




MIGRACIÓN AL SOFTWARE LIBRE GUÍA DE BUENAS PRÁCTICAS

 **AIMME**
INSTITUTO TECNOLÓGICO
METALMECÁNICO

 **AIMPLAS**
INSTITUTO TECNOLÓGICO
DEL PLÁSTICO

 **ITI**
INSTITUTO TECNOLÓGICO
DE INFORMÁTICA

 **UNIVERSITAT
POLITÈCNICA
DE VALÈNCIA**



**PROYECTO
SOURCE PYME**

SOFTWARE LIBRE EN LA EMPRESA

 **GENERALITAT VALENCIANA**
CONSELLERIA D'INDÚSTRIA, COMERC I INNOVACIÓ



IMPIVA

 **UNIÓN EUROPEA**
FONDO EUROPEO DE
DESARROLLO REGIONAL


Autor: Daniel Sáez (dsaez@iti.upv.es)
Autor: Martín Peris (marpemar@iti.upv.es)
Autor: Ricard Roca (ricardroca@iti.upv.es)
Autor: David Anes (danes@iti.upv.es)
Organización: Instituto Tecnológico de Informática
Versión: 1.0
Fecha: Revisado 13-11-2007 18:15
Licencia: CC-by-sa 2.5


0. Licencia

Usted es libre de:

-  Copiar, distribuir y comunicar públicamente la obra
-  Hacer obras derivadas

Bajo las condiciones siguientes:

 **Reconocimiento.** Debe reconocer los créditos de la obra de la manera especificada por el autor o el licenciador (pero no de una manera que sugiera que tiene su apoyo o apoyan el uso que hace de su obra).

 **Compartir bajo la misma licencia.** Si altera o transforma esta obra, o genera una obra derivada, sólo puede distribuir la obra generada bajo una licencia idéntica a ésta.

- Al reutilizar o distribuir la obra, tiene que dejar bien claro los términos de la licencia de esta obra.
- Alguna de estas condiciones puede no aplicarse si se obtiene el permiso del titular de los derechos de autor
- Nada en esta licencia menoscaba o restringe los derechos morales del autor.

Para ver la licencia completa, visite:

<http://creativecommons.org/licenses/by-sa/2.5/es/legalcode.es>



Aviso Legal

Microsoft Windows y el logotipo de Windows son marcas registradas de Microsoft Corp. Apple, Apple Computer, MacOS X y AppleTalk son marcas registradas de Apple Computer Inc.

Linux es una marca registrada de Linus Torvalds.

Otras marcas, logotipos o nombres comerciales de software aparecidos en esta guía son propiedad de sus respectivos dueños.

1. Índice

Contenido

- 0. Licencia
- 1. Índice
- 2. Prólogo
- 3. Motivación
- 4. Requisitos
 - 4.1. Estado actual
 - 4.1.1. Descripción general de la empresa
 - 4.1.2. Aspectos técnicos
 - 4.1.3. Aspectos de recursos humanos
 - 4.1.4. Aspectos Legales
 - 4.1.5. Recursos temporales
 - 4.1.6. Recursos económicos
 - 4.2. Objetivos
- 5. Planificación
 - 5.1. Planificación técnica
 - 5.1.1. Cosas a tener en cuenta
 - 5.1.2. Inventario
 - 5.1.3. Diagrama de red
 - 5.1.4. Diagrama de estructura
 - 5.1.5. Elección de la estrategia de migración
 - 5.2. Planificación de comunicaciones
 - 5.3. Planificación de recursos humanos
 - 5.3.1. Miedo a lo desconocido

- 5.3.2. El temor de que el CV pierda importancia
 - 5.3.3. Saber es poder
 - 5.4. Plan de contingencia
 - 5.5. Planificación temporal
 - 5.5.1. Planificación de pruebas
 - 5.6. Plan de evaluación
 - 5.7. Planificación económica
- 6. Implantación
 - 6.1. Formación
 - 6.1.1. ¿Cómo realizar la formación?
 - 6.2. Implantación técnica
 - 6.2.1. Instalando muchos equipos
 - 6.2.2. Migrando datos de usuarios a sistemas GNU/Linux
 - 6.2.3. Realización de copias de seguridad
 - 6.2.4. Emulación de aplicaciones
 - 6.2.5. Servidores de archivos
 - 6.2.6. Bases de datos
 - 6.2.7. Sistemas de monitorización y administración
 - 6.2.8. Otros elementos a migrar
 - 6.3. Consejos de implantación
 - 6.3.1. Introducir nuevas aplicaciones en un entorno familiar
 - 6.3.2. Lo fácil primero
 - 6.3.3. Mirar hacia adelante
- 7. Evaluación
- 8. Casos de éxito
 - 8.1. Proyecto SourcePYME
 - 8.1.1. EBIR Iluminación
 - 8.1.2. Resimol
 - 8.1.3. Mipesa Mecanizados, S.L.
 - 8.1.4. Instituto Valenciano de la Edificación (I.V.E.)
 - 8.2. Casos de éxito en España
 - 8.2.1. INES: Iniciativa Española de Software y Servicios
 - 8.2.2. Proyecto MORFEO

- 8.3. Distribuciones españolas de GNU/Linux
 - 8.3.1. Lliurex
 - 8.3.2. LiNEX
- 8.4. OpenBravo
- 8.5. Ikusnet
- 9. Promotores de Software Libre
- 10. Software recomendado por SourcePYME
 - 10.1. Software de servicios y administración
 - 10.2. Aplicaciones de escritorio
- 11. Directorio de software
 - 11.1. Servicios y administración
 - 11.1.1. Servidores Web
 - 11.1.2. Servidores DNS
 - 11.1.3. Servidores FTP
 - 11.1.4. Servidores DHCP
 - 11.1.5. Proxys
 - 11.1.6. Cortafuegos (firewalls)
 - 11.1.7. Antivirus
 - 11.1.8. Servidores de impresión
 - 11.1.9. Interconectividad con Microsoft Windows
 - 11.1.10. Sistemas de autenticación y directorio
 - 11.1.11. Servidores de correo electrónico
 - 11.1.12. Servidores de mensajería instantánea
 - 11.1.13. Sistemas de sincronización y copias de seguridad
 - 11.1.14. Sistemas de redes, monitorización y alarmas
 - 11.1.15. Software de virtualización y emulación
 - 11.1.16. Servidores RDBMS
 - 11.1.17. Software de administración y control remoto
 - 11.1.18. Software de gestión de negocios (ERP y CRM)
 - 11.1.19. Sistemas de creación de portales e intranets
 - 11.1.20. Software de gestión de proyectos y groupware
 - 11.1.21. Sistemas de formación (web)
 - 11.1.22. Servidores de control de versiones
 - 11.1.23. Sistemas de control de incidencias
 - 11.2. Aplicaciones de escritorio

- 11.2.1. Ofimática
- 11.2.2. Internet
- 11.2.3. Reproducción Multimedia
- 11.2.4. Edición multimedia
- 11.2.5. Gestión
- 11.2.6. Diseño e imagen
- 11.2.7. Desarrollo de aplicaciones
- 11.3. ¿Dónde encontrar más aplicaciones?
 - 11.3.1. Forjas de software
 - 11.3.2. Directorios de software en línea
 - 11.3.3. Repositorios de software
- 12. Bibliografía

2. Prólogo

El Software Libre es gratuito, seguro, aporta calidad, se basa en estándares abiertos, tiene libertad para utilizarse en cualquier sitio, favorece la cultura de la colaboración, aumenta la capacidad tecnológica, proporciona ahorro de gastos en informática, reduce la dependencia de proveedores y fomenta el desarrollo de la empresa local.

Todas estas son algunas de las ventajas que les aporta el uso del software libre a las pymes de la Comunidad Valenciana que optan por incorporarlo, unos beneficios que el proyecto SourcePYME, coordinado por AIMME en cooperación con AIMPLAS, ITI y UPV y promovido por el IMPIVA, trata de fomentar, facilitando el uso de este software en las pymes con el objetivo de que éstas ganen capacidad tecnológica y reduzcan costes.

En la Conferencia Internacional de Software libre 3.0, celebrada recientemente en Badajoz, un experto afirmó que en Extremadura han llevado a cabo la introducción del software libre en los ámbitos educativos (Linex) y empresariales (LinexPyme), simplemente porque han sido los primeros en pasar de las palabras a la acción. Esto significa que todo el mundo habla del software libre, pero en realidad pocos ámbitos los están incorporando en la vida cotidiana. En el caso de la Comunidad Valenciana, también se empezó a extender hace unos años en el ámbito educativo (Lliurex) y, en el empresarial, se está llevando a cabo desde finales de 2005 a través del Proyecto SourcePYME.

Sin embargo, durante los primeros meses de la puesta en marcha de este proyecto, han ido aflorando diversos problemas que suponen un freno a la innovación (hacer viable la tecnología comercialmente, es decir, que exista oferta y demanda) y a la difusión (hacer viable la tecnología económicamente, o lo que es lo mismo, que se gane dinero con ello).

En el caso de la demanda, es decir, de las empresas usuarias de software libre, los problemas identificados están relacionados con la piratería, el miedo al cambio y la desconfianza. En España, el índice de piratería se sitúa en el 46 %, lo que representa unas pérdidas para el sector TIC de unos 600 millones de euros; las empresas confunden el software libre con el gratuito y algunas compañías descartan su implantación, bien porque no hay software libre con similares niveles de calidad o bien porque desconfían que haya empresas detrás que garanticen el mantenimiento y soporte de dicho software.

Para evitar todos estos inconvenientes, por una parte habría que eliminar por completo el software pirata y optar por utilizar por ejemplo un 60 % de software libre y un 40 % de software propietario con licencia y, por otra, se podría optar por migrar a software libre decenas de aplicaciones ejecutándolas sobre Windows.

Los problemas identificados en el caso de la oferta, es decir, de las empresas proveedoras de aplicaciones y servicios de software libre, se resumen básicamente en dos: el miedo al cambio y la falta de espíritu colaborativo. Y es que generalmente, las empresas de informática están acostumbradas a desarrollar software a medida, sin suministrar las fuentes a sus clientes manteniendo un modelo de pago por licencias de uso. Con la aparición del software libre, algunas compañías están viendo que un modelo de negocio sostenible es basarse en el cobro por servicios y no por licencias, de forma que están procediendo a liberar parte del código que hasta la fecha ha sido cerrado. Por ello, una posibilidad que inicialmente podría ser utilizada por muchas empresas sería ofertar dos soluciones a sus clientes, la propietaria (con coste de licencia) y la libre (sin coste de licencia), aunque existen muchos otros modelos de negocio asociados al software libre.

Por otra parte, las empresas informáticas también están acostumbradas a desarrollar software de forma individual, y en este sentido una mejor opción sería apostar por el modelo colaborativo para desarrollo de software con el fin de reaprovechar el trabajo de otros, colaborar con ellos y trasladar al cliente final los ahorros en licencias.

En resumen, nos encontramos ante la revolución del conocimiento abierto que está cambiando los paradigmas de crecimiento de la mayoría de sectores empresariales, entre ellos el sector informático. Las empresas que sepan adaptarse, y vean en ello una oportunidad, mediante la cooperación competitiva pueden llegar a alcanzar metas más lejanas que de forma individual. Tal y como decía John Perry Barlow, en la pasada edición del PowerfulIdeasSummit en la Universidad Politécnica de Valencia, “en una jerarquía el poder lo tiene quien guarda secretos; en una red el poder lo obtiene quien disemina información”.

Santiago Bonet Boldú

Responsable de la Unidad de Tecnologías de la Información y las Comunicaciones (TIC) de AIMME
sbonet@aimme.es

Eduardo Requena Algueró

Responsable de la Unidad de Tecnologías de la Información y las Comunicaciones (TIC) de AIMPLAS
erequena@aimplas.es

Daniel Sáez Domingo

Responsable del Área de Innovación Tecnológica
ITI - Instituto Tecnológico de Informática
dsaez@iti.upv.es

3. Motivación

Nos encontramos en uno de esos momentos en los que toda la industria puede estar cambiando de paradigma. En una industria cada vez más competitiva las empresas deben adaptarse y ajustarse a las variaciones para poder mantener su posición de competitividad. Durante años, en el mundo del software, se ha vivido bajo un modelo de negocio muy lucrativo para las empresas de desarrollo de software y muy poco ventajoso para el resto de usuarios de software, el modelo de software propietario.

