰ Vid. http://sourceforge.net/softwaremap/trove_list.php?form_cat=13 (visita 31 mayo 2004)

para considerar las posibilidades que ofrecen para el empleo y desarrollo de software bajo sus términos. Entre ellas, merecen considerarse la Mozilla Public License (MPL), la Sun Public License (SPL) y la Artistic License.

La Artistic License impone a los desarrolladores dejar disponible las modificaciones ulteriores al programa licenciado bajo sus términos desde el instante mismo en que se introducen tales modificaciones; ello se logra mediante diversos expedientes: en el caso de haber mediado distribución, a través de un mecanismo que garantice dominio público del código fuente, por ejemplo mediante su liberación en un foro de discusión usenet o en un medio equivalente. Tal liberación no es menester si el desarrollo tiene un uso interno, o bien se alcanza acuerdo sobre el particular con el titular del copyright.

En cambio, en el caso de las SPL y MPL las modificaciones posteriores al código fuente pueden quedar liberadas o no, pero en todo caso debe consignarse su autoría. Asimismo, admite el concurso de licencias. Ahora bien, de distribuirse el desarrollo posterior debe brindarse acceso al usuario al código fuente con las explicaciones del caso, cuando menos “temporalmente”.

Conclusiones

Sobre la base de lo expuesto, es posible formular las siguientes conclusiones:

1. El Software Abierto no reniega de los derechos de autor sobre las obras, antes, al contrario, opera sobre el marco jurídico de los derechos de autor, normativa que presenta un alto nivel de aproximación entre las diversas legislaciones, gracias a los diversos tratados internacionales habidos en la materia (Convenio de Berna, Tratados OMPI sobre Internet, etc.).
2. En general, las licencias de Software Abierto confieren al usuario derecho a usar, copiar, distribuir y modificar el código fuente. A su vez, establecen cláusulas limitativas de responsabilidad y exención de garantías. Sin embargo, existe una amplia gama de licencias disponibles, siendo las más habituales BDS, GPL y LGPL, las que, en conjunto, representan cerca del 90% del software de código abierto o libre desarrollado.
3. En general, a la luz del derecho interno y tras la revisión de las licencias de Software Abierto precedentemente consignadas, es posible afirmar que, salvo ligeros reparos, las prácticas de licenciamiento del Software Abierto, por lo que concierne a los derechos de autor, se avienen con nuestra legislación interna.

6. INTEROPERABILIDAD

Los estándares de facto

Con la aparición de la Internet se hace aún más evidente la importancia y potencial que tiene la interacción entre sistemas. En este sentido, uno de los elementos claves para lograr esta comunicación entre sistemas y entre personas, tiene relación con la utilización de protocolos y lenguajes estandarizados y ampliamente difundidos.

En el mundo del Software Propietario con frecuencia se busca establecer ciertos protocolos y formatos como “estándares de facto”, con lo que se logra una cierta ventaja competitiva, al existir barreras de entrada para competidores. Si bien la existencia de estos “estándares de facto”, puede resultar atractivo para un proveedor al establecer un *lock in*, desde el punto de vista de la comunidad de usuarios representa una importante barrera de salida.

Por una parte, un proveedor dominante tiene completo control sobre los productos o servicios con que es posible interactuar, en función de alianzas comerciales, pago por los derechos de uso de dichos protocolos o formatos y, en general según su curva de maximización económica.

A su vez, se hace muy difícil la migración de la información de los sistemas de un proveedor a los de otro, puesto que no se conoce la estructura interna en la que se almacena o transmite la información, por lo que incluso frente a un cambio de proveedor, se sigue requiriendo de la participación del proveedor original, para posibilitar la transferencia de los datos al nuevo sistema.

Software y estándares abiertos

Por el contrario, uno de los elementos característicos del Software Abierto es la utilización de estándares abiertos. Esto se debe por una parte a la creencia de que esta es la manera de favorecer la interoperabilidad de sistemas y también a la realidad de que gran parte de los proyectos son iniciados por personas, que en general poseen un mínimo nivel de incidencia en el mercado, por lo que su mejor posibilidad es usar estos estándares.

En este sentido, cada vez se hace más frecuente que las implementaciones de referencia, de ciertos protocolos o estándares, se dejen disponibles bajo modelos de Software Abierto, justamente con el objetivo de acelerar la difusión, uso e integración con los sistemas existentes.

A su vez, también se dan ciertos casos, donde a partir de soluciones de Software Abierto se establecen estándares. Este es el caso del programa de Mensajería Instantánea Jabber, que dio origen al protocolo Extensible Messaging and Presence Protocol (XMPP).

Interoperabilidad de aplicaciones

Sin duda, uno de los grandes desafíos del Software Abierto tiene relación con la interoperabilidad con aplicaciones propietarias, especialmente las más ampliamente difundidas, para dar facilidades de migración a estos sistemas de software abierto y para asegurar a quien implementa soluciones abiertas que podrá interactuar naturalmente con soluciones propietarias ampliamente difundidas. Al respecto, existen múltiples casos interesantes de soluciones de Software Abierto que cumplen con estos principios, entro de los cuales podemos mencionar:

- OpenOffice: esta suite de aplicaciones de escritorio en la actualidad puede abrir y guardar documentos y archivos en los formatos nativos de la suite Microsoft Office.
- Samba: este proyecto permite que los sistemas Windows y Unix compartan información, tanto a nivel de carpetas compartidas, como información de cuentas de usuario.
- Gaim: esta aplicación de mensajería instantánea, es capaz de interactuar con los principales protocolos y servicios, como AIM, ICQ, MSN Messenger, Yahoo, IRC, etc. Además es una aplicación de Software Abierto que opera en varias plataformas como Linux, BSD, MacOS X y Windows.

En la actualidad, la interoperabilidad es sin lugar a dudas un atributo extremadamente importante al momento de evaluar la compra o implementación de sistemas. Al respecto, la utilización de estándares abiertos, se ha transformado en un mecanismo clave, por parte de las empresas proveedoras de tecnología, en la búsqueda de mejorar la vida útil de sus soluciones y dar mayores niveles de apertura a sus sistemas.

En este sentido, existe un gran número de soluciones de Software Abierto que se basan en estándares abiertos. Este hecho, sumado a la disponibilidad del código fuente, implican una importante garantía en lo relacionado con la integración con otros sistemas y la utilización de la información almacenada en otros contextos.

Para mayor profundidad, en los Anexos 2 y 3 se tratan varios tópicos de detalle relacionados con Software Abierto e interoperabilidad.

7. BASES DE UNA ESTRATEGIA PARA CHILE

Desafíos Para La Adopción Del Software Abierto

Toda innovación encuentra siempre distintos tipos de desafíos que debe enfrentar para ser aceptada. El Software Abierto no es una excepción. A continuación se mencionan diversas áreas que deben trabajarse para lograr un ambiente favorable para que el Software Abierto sea una alternativa real para el mercado chileno, facilitando cuanto antes la masificación del uso de tecnologías de información a nivel nacional.

a.- Cumplimiento de estándares internacionales en lo relacionado a sitios Web.

En la actualidad existe una considerable cantidad de sitios web que solamente operan correctamente con algunos navegadores de Internet propietarios. Este hecho constituye una importante barrera para la utilización de Software Abierto.

Al respecto, es importante recalcar de que no se busca que los sitios web sean diseñados únicamente para operar con Software Abierto, sino que por el contrario, cumplan con los estándares internacionales, que aseguran una correcta operación tanto en los navegadores web basados en Software Abierto, como en los propietarios.

b.- Compatibilidad con dispositivos y componentes de hardware:

Existe aún hoy un número limitado de dispositivos y periféricos que puede no operar correctamente bajo Software Abierto. Si bien esta es una tendencia que se ha ido revirtiendo rápidamente en el tiempo, constituye un área donde se debe mejorar.

A su vez considerando la velocidad de penetración de soluciones de software abierto, el mercado comienza a reaccionar, tanto desde el punto de vista de los proveedores, que se están preocupando de hacer compatible sus dispositivos con soluciones de Software Abierto, como de parte de los grades clientes que empiezan a solicitar esta compatibilidad.

c.- Compatibilidad de formatos y comunicación.

Otro de los elementos que hoy en día dificultan en parte la utilización de Software Abierto, tiene relación con la posibilidad de leer y modificar ciertos formatos de archivos generados por programas propietarios, en sistemas basados en Software Abierto. Si bien hoy en día existen ciertas aplicaciones basadas en Software Abierto que son capaces de leer una amplia gama de formatos tanto propietarios como libres, existen ciertas funcionalidades avanzadas que no funcionan completamente en las aplicaciones basadas en Software Abierto, y que pueden afectar negativamente la experiencia inicial del usuario al usar el nuevo ambiente.

d.- Percepción negativa inicial sobre Software Abierto.