¿En qué consiste el modelo de software propietario? El modelo de software propietario consiste básicamente en que un programador escribe un programa (código fuente) lo compila y obtiene un fichero ejecutable (un conjunto de ordenes que son entendidas y llevadas a cabo por una computadora), este ejecutable se vende a un usuario que de esta manera puede comenzar a utilizar el programa. Pero al usuario se le ponen ciertas restricciones para usar ese software, como por ejemplo el pago de una licencia de uso cada cierto tiempo o que no pueda redistribuir el programa (dado que esto se considera “piratería” y es un delito).

¿Qué desventajas tiene este modelo de software? El software propietario tiene varias desventajas, entre ellas desventajas económicas como por ejemplo el pago de elevadas sumas en concepto de “licencia de uso”. Aunque también hay otras desventajas más graves. El no tener acceso al código fuente provoca que el usuario sea completamente dependiente del proveedor de software. ¿Qué significa esto? Significa que si el software tiene defectos o se nos ocurre alguna mejora, debemos ponernos en contacto con el proveedor para que lo corrijan o lo amplíen. El proveedor y solo el proveedor es capaz de subsanar un error o ampliar el funcionamiento. Otra desventaja de no tener acceso al código fuente es que nos impide aprender de él al no saber cómo se hacen las cosas.

Si la dependencia tecnológica de un único proveedor, los precios abusivos y las barreras intelectuales que plantea este modelo de negocio no le parecen claras desventajas, éste es el punto adecuado para dejar de leer esta guía.

Existe una alternativa, el software libre. El software libre es una cuestión de libertad, no de precio. Para comprender este concepto, se debe pensar en la acepción de libre como en “libertad de expresión” y no como en “barra libre de cerveza”¹.

Con software libre se quiere hacer referencia a la libertad de los usuarios para ejecutar, copiar, distribuir, estudiar, modificar, mejorar el software e incluso venderlo.

El movimiento del software libre empezó en 1984, cuando Richard Stallman dio a conocer el proyecto GNU², para crear una versión del sistema operativo Unix que respetara las libertades de aquellos que lo utilizaran. La consecuencia de este proyecto fue la creación en 1985 de la *Fundación del Software Libre*³.

Cuando hablamos de libertad, nos referimos especialmente a cuatro clases de libertad para los usuarios de software:

- **Libertad 0:** la libertad para ejecutar el programa sea cual sea nuestro propósito. (privado, educativo, público, comercial, etc.)
- **Libertad 1:** la libertad para estudiar el funcionamiento del programa y adaptarlo a nuestras necesidades. El acceso al código fuente es condición indispensable para esto.
- **Libertad 2:** La libertad para redistribuir copias y ayudar así a tu vecino.
- **Libertad 3:** La libertad para mejorar el programa y luego publicarlo para el bien de toda la comunidad. El acceso al código fuente es condición indispensable para esto.

Llegados a este punto, y debido al modelo de software al que estamos acostumbrados, nos pueden asaltar muchas dudas y muchas personas pueden sentirse tentadas a lanzar afirmaciones como las siguientes:

“Nadie lo usará debido a que es gratuito, ya que esto significa que no cuenta con ninguna asistencia”

“Se tiene que cobrar por el programa para pagar el servicio de soporte”.

Si se prefiere pagar por un software libre y su servicio en lugar de obtener el software libre gratis y sin servicio, entonces una empresa que ofrezca servicio a las personas que utilicen software libre obtenido gratuitamente debería tener beneficios.

Debemos distinguir entre asistencia bajo la forma de un trabajo real de programación y lo que es simplemente llevar de la mano al usuario. En relación a la primera no se puede depender del vendedor de software. Si su problema no es compartido por un número suficiente de clientes, el proveedor le ignorará.

Si su negocio requiere poder confiar en la asistencia, la única manera es tener el código fuente y todas las herramientas necesarias. De este modo, se puede contratar a cualquier

¹ Software libre viene del Inglés “free software” donde “free” significa “libre” o “gratis” según el contexto, la definición que nos interesa es la de “libre”.

² Del acrónimo recursivo en inglés *GNU is Not Unix*, GNU no es Unix

³ La FSF es una organización sin ánimo de lucro con la misión de difundir y proteger el software libre en todo el mundo.

persona disponible para corregir el problema y no estar a merced de ningún proveedor (Independencia Tecnológica).

Por otra parte los usuarios sin conocimientos informáticos necesitan que se les lleve de la mano: hacer cosas por ellos, que ellos mismos podrían hacer fácilmente, pero que no saben cómo hacerlas. En este sentido hay compañías que ofrecen servicios de asesoría y reparación. Si los usuarios están de acuerdo en gastar dinero y obtener un producto con servicio, estarán igualmente de acuerdo en adquirir el servicio habiendo obtenido el producto de forma gratuita.

“¿Por qué voy a invertir dinero en algo que puede aprovechar mi competencia?”

Estamos malinterpretando el significado de competencia. La competencia hace que las cosas se hagan mejor. Si una empresa de la competencia es mejor, entonces intentemos hacerlo nosotros mejor aún, pero no les pongamos la zancadilla, porque si hoy ponemos la zancadilla, mañana nos la pondrán a nosotros. La competencia se puede comparar con una carrera en la que se premia al ganador. Al premiar al ganador, se está alentando a todos a correr más rápido. Cuando el capitalismo realmente trabaja de esta manera, hace un buen trabajo. Pero nos equivocamos al dar por hecho que esto siempre es así. Si los corredores de la carrera olvidan por qué se otorga el premio y se centran en ganar y solo en ganar, sin importar como, pueden encontrar otras estrategias, como por ejemplo atacar a los demás corredores. Si los corredores se enredan en una pelea, todos llegarán tarde a la meta. El software propietario y secreto es el equivalente moral de los corredores en una pelea a puñetazos.

Hay que tener en cuenta, además, que si el desarrollo de software libre parte de nuestra empresa, y dado que somos los responsables de él, las empresas que utilicen este software confiarán generalmente en nosotros para realizar tareas de mantenimiento, adecuación y gestión de dicho software.

El software libre también tiene limitaciones, algunos componentes de software que son de interés para las empresas todavía no están desarrollados completamente, pero cada vez más gente es consciente de que el cambio está aquí y el futuro ya ha llegado.

Apostemos por el cambio.

4. Requisitos

Un factor crucial para el éxito de la migración es el análisis en profundidad de la situación de partida. Esta tarea usualmente consumirá gran parte de los recursos iniciales del proyecto, tanto en tiempo como en mano de obra. De todas maneras, un conocimiento detallado de los documentos o las aplicaciones de base de datos evita realizar ajustes imprevistos durante la migración y permite el establecimiento de planes de actuación con suficiente antelación. Además, la determinación de la situación de partida es también la base para identificar los requisitos funcionales del nuevo sistema. Aspectos importantes a tener en cuenta en este contexto incluyen, por ejemplo, los siguientes:

- Bases de datos y estructuras de datos
- Documentos y formatos de documentos
- Aplicaciones y sus interfaces
- Funcionalidades disponibles
- Disponibilidad de datos y aplicaciones
- Atajos y problemas
- ...

En este punto vamos a dar una visión global de qué es lo que debemos saber sobre la empresa, sus sistemas de información y su funcionamiento, para maximizar las posibilidades de éxito en una migración a software libre.

Hay que tener en cuenta que la presente guía es sólo una *guía de buenas prácticas*, en ella se describen una serie de pasos para llevar a cabo una migración pero sin entrar en los detalles. Para más detalles técnicos se recomienda la lectura de la bibliografía citada en el punto 5.

Para clarificar un poco el contenido de esta guía, se van a utilizar ejemplos basados en una empresa ficticia en aquellos puntos que se consideren de interés concreto.

4.1. Estado actual

La primera tarea a llevar a cabo es la de determinar cuál es el estado actual de la empresa, intentando recopilar la mayor cantidad de información posible. Esta información nos permitirá conocer en profundidad todo el entorno que estemos intentando migrar, lo que nos permitirá elaborar los informes necesarios para que las personas encargadas de tomar las decisiones puedan tomar las decisiones óptimas en cada caso. Para esta recopilación de datos podemos servirnos de plantillas de preguntas e incluso de software especializado.

4.1.1. Descripción general de la empresa

Cuanto más profundamente se comprenda la actividad de la empresa, más posibilidades hay de encontrar la solución óptima para una migración. Sobre todo si la migración se va a llevar a cabo por personal externo a la compañía.

Por eso se debe realizar una descripción previa de la empresa en la que se mencione a qué actividad se dedica la empresa, cuántos años de experiencia tiene en el sector, cuántos empleados tiene, cuales son sus objetivos y toda la información que se considere de interés.

En los siguientes puntos se comentan los diferentes aspectos en los que se debe profundizar para poder llevar a cabo una buena planificación que garantice el éxito de la migración.

4.1.2. Aspectos técnicos

4.1.2.1. El inventario del software

Realizar un inventario de software de la organización. Esto es, un listado con todos los programas que se utilicen en los equipos a migrar. El inventario dependerá del tipo de sistemas que se vayan a migrar, si se van a migrar los servidores se hará un inventario del software que se utiliza en dichos servidores, si se migran los equipos de escritorio hacer un inventario de todo el software que hay en esos equipos. Esto nos servirá para identificar todas las aplicaciones, servicios y configuraciones especiales que se necesitan tener en cuenta en el plan de migración.

Las cuestiones clave que debe responder el inventario son las siguientes:

- ¿Qué aplicaciones de terceras partes están instaladas y se utilizan?

Esto generará una lista de software incluyendo las versiones utilizadas y los potenciales parches aplicados.

- ¿Qué software desarrollado por la empresa se utiliza?

Esto resultará en una lista de software desarrollado y mantenido internamente en la compañía, que puede necesitar ser portado a GNU/Linux o a un entorno independiente de la plataforma.

- ¿Qué aplicaciones requieren acceso a datos externos?

Esto resultará en una lista de software que accede a servidores de ficheros, servidores de aplicaciones, servidores web, bases de datos, mainframes o cualquier otra implementación de proceso de datos.

- ¿Hay definidos grupos de usuarios? ¿Cómo se caracterizan?

Esto debería proporcionar una visión global de si hay grupos o usuarios típicos, y de ser así cómo se agrupan. La agrupación se puede hacer por departamentos, aplicaciones que se utilizan, tipo de trabajo o responsabilidad en el negocio.

- ¿Qué software relacionado con la seguridad se utiliza? ¿Qué procesos y reglas de seguridad se aplican?

Esto dará una vista de qué productos se utilizan para asegurar los PC, como por ejemplo antivirus, seguridad de escritorio, escaneo de puertos, así como reglas de cómo se instalan dichas aplicaciones, cómo se mantienen, actualizan y cómo se instruye al usuario para que las utilice. También se deben incluir las políticas de aplicación de parches de seguridad de cualquier componente del sistema operativo o cualquier aplicación instalada.

Este inventario nos servirá después para identificar aplicaciones que pueden ser migradas, las que no, las que pueden serlo parcialmente y las que no se utilizan o no son necesarias, y proporciona una información de partida para poder realizar una migración consistente y homogénea.

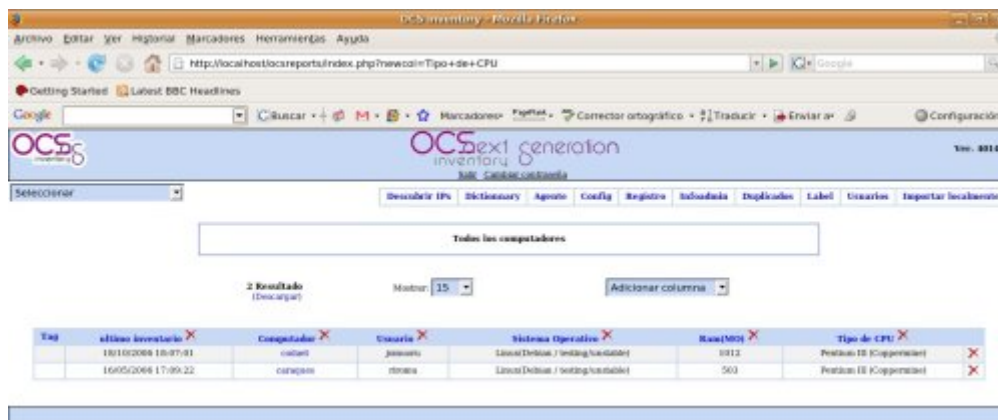
Importante

¿Qué es una aplicación *no migrable*? Vamos a considerar una aplicación *no migrable* cuando una o más de las siguientes afirmaciones a cerca de un software sean ciertas:

- No existe una versión software libre o una alternativa a la aplicación.
- No es factible portar la aplicación a software libre.
- Las restricciones en la licencia de la aplicación hagan su migración imposible o muy cara.

El inventario de software puede ser realizado a mano, examinando el contenido de los equipos, pero cuando se dispone de una gran cantidad de equipos o ningún control sobre

el software que los usuarios han podido ir instalando, el proceso puede ser demasiado costoso. Para hacer este proceso menos costoso, se recomienda el uso de sistemas de inventariado automático. Se pueden utilizar aplicaciones específicas para realizar esta tarea, como el *OCS Inventory*⁴.



The screenshot shows the OCS Inventory web interface. The browser address bar displays the URL: `http://local/ocsreports/index.php?menu=Tipo+de+CPU`. The page header includes the OCS Inventory logo and navigation tabs such as 'Descubrir IPs', 'Diccionario', 'Agentes', 'Config', 'Registro', 'Indicadores', 'Duplicados', 'Labels', 'Usuarios', and 'Importar localmente'. Below the header, there is a search bar and a table of results. The table has 7 columns: 'ID', 'último inventario', 'Computador', 'Usuario', 'Sistema Operativo', 'RAM(MB)', and 'Tipo de CPU'. Two rows of data are visible.

ID	último inventario	Computador	Usuario	Sistema Operativo	RAM(MB)	Tipo de CPU
18102088	18-07-01	cafeal	jesusito	Linux Debian / testing/instable	512	Pentium III (Kopernicus)
16052066	17-09-22	caracas	stina	Linux Debian / testing/instable	503	Pentium III (Kopernicus)

Figura 1: Ejemplo de OCS Inventory, accediendo a los datos de inventario vía Web.

En caso de no poder llevar a cabo un inventariado automático puede ser de utilidad establecer una categorización de software para así poder llevar un orden en el inventariado y poder identificar mejor los grupos de aplicaciones de interés. Una posible categorización puede la que se muestra a continuación.

⁴ En <http://www.ocsinventory-ng.org/> podrá encontrar más información. Se pueden encontrar más proyectos similares en <http://www.sourceforge.net> haciendo una búsqueda con las palabras clave “software inventory”*

Categorías

- Sistemas
 - Sistemas operativos
 - Antivirus
 - Backup
 - Compatibilidad Windows
 - Proxy/Firewall
 - Servidor Web/FTP
 - Servidores correo electrónico
 - Comunicaciones
 - Clientes correo electrónico
 - Clientes FTP/SCP
 - Control remoto
 - Envío / Recepción Faxes
 - Mensajería instantánea
 - Navegador Web

- Ofimática
 - Agendas y calendarios
 - Compresores
 - Diagramas
 - Diccionarios
 - Encriptación
 - Multimedia
 - Paquetes
 - PDF
 - Traductores

Categorías (continuación)

- Diseño
 - 3D
 - CAD / CAM / CAE
 - Editores de imágenes simples, vectoriales o avanzados
 - Bases de Datos
 - Servidores de bases de datos
 - Gestión
 - CRM y ERP
 - e-Learning
- Finanzas
 - Gestión de la producción (GPAO)
 - Gestión de proyectos
 - Gestión del conocimiento
 - Trabajo en grupo
 - OLAP
 - Punto de venta
 - Gestión documental

4.1.2.2. El inventario de hardware

Realizar un inventario completo de los sistemas que se hayan seleccionado para la migración. El inventario de hardware permitirá identificar cualquier incidencia con el soporte de hardware y nos ayudará a definir reglas para comprar o reemplazar sistemas en un futuro.

Las preguntas a realizar en este área serian las siguientes:

- ¿Qué hardware está en uso actualmente? Indicar el Vendedor, tipo y modelo.
- ¿El tipo de hardware está estandarizado? Si todas las máquinas son iguales, entonces el soporte de drivers y sistema operativo debería ser significativamente más sencillo.
- ¿Qué dispositivos periféricos están actualmente instalados y son requeridos por los usuarios? Esto incluye cualquier tipo de impresoras, escáneres o dispositivos especiales.

- ¿El soporte de GNU/Linux está incluido en los requisitos a los vendedores de hardware cuando se adquiere hardware nuevo?
- ¿Cuáles son los componentes clave del hardware requeridos actualmente por los usuarios? Por ejemplo, las máquinas pueden llevar tarjetas de sonido incorporadas, pero los drivers no están instalados dado que los usuarios no van a utilizar el sonido. De esta manera el soporte de sonido en GNU/Linux no estaría incluido.
- ¿A qué tipo de dispositivos extraíbles se debe dar soporte? Por ejemplo la sincronización de calendarios u otros datos con PDAs o smartphones puede ser un requisito. También, lápices USB, dispositivos Bluetooth y discos duros externos firewire se han hecho muy populares, aunque algunas organizaciones no los permiten debido a motivos de seguridad. Puede ser necesario dar soporte a dichos dispositivos o explícitamente *no* incluir el soporte para alguno de estos dispositivos.

Consejo

Se recomienda la utilización de sistemas de inventario automático. Como GLPI o OCSInventory. Se pueden encontrar diversos sistemas de estas características en <http://www.sourceforge.net> estableciendo “hardware inventory” como criterio de búsqueda.

Al realizar este inventario también se deben tener en cuenta las máquinas retiradas, normalmente la mayoría de herramientas basadas en software libre suelen requerir máquinas con pocos recursos, por ejemplo las herramientas de gestión de red (firewall, router, etc...) o servidores de impresión, incluso servidores de bases de datos o servidores web pueden ser ejecutados completamente en modo texto, de esta manera una máquina retirada por no poder ejecutar fluidamente el “pesado” software propietario puede convertirse en un servidor más que eficiente.

Se debe proporcionar el máximo nivel de detalle en el listado de hardware ya que esto nos permitirá saber de antemano si el hardware del que se dispone está soportado en Software libre de manera nativa o no. Por ejemplo algunas tarjetas de red inalámbrica (Wireless, también conocido como Wi-Fi) no disponen de un driver nativo para sistemas operativos basados en software libre, pero se pueden hacer funcionar gracias a emuladores que permiten ejecutar el driver propietario.

Si no se dispone de un sistema automático de inventariado de hardware se propone la siguiente plantilla para rellenar con los datos de cada equipo.

Plantilla

- Nombre del equipo:
- Comentarios:
- Sistema Operativo:
 - Nombre:
 - Versión:
 - Service Pack:
- Procesador:
 - Tipo:
 - Velocidad:
 - Num. procesadores:
- Memoria RAM (Una lista con la siguiente información):
 - Descripción (DIMM, SIM...):
 - Capacidad:
 - Velocidad (MHz):
 - Número de Ranuras:
- Almacenamiento (Lista):
 - Fabricante:
 - Modelo:
 - Descripción (IDE, SCSI...):
 - Tipo (CD-ROM, HD, DVD-ROM...):
 - Tamaño (MB):
- Particiones (Lista):
 - Letra:
 - Tipo (Primaria, Secundaria...):
 - Sistema de archivos (Ext3, NTFS, VFAT...):
 - Tamaño (MB):
- Dispositivos de entrada (Lista):

Plantilla (continuación)

- BIOS:
 - Número serie:
 - Fabricante:
 - Modelo:
 - Versión:
 - Fecha:
- Sonido:
 - Fabricante:
 - Nombre:
 - Descripción:
- Tarjeta vídeo:
 - Nombre:
 - Chipset:
 - Memoria (MB):
 - Resolución:
- Monitor
 - Nombre:
 - Fabricante:
 - Tipo (CRT, TFT...):
 - Resolución:
- Red (Lista)
 - Descripción:
 - Tipo (Ethernet, Fast Ethernet..):
 - Velocidad:
 - Dirección MAC:
 - Dirección IP:
 - Máscara:
 - Puerta de enlace:
 - Numero de red:

Plantilla (continuación)

- Puertos (Lista)
 - Tipo (USB, paralelo, FireWire...):
 - Nombre:
 - Libre (Si, No):
 - Descripción:
- Impresora
 - Nombre:
 - Fabricante:
 - Modelo:

Importante

Se deben incluir también en el inventario todos aquellos dispositivos pertenecientes a terceros, como por ejemplo un Router perteneciente al ISP.

4.1.2.3. Diagrama de estructura

Es conveniente tener una idea clara de donde están ubicados todos los equipos que se van a migrar y realizar un diagrama de estructura que describa estas localizaciones. Algunas cuestiones que se pueden plantear para entender mejor este concepto son:

- ¿Todos los ordenadores se encuentran en la misma sala?
- ¿Existe una sala en la que se encuentren todos los servidores de la empresa (en caso de que hayan)?
- ¿Hay equipos distribuidos en diferentes salas, despachos, pisos de un edificio o incluso diferentes edificios?
- ¿Cómo están distribuidas las impresoras y otros periféricos de uso común?

Esta información puede resultar relevante si la empresa posee gran cantidad de equipos y están repartidos en diferentes localizaciones, a la hora de planificar las tareas de migración.

En un diagrama de estructura se muestra la distribución de los equipos en la empresa, mediante iconos fácilmente identificables. En cierto sentido es parecido a un plano arquitectónico, aunque en este tipo de planos no es necesario que las medidas sean exactas,

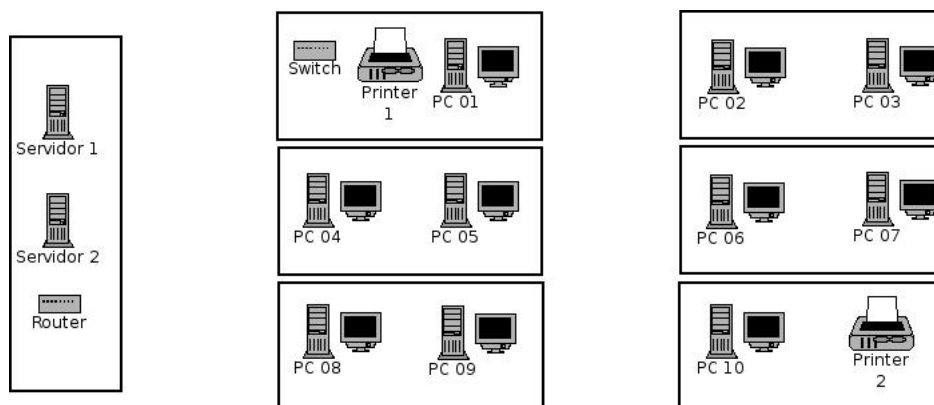


Figura 2: Ejemplo de diagrama de estructura

solo con representar esquemáticamente la distribución física de los equipos en la empresa es suficiente.

Consejo

Para realizar estos diagramas se puede utilizar una herramienta del estilo de Dia o TCM.

4.1.2.4. Diagrama de red

Un diagrama de red representa los nodos y las conexiones entre nodos en una red de ordenadores o, más generalmente, en cualquier red de telecomunicaciones.