En los inicios de masificación del Software Abierto, existe un cierto escepticismo respecto al Software Abierto y lo real que es como alternativa, frente a las soluciones propietarias. Esta percepción va cambiando en la medida de establecerse las primeras referencias exitosas y la

mayor oferta de proveedores experimentados en este tipo de soluciones. Sin duda, este es un aspecto que debe trabajarse, de forma de ir generando una creciente confianza, en especial por parte de las empresas e instituciones que compran tecnología.

Las empresas establecidas y organizaciones relacionadas con TIC's, deben comenzar a explorar con mayor energía las posibilidades del Software Abierto, en especial, con una perspectiva de mediano/largo plazo, analizando las múltiples oportunidades de nuevos negocios que se les abren.

Por otra parte, la comunidad que originalmente está a favor del Software Abierto proveniente principalmente del mundo académico, debe hacer un esfuerzo por entender y promover los beneficios económicos y prácticos que el Software Abierto trae a las instituciones y a los países, dejando de lado aspectos netamente técnicos o filosóficos que, lejos de seducir a potenciales usuarios, los alejan de esta alternativa.

Entendiendo el tema y sus implicancias

Como un paso necesario para esta etapa está el entender el problema analizando distintas fuentes como la información proporcionada en este documento y las experiencias de los países que están liderando este tema a nivel mundial. Urge definir una estrategia a nivel nacional, considerando que sólo las leyes del mercado parecen no resultar suficientes frente a razones estructurales propias de una innovación de estas características.

En la actualidad una de las principales limitantes es la falta de desarrollo de una oferta de servicios en torno al Software Abierto, porque se estima que aún hay una demanda pequeña para este tipo de soluciones. Por otro lado se ve como un impedimento la implementación de estas soluciones por la falta de una masa crítica de empresas relacionados con Software Abierto que prestan servicios, como soporte, consultoría y apoyo en migración.

Sin embargo, considerando lo importante y urgente que son las necesidades tecnológicas de nuestro país, resulta clave elaborar una evaluación seria por parte de las autoridades de gobierno para analizar y liderar un proyecto de esta envergadura, no sólo en su rol de principal comprador de tecnología, sino también en su responsabilidad de velar por la maximización de recursos en TICs para hacer más competitivo al país.

Dentro de esta evaluación, a nivel de gobiernos hay variadas experiencias ya realizadas interesantes de revisar, como los casos del gobierno de Gran Bretaña, que realizó pilotos en conjunto con IBM y Sun Microsystems, de Tailandia que logró la liberación de software propietario masivo a costos muy convenientes, del Estado de California que calculó miles de millones de dólares en ahorros al adoptar Software Abierto como su estándar.

Algunas Medidas de Corto Plazo

No es la intención del presente documento elaborar una lista exhaustiva de actividades que habría que realizar para enfrentar la oportunidad que el Software Abierto representa para Chile, solucionando todos los aspectos descritos en el presente documento. Sólo pretendemos llamar la atención sobre algunos caminos posibles que debieran estar en toda evaluación del sector público y privado en esta tarea.

En el corto plazo, sin duda resulta fundamental realizar la evaluación general mencionada anteriormente. Considerando el atraso que tenemos en este tema, deberían en paralelo también realizarse pilotos controlados para asegurar una adecuada acumulación de experiencias para proyectos escalables. En este proceso participarían también oferentes que irían aumentando su experiencia en los desafíos de detalle, claves para llevar estas iniciativas a mayor escala.

Por otra parte, resulta necesario tomar medidas de forma de asegurar que la mayor parte de los sitios Web de nuestro país, cumpla con los estándares internacionales de la W3C, de forma que sean accesibles en toda su funcionalidad desde cualquier navegador que cumpla con dichos estándares. Al respecto, resulta importante asegurar por parte de toda institución pública el cumplimiento de la “Guía Web” lanzada por el gobierno y que sin duda constituye un aporte real y efectivo en esta línea.

Es necesario complementar este tipo de iniciativas gubernamentales con una educación a los decisores privados para que vayan entendiendo los aspectos relacionados con el Software Abierto, considerando que su adopción traerá mayor competitividad a sus empresas y, el tema ya se instaló como una realidad de cambio de la ecuación económica en la industria del software.

Estrategia de Largo Plazo

Dentro de la estrategia de mediano y largo plazo, se hace fundamental la incorporación de todos los actores a esta estratégica discusión. Aquí se cuentan las autoridades de gobierno que deben ejercer una posición de liderazgo, al mundo privado, inicialmente los CIOs, a la industria TI para que esté atenta a las oportunidades de negocios que se les abre tanto nacional como internacionalmente y, al mundo académico, que puede y debe jugar un papel fundamental en la materialización del aprovechamiento de esta oportunidad.

A mediano plazo se irán desarrollando numerosas medidas de diversa índole. El aprovechamiento óptimo de recursos para lograr una masificación real de las TI a todo nivel hará que las diversas áreas de la actividad nacional se incorporen a evaluar soluciones de Software Abierto. Las prioridades estarán dadas por aquellas áreas donde existan soluciones más maduras con múltiples referencias, como sistemas operativos y soluciones de oficinas, lo que ya posibilitaría una real masificación de computadores personales conectados a Internet.

No ser líderes tiene al menos algunas ventajas. Hoy podemos ir copiando experiencias ya exitosas realizadas en países o estados como California, Tailandia, Pakistán, Brasil, Inglaterra y muchos otros, que nos llevan una considerable ventaja en esta estratégica área.

ANEXOS

Anexo 1.- Definiciones Básicas

Las tecnologías de información suelen tener términos de uso exclusivo para iniciados. Estos términos y siglas han aumentado permanente en el tiempo y son de naturaleza dinámica. Con el objeto de facilitar el entendimiento del presente documento y de entregar distinciones para la discusión que se pretende facilitar, hemos incorporado este capítulo que da respuesta a inquietudes básicas sobre el Software Abierto y el Software Libre.

¿Qué es Software Abierto?

En el caso de compra un programa para un sistema propietario, antes de usarlo, el usuario se somete a la licencia mediante la cual la empresa le cede el derecho de uso limitado de su programa: uso en una sola computadora y otros puntos según cada licencia en sí. Este software no se puede copiar, ni usarlo en más de un PC. El programa pertenece siempre a la empresa que lo creó, y ésta decide si hace actualizaciones y cuánto va a cobrar por ellas.

A diferencia de lo descrito anteriormente, el Software Abierto es aquel que no posee estas restricciones y permite según su licencia instalarlo en cuantas computadoras se necesite, modificar su código fuente según las necesidades de cada usuario particular y redistribuir el mismo, entre otras cosas.

El Software Abierto puede entregarse en forma gratuita o pagada, sin embargo, el código fuente está abierto a modificaciones y se distribuye bajo diferentes licencias de entre las cuales la GPL (GNU Public License) es la más usada.

En resumen, la diferencia entre Software Abierto (u Open Source) y el Software Propietario radica en la licencia bajo la cual se puede usar. No hay necesariamente una diferencia de tecnologías o de otro tipo, pero el Software Abierto permite modelos de desarrollo que han demostrado ser sumamente efectivos y que no son posibles de implementar en un desarrollo cerrado.

¿Cuál es la diferencia entre "software libre", "código de fuentes abiertas", "free software" y "open source software"?

"Free Software" (en español: Software libre) es el término empleado por la Free Software Foundation (FSF), la cual creó la licencia GPL e impulsa el proyecto GNU. Debido a que la palabra "free" en inglés es ambigua (puede significar tanto "gratis" como "libre"), y que la posición de la FSF es bastante inflexible, surgió otra organización llamada Open Source Initiative (OSI), la cual usa el término "Open Source Software" ("Código de Fuentes Abiertas" en español). Esta última pretende resaltar los beneficios comerciales del Software Abierto más que una creencia fundamental de que el software debe ser libre.

La FSF define 4 libertades que un software debe cumplir para ser considerado Software libre o Free Software. A su vez, la OSI define 10 requisitos para que un software sea Open Source (Código de Fuentes Abiertas), e incluso certifica las licencias que cumplen con estos requisitos.

En la práctica, toda licencia de software aceptada por la FSF como de Software libre también es aceptado por la OSI como licencia Open Source, por lo cual son términos equivalentes salvo por las motivaciones y el discurso de quienes acuñaron ambos términos. Incluso en muchas ocasiones se habla de FLOSS (Free / Libre / Open Source Software), para no encasillarse en ninguna de las definiciones particulares sino hablar del fenómeno en general. Para efectos de este documento usaremos el término Software Abierto.

¿Qué es Linux?