Iconos fácilmente identificables se utilizan para representar aplicaciones de red usuales, como por ejemplo un enrutador, y el estilo de líneas entre los nodos indican el tipo de conexión. Las nubes se utilizan para representar redes externas a la red que se está dibujando con el objetivo de representar las conexiones entre dispositivos internos y externos, sin indicar los detalles de la red exterior. Por ejemplo, en la hipotética red de área local (LAN) que hay más abajo, hay 3 ordenadores personales y un servidor conectado a un *switch*, al servidor también se conecta una impresora y una pasarela *router*, la cual está conectada a través de un enlace WAN a Internet.

Dependiendo de si el diagrama está previsto para un uso formal o informal, ciertos detalles pueden estar ausentes y ser determinados por el contexto. Por ejemplo, el diagrama de ejemplo no indica el tipo de conexión física entre los PCs y el *switch*, pero dado que se trata de una LAN moderna, se puede asumir que se utilizan el estándar *ethernet*.

A diferentes escalas, los diagramas pueden representar varios niveles de granularidad de red. A nivel de LAN, los nodos individuales pueden representar dispositivos físicos individuales, como *hubs* o *servidores de ficheros*, mientras que a nivel de WAN, los nodos individuales pueden representar ciudades enteras.

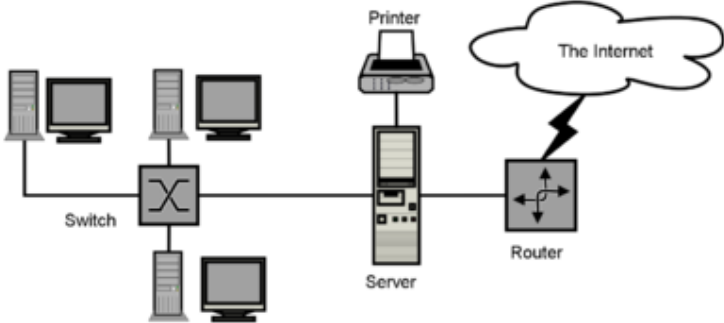


Figura 3: Ejemplo de diagrama de red

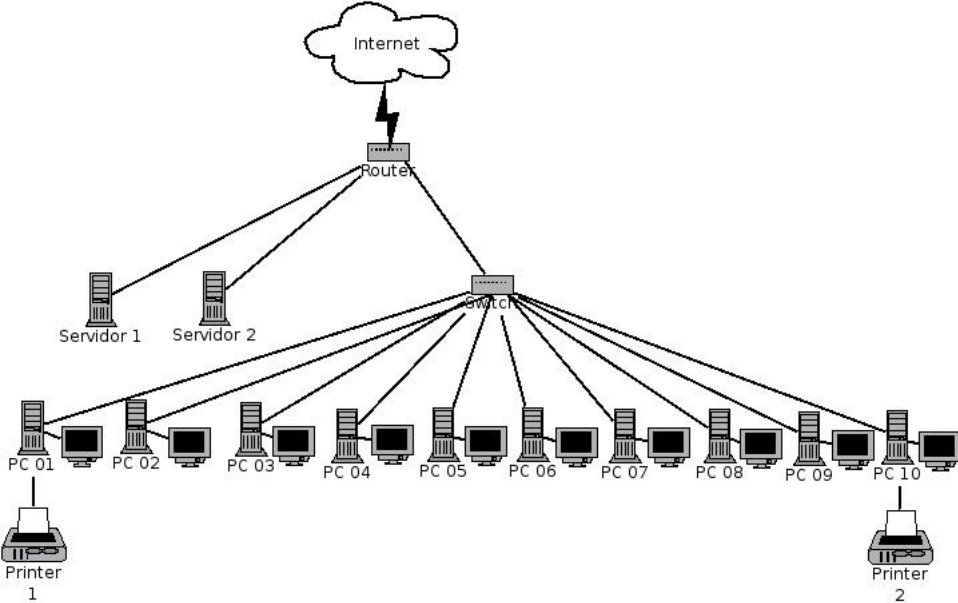


Figura 4: Diagrama de red de una organización conectada a Internet

A partir de un cierto tamaño las redes se convierten en algo difícil de visualizar sin ayudas gráficas. Cuanto más grande es la red más difícil es entenderla como un todo. Los diagramas de red ayudan entender mejor las redes de conexión partiéndolas en trozos más manejables. Una red grande puede ser resumida por un punto de vista muy amplio, por ejemplo mostrando solo las grandes oficinas con los *backbones* principales entre ellas. Después cada oficina puede ser expandida en otro diagrama que revele mas detalles sobre la red.

Uno de los aspectos negativos de los diagramas de red es que requieren una inversión de tiempo en su creación y una vez creados requieren también esfuerzo para mantenerlos actualizados. Pero son una herramienta de mucho valor en situaciones como las de una migración a software libre.

Consejo

Los diagramas de red pueden expresar más en pocos minutos que hablar sobre la red durante días enteros. Existe software que facilita la tarea de crear diagramas de red como por ejemplo Dia. Se puede encontrar ejemplos de diagramas de red realizados por diferentes personas en <http://www.ratemynetworkdiagram.com>

4.1.2.5. Listado de formato de datos

Para la mayor parte de aplicaciones cliente-servidor, el único requisito es la disponibilidad de un reemplazo funcional de la parte cliente que se ejecute nativamente en GNU/Linux. Un ejemplo puede ser una aplicación que utiliza una interfaz Web para acceder a datos almacenados en el servidor. Siempre que la interfaz Web pueda ser ejecutada en un Navegador Web para GNU/Linux, entonces la migración de la parte del cliente se convierte en algo trivial.

Para algunas aplicaciones (mayoritariamente aplicaciones locales y nativas), los datos pueden estar almacenados en *formatos propietarios* que requerirán un proceso de conversión. Las aplicaciones en esta categoría pueden incluir sistemas de correo electrónico (como por ejemplo Lotus Notes) o suites de productividad (Como Lotus Smartsuite o Microsoft Office). Sin entrar en aspectos técnicos durante la toma de requisitos se debe explorar el uso actual de dichas aplicaciones.

Como ejemplo, algunas de las cuestiones a tratar en este punto son:

- ¿Se utiliza Microsoft Office? De ser así, ¿Qué componentes y con qué frecuencia?
- ¿Se utilizan macros en Microsoft Office? Si es así, ¿Qué tipo de macros, para qué componentes y con qué frecuencia?
- ¿Se utiliza Microsoft Outlook? ¿Qué componentes y con qué frecuencia?
- ¿Se utiliza Microsoft Project? ¿Qué componentes y con qué frecuencia?

- ¿Se utiliza algún lenguaje de programación (como Visual Basic) para automatizar tareas en o entre aplicaciones?
- ¿Se utiliza Lotus Smartsuite? ¿Qué componentes y con qué frecuencia?
- ¿Se utiliza Lotus Notes? ¿Qué componentes y con qué frecuencia?
- ¿Se comparten datos con organizaciones externas? ¿En qué formatos y con qué frecuencia?

Algunas respuestas a las preguntas anteriores requerirán una revisión en profundidad de la infraestructura subyacente. Por ejemplo, el uso de Microsoft Outlook en el cliente frecuentemente nos lleva a que se utilice Microsoft Exchange en el servidor, mientras que Lotus Notes en el cliente usualmente indica Lotus Domino en el servidor. Para escenarios como estos, el software instalado en los servidores se deben tener en cuenta cuando se diseña una nueva pila de clientes y la migración de las cuentas de los usuarios ha de ser planificada. En caso de que no exista una alternativa libre (o cliente para GNU/Linux) para la comunicación con el servidor, puede ser necesario tener en cuenta la migración del servidor a una solución compatible con GNU/Linux antes de que la migración de los clientes a GNU/Linux pueda comenzar.

4.1.2.6. Aplicaciones a utilizar

Después de una migración, los usuarios en la mayoría de los casos tendrán que adaptarse a aplicaciones diferentes pero funcionalmente equivalentes a las actuales. Para poder “puentear” este salto, el cual puede llevar a una pérdida de productividad, es de utilidad desarrollar una estrategia mediante la cual los usuarios se familiaricen con las nuevas aplicaciones.

Algunas aplicaciones que se ejecutan de modo nativo en GNU/Linux también están disponibles nativamente para Windows. Estas aplicaciones proporcionan la oportunidad de minimizar los efectos de la transición y los requisitos de reentrenamiento de los usuarios provocados por una migración de Sistema Operativo. De esta manera es posible migrar aplicaciones que serán soportadas en GNU/Linux antes incluso de la migración del propio sistema operativo.

Sugerencia

Es importante ir siempre de lo fácil a lo difícil. Intentar en la medida de lo posible realizar una migración lo más escalonada posible, es conveniente acostumbrarse al uso de las nuevas aplicaciones antes de hacer efectiva la migración, para evitar pérdidas de productividad.

El beneficio de tales cambios antes de la migración es que de esta manera se le permite a los usuarios acostumbrarse a las nuevas aplicaciones antes de que se lleve a cabo la migración del Sistema Operativo.

Una vez que se instale el nuevo Sistema Operativo, los usuarios no experimentarán ningún cambio en absoluto en lo referente a las aplicaciones que utilizan.

La siguiente tabla proporciona algunos ejemplos de aplicaciones que pueden ser utilizadas como “puente” entre Microsoft Windows y GNU/Linux. Los equivalentes basados en GNU/Linux que se muestran en la segunda columna son ejemplos de aplicaciones que tienen versiones nativas tanto para Windows como para GNU/Linux.

Aplicación en Windows	Aplicación Puente
Microsoft Internet Explorer	Mozilla Firefox
Microsoft Outlook, Outlook Express	Mozilla Thunderbird, Evolution
Microsoft Word	OpenOffice.org Writer
Microsoft Excel	OpenOffice.org Spreadsheet
Microsoft PowerPoint	OpenOffice.org Impress
Adobe Photoshop	The GIMP
Cliente de Mensajería (MSN, Yahoo...)	Pidgin (Antes conocido como GAIM)

Las funcionalidades proporcionadas por las aplicaciones tales como navegadores de sistemas de ficheros, archivadores, y visores fuerzan que el diseño de estas herramientas esté más íntimamente ligado al Sistema Operativo de la máquina. No se pueden considerar como “aplicaciones puente” en el sentido en el que se describen las aplicaciones anteriores. Uno de los motivos por los que se dice que GNU/Linux está alcanzando la equivalencia a Windows es la disponibilidad de múltiples aplicaciones de utilidad. En muchos casos, estas aplicaciones tienen conjuntos de funcionalidades más potentes que sus aplicaciones equivalentes en Windows hoy en día.

Desafortunadamente no es posible encontrar aplicaciones “puente” (el caso ideal) o aplicaciones funcionalmente equivalentes para satisfacer todos los escenarios. Por ejemplo, aplicaciones ERP o CRM son especialmente susceptibles de implementar un cliente ligero para los cuales no hay equivalente entre Windows y GNU/Linux. Las empresas fabricantes de software no están respondiendo a esto implementando un cliente ligero para cada plataforma, si no centrándose en los Navegadores Web como contenedor para extender el funcionamiento de sus aplicaciones a las plataformas alternativas de sus clientes (como GNU/Linux).

Si una solución basada en Navegador Web no es factible, la opción de “puentear” las aplicaciones al nuevo escritorio mediante la migración en primer lugar a un cliente web multiplataforma no puede ser utilizada. En este caso puede que se tenga que migrar la aplicación a una versión más reciente y que sí soporte clientes multiplataforma o puede que se tenga que considerar la opción de cambiar a otro proveedor de software que sí proporcione clientes multiplataforma.