Linux es un sistema operativo gratuito, basado en Unix, con una estructura sólida y entorno de ventanas semejante a Windows. Es el más famoso representante de Software Abierto, que tiene una aceptación generalizada a nivel mundial. Fue creado a principios de los noventa por Linus Torvalds en Finlandia. No está desarrollado por una empresa en particular, sino está desarrollado por un grupo multidisciplinario geográficamente disperso por todo el mundo. Se distribuye gratuitamente bajo el concepto de Software Abierto. Linux es un sistema operativo multitarea real, multiplataforma y multiusuario. Corre en diferentes plataformas de hardware, incluyendo procesadores de 32 y 64 bits.

¿Cómo se obtiene Linux?

Linux puede “bajarse” gratuitamente desde los servidores de algunas organizaciones, fundaciones o desde los sitios web de las empresas que crean las diferentes distribuciones de Linux. Puede comprar una distribución empaquetada o usar alguna distribución que acompañan a algunas de las revistas sobre Linux (Todo Linux, Solo Linux, Solo Linux Programadores, etc.)

¿Qué significa una distribución?

Una distribución es un compilado de distintas partes de software open source ordenadas en forma coherente en uno o mas medios (típicamente CDs) que permiten instalar el Sistema Operativo y varias aplicaciones para el mismo en una o varias PCs.

¿Qué es X-Windows?

Linux por sí solo no provee interfaz gráfica. Linux es sólo el núcleo de un sistema operativo y solo se encarga del manejo interno del sistema (dispositivos, redes, etc.). X-Windows es un sistema gráfico semejante a Microsoft Windows, para manejo de sistemas Unix/Linux. El sistema es de ventanas con manejo por teclado y Mouse, desde 256 colores hasta color verdadero y totalmente configurable a gusto del usuario.

¿Qué son KDE y Gnome?

KDE es la sigla de K Desktop Environment un sistema gráfico de escritorio especialmente diseñado para Linux fácil de configurar y con variadas opciones para el usuario. Gnome significa GNU Object Model Environment que es otro sistema gráfico de fácil uso. Ambos funcionan sobre X-Windows para proveer funcionalidades como un escritorio con archivos y carpetas y otras funcionalidades necesarias para una experiencia comparable con Macintosh OSX o Microsoft Windows.

¿Qué son los paquetes?

Los programas y aplicaciones para Linux pueden distribuirse en varios formatos, por ejemplo para sistemas que usan paquetes tipo RedHat (.rpm) o Debian (.deb). Los paquetes permiten una correcta instalación de los programas dentro del sistema, se puede decir que cumplen una función semejante a los instaladores que poseen los otros Sistemas Operativos propietarios.

¿Qué aplicaciones están disponibles en Linux?

Existen múltiples aplicaciones de Software Abierto para ser usadas en Linux, desde suites ofimáticas gratuitas (StarOffice/Open Office; AbiSuite; KOffice), programas técnicos (CAD 2D; Diseño), sistemas de bases de datos relacionales (PostgreSQL, MySQL), programas de edición de documentos, aplicaciones multimedia, audio, mensajería instantánea, educativos, palm, radio y televisión y, juegos.

Anexo 2. Mitos y Realidades

La lucha entre seguidores y detractores del Software Abierto tiene, además de aspectos de principios, tremendas implicancias económicas. Esta lucha ha llevado a los promotores del Software Propietario y del Software Abierto a sembrar dudas respecto a la aplicabilidad y/o conveniencia de uno u otro tipo de software.

Considerando que dentro de los propósitos de este documento está el entregar información que enriquezca el análisis y la discusión sobre las implicancias de utilizar Software Abierto, se ha incorporado este capítulo que comenta alguno de sus mitos y realidades. Adicionalmente se ha buscado entregar respuesta a las inquietudes más comunes relacionadas con el Software Abierto.

A2.1.- Mitos

A continuación se enumeran algunos de los mitos sobre el Software de Código Abierto (Open Source Software, OSS) y el Software de Código Cerrado (Close Source Software, CSS). Dentro de este capítulo se compararán las soluciones ofrecidas sobre la plataforma GNU/Linux y por la plataforma Microsoft Windows, representativos ejemplos de OSS y CSS, respectivamente.

A2.1.1.- Mitos sobre dificultades en el uso

MITO 1: *Linux es imposible de usar como desktop, o bien, Linux no es amistoso*

Antes Linux se manejaba en forma casi exclusiva con comandos escritos en el teclado. Sin embargo, hoy en día existe una serie de entornos gráficos tales como GNOME y KDE, que están incluidos en los CD de distribución de Linux. Estas soluciones incluso incorporan características que Windows no tiene (como escritorios múltiples) y que han producido innovaciones interesantes. Esto hace que Linux sea una buena opción para software de escritorio. Para personas sin experiencia con Windows, el uso de Linux no representa problema ya que su ambiente de trabajo es amistoso y no requiere del uso de consolas de comandos.

Al respecto, en un estudio realizado en el 2003 por la empresa alemana Revelantive A.G., se concluye que el desempeño (tiempo necesario en realizar una tarea), de un usuario utilizando un desktop Linux (KDE) en comparación con otro que utiliza Windows XP es prácticamente equivalente.

MITO 2: *Linux es difícil de instalar*

Las nuevas versiones de Linux (como Mandrake, SuSE, linspire (ex Lindows)) son muy simples de instalar, ya que cuentan con instaladores y procesos de configuración automáticos, simples y amistosos. Por otra parte, sistemas como Windows no son necesariamente fáciles de instalar, sino que en la mayoría de casos vienen pre-instalado en el computador que un usuario compra.

MITO 3: *En Linux es difícil instalar software*

En Linux existen formas fáciles y visuales de instalar software. En la actualidad, en la mayoría de las distribuciones es mucho más simple instalar software y mantenerlo actualizado. Ya no es necesario ir a un sitio web ni tener un disco de instalación, sólo se requiere seleccionarlo de una lista y presionar un botón para instalar.

A2.1.2.- Mitos sobre seguridad

MITO 4: *Linux no ofrece seguridad a los usuarios*

Al estar el código fuente disponible se ofrece la oportunidad de que los usuarios, al detectar los *bugs* del software, puedan dar aviso a múltiples instancias de solución. Con software cerrado, el usuario está obligado a esperar que la empresa distribuidora del software detecte y haga un reconocimiento del problema y que entregue una actualización para éste. Además, existen características técnicas en la arquitectura de los sistemas tipo UNIX (como Linux) que potencian la seguridad, como políticas de permisos.

MITO 5: *Linux está diseñado por amateurs y no tiene soporte*

Eso quizás fue en sus inicios. Hoy en día Linux cuenta con el soporte directo de empresas tales como IBM, Intel, HP y Sun, entre otros. Por otra parte, en la gran mayoría de casos, el soporte que ofrece Windows no lo da directamente su fabricante, sino usuarios capacitados que prestan ayuda. Como dato curioso, Microsoft ya no da soporte para Windows NT ni otros sistemas que suman más de un 49%¹¹ de la base de productos instalados de la familia Microsoft Windows. En general, los desarrolladores de Software Abierto tienen un promedio de más de 11 años de experiencia trabajando en desarrollo de software, y muchos reciben sueldos directamente por su actividad en Software Abierto¹².

MITO 6: *Linux no tiene soporte de hardware*

En cierta medida, es cierto que hay algunos dispositivos específicos que aún no se soportan debido principalmente a que fabricantes no hacen los drivers respectivos, ni permiten acceso a la información para que otros los hagan. Sin embargo en la medida que el Software Abierto está representando un mercado crecientemente amplio y atractivo, esta situación está cambiando aceleradamente. Por ahora, la gran mayoría de hardware que posee un usuario “normal” funciona, al menos a un nivel razonable.

MITO 7: *Linux es tecnología atrasada*

Cuando salió la primera versión oficial (1.0) de Linux en el año 94', era similar al DOS: sólo una consola de comandos y funcionaba en un hardware muy específico. A esa altura ya estaba Windows95 en el mercado, el cual era mucho más amistoso. Para el año 99', salieron las primeras versiones oficiales de interfaces gráficas (GUI). En la actualidad, las GUI de Linux soportan múltiples escritorios, transparencias, gran facilidad de uso e instalación de

¹¹ http://news.com.com/2102-1032_3-5378366.html

¹² Lakhani, Wolf, Bates y DiBona (2002) “The Boston Consulting Group Hacker Survey”. O'Reilly Open Source Conference.

dispositivos, y compatibilidad con una enorme cantidad de hardware. Linux se puede instalar no solo en equipos de escritorio, sino que en mainframes, ppc (mac), equipos con hasta 32 procesadores. Hoy ya existen para Linux miles de aplicaciones. Muchas de estas cosas aun no suceden en el mundo de Windows. Sabemos que la tecnología avanza muy rápido y que, si bien aún existen necesidades específicas (como el soporte de webcam para messenger) no satisfechas, están en el proceso de desarrollarse (amsn), por lo que pronto estarán disponibles.