4.1.2.7. Funcionalidades necesarias

En este punto ya se dispone de un listado de aplicaciones que se utilizan y seguramente exista una alternativa para GNU/Linux en la mayoría de los casos, bien proporcionada por el mismo fabricante o bien desarrollada por la comunidad de software libre, pero para facilitar la tarea de decidir que software se adapta mejor a las necesidades concretas de la empresa es recomendable elaborar un listado de funcionalidades requeridas para el nuevo software. Este listado de funcionalidades nos permitirá más adelante cotejar las diferentes alternativas y decidir cual es la más conveniente.

Para la elaboración de la lista de funcionalidades necesarias se puede utilizar una plantilla como la siguiente:

- Título del software:
- Categoría (p.e. Diseño):
- Subcategoría (p.e. CAD/CAM/CAE):
- Descripción de funcionalidades:

4.1.2.8. Aplicaciones que querría utilizar

Se conoce muy bien qué software se utiliza en la empresa, pero una migración puede ser más que un mero cambio de sistema operativo o de aplicaciones por otras equivalentes.

Por eso se debe plantear también si existe alguna aplicación que se desearía utilizar pero que actualmente no se hace.

- ¿Existen aplicaciones que resuelvan mejor la problemática de la empresa?
- ¿Por qué no se utilizan? ¿Precio elevado de las licencias? ¿Desconocimiento en el uso?

Importante

Una migración no solo pretende cambiar unas aplicaciones propietarias por otras libres, también se busca mejorar, tanto en eficiencia como en calidad y prestaciones.

4.1.3. Aspectos de recursos humanos

Los proyectos de migración solo pueden tener sentido y solo pueden tener éxito a nivel de recursos humanos si los beneficios pueden ser claramente identificados y comunicados como algo esencial y necesario. Estos beneficios se derivan de la definición de los objetivos que se explica en el punto “[Objetivos](#)” de este documento.

El personal participante debe estar convencido de los beneficios del proyecto de migración para que apoyen e introduzcan el proyecto en cada departamento de la empresa. Al mismo

tiempo, los límites del software libre deben ser comunicados claramente y las razones para introducir software libre en la empresa deben ser explicadas.

El objetivo es asegurar un alto grado de aceptación y de esta manera fomentar la motivación y la satisfacción entre el personal de la empresa. Se debe prevenir que personal insatisfecho (gente con falta de información, motivación o formación) comprometa el éxito de toda la migración y difunda los fallos, de haberlos. A largo plazo, esto puede afectar a la eficiencia y el rendimiento de toda la empresa. Mas allá del “ejercicio obligatorio” de asegurar información sobre el estado del proyecto, los responsables también deben realizar un “ejercicio voluntario” de monitorizar el nivel de satisfacción del personal durante el proceso de migración para poder tomar las medidas oportunas, en caso de ser necesario.

Aunque el desarrollo de los conceptos y de las medidas de implementación son inicialmente trabajo de la gente encargada de gestionar la migración, esta solo se podrá desarrollar, implementar y, por supuesto, mejorar de manera continuada junto con todo el personal. El soporte, consejos o experiencias de organizaciones externas pueden ser de gran ayuda en la fase inicial.

4.1.3.1. Factores humanos

Dado que la migración de los escritorios afecta directamente a los usuarios finales, considerar los aspectos de recursos humanos en la estrategia de gestión del cambio es muy importante.

Es de esperar que un cambio radical en la interfaz de escritorio, a la que están acostumbrados los usuarios, causará distintos tipos de reacciones: desde aceptación entusiasta hasta rebelión extrema. Así que es muy importante mantener a los usuarios finales informados acerca de los nuevos desarrollos de una manera clara y concisa. Un plan de comunicaciones es clave.

Generalmente no es una buena idea hacer algún cambio en los entornos de trabajo de los usuarios finales sin su conocimiento, así que un buen plan de comunicaciones contribuirá a reducir los aspectos negativos del cambio para los usuarios y hará que lo acepten mejor.

Consejo

Un buen plan de comunicaciones combinado con la formación adecuada en las nuevas aplicaciones debería minimizar el número de usuarios que se oponen y repudian el cambio, haciendo al mismo tiempo que los usuarios afronten el cambio con aceptación en lugar de insatisfacción.

Con respecto al personal de soporte de TI, estos mismos aspectos son incluso más importantes. La decisión estratégica de cambiar de sistema operativo y la manera en la que se gestionan los clientes pueden causarles la impresión de que se desaprueba el trabajo que

han hecho. Al personal técnico puede darle la sensación de que sus conocimientos actuales están siendo devaluados. Probablemente no será fácil convencer a un administrador de sistemas Windows de que migrar a clientes GNU/Linux es una buena estrategia, a menos que se les convenza de que la organización esta preparada para realizar una importante inversión en ampliar y actualizar sus conocimientos actuales.

- Desarrollar un plan de comunicaciones
- Seleccionar un grupo piloto que se ajuste de la mejor manera posible a la tarea de la migración

Para ayudar a que los usuarios acojan el cambio con mayor aceptación puede contemplarse la posibilidad de otorgar incentivos tales como una renovación del hardware. En algunas administraciones que decidieron migrar a software libre se incentivó al personal a participar en el grupo piloto cambiando sus monitores CRT (Tubo de Rayos Catódicos) por un monitor TFT (Monitor plano), fomentando así una actitud positiva desde el principio al cambio a Software Libre.

4.1.3.2. La importancia de la formación

En lo que a la formación se refiere, los administradores deben estar integrados en una etapa temprana del proyecto y la formación de los futuros usuarios se debe realizar lo antes posible. Un plan específico de formación para cada grupo de usuarios debe ser desarrollado teniendo en cuenta tanto sus habilidades actuales, experiencia y cualificación así como los componentes específicos del trabajo que van a desarrollar.

Nota

Se debe tener especial atención con la formación en el lugar de trabajo a los encargados de ofrecer soporte a los usuarios durante la migración a software libre.

Además las experiencias de migraciones piloto u otros proyectos de migración deben ser tenidas en cuenta para hacer uso de las lecciones aprendidas.

Nota

La formación se vuelve aún mas importante si actualmente no se poseen conocimientos sobre software libre y una vez finalizada la migración no se va a tener soporte.

Durante el curso de la migración, la formación a los usuarios será necesaria en muchos casos. Dado que los cursos o clases son normalmente de coste elevado debido a tener que pagar a un profesor, y las horas de trabajo que pierden los empleados, se pueden plantear maneras de reducir un poco estos costes.

- a) Las aplicaciones puente pueden separar la formación de la migración

Refiriéndose a las aplicaciones puente mencionadas anteriormente, la estrategia de reemplazar las aplicaciones actuales con equivalentes de software libre que estén disponibles tanto en Windows como en GNU/Linux puede reducir los costes en formación.

- b) Aprender un nuevo “look and feel”

Otra estrategia para ahorrar costes es intentar mantener la apariencia de las aplicaciones actuales y del escritorio. Es posible personalizar ciertos aspectos de los escritorios GNOME y KDE (GNOME y KDE son entornos de ventanas de los escritorios GNU/Linux) para emular la apariencia del escritorio de Windows y de las aplicaciones basadas en Windows. Hay multitud de temas de escritorio disponibles para ser descargados y modificados libremente.

Consejo

Se pueden encontrar ejemplos para GNOME en <http://www.gnome-look.org> y para KDE en <http://www.kde-look.org>

- c) Acciones cotidianas

Emular las acciones es también una buena práctica. Un buen ejemplo es configurar un “doble clic” en lugar de un solo “clic” como evento para abrir iconos del escritorio.

- d) Sistema de ficheros: todo ha cambiado de sitio.

Los usuarios de Windows están acostumbrados al sistema de ficheros jerárquico basado en los puntos de montaje de sistemas de ficheros como C: o D:. El sistema de ficheros jerárquico de GNU/Linux difiere de esta convención, viéndose un único sistema de ficheros, sobre el que hay puntos de montaje de unidades físicas u otros sistemas de ficheros, como puede ser */mnt/floppy*, */mnt/cdrom* o */home*, por ejemplo. Los usuarios que se enfrenten a una migración pueden encontrar mucha confusión a la hora de entender la nueva jerarquía del sistema de ficheros de GNU/Linux. Para suavizar esta transición, un método recomendado es migrar el contenido completo de los archivos existentes en la carpeta “Mis Documentos” de los usuarios a carpetas de nombre similar en el directorio por defecto del usuario en GNU/Linux. Dentro de */home/nombre-de-usuario/Mis Documentos* el contenido y estructura de archivos y directorios aparecerán exactamente igual a como lo hacían en la carpeta original de Windows.

- e) Tomar contacto con GNU/Linux antes de la migración

Actualmente la mayor parte de distribuciones de GNU/Linux proporcionan “Live CD” o CD’s que cargan una distribución de GNU/Linux al encender el

ordenador. Uno de los pioneros en crear estas distribuciones “Live CD” fue Knoppix.

Qué es Knoppix

KNOPPIX es un CD auto-ejecutable con una colección de software GNU/Linux, detección de hardware automática, y soporte para la mayor parte de tarjetas gráficas, tarjetas de sonido, SCSI, dispositivos USB y otros periféricos. KNOPPIX puede ser utilizado como una demostración de GNU/Linux, CD educacional, sistema de rescate, o adaptado y usado como una plataforma para demostraciones de productos software comerciales. No es necesario instalar nada en el disco duro. Gracias a la descompresión “al vuelo” el CD puede incorporar hasta 2GiB de software ejecutable instalado en él.

<http://www.knoppix.com>

Un “Live CD” se puede utilizar para proporcionar al usuario la oportunidad de ejecutar un sistema GNU/Linux en su escritorio. Se puede usar para testear y evaluar la interfaz de usuario, aplicaciones, y otras facetas del cliente GNU/Linux, antes de hacer efectiva la migración. Y todo esto se puede llevar a cabo sin dañar la instalación del sistema operativo actual de la máquina. Otro beneficio de utilizar un “Live CD” como parte del plan de migración es la detección de problemas con el hardware o dispositivos. Si se es capaz de personalizar los módulos de los “driver” cargados por un “Live CD” entonces se debe ser capaz también de validar la correcta detección de hardware y soporte de dispositivos en las máquinas sujetas a la migración antes de la migración real.

Advertencia

La formación a los usuarios es un aspecto clave en el éxito de una migración. El mayor esfuerzo, tanto económico como temporal, se debe realizar en este área.

4.1.4. Aspectos Legales

En este punto se van a observar los diferentes puntos a tener en cuenta sobre aspectos legales en una migración a Software Libre.

4.1.4.1. Contratos actuales (mantenimiento y otros)

Antes de la migración se debe pensar en los contratos (de mantenimiento u otros) que pueda tener la empresa. Si la empresa utiliza un software hecho a medida por algún proveedor de software se puede intentar negociar con el proveedor de software el que libere la aplicación bajo una licencia libre como la GPL, ya que al fin y al cabo el software ha sido desarrollado para la empresa en concreto. Pero los proveedores de software no

suelen colaborar en este aspecto y no suelen estar por la labor de liberar sus aplicaciones propietarias.

Otro aspecto a tener en cuenta es la existencia de contratos de mantenimiento de software propietario, si ese software es elegido para ser migrado dichos contratos deben ser extinguidos, lo cual puede acarrear alguna penalización que debe ser estudiada a fondo por la empresa.

Si no se dispone de personal cualificado en la empresa para modificar y adaptar las aplicaciones, en caso de ser necesario, se puede optar por la realización de un contrato con algún proveedor de software libre que se encargue de la modificación y adaptación de las aplicaciones a las necesidades de la empresa. Normalmente las aplicaciones distribuidas bajo una licencia libre suelen ir sin “garantías”, así que si se desea soporte técnico o mantenimiento se debe contratar. Ahí es donde está el negocio en el software libre.

4.1.4.2. Licencias de software

Al analizar las licencias de uso, tanto las de software propietario como las de software libre, merece especial consideración que se estudie sus elementos subjetivos o personales: qué sujetos son las partes de la licencia de software. Por un lado, se encuentra el proveedor-licenciante, quien concede un derecho de uso sobre el software al usuario-licenciatarario (en una licencia de software libre, concede además el derecho a modificar y redistribuir el software, con o sin modificaciones). Por otro lado, tenemos a ese usuario-licenciatarario, quien a su vez adquiere tal derecho de uso, abonando o no un precio por ello.