A2.1.3.- Mitos sobre temas comerciales

MITO 8: *Linux no cuenta con aplicaciones para oficina*

Linux cuenta con una vasta gama de aplicaciones que son alternativas viables para oficina, y equivalentes a los programas desarrollados para Windows. En la gran mayoría de casos, se mantiene la compatibilidad entre ambos sistemas operativos mediante el uso de los mismos formatos. A manera de ejemplo, las alternativas a Microsoft Office son OpenOffice.org, Koffice entre otras. Asimismo existen opciones para manejar música, imágenes, mensajes, etc.

La gran mayoría de estas aplicaciones vienen en los CD de distribuciones de Linux, por lo que no requieren ser instaladas aparte del sistema operativo, contrario con lo que ocurre en el software propietario donde cada aplicación debe ser considerada, evaluada, comprada e instalada aparte por el modelo de negocios a que está sujeto el software.

MITO 9: *Linux es más costoso a largo plazo*

Se cree que el costo asociado al mantenimiento y capacitación para Software Abierto es más caro que en sistemas cerrados. Hay sin embargo, varios estudios que señalan que los gastos totales asociados a Software Abierto están entre un 25% y 50% del precio asociado a Software Propietario.

Además, se debe destacar que la metodología de cálculo del costo total de una solución (Total Cost of Ownership, TCO), puede no ser tan válida en los países en desarrollo. Esto puesto que por una parte, los costos de licencias son bastante similares en todos los países, a pesar de que los niveles de riqueza difieren significativamente, comparando países desarrollados y en desarrollo. Por ejemplo, en un estudio del Internacional Institute of Infonomics¹³, se muestra que una solución propietaria que cueste US\$560, corresponde a 0.2 meses de PGB per cápita en EEUU y a 1.6 meses PGB per cápita en Chile.

Por otra parte, entre un 60% y 85% de los costos del TCO, provienen de ítems relacionados con servicios, como mantención, soporte, capacitación, etc. Sin embargo, en los países en desarrollo, estos costos son significativamente más baratos que en los países desarrollados.

Por lo tanto, si combinamos ambos factores, podemos concluir que en el caso de los países en desarrollo, siguiendo el modelo de TCO, los costos de licencias tienen un peso mucho mayor en los costos de las soluciones tecnológicas de lo que se considera normalmente en los

¹³ <http://www.flossproject.org/papers/20040527/usuario-RishabGHOSH-final.pdf>

estudios. Esto sin duda es un elemento a considerar al momento de evaluar la factibilidad económica de utilizar Software Abierto.

MITO 10: *Linux no es apto para aplicaciones críticas*

Existe un gran número de ejemplos “críticos” controlados por Linux, algunos de ellos son:

- Metro de Sao Paulo (2.5 millones de usuarios / día)
- Boeing
- Amazon
- Google
- NASA
- Shell
- ILM, Pixar, Digital Domain (animación cinematográfica)
- Administración pública de Alemania, China, Noruega, Perú, Brasil, Tailandia, Taiwán, etc.

A2.1.4.- Los mitos de soporte y garantías

MITO 11: *No existe un responsable a quien demandar ante problemas de garantía o insatisfacción*

En el caso del Software Propietario existe alguien a quien se pueda demandar, a pesar que los pronósticos no son alentadores. Primero, está el tema de la licencia de usuario final que inevitablemente debió haber aceptado para que algo saliera mal, de lo contrario simplemente no se hubiese podido usar el software. Según esa licencia, la empresa proveedora no se hace responsable por más de lo que costó la licencia y en algunos casos por cifras aún menores. Adicionalmente, el costo de lograr demostrar negligencia es muy elevado tanto en tiempo como dinero, ya que involucra largos procesos judiciales, y aún así nada garantiza algún resultado alentador.

MITO 12: *No existe un apoyo para tareas diarias y actualizaciones*

Usando software propietario, en el mejor de los casos se tiene acceso a un call center mediante el cual se pueden recibir instrucciones básicas. Éstas sirven cuando no hay alguien entendido en la materia dentro de la empresa, pero en el caso de una empresa con administradores internos que tienen cierta idea de cómo solucionar los problemas más básicos, ese apoyo pronto deja de ser un aporte real. La calidad de las actualizaciones disponibles para Software Abierto y propietario tienden a tener tiempos de respuesta parecidos, con una creciente cantidad de empresas dedicada a soporte de Software Abierto.

MITO 13: *El cliente no tiene acceso a servicio "premium"*

La teoría dice que siempre es posible optar a un mejor servicio pagando más, al menos hasta cierto límite. Pero, ¿por qué eso podría ser posible solamente (o en mayor medida) para programas de software propietario? Si consideramos que existe mayor competencia para proveer servicios sobre un Software Abierto, la balanza tiende a inclinarse a favor del éste.

Una ventaja de los proveedores de software propietario es que tienen canales consolidados a través de los cuales entregan soporte y capacitación a sus clientes. Estos canales funcionan de manera relativamente abierta, incluyendo programas de certificación, informaciones y cursos en los cuales pueden participar muchos interesados. Frente a la creciente demanda por Software Abierto, existe ya una capacidad de oferta para satisfacer las necesidades de soporte avanzado en los clientes, capaz de incorporar nuevos productos a sus portafolios.

MITO 14: *No hay empresa de software que ofrezca un servicio serio para soporte de Software Abierto*

Este argumento tiene dos partes. La primera es que existe poca oferta, y la segunda es que dentro de la oferta existente no hay ninguna que infunda la confianza suficiente. Efectivamente, existe menor oferta, y la razón más fuerte es que la demanda es aún baja. Si bien esto se puede presentar como un problema difícil de resolver, también se puede enfocar como una gran oportunidad para oferentes de soporte en esa área.

La segunda parte del argumento – que la confianza generada por los que ofrecen soporte no es suficiente-, parece ser la que tiene el mayor peso. Sin embargo, hay grandes proveedores de soluciones (IBM, Novell, Oracle, HP) que han comenzado a entregar servicios basados en Software Abierto, en particular Linux. Es de esperar que la oferta aumente, sobre todo considerando que en toda la industria del software se nota un claro acercamiento a la visión de software como servicio más que como producto.

A2.2.- Realidades

En el mundo, existe Software Abierto para servidores, para estaciones de trabajo, para dispositivos embebidos, y para otras aplicaciones específicas o verticales. Aquí se presentan algunos hechos:

A2.2.1.- Servidores

- 1. En servidores, los sistemas basados en Software Abierto son extremadamente competitivos.** Existe una alta y creciente penetración de mercado de Linux, de los distintos sistemas BSD, de servidores de correo libres, de sistemas DNS, entre otros. El sistema de correos mundial, por ejemplo, es movido mayoritariamente por servidores de correo basados en Software Abierto. Los mayores buscadores, Google entre ellos, están también basados en Software Abierto. Linux, uno de los más exitosos sistemas operativos, tiene hoy cerca del 25% de los servidores mundiales, y se estima que esta rápidamente creciendo, siendo instalado en cerca de la mitad de los nuevos servidores instalados en el mundo.
- 2. Excepto algunos pocos sistemas propietarios, es posible hacer de todo con 100% Software Abierto.** Como se mencionó, existen en Software Abierto sistemas operativos, servidores de correo, servidores de nombre y directorio, servidores de páginas Web, servidores de bases de datos, servidores gráficos, de video, de radio, de todo tipo de servicios. Esto técnicamente permitiría reemplazar con Software Abierto la funcionalidad de cualquier sistema basado en sistemas propietarios. Es solo imposible la migración

cuando se debe mantener algún producto propietario y no se cuenta con la capacidad para rescribir o modificar sustancialmente alguna aplicación. La conveniencia económica de migrar o no estará definida por la evaluación de cada proyecto específico.

3. **Es posible ser compatible con clientes propietarios usando servidores de Software Abierto.** Un servidor basado en Software Abierto es perfectamente capaz de servir a clientes propietarios de cualquier sistema operativo. Existen servidores de archivos, de impresoras, de correo, de autenticación y de otros servicios necesarios en una red.
4. **Normalmente es posible hacer las cosas con menos hardware con Software Abierto.** Los sistemas operativos libres ocupan por lo general menos recursos para la misma tarea. Los fabricantes de bases de datos recomiendan hardware más sencillo para Software Abierto. Además, es posible instalar solo lo necesario, muchas veces reduciendo el tamaño de la instalación.
5. **Los servidores basados en Software Abierto tienen muy buena estabilidad.** No es raro encontrar servidores basados en Software Abierto que llevan un par de años sin una caída. Servidores bien configurados sólo es necesario apagarlos para hacer upgrades de hardware. Es además posible hacer casi todo tipo de actualizaciones sin reiniciar. Solo es necesario reiniciar el sistema cuando se hacen actualizaciones mayores (cambios de versión).
6. **Existe soporte pagado de calidad para servidores de Software Abierto.** Diversas empresas en Chile, en Sudamérica y en el mundo dan soporte pagado para sistemas operativos y programas de Software Abierto, así como para instalaciones completas. Es posible comprar soporte en forma anual, mensual, en forma de consultoría y de capacitación.

A2.2.2.- Computadores Clientes

7. **Las aplicaciones de Software Abierto existen y tienen buenas interfaces.** Existen sistemas de oficina gráficos que permiten una transición razonable para los usuarios ya familiarizados con sistemas propietarios. Así mismo, es posible usar Software Abierto en el equipo del hogar y conectarse a Internet vía telefónica, cable módem o ADSL.
8. **Existen aplicaciones de oficina que resuelven prácticamente todos los problemas/tareas de una oficina típica.** En Software Abierto existen muy buenos procesadores de texto, planillas de cálculo, editores de presentaciones, navegadores, clientes de mensajería instantánea, de correo, de video, entre otros. Por lo que es factible utilizar un sistema basado en Software Abierto para tareas diarias de oficina.
9. **En general, se necesita al menos el mismo nivel de hardware para un cliente de Software Abierto sobre todo RAM.** Los requerimientos de hardware necesario para un cliente de oficina o casa son similares a los de los sistemas propietarios. Cualquier PC a la venta hoy en el mercado es capaz de ejecutar una distribución de Software Abierto sin problemas.

10. Hay claras incompatibilidades a nivel de navegación con sitios expresamente diseñados para ser usados solo desde Windows. Existe una cierta cantidad de sitios expresamente diseñados para ser usados solo desde sistemas propietarios. Esas incompatibilidades suelen ser detalles pequeños, que sin embargo crean molestias al usuario, inhibiendo ciertas transacciones.

Muchas de esas incompatibilidades son fáciles de solucionar con información correcta. En Chile, la mayoría de esas incompatibilidades están en sitios transaccionales (bancos, servicios de pago en línea) y en algunos servicios públicos del gobierno como Chilecompras, Sence y, SII, entre otros. Cabe destacar que el Gobierno ha editado el año 2004 una guía de buenas prácticas para construcción de sitios web institucionales, que específicamente trata el tema de la compatibilidad con todas las plataformas y, de ser seguidas sus recomendaciones, asegurará una buena navegación para estaciones libres¹⁴.

11. Existe una incipiente red de soporte de amistades y relaciones. En el mundo no propietario existe una enorme red de soporte informal que se ayuda mutuamente en mantener los sistemas, sobre todo en las casas, funcionando. Es normal para los usuarios de las casas saber a que conocido, vecino o pariente llamar para que les ayuden a solucionar un problema con su sistema. A nivel de Software Abierto esta red esta comenzando a aparecer, aunque necesita aun un tiempo de maduración para ser realmente efectiva.

12. Existen oportunidades nuevas para reutilizar hardware usando Software Abierto. Una característica interesante de los sistemas basados en Software Abierto es que el costo de implementar tecnologías nuevas es usualmente muy bajo. Proyectos que serian muy costosos usando software propietario, como Alta Disponibilidad y sistemas de Clientes Livianos, se hacen factibles usando Software Abierto. Los sistemas de “clientes livianos” son particularmente interesantes como herramientas de ahorro de costos, ya que un PC obsoleto es perfectamente útil como cliente liviano en este nuevo mundo.

¹⁴ Puede ser revisada en <http://www.guiaweb.gov.cl>

Anexo 3.- Preguntas Frecuentes

Al igual que los mitos, existen preguntas de uso doméstico que al no encontrar respuestas concretas pueden ser una barrera para la adopción del uso del Software Abierto por parte de los usuarios individuales y Pymes que requieren urgentemente aumentar su acceso a Tecnologías de Información. Esta sección busca dar respuestas a algunas de estas preguntas.

¿Qué tipo de PC se necesita para Instalar Linux?

Actualmente Linux funciona casi en todos los PCs. Es decir, en la mayor parte de los casos, cualquier equipo que funcione con Windows funciona también con Linux. Además de los equipos de escritorio, Linux funciona en notebooks, ppc (mac, ibook, powerbooks, etc.) incluyendo la nueva generación de procesadores de 64-bit. La configuración mínima de un PC sería: procesador 80386, memoria RAM 4MB, espacio en disco duro desde 200 MB, monitor color VGA, mouse y teclado. La configuración mínima recomendada para correr X-Windows como entorno gráfico es: procesador pentium o superior, RAM 64MB, espacio en disco duro de 1GB o más, monitor SuperVGA, mouse y teclado. Claramente, para aprovechar todas las características estéticas y software de última generación, es útil tener máquinas más modernas y con mejores características.

¿Se pueden usar programas para Windows en Linux?

Si, ya que existen varios emuladores de las APIs y otras librerías de Windows para Linux. Los emuladores más usados son Wine, Crossover, VMWare y Cedega. Obviamente, no todos los programas diseñados para Windows corren en los emuladores para Linux, pero teóricamente la gran mayoría de ellos pueden hacerlo. Wine es un emulador genérico de licencia GPL. Crossover es un emulador que permite instalar aplicaciones Windows (como MS Office) en Linux. VMWare permite correr ambos sistemas operativos simultáneamente, y comunicados entre sí. Cedega permite correr juegos de última generación con soporte DirectX. Los tres últimos tienen licencias comerciales. Es decir, es software propietario que funciona sobre una plataforma de Software Abierto.

¿Se puede usar Linux y Windows en el mismo PC?

Si, se puede usar ambos sistemas operativos en el mismo PC de dos formas diferentes. La primera es particionando el disco duro para hacerle un espacio a Linux, y la otra, sin particionar el disco duro mediante la instalación genérica de un "win-Linux", es decir un Linux adaptado para correr desde Windows. En caso de duda, se recomienda la primera opción. En varias distribuciones Linux se puede hacer esto de manera muy fácil al momento de instalar.

¿Se puede recibir y enviar e-mails en Linux?

Si. No solo puede usarse correo electrónico, sino también navegar, usar chat y mensajería instantánea. Existe una gran cantidad de programas, también abiertos, que cumplen con todas estas necesidades.

Los programas de Software Abierto tienen versiones 0.8, 1.2, o similar. ¿Éstos programas son menos desarrollados e inestables que los que tienen versión 2000, 2004, XP, MX, Plus, Lite, o similares?

En Linux las versiones siguen otros principios y no se usa la versión de los programas como estrategia de comercialización. Para que un programa llegue a una versión 1.0, ya ha pasado por muchas revisiones, un desarrollo exhaustivo y muchas pruebas de estabilidad bajo distintos ambientes. Por ejemplo, el cliente de correo Mozilla Thunderbird 0.8, es estable y más confiable que la mayoría de los clientes de los correos propietarios. (Thunderbird también funciona sobre Windows).

¿Se puede usar un PC con Linux para abrir archivos de Word, Excel y/o Powerpoint?

En general si. Sólo los archivos con macros presentan incompatibilidades, y en general son de muy poco uso. De hecho, en muchos casos las macros de Word, Excel y Powerpoint son deshabilitadas para evitar propagación de virus y por ende no se usan. Los archivos normales de texto, presentaciones y planillas pueden ser abiertos por diversos programas, siendo el más conocido OpenOffice.org.

¿Se puede usar un PC con Linux para escribir archivos de Word/Excel/Powerpoint?

Si. OpenOffice.org puede crear archivos en diversos formatos. Los formatos clásicos de Word, Excel, Powerpoint y otros, ya que están todos soportados.

¿Se puede usar un PC con Linux para conectarse a Internet por teléfono?

Si, siempre que se tenga un modem compatible con Linux. La gran mayoría de los modems que vienen en una caja externa, conectado al PC por un cable, son 100% compatibles. De los que vienen integrados en la placa madre de los PCs, casi todos son compatibles. Los de tarjeta ISA de 33.6kpbs o menos también son en su gran mayoría compatibles. Los más problemáticos son los 'Winmodems' en tarjetas ISA o PCI de hace un par de años y de más de 33.6kpbs, ya que es altamente probable que se necesite conseguir un *driver* en <http://www.linmodems.org> o bien conseguirse otra tarjeta más compatible.

¿Se puede usar un PC con Linux para conectarse a Internet por banda ancha?