El proveedor-licenciante ha de encontrarse facultado para conceder licencias de software, bien por ser su autor, el titular de sus derechos de explotación o, como mínimo, de un derecho a su distribución. Por su parte, con relación al usuario-licenciatarario, es importante saber si se trata de un empresario o de un consumidor, pues de ello dependerá el régimen legal aplicable a la licencia.

El proveedor-licenciante es quien concede la licencia al usuario para utilizar el software, proporcionándole una copia del software licenciado.

Pueden conceder licencias de uso el autor o autores originales del software, la persona física o jurídica que sea titular de los derechos de explotación, o aquella que como mínimo tenga el derecho a distribuir el software objeto de la licencia en cuestión. Esta diversidad de sujetos con capacidad para conceder licencias de software es lo que explica que denominemos a esta parte proveedor del software o simplemente licenciante. Se trata de expresiones más genéricas que permiten abarcar a todos los que pueden otorgar licencias sobre el software, a diferencia de otras comúnmente empleadas como autor, titular o propietario del software.

El usuario-licenciatarario es la otra parte del contrato de licencia de software. Es la persona (física o jurídica) que adquiere el derecho a usar el software por medio de la licencia,

según los términos y condiciones que se establecen en la misma (casi siempre impuestos por el proveedor del software).

El usuario-licenciatarario tiene como principales obligaciones pagar el precio de la licencia (cuando es de pago) y respetar las limitaciones de uso que le impone la licencia de software, un software cuya propiedad no le pertenece.

En el caso de que el usuario sea licenciatarario de un software propietario, en principio serán pocos sus derechos como usuario (básicamente ejecutar el programa, aprovechar sus aplicaciones y poder hacer una copia de seguridad del mismo), mientras que las limitaciones son muchas. Por el contrario, si es licenciatarario de un software libre, las libertades del usuario-licenciatarario son mucho más amplias, y por ende, las limitaciones son menores: puede usar el software libremente, modificarlo y redistribuirlo con o sin modificaciones. Si el usuario está facultado para modificar y modifica el software, puede pasar a ser el autor de una obra derivada, según el artículo 11 de la Ley de la Propiedad Intelectual (es decir, de la traducción o adaptación del software). Por su parte, si el usuario está facultado para redistribuir y redistribuye el software, se convertirá también en proveedor de software.

Para ser usuario-licenciatarario no se requiere ningún requisito especial en principio, aparte de las exigencias sobre capacidad legal genéricas: que el usuario persona física sea mayor de edad o, si se trata de una persona jurídica (empresa, administración, asociación sin ánimo de lucro, etc.), que ésta se halle válidamente constituida). Es importante tener en cuenta si se emplean o no condiciones generales en las licencias de software (en casi todos los casos se emplean) y si el usuario-licenciatarario es un consumidor o un empresario, porque varía el régimen legal al que está sujeto el contrato de licencia.

A veces, en el propio texto de la licencia de uso se contemplan derechos y limitaciones distintas según si el usuario es un consumidor que va a destinar el software a un uso particular o un empresario/profesional que va a destinar el software a su actividad. Se trata de las llamadas licencias duales.

Si el usuario es un consumidor, se entiende que se halla en una posición especialmente débil, por lo que debe tener una protección legal frente a posibles abusos del proveedor del software, al igual que sucede en muchos otros contratos que celebran los consumidores.

Se debe tener en cuenta que, aunque las normas protectoras del consumidor no se apliquen cuando el usuario sea un empresario o profesional, ello no significa que el proveedor del software puede imponer sin más a este usuario cláusulas especialmente injustas o abusivas. Ocurre, no obstante, que el usuario empresario no tiene la protección legal que supone que ciertas cláusulas abusivas se consideran automáticamente nulas por disponerlo así la ley.

Si el usuario (empresario o profesional) cree que una cláusula de la licencia es abusiva y se niega a respetarla, deberá demandar el contrato de licencia ante los tribunales. La cláusula podrá ser anulada por estimar que es contraria a la regla general de buena fe que ha de regir en el cumplimiento de los contratos. No obstante, ello dependerá del

examen de las circunstancias de cada caso en concreto y será el juez quien decida si se trata o no de una cláusula contraria a la buena fe entre las partes.

Aun siendo las partes de la licencia las mismas (proveedor-licenciante y usuario-licenciatarío) tanto para software propietario, como para software libre, las diferencias tan importantes sobre los derechos que otorgan unas y otras al usuario hace que sea conveniente tener en cuenta las siguientes consideraciones:

- a) Las licencias de software libre son el medio o instrumento legal, no para que el proveedor del software pueda rentabilizar al máximo sus derechos exclusivos de explotación, sino para garantizar a los usuarios las libertades de uso, modificación y redistribución. En el supuesto de que modifique el software, el usuario será el autor de un programa derivado. Por tanto, el usuario-licenciatarío también puede convertirse a su vez en proveedor-licenciante de otros usuarios; bien licenciando el mismo software que se le ha licenciado a él, bien por licenciar un software derivado del original. Estos otros usuarios pueden, a su vez, modificar y distribuir el programa de nuevo, y así sucesivamente.
- b) Debemos tener en cuenta que las legislaciones sobre propiedad intelectual, incluida la Ley de la Propiedad Intelectual en España, conceden al proveedor de software unos derechos exclusivos en virtud de los cuales ninguna otra persona puede hacer nada con el software si no cuenta con la expresa autorización (licencia) del proveedor. Por ello, el usuario de un software libre puede beneficiarse de las libertades de uso, modificación y redistribución sólo si el proveedor del software le ha concedido realmente tales libertades por medio de una licencia de uso.
- c) Además, ni en España ni en otros países es necesario inscribir el software en el Registro de la Propiedad Intelectual para que su autor sea reconocido como tal. En principio, para ello basta con que el autor pueda probar ser el creador de un software original (o derivado, con la autorización del autor del software original). Esto propicia que existan riesgos de que alguien intente apropiarse de un software libre, que reclame la exclusividad en su explotación y se atreva a sostener que los usuarios utilizan ese software sin tener derecho.
- d) Si alguien intenta apropiarse ilegítimamente del software o pretende restringir las libertades que tienen los usuarios sobre el mismo, el verdadero autor del software es quien podrá reaccionar y ejercer las medidas legales oportunas para impedir esta apropiación indebida. A diferencia del software propietario, el autor no reaccionará tanto para proteger sus derechos exclusivos, sino más bien para que

los usuarios puedan continuar disfrutando de las libertades (de uso, modificación y distribución) sobre el software.

- e) Por otra parte, quien pretenda divulgar su software como libre debe garantizar que ese software es verdaderamente libre y que no infringe los derechos de otro software (sea libre o propietario).
- f) La concesión de una licencia de software libre implica que su titular comparte con los usuarios los principales derechos de explotación sobre el mismo. Pero en España (y en el resto de la Europa “Continental”), el hecho de que ceda a una multitud de usuarios los derechos de modificar o distribuir el software no significa que el software libre pase al dominio público. El software libre no es un software sin propietario, sino que el autor conserva su condición de autor del software y, en particular, los derechos morales sobre el software.

4.1.4.2.1. Licencias de Software Libre

Una licencia es aquella autorización formal con carácter contractual que un autor de un software da a un interesado en ejercer “actos de explotación legales”. Pueden existir tantas licencias como acuerdos concretos se den entre el autor y el licenciatarario. Desde el punto de vista del software libre, existen distintas variantes del concepto o grupos de licencias:

- Las libertades definidas anteriormente están protegidas por licencias de software libre, de las cuales una de las más utilizadas es la Licencia Pública General GNU (GPL). El autor conserva los derechos de autor (copyright), y permite la redistribución y modificación bajo términos diseñados para asegurarse de que todas las versiones modificadas del software permanecen bajo los términos más restrictivos de la propia GNU GPL. Esto hace que no sea imposible crear un producto con partes no licenciadas GPL: el conjunto tiene que ser GPL.
- Licencias estilo BSD, llamadas así porque se utilizan en gran cantidad de software distribuido junto a los sistemas operativos BSD. El autor, bajo tales licencias, mantiene la protección de copyright únicamente para la renuncia de garantía y para requerir la adecuada atribución de la autoría en trabajos derivados, pero permite la libre redistribución y modificación, incluso si dichos trabajos tienen propietario. Son muy permisivas, tanto que son fácilmente absorbidas al ser mezcladas con la licencia GNU GPL con quienes son compatibles. Puede argumentarse que esta licencia asegura “verdadero” software libre, en el sentido que el usuario tiene libertad ilimitada con respecto al software, y que puede decidir incluso redistribuirlo como no libre. Otras opiniones están orientadas a destacar que este tipo de licencia no contribuye al desarrollo de más software libre.

- Licencias estilo MPL y derivadas, Esta licencia es de Software Libre y tiene un gran valor porque fue el instrumento que empleó Netscape Communications Corp. para liberar su Netscape Communicator 4.0 y empezar ese proyecto tan importante para el mundo del Software Libre: Mozilla. Se utilizan en gran cantidad de productos de software libre de uso cotidiano en todo tipo de sistemas operativos. La MPL es Software Libre y promueve eficazmente la colaboración evitando el efecto “viral” de la GPL (si usas código licenciado GPL, tu desarrollo final tiene que estar licenciado GPL). Desde un punto de vista del desarrollador la GPL presenta un inconveniente en este punto, y lamentablemente mucha gente se cierra en banda ante el uso de dicho código. No obstante la MPL no es tan excesivamente permisiva como las licencias tipo BSD. Estas licencias son denominadas de copyleft débil. La NPL (luego la MPL) fue la primera licencia nueva después de muchos años, que se encargaba de algunos puntos que no fueron tenidos en cuenta por las licencias BSD y GNU. En el espectro de las licencias de software libre se la puede considerar adyacente a la licencia estilo BSD, pero perfeccionada. Hay que hacer constar que el titular de los derechos de autor (copyright) de un software bajo licencia copyleft puede también realizar una versión modificada bajo su copyright original, y venderla bajo cualquier licencia que desee, además de distribuir la versión original como software libre. Esta técnica ha sido usada como un modelo de negocio por una serie de empresas que realizan software libre (por ejemplo MySQL); esta práctica no restringe ninguno de los derechos otorgados a los usuarios de la versión copyleft. También podría retirar todas las licencias de software libre anteriormente otorgadas, pero esto obligaría a una indemnización a los titulares de las licencias en uso. En España, toda obra derivada está tan protegida como una original, siempre que la obra derivada parta de una autorización contractual con el autor. En el caso genérico de que el autor retire las licencias “copyleft”, no afectaría de ningún modo a los productos derivados anteriores a esa retirada, ya que no tiene efecto retroactivo. En términos legales, el autor no ha derecho a retirar el permiso de una licencia en vigencia. Si así sucediera, el conflicto entre las partes se resolvería en un pleito convencional.

Nota

Existen otras muchas licencias, como por ejemplo la licencia de Apache o la licencia MIT/X11.

Puede encontrar más licencias de software libre en la web del OpenSource Initiative (<http://www.opensource.org/licenses/alphabetical>), un organismo sin ánimo de lucro que se encarga de revisar y aprobar licencias compatibles con la filosofía del software libre.

4.1.4.2.2. Comparación con el software Open Source

Aunque en la práctica el software Open Source (de código abierto) y el software libre comparten las mismas licencias, la FSF opina que el movimiento Open Source es filosó-

ficamente diferente del movimiento del software libre. Apareció en 1998 con un grupo de personas, entre los que cabe destacar a Eric S. Raymond y Bruce Perens, que formaron la Open Source Initiative (OSI). Ellos buscaban:

- 1) darle mayor relevancia a los beneficios prácticos del compartir el código fuente.
- 2) interesar a las principales casas de software y otras empresas de la industria de la alta tecnología en el concepto.

Estos defensores ven que el término “open source” evita la ambigüedad del término inglés free en free software. El término “open source” fue acuñado por Christine Peterson del think tank Foresight Institute, y se registró para actuar como marca registrada para los productos de software libre.

Mucha gente reconoce el beneficio cualitativo del proceso de desarrollo de software cuando los desarrolladores pueden usar, modificar y redistribuir el código fuente de un programa. (Véase también La Catedral y el Bazar). El movimiento del software libre hace especial énfasis en los aspectos morales o éticos del software, viendo la excelencia técnica como un producto secundario deseable de su estándar ético. El movimiento Open Source ve la excelencia técnica como el objetivo prioritario, siendo la compartición del código fuente un medio para dicho fin. Por dicho motivo, la FSF se distancia tanto del movimiento Open Source como del término “Open Source”.

Puesto que la OSI sólo aprueba las licencias que se ajustan a la OSD (Open Source Definition), la mayoría de la gente lo interpreta como un esquema de distribución, e intercambia libremente “open source” con “software libre”. Aun cuando existen importantes diferencias filosóficas entre ambos términos, especialmente en términos de las motivaciones para el desarrollo y el uso de tal software, raramente suelen tener impacto en el proceso de colaboración.

Aunque el término “open source” elimina la ambigüedad de libertad frente a precio (en el caso del Inglés), introduce una nueva: entre los programas que se ajustan a la Open Source Definition, que dan a los usuarios la libertad de mejorarlos, y los programas que simplemente tienen el código fuente disponible, posiblemente con fuertes restricciones sobre el uso de dicho código fuente. Mucha gente cree que cualquier software que tenga el código fuente disponible es open source, puesto que lo pueden manipular (un ejemplo de este tipo de software sería el popular paquete de software gratuito Graphviz, inicialmente no libre pero que incluía el código fuente, aunque luego AT&T le cambió la licencia). Sin embargo, mucho de este software no da a sus usuarios la libertad de distribuir sus modificaciones, restringe el uso comercial, o en general restringe los derechos de los usuarios.

4.1.5. Recursos temporales

Como se puede observar, en la mayoría de los casos una migración no es una tarea sencilla o rápida, requiere mucha planificación y tener en cuenta muchos factores a la hora de tomar decisiones. Otro factor mas a tener en cuenta es el factor “tiempo”.

Nota

En general las prisas no son buenas, y más en una migración a nuevas aplicaciones basadas en software libre. Todo requiere su tiempo. Se debe ir paso a paso para garantizar el éxito.

Algunas cosas a tener en consideración son:

- ¿De cuánto tiempo se dispone para llevar a cabo la migración?
- Conociendo el nivel de los usuarios ¿Cuánto tiempo se va a dedicar a la formación?
- ¿Existen procesos (industriales o de cualquier otro tipo) involucrados en la migración que no se puedan detener?
- ¿Existen procesos o situaciones que nos acoten el tiempo disponible para llevar a cabo la migración? Por ejemplo el envío semanal de una copia de seguridad de datos a una oficina central o similares.

4.1.6. Recursos económicos

Es importante determinar el esfuerzo económico que puede suponer el realizar una migración y contrastarlo con el coste que supondría mantener un sistema compuesto enteramente por software propietario.

Importante

No se debe basar la decisión de realizar una migración únicamente en el factor económico.

Aunque por lo general el software libre es más rentable a medio/largo plazo que el software privativo se deben evaluar más criterios para decidir si se lleva a cabo una migración.

Por ello otro de los factores importantes a tener en cuenta en una migración es el factor económico. En concreto el aspecto que mejor refleja el coste de una migración es el que se conoce como TCO: Total Cost of Ownership. El TCO define el coste total de propiedad de una tecnología concreta sobre su periodo de vida útil.

Los componentes que forman el TCO son todos aquellos costes que se producen como consecuencia de la introducción de una nueva tecnología. A grandes rasgos se puede hablar de dos tipos de costes, los directos e indirectos. Los costes directos, normalmente, son aquellos costes conocidos y que implican una contraprestación económica (por ejemplo la compra de un nuevo PC para la empresa). Por otra parte, los costes indirectos son los que no tienen una contraprestación económica conocida y no son tan fácilmente identificables como los costes directos (por ejemplo la producción “perdida” a causa de las horas invertidas por el usuario en la instalación y configuración del nuevo PC adquirido por la empresa).

De esta manera vamos a intentar hacer una clasificación general de los costes asociados a una migración:

- Costes Directos
 - Licencias y soporte de software
 - Costes hardware
 - Costes de soporte
 - Costes de formación
 - Costes de personal
- Costes Indirectos
 - Costes de soporte
 - Downtime

4.1.6.1. Costes directos

4.1.6.1.1. Licencias y soporte de software

El software propietario que se utiliza actualmente tiene asociado un coste por licencia (por puesto de trabajo, por acceso, etc...) que, en función del volumen de la empresa, puede suponer un porcentaje elevado de los costes totales de un sistema de información.

Las distribuciones de GNU/Linux y la mayoría de aplicaciones incluidas en dichas distribuciones son de Código Abierto y se licencian bajo la GPL. Esto significa que es distribuido libremente. Así pues no suele haber costes de licencia, como tal, asociados al software distribuido bajo esta licencia.

Aun así, las distribuciones empaquetadas por empresas distribuidoras no tienen porqué ser gratuitas. Normalmente hay un modelo de precios “por puesto” establecido por las empresas distribuidoras de software libre. Pero esta tasa se paga en concepto de soporte técnico y no en concepto de licencia de uso. Usualmente esta tasa otorga el derecho a soporte técnico durante un año, aunque es posible contratar niveles de soporte extra con las empresas distribuidoras de software libre.

Dada la naturaleza de código abierto del sistema operativo GNU/Linux y su software relacionado, también es posible utilizarlo completamente libre de costes de licencia o de

soporte. En este caso el soporte depende completamente de la comunidad software libre y del personal bien cualificado del que se disponga en la plantilla de la empresa.

4.1.6.1.2. Costes de Hardware

La mayor parte de distribuciones de GNU/Linux y el software libre en general son capaces de ser ejecutados satisfactoriamente sobre hardware viejo. Dependiendo de las necesidades y exigencias de las aplicaciones a utilizar es incluso posible reutilizar hardware que haya sido retirado porque no era lo suficientemente potente como para satisfacer las necesidades de rendimiento requeridas por las últimas versiones de aplicaciones no libres.

De todas maneras, las últimas versiones de las distribuciones de software libre tienen requisitos mínimos de memoria que se asemejan a los requisitos de memoria de las aplicaciones privativas. Aún así, siendo solo requisitos de memoria, se puede seguir utilizando hardware retirado para ejecutar estas distribuciones. Y la tendencia parece que continuará siendo esta debido al uso intensivo que hacen las aplicaciones privativas de formatos de datos más complejos de lo necesario (conocidos también como Rich Data Formats).

Consejo

Dependiendo del tipo de migración que se vaya a llevar a cabo y del estado del hardware del que se dispone actualmente, se debe plantear la adquisición de nuevo hardware, el mantenimiento del hardware actual o la recuperación de hardware retirado.

4.1.6.1.3. Costes de soporte

En una empresa mediana/grande, mantener cualquier equipo de escritorio operativo, libre de fallos y de agujeros de seguridad es, normalmente, uno de los costes totales más altos. Esta situación no es diferente en una estrategia basada en GNU/Linux y software libre. Pero, el hecho de utilizar sistemas operativos de tipo UNIX introduce muchas estrategias de ahorro de costes a tener en cuenta. Por ejemplo, al poder acceder remotamente a los equipos y gestionarlos utilizando protocolos como *telnet*, *ssh* o similares, es posible instalar *scripts* en las estaciones de trabajo que pueden ser ejecutados remotamente de una manera sencilla.

El uso de *scripts* remotos posibilita la monitorización de todas las estaciones de trabajo para prevenir fallos y ejecutar tareas en todos los equipos de manera centralizada. Por ejemplo, es posible aplicar la corrección de un error en todos los equipos a la vez sin que la producción del usuario final se vea afectada o interrumpida por el cambio.

Los costes de soporte incluyen la instalación y configuración, mantenimiento y solución de problemas derivados de la migración.

Importante

Si la empresa no dispone de un departamento de TI se deben contratar los servicios profesionales de consultoras u otras empresas de servicios de software libre que puedan implantar la base tecnológica necesaria.

La ventaja que ofrece el software libre en el aspecto del soporte es la existencia de multitud de comunidades de usuarios que ofrecen soporte sobre la aplicación software libre o sistema operativo en cuestión mediante foros, documentación o charlas completamente libres de coste.

Los fabricantes de software propietario son conscientes de la importancia de este canal de soporte que ha establecido el software libre. Por eso Microsoft creó sus propias comunidades de usuarios para poder ofrecer soporte de manera eficiente.

También es cierto que, aunque el coste de soporte sea cual sea el tipo de software (libre o propietario) es gran parte del TCO, las soluciones basadas en software libre configuradas apropiadamente requieren un coste mínimo de mantenimiento.

4.1.6.1.4. Costes de formación

Este es otro apartado fundamental en el que se debe invertir, y el éxito de la migración depende en gran medida del esfuerzo que se realice en cuanto a la formación de los usuarios. Se debe tener en cuenta que la filosofía del software libre, así como las metodologías de uso de las aplicaciones, son bastante diferentes a las que se está acostumbrado en Software Propietario, y esto puede generar confusión y rechazo en los usuarios si no se invierten los recursos necesarios para dar a conocer las nuevas aplicaciones.

En este apartado se deben contemplar los costes asociados, si procede, de posibles cursos externos de formación para los usuarios, o por ejemplo el sueldo del profesorado contratado para impartir la formación internamente.

4.1.6.1.5. Costes de personal

En este apartado se deben reflejar todos los gastos de personal relacionados directamente a la migración, es decir, los salarios del personal de TI que lleve a cabo la migración.

La actual penetración de GNU/Linux y el software libre en general está haciendo que el dominio de estos sistemas sea ya cada vez más extenso por parte de muchos administradores de sistemas. Actualmente, las empresas que necesiten un administrador de sistemas GNU/Linux no tendrán problemas en encontrar a personal cualificado. De igual manera se puede encontrar personal cualificado para administrar, programar o crear aplicaciones de software libre, tanto en el área de servidores como en entornos de escritorio.

4.1.6.2. Costes indirectos

Estos gastos son algo mas difíciles de predecir ya que cuando se habla de costes indirectos se suele hacer en el sentido de pérdidas de productividad en la empresa debidas a la migración.

Nota

Un buen estudio y plan de migración conducirá inexorablemente a una migración exitosa e implícitamente a una reducción sustancial en los costes indirectos.

Aunque como ya se ha comentado estos costes son difíciles de establecer a priori se van a comentar dos de los más importantes.

4.1.6.2.1. Costes de soporte

Los usuarios de las tecnologías en empresas normalmente se apoyan en los técnicos informáticos y en compañeros de trabajo para la resolución de problemas. Este hecho implica el conocimiento de la tecnología por parte de los usuarios de la empresa. Este concepto pretende abarcar el coste de las pérdidas de productividad derivadas bien del desconocimiento del uso de la tecnología, bien sea por una errónea utilización del sistema o por errores del propio sistema.

4.1.6.2.2. Costes de inoperatividad del sistema

En este apartado se suman los costes derivados de la pérdida productividad en la empresa debidas a inoperatividad del sistema. Existen muchas causas por las cuales el sistema puede quedar temporalmente inoperativo, entre ellas la propia migración. Es decir, dependiendo de la planificación que se haga se puede dar el caso de que la empresa no pueda producir normalmente durante el proceso de migración.