Si. Están soportados todos los modems ADSL y Cable que se conectan usando una tarjeta Ethernet (red) o USB. Si el proveedor entrega un modem que sólo se conecta vía USB, se puede pedir que lo cambie por uno Ethernet, que es más compatible.

¿Se puede usar un PC con Linux para jugar?

Si, aunque la oferta de juegos para Linux es más restringida en comparación a la oferta que existe para sistemas propietarios como Windows. Para jugarlos, lo más probable es que se deba hacer un *doble boot* entre Windows y Linux, lo que implica apagar y prender el equipo para cambiar de un sistema al otro, o bien usar algún emulador como Wine.

¿Se puede usar un PC con Linux para imprimir en la impresora que ya está en el hogar?

Si, siempre que la impresora sea compatible. Actualmente, casi todas las impresoras que se venden en el mercado son compatibles, sin embargo, las impresoras más antiguas pueden no serlo. Lo más fácil es consultar en <http://www.Linuxprinting.org/> por si es compatible.

En todo caso, al tener una impresora relativamente estándar es casi seguro que sea soportada, no importa cuan antigua sea.

¿Se puede usar un pendrive/memory stick/compact flash en Linux?

Si, Linux soporta todo tipo de memorias vía USB, así como disqueteras, zip, jaz y copiadore de CD/DVD, tanto internos como externos. También existe soporte para tarjetas de televisión y radio, aunque algunas específicas no son triviales de configurar. Normalmente, las tarjetas de televisión que se venden masivamente en Chile (chip BT848), están muy bien soportadas y las distribuciones más comunes las reconocen automáticamente, por lo que no se necesitan configurar. También existe soporte para las PDAs (palm).

¿Es Linux para cualquier usuario?

Linux no necesariamente es la mejor solución en todos los casos. A continuación se indican argumentos en pro y contra del uso de Linux para situaciones específicas.

No usar Linux:

- Si alguna aplicación importante y necesaria para el trabajo del usuario no está disponible en Linux ni hay sustitutos adecuados en el mercado.
- Si el usuario trabaja en un ambiente comercial/industrial/profesional donde no se usa Linux y no dispone de apoyo técnico por parte de su personal técnico ni piensa poder proporcionar dicho apoyo por usted mismo.
- Si el objetivo único o principal del equipo son juegos muy sofisticados y el usuario desea pagar por licencias comerciales de emuladores.
- Si alguna pieza del equipo no está soportada en Linux y no se puede prescindir de ella ni se esta dispuesto a comprar otra compatible.

Usar Linux si:

- ◆ El usuario no está dispuesto a invertir en software comercial, y desea tener todas las herramientas para desarrollo, trabajo de oficina, gráfica, multimedia, etc.
- ◆ Se necesita estabilidad: es decir, no quiere tener que estar reiniciando el equipo con cada instalación, y menos tener que formatear por problemas varios. Igualmente, no quiere tener que depender de que se caiga su equipo cuando más lo necesita.
- ◆ Si el equipo debe hacer tareas de alto rendimiento y sobrecarga de trabajo.
- ◆ Se desea un ambiente libre de virus, troyanos, gusanos, spyware, y otros.
- ◆ Si se es nuevo en el mundo de la computación.
- ◆ No desea depender de una sola herramienta, software, proveedor o soporte.
- ◆ Desea que el sistema necesite poca mantención, además de fácil y gratuita. Por ejemplo, al salir una nueva versión de Linux, no es necesario formatear e instalar todo de cero,

simplemente se instala como cualquier programa en la mayoría de las distribuciones.

Otras razones para usar Linux:

- Las distribuciones de Linux no son costosas: pueden obtenerse gratis de los lugares apropiados de Internet o bien comprar los CDs de costo relativamente barato en comparación con otros sistemas comerciales.
- Linux es un sistema operativo completo con tres grandes características: **Estable** (es muy raro que colapse), **Fidedigno** (un servidor Linux puede funcionar durante cientos de días sin necesidad de reiniciarse, lo que no es común en otros sistemas) y extremadamente **Potente**.
- Tiene un entorno de programación completo, incluyendo C, C++, Pascal, compiladores Fortran, utilidades como Qt y lenguajes de guiones (scripts) como Perl, gwak y sed, un simple compilador C para Windows cuesta cientos de dólares.
- Presenta una excelentes prestaciones para conexión en red, compartir Internet, comunicación entre equipos seguras (ssh), manipular su equipo remotamente (incluso quemar CDs o poner música).
- Además del Software Abierto, hay también bastante software comercial para Linux.
- Se actualiza y mantiene fácilmente. Por ejemplo, las particiones de Linux no se fragmentan, por lo que no se necesitan programas para defragmentar. El sistema tiende a mantenerse limpio (caché, temporales, cookies, etc).
- Admite múltiples procesadores como norma.
- Es realmente multitarea y multiusuario. En Linux pueden estar trabajando varias personas simultáneamente en un solo equipo, sin interrumpirse unos con otros. Incluso existen configuraciones en las cuales múltiples monitores y teclados ocupan el mismo computador y parecen computadores independientes.

Y además, por instalar Linux no se tiene que renunciar a Windows. Es posible gestionar Linux más otros sistemas operativos en el mismo computador.

Anexo 4.- Casos de Éxito

Buenas Prácticas en Chile

SOFOFA

La Sociedad Fomento Fabril se ha preocupado de fomentar la incorporación de herramientas para apoyar la productividad basadas en código abierto y de distribución gratuita. Desde mediados del año 2003, la política interna de Sofofa en esta materia considera la habilitación del paquete de herramientas “OpenOffice” en los computadores de las secretarías. Comprobándose la integración satisfactoria del procesador de textos “Writer” y la Hoja de Cálculos “Calc” con los documentos generados mediante el resto del software comercial disponible en la institución.

La buena experiencia con Open Office los ha motivado a publicar en el sitio web corporativo (www.sofofa.cl) información acerca de esta herramienta y habilitar un link para que cualquier empresa interesada pueda descargarla e instalarla.

EDULINUX

El Instituto de Informática Educativa (IIE), apoyado por la Coordinación Nacional de la Red Enlaces (<http://www.eduLinux.cl>, <http://www.redenlaces.cl/>), realizó un estudio exploratorio para determinar la factibilidad de instalar Linux en establecimientos educacionales, con el objetivo de reaprovechar el equipamiento de laboratorios que fueron implementados hace cinco o seis años y que por tanto cuentan con computadores de bajas características técnicas que no soportan las últimas versiones de navegadores y software de productividad.

La solución encontrada se basa en la implementación de laboratorios en modelo cliente - servidor, donde los computadores que forman parte de la red son sólo terminales y las aplicaciones (software) no se ejecutan directamente en ellos, sino que en un servidor central. Esto permite que equipos (clientes) de bajas características puedan utilizarse para ejecutar aplicaciones que requieren de mayor poder de cómputo.

Durante el año 2004, el IIE desarrolló y empaquetó la primera versión de esta solución llamada EduLinux que incluye un conjunto de CD's de instalación, manual de instalación, una ficha de datos de instalación y un diskette de arranque para equipos Olivetti M-24 (equipos altamente presentes en escuelas y liceos de Chile), aunque EduLinux funciona con cualquier tipo de computadores cliente que cumplan con las condiciones técnicas mínimas (CPU Pentium I o superior, 16 MB RAM, 1 MB de video, tarjeta de red).

INFOCENTROS INJUV

El programa Centros de Información y Difusión Juvenil nace en septiembre de 1993. En ese momento su misión consistió en entregar a los jóvenes información organizada y relevante para la vida cotidiana y capacitarlos en la búsqueda, registro, análisis de información y en el uso de los servicios públicos.

Durante el año 1997 el INJUV tomó la decisión de rediseñar el programa asumiéndolo como un Sistema Nacional de Información Juvenil. De esta forma, el programa autodefinió su misión de la siguiente manera: "Generar al interior del Instituto Nacional de la Juventud, un Sistema Nacional de Información que contribuya a la formación e inserción social de los jóvenes mejorando su calidad de vida a través del conocimiento que posean sobre los Beneficios, Oportunidades y Servicios (BOS) que la sociedad les ofrece de modo tal, que puedan contar con un conocimiento relevante y pertinente para la toma de decisiones en los distintos ámbitos de participación y expresión juvenil."

Para el 2004, el Sistema de Información para la Juventud tiene 93 Infocentros operativos en un 100%, en todas las regiones de Chile, de los cuales 43 fueron completamente instalados el último trimestre del 2004. Actualmente, existen 110 Infocentros en la red de la Injuv, donde 93 Infocentros utilizan Open Office y 17 de ellos están en proceso de implementación y correrán con el Sistema Operativo Linux.

Respecto al número de capacitados por servicios prestados, más de 230.000 jóvenes, en todas las regiones del país, fueron capacitados en el uso de OpenOffice -uso de Planilla de Cálculo, procesador de Texto- debido a que Open Office es la plataforma tecnológica de los Infocentros.

Dentro de las metas y compromisos oficiales tomados con la DIPRE y la Campaña de Alfabetización Digital, las metas para el 2004 eran de 10.000 personas, sin embargo estas fueron ampliamente superadas con 12.000 personas capacitadas en alfabetización digital en Open Office. Para el 2005, las proyecciones prevén que las cifras serán superiores a la alcanzada hasta ahora.

BIBLIOTECA NACIONAL

Con el objetivo de Promover el conocimiento, la creación, la recreación y la apropiación permanente del patrimonio cultural y la memoria colectiva del país, contribuyendo a los procesos de construcción de identidades y al desarrollo de la comunidad nacional y de su inserción en la comunidad internacional, se instalaron mas 20 terminales de autoconsulta con LTSP, dichos terminales estan instalador en el Hall de la biblioteca, estos equipos remplazaron a los antiguos terminales IBM de pantalla verde.

NETLINE (ISP)

Incorpora dentro de su organización 30 Linux Terminal Server para su call center , cada escritorio esta conectado a un servidor central, el cual cuanta con un sistema de clustering de servicios.

Buenas Prácticas Internacionales

Uruguay

- Poder Judicial

<http://www.Linux.net.uy/SoftwareLibreYeElEstado>

Argentina

- Cervecera Quilmes (siguiendo recomendación de Gartner)
La empresa de 2500 funcionarios migró todos sus equipos a Linux.

Venezuela

- Banco Mercantil (IBM)
<http://www.pc-news.com/detalle.asp?sid=2&id=36&Ida=257>
<http://www.bancomercantil.com/>
- Noticia: Gobierno decretará uso obligatorio de Software Abierto para la administración pública
<http://www.rnv.gov.ve/noticias/index.php?act=ST&f=2&t=8930>

Brasil

- Según estudios recientes el 78% de las empresas usan Linux
- Brasil se aleja de Microsoft; para acortar la brecha tecnológica en su país, el Presidente Lula da Silva y su CIO están impulsando la migración a Open Source. Infoweek
http://cma.zdnet.com/texis/techinfobase/techinfobase/+4wo_qc6X+9mvvnWqn/cdisplay.html?source=alac
- “IBM Global Services moves Linux beat to Brazil”
http://news.com.com/2102-7344_3-5233011.html?tag=st.util
- IBM firma contrato para entrenar a 980 empleados del gobierno de Sao Pablo, Brasil.
http://news.com.com/IBM+signs+Brazilian+Linux+training+pact/2110-7344_3-5470187.html

Colombia

- Colegio "Gimnasio Norte del Valle (GNV)"
<http://www.kazak.ws/semilla/colegios.php> (Hay un video sobre la experiencia)
Todos los niños del colegio, incluyendo los más pequeños, tienen acceso a un gran nivel de tecnología de software escritorio, aplicándola a la educación en varias materias (matemáticas, dibujo, etc.), además de computación propiamente tal, usando exclusivamente Software Abierto.

Estados Unidos

- AT&T Planea sustituir Windows por Linux
El mayor operador estadounidense esta estudiando la posibilidad de migrar a Linux o Mac en los mas de 70.000 equipos de sus empleados. La compañía calcula un ahorro del 50% al migrar a Linux. Eslambolchi detalla también que además de ese ahorro, AT&T se queja de que ha tenido más ataques de virus en los computadores personales durante los últimos seis meses que en los 10 años previos.
- **Estado de California**
Administración Pública: Schwarzenegger se va a Linux. Un documento, que lleva por nombre "*California Performance Review*" (<http://report.cpr.ca.gov>), redactado por un grupo de veintiún expertos y observadores independientes, recomienda a la administración del estado de California, liderada por Arnold Schwarzenegger, adoptar Linux y permitir avisos publicitarios en los sitios estatales. Su puesta en práctica supondría un ahorro de US\$32.000 millones en cinco años a su administración.
Qué proponen los expertos: adopción del Software Abierto, migración de los sistemas de comunicación telefónica a Voz sobre IP, creación de una división centralizada de

administración de sistemas de información y venta de espacios publicitarios en los sitios web de la administración.

- Varios casos notables de éxito en el gobierno norteamericano pueden ser encontrados en LinuxJournal

Roma

- Migración similar a la de Munich en un total de 9500 computadores
http://www.repubblica.it/2004/b/sezioni/scienza_e_tecnologia/Linuxroma/Linuxroma/Linuxroma.html

Munich

Prácticas de migración masiva usando suse.

- News (EN) Munich Votes for Linux Migration Plan
<http://it.slashdot.org/article.pl?sid=04/06/16/2254253&tid=163>
- Article (DE) Münchner Stadtrat segnet Konzept zur Linux-Migration ab
<http://www.heise.de/newsticker/meldung/48313>
- Site (DE) The project "LiMux - the IT evolution" with the state capital
<http://www.muenchen.de/Rathaus/referate/dir/limux/89256/index.html>
- Article (EN) Limux - the IT-Evolution
<http://europa.eu.int/ida/en/document/3223/470>

Berlín

- Berlin Budenstag (Parlamento) migró todos sus servidores.
- Berlin Schoeneberg esta iniciando la migración de sus PCs de escritorio de oficina.
- Sixt Car Rental migró exitosamente en 1996
- El gobierno de la ciudad de Schwaebisch (Swabian) migró a Linux cerca de 2000 PCs
- El Proyecto KDE se encuentra activamente haciendo *lobbying* de sus productos *kgroupware*, los que integran seguridad de correo (OpenPGP), a los legisladores

Israel

- <http://www.theinquirer.net/?article=13808>
El argumento fue bastante simple, si es más barata la migración a OpenSource que el costo permanente en licencia, se usa OpenSource.

Thailand - Linux para la población:

- En Thai...
<http://Linux.thai.net/>
<http://opentle.org/>
<http://opensource.thai.net/>
- Microsoft Loses to Linux in Thailand Struggle (LinuxInsider - Diciembre 11, 2003)
<http://www.Linuxinsider.com/story/32110.html>
- Foro sobre la lucha que dio Microsoft después de la iniciativa Tailandesas
<http://lwn.net/Articles/57940/>
- Microsoft slashes prices in Thailand (June 19, 2003)
<http://asia.cnet.com/news/systems/0,39037054,39136847,00.htm>
- Windows global pricing cracked by Thailand: Gartner

<http://asia.cnet.com/news/industry/0,39037106,39147732,00.htm>

Anexo 5.- Algunas Empresas Chilenas en Software Abierto

En esta sección se enumeran empresas proveedoras de servicios informáticos que tienen una práctica madura trabajando con Software Abierto.

Empresa	Actividad	URL
ADEMN	Asesoría y migraciones a Linux y Oracle, instalación y configuración de redes, servidores, software.	www.ademn.com
Blue Company	Soluciones tecnológicas basadas en Linux y Open Source para PYMES.	www.bluecompany.biz
Corporación Linux	Consultoría y servicios basados en aplicaciones Open Source.	www.corporacionLinux.cl
Dicec	Servicios basados en Linux Suse y RedHat.	www.dicec.cl
DSI	Desarrollo de software y aplicaciones basadas en Linux y OpenSource.	www.dsi.cl
IBM	Soluciones Linux.	www.ibm.cl
LinuxCenter	Asesorías, soporte y capacitación basados en Linux.	www.Linuxcenter.cl
NetLinux	Soporte y Administración de servidores Linux.	www.netLinux.cl
NetOffice	Soporte, asesoría e integración multiplataforma.	www.netoffice.cl
NOVELL Chile	Soluciones Linux.	www.novell.com/Linux
Microserv	Servicios basados en Linux.	www.microserv.cl
Mundo Linux	Instalación de servidores Linux, sistemas de venta y manejo de documentos basados en Open Source	www.mundoLinux.cl
Sonda	soporte, consultoría, implementación de soluciones de correo y de seguridad basados en Red Hat Linux	www.sonda.com
Igloo Sistema Ltda	Soluciones en Debian	www.igloo.cl
Pegasus	Soluciones en Debian, capacitación.	www.pegasus.cl
Gestión Integral	Soluciones y servicios basados en Linux, Solaris y FreeBSD.	www.gestion-integral.com
Cosertek Ingeniería	Plan de Migración por capas y áreas, para hacerlo fácil para el usuario común.	www.cosertek.cl
0Day	Consultarías	www.0day.cl

REFERENCIAS

EFECTOS ECONOMICOS

www.cyber.com.au/cyber/about/Linux_vs_windows_tco_comparison.pdf

<http://management.itmanagersjournal.com/management/04/05/10/2052216.shtml>

SEGURIDAD

Kerckhoffs, A. (1883). **La cryptographie militaire**. Journal des sciences militaires, IX(1):5-83.

Anderson, R. (2002). **Security in Open versus Closed Systems - The Dance of Boltzmann, Coase and Moore**. In Papers of Open Source Software: Economics, Law and Policy, Toulouse, France. Institut d' Economie Industrielle, Université des Sciences Sociales.

Raymond, E. S. (1998). **The Cathedral and the Bazaar**, capítulo 3, pags 27-78. O Reilly and Associates, Sebastopol, California.

Serrão, C., Neves, D., and Trezentos, P. (2003). **Open Source Security Analysis - Evaluating Security of Open Source Vs. Closed Source Operating Systems**. In ICEIS 2003, Proceedings of the 5th International Conference on Enterprise Information Systems, volume III - Information Systems Analysis and Specification, pages 433-440, Angers, France.

Price Waterhouse Coopers (2004). **DTI Information Security Breaches Survey**. Technical report, Department of Trade and Industry, United Kingdom.

Gordon, L. A., Loeb, M. P., Lucyshyn, W., and Richardson, R. (2004). **Ninth Annual CSI/FBI Computer Crime and Security Survey**. Technical report, Computer Security Institute / Federal Bureau of Investigation, USA.

Reasoning Inc. (2003). **How Open-Source and Commercial Software Compare: A Quantitative Analysis of TCP/IP Implementations in Commercial Software and in the Linux Kernel**. Technical report, Reasoning Inc. (<http://www.reasoning.com/downloads/opensource.html>)

Reasoning Inc. (2003). **How Open-Source and Commercial Software Compare: A Quantitative Analysis of Database Implementations in Commercial Software and in MySQL 4.0.16**. Technical Report, Reasoning Inc. (<http://www.reasoning.com/downloads/mysql.html>)

Anderson, R. (2003). **Cryptography and Competition Policy - Issues with Trusted Computing**. In 2nd Annual Workshop Economics and Information Security.

Anderson, R. (2003). **Trusted Computing and Competition Policy - Issues for Computing Professionals**. Upgrade: The European Journal for the Informatics Professional, 4(3):35-41. This is a shortened version, special for Upgrade, of a paper entitled "Cryptography and Competition Policy - Issues with Trusted Computing", which can be found at <http://www.ross-anderson.com/>

LEGALIDAD

Wheeler, D. A. (2004). **Make Your Open Source Software GPL-Compatible. Or Else.** <http://www.dwheeler.com/essays/gpl-compatible.html>.

Barlow, J. P. (1994). **The Economy of Ideas**.

http://www.wired.com/wired/archive/2.03/economy.ideas_pr.html. Bessen, J. and Maskin, E. (2000). Sequential Innovation, Patents, and Imitation. <http://www.researchoninnovation.org/patent.pdf>.

Boldrin, M. and Levine, D. K. (2001). Perfectly Competitive Innovation. Available online at <http://levine.sscnet.ucla.edu/papers/pci23.pdf>.

Hall, B. H. (2003). Business Method Patents, Innovation and Policy. Competition Policy Center. Paper CPC03-039.

Malcolm, J. (2003). Problems in Open Source Licensing. In Australian Linux conference.

Moglen, E. (1999). Anarchism Triumphant: Free Software and the Death of Copyright. First Monday, 4(8).

Perchaud, S. (2003). Software Patents and Innovation

Aragónés, J. (2004). La GNU GPL en el Ordenamiento Jurídico Español. In Proceedings of the International Open Source Conference.

Gritzalis, D., editor (2002). Secure Electronic Voting, volume 7 of Advances in Information Security. Kluwer Academic Publishers, Boston, MA, USA.

Lessig, L. (2004). Free Culture - How Big Media uses Technology and the Law to Lock Down Culture and Control Creativity. The Penguin Press.

Malcolm, J. (2003). Problems in Open Source Licensing. In Australian Linux conference. Metzger, A. and Jaeger, T. (2001). *Open Source Software and German Copyright Law.* International Review of Industrial Property and Copyright Law (IIC), 32(1):52.

Welte, H. (2004). netfilter project was granted a preliminary injunction against Sitecom GmbH. Press Release.

DESAFÍOS

Microsoft Sees Open-Source Threat Looming Ever Larger (eWeek September 5, 2004)
http://www.eweek.com/print_article/0,1761,a=134769,00.asp

Laptopia: Will your next laptop just work with Linux? (LinuxJournal January 01, 2004)
<http://www.Linuxjournal.com/article.php?sid=7198>

The future of Linux multimedia (NewsForge October 02, 2004)
<http://www.newsforge.com/article.pl?sid=04/09/29/1420201>

MITOS Y REALIDADES

Facts (Microsoft) Costos Reales (desde el punto de vista de Microsoft) del uso de soluciones GNU/Linux
<http://www.microsoft.com/mscorp/facts>

Linux vs. Windows: Open to Interpretation (Internet Works - July 2004)
Estudios de costos hechos por organismos independientes, aunque auspiciados por Microsoft,
<http://www.iwks.com/opinion/default.asp?pagetypeid=2&articleid=31871&subsectionid=649>

Windows vs. Linux, Mitos y Realidades (Visión de Microsoft)
<http://www.microsoft.com/spanish/msdn/comunidad/mtj.net/voices/art184.asp>
<http://www.sysinternals.com/>

Wikipedia - Software libre
http://es.wikipedia.org/wiki/Software_libre

Introducción al Software libre y GNU/Linux - Mitos
<http://www.ututo.org/docs/descarga/2003/curso/htmls/introduccion/x185.html>

Mitos y leyendas del software open source
http://www.evoluty.com/esp/columns/20020915_open_source.html

PREGUNTAS Y RESPUESTA FRECUENTES

<http://www.tldp.org/FAQ/Linux-FAQ/>

<http://www.Linux-es.org/Faq/Files/Html/index.html>

OTRA INFORMACION RELEVANTE

Alternativas de Software

Alternativas Libres <http://alts.homeLinux.net/index.php>

Open source replacements for proprietary software
http://www.asiaosc.org/enwiki/page/Open_source_replacements_for_proprietary_software.html

The table of equivalents / replacements / analogs of Windows software in Linux.
<http://Linuxshop.ru/Linuxbegin/win-lin-soft-en/table.shtml>

Use Open Source Software without installing Linux
http://www.asiaosc.org/article_223.html

OpenSource en el Desktop

User Guide to Using the Linux Desktop
Introductory training materials for end-users and trainers to use the Linux operating system.
<http://www.iosn.net/training/end-user-manual/>

Linux in Government: How to Misunderstand the Enterprise Linux Desktop
<http://www.Linuxjournal.com/article.php?sid=7692>

Linux poised for move from data centre to desktop - report
http://www.theregister.co.uk/2004/09/10/butler_on_enterprise_Linux/

Linux in the Enterprise: A Viable Alternative for Server and Desktop Operating Systems?
<http://www.butlergroup.com/reports/Linux/>

An Alternative to Windows
<http://www.technologyreview.com/articles/04/09/roush0904.asp>

OpenSource en el Tercer Mundo

Promoting the adoption and use of FOSS in developing countries
<http://business.newsforge.com/article.pl?sid=04/03/24/1751217&tid=110&tid=85&tid=35&tid=3&tid=31>

Constraints against the adoption and use of FOSS in developing countries
<http://business.newsforge.com/article.pl?sid=04/03/24/1744251&tid=110>

Free software and good user interfaces (Abril 2002)
<http://www106.pair.com/rhp/free-software-ui.html>

The State of OSS Documentation

http://www.osnews.com/story.php?news_id=5180

SITIOS WEB DE ORGANIZACIONES DE SOFTWARE ABIERTO

Foundation for a Free Information Infrastructure (FFII) <http://www.ffi.org/>

Public Patent Foundation <http://www.pubpat.org/>

Center for the Public Domain <http://www.centerforthepublicdomain.org/>

Digital Consumer.org <http://www.digitalconsumer.org/>

Stanford Copyright and Fair Use (Stanford University Libraries) <http://fairuse.stanford.edu/>

Chilling Effects Clearinghouse <http://www.chillingeffects.org/>

Internet Free Expression Alliance <http://www.ifea.net/>

Creative Commons <http://creativecommons.org/>

How to fight software patents - singly and together Richard M. Stallman (NewsForge - September 09, 2004)
<http://www.newsforge.com/article.pl?sid=04/09/09/1612239>

Electronic Frontier Foundation <http://www.eff.org/>