En el mundo del software y los sistemas operativos privativos se vive una situación que provoca muchísimas pérdidas a las empresas, los virus. Estos virus pueden provocar que el sistema quede inoperativo temporalmente, con las consiguientes pérdidas para la empresa. En el mundo del software libre y los sistemas operativos libres, como GNU/Linux, los virus tal como se entienden en los sistemas privativos no existen debido a las restricciones establecidas y las medidas de seguridad tomadas por naturaleza. Pero esto no significa que el software libre esté exento de errores o programas maliciosos que puedan aprovechar una vulnerabilidad del sistema para dejarlo inoperativo o causar daños.

4.2. Objetivos

Se debe identificar cuales van a ser los objetivos iniciales perseguidos por la migración. Aunque estos puede que cambien durante la planificación al encontrarnos con posibles obstáculos como por ejemplo aplicaciones *no migrables*.

- ¿Cual va a ser la finalidad de la migración?
- ¿Reducir costes?
- ¿Mejorar el sistema?
- ¿Obtener independencia frente a distribuidores?
- ¿Regularizar la situación de la empresa con respecto a las licencias de software?

5. Planificación

El trabajo del proyecto comienza estableciendo un plan que describa el camino a seguir para llegar al objetivo. El plan de migración debería contener como mínimo la siguiente información: fecha final del proceso de migración, recursos materiales y humanos, participación de terceras partes, hitos durante el proceso de migración y costes. La planificación del proyecto es también la base para una gestión eficiente de la migración.

Como en cualquier implantación de un nuevo sistema de trabajo, se debe estudiar muy detenidamente toda la información disponible y planificar todos los pasos a seguir para garantizar el éxito.

Una vez se ha llevado a cabo la toma de requisitos, ya se conoce perfectamente el estado de la empresa en cuanto a software se refiere. Es el momento de empezar a planificar la estrategia que se va a seguir para llevar la migración a buen término y lograr los objetivos establecidos en el punto anterior.

Este es el momento de tomar decisiones en base a la información recogida y de estas decisiones depende en gran medida el éxito de la migración.

5.1. Planificación técnica

Comenzaremos por la parte técnica de la planificación, en este punto se debe decidir que tipo de migración se va a llevar a cabo y cómo. Esto nos servirá para dividir la migración en pequeños pasos o tareas que hagan la gestión del proyecto mucho más fácil. Cuanto más nivel de detalle se alcance en la descripción de las tareas, más sencillo será después planificar que recursos humanos y temporales asignarle.

5.1.1. Cosas a tener en cuenta

Este documento no pretende ser un manual exhaustivo de cómo realizar una migración en términos técnicos. Solo es una guía en la que se intenta establecer una serie de pasos o procedimientos que ayudarán a planificar y ejecutar una migración a software libre.

Nota

Para más información o detalles técnicos, por favor, consultar la bibliografía.

5.1.1.1. Tipos de migración

Existen diferentes tipos de migración. No siempre es posible llevar a cabo todos los tipos de migración y se debe decidir cual conviene más en cada caso concreto.

5.1.1.1.1. Migración de los servicios (servidores)

En este tipo de migraciones solo las aplicaciones de los servidores se migran, esto es posible solamente si existe un reemplazo compatible (en la mayoría de casos para aplicaciones de servidores como correo electrónico, paginas web, etc... sí que existen alternativas libres) con los clientes.

Por ejemplo, si nuestro servidor ofrece servicio de autenticación de usuarios en un dominio Microsoft Windows, carpetas compartidas, servicios de correo electrónico y páginas web se puede migrar a un entorno con GNU/Linux como sistema operativo, OpenLDAP⁵ y Samba⁶ para la autenticación de usuarios en dominios Microsoft Windows y carpetas compartidas, Postfix⁷ o Sendmail⁸ para los servicios de correo electrónico y Apache Web Server⁹ o LightHTTPD¹⁰ como servidor de páginas web o directorios WebDAV.

La ventaja de este tipo de migración es que las aplicaciones instaladas en los clientes no se alteran en ningún momento, es decir, los usuarios de las aplicaciones cliente no notan ningún cambio. Además estos usuarios no necesitarán formación dado que continúan manejando las mismas aplicaciones. Esta es una gran ventaja, ya que los usuarios, al no tener que aprender a usar nuevas herramientas, seguirán siendo, al menos, tan productivos como antes de la migración del servidor.

Además, es muy probable que la productividad de los usuarios aumente, ya que en términos generales, los servidores basados en GNU/Linux soportan una carga mayor que aquellos utilizando software privativo. De esta manera, servidores que antes de la migración soportaban una carga alta de transacciones, al ser migrados podrán soportar aún más transacciones con el mismo hardware, con lo que los usuarios notarán una disminución del tiempo de respuesta del servidor, y por tanto la productividad de estos usuarios puede llegar a aumentar de manera considerable, ya que podrán realizar más tareas en mismo tiempo.

⁵ <http://www.openldap.org>

⁶ <http://www.samba.org>

⁷ <http://www.postfix.org>

⁸ <http://www.sendmail.org>

⁹ <http://www.apache.org>

¹⁰ <http://www.lighttpd.net>

Los únicos usuarios que necesitan formación en las nuevas aplicaciones (si no la poseen ya) son los técnicos encargados del mantenimiento y buen funcionamiento de los servidores. Por lo general éste colectivo de profesionales suele ser mucho más receptivo a los cambios, debido a su mayor conocimiento de los sistemas, que los usuarios finales.

Consejo

Siempre que se realice una migración de algún servidor, es importante que el técnico o administrador encargado de dicho servidor sea participativo en la migración. De esta manera podrá aprender durante la migración las tareas básicas de administración del nuevo sistema.

Si conseguimos que el administrador sea participativo, podemos conseguir que los costes de formación sean nulos para tareas sencillas, y para tareas más sofisticadas podemos reducirlos considerablemente.

En caso de que los servidores necesiten ofrecer servicios que únicamente dispongan de software propietario, podemos realizar una **migración parcial**. Para realizar esta tarea, podemos migrar el servidor a software libre, reemplazando todos los posibles servicios que tengan una alternativa basada en software libre. Para aquellos que no exista una alternativa viable, o que no se deseen cambiar, podemos realizar una migración parcial por virtualización. De esta manera, corremos otro sistema operativo encima del servidor con software libre, sobre el que se instalan únicamente los servicios que no se deseen/puedan migrar. Para realizar esta virtualización, podemos utilizar máquinas virtuales como QEmu¹¹ o Innotek VirtualBox¹². Este tipo de soluciones es muy atractiva ya que aislamos el software propietario del libre, con lo que los fallos de seguridad de esa máquina virtual quedan completamente aislados del sistema operativo real.

5.1.1.1.2. Migración de los usuarios (clientes)

Se puede llegar a este tipo de migración de tres maneras, una es que se haya realizado la migración de los servidores en una etapa anterior, otra es que se disponga de aplicaciones para los clientes compatibles con las aplicaciones propietarias instaladas en los servidores y sean software libre. El otro supuesto que puede desembocar en este tipo de migración es que la empresa no disponga de servidores, con lo cual los “clientes” son máquinas aisladas y su software no mantiene ninguna relación con otra máquina externa.

En este tipo de migración son sólo las máquinas cliente las que migran su software. La desventaja es que son los usuarios finales los que padecen el cambio y por lo tanto éste debe ser gestionado de la mejor manera posible para evitar el posible rechazo de las nuevas aplicaciones por parte de los usuarios, haciendo especial hincapié en la formación.

¹¹ <http://fabrice.bellard.free.fr/qemu/>

¹² <http://www.virtualbox.org>

Advertencia

Si la migración de software se hace de forma abrupta, es muy probable que los usuarios rechacen el cambio o que incluso se opongan a él.

Hay que tener en cuenta, además, que realizar este tipo de migración software puede producir un decremento de la productividad de los usuarios, ya que durante un tiempo (en general entre una y dos semanas) los usuarios tendrán problemas de adaptación. Afortunadamente, las HIG¹³ implementadas por el software libre, sobre todo en los entornos de escritorio, son seguidas a rajatabla en muchas aplicaciones libres, por lo que este paso, aunque tarden un par de semanas en adaptarse los usuarios, implicaran un notable incremento de productividad posteriormente. Por tanto, podemos considerar la penalización de productividad inicial como un pequeño obstáculo debido al cambio, pero que a medio o largo plazo proporcionará beneficios de productividad notables.

En el caso de que haya aplicaciones indispensables que no puedan migrarse, se puede optar por una adaptación parcial, como hemos comentado anteriormente en el caso de la migración de servicios. Para conseguir esto, disponemos de más herramientas que en el caso del servidor, ya que además de la estrategia de la virtualización, podemos utilizar aplicaciones de emulación.

5.1.1.1.3. Migración completa

Este tipo de migración es una combinación de los dos anteriores. Se trata de hacer la migración tanto de los servidores como de los clientes. En este caso se debe destacar que todo ha de estar muy bien planificado de antemano ya que en una migración no se pueden dejar cabos sueltos. Se debe estar bien seguro de los pasos a seguir y las acciones a tomar puesto que tanto el cliente como el servidor cambian al mismo tiempo y pueden surgir errores o incompatibilidades inesperadas que hagan peligrar el éxito de la migración.

Advertencia

Para realizar este tipo de migración, ha de considerar siempre realizarlo en dos pasos, migrando inicialmente los servicios y, posteriormente, los usuarios.

¹³ HIG, siglas del inglés, *Human Interface Guidelines* (guía para interfaces humanas), son las especificaciones de usabilidad de las aplicaciones para que estas se puedan utilizar de la manera más amigable. En el caso del escritorio GNOME, podemos encontrar estas guías en <http://developer.gnome.org/projects/gup/hig/2.0/>

5.1.1.1.4. Migración de aplicaciones

En los anteriores tipos de migración, se considera que se cambia tanto de aplicaciones como de sistema operativo, ya sea en los servidores, en los clientes o en ambos.

En este caso, solo cambian algunas aplicaciones. Se suele dar este caso sobre todo cuando al analizar las aplicaciones que se utilizan en la empresa aparecen muchas aplicaciones *no migrables* u otros factores que no permiten una migración total. De esta manera se escogen las aplicaciones que tienen una clara alternativa en software libre y se migran, dejando las demás inalteradas.

Importante

Aunque este tipo de migraciones son mucho más sencillas y rápidas que las migraciones que hemos comentado anteriormente, pueden suponer un ahorro importante respecto al coste económico necesario para realizarlas.

Para comprobarlo, nos serviremos de un ejemplo. Una licencia de Microsoft Office cuesta aproximadamente 600 euros, mientras que OpenOffice.org no sólo es gratuito, sino que podemos actualizarlo siempre sin ningún coste y el entorno de trabajo es prácticamente idéntico al del software privativo.

5.1.1.2. Estrategias de migración

En este apartado vamos a ver las distintas posibilidades que tenemos para realizar la migración. Trataremos de describir las alternativas, mostrando tanto las ventajas como las desventajas de cada una de ellas. De esta manera, podremos elegir la alternativa adecuada a la hora de realizar la migración en nuestra empresa.

Tenga en cuenta que no existen únicamente estas posibilidades, y que en función de sus necesidades puede optar por una en concreto, o mezclar varias alternativas. Por ejemplo, si dispone de dos departamentos en su empresa que desea migrar, puede utilizar estrategias de migración distintas para cada departamento, pero siempre teniendo una estrategia global de migración para no perder de vista las operaciones que desea realizar. De esta manera en departamentos con muy pocos equipos a migrar puede utilizar el sistema de migración en un solo paso. En cambio, en el área de sistemas puede utilizar una migración progresiva en grupos.

Antes de describir las diferentes estrategias para realizar la migración, es importante seguir estos consejos:

- Recuerde siempre dedicar los medios necesarios para informar y formar a los usuarios y de este modo evitar el rechazo.
- Motive a sus empleados a usar los nuevos sistemas, esto hará la transición más llevadera y permitirá que los usuarios se esfuercen en aprender a utilizar el nuevo software.

- Dedique tiempo a escoger la estrategia de migración adecuada para su empresa. Estudie las ventajas y desventajas de cada estrategia e identifique los posibles riesgos y cómo solucionarlos.

5.1.1.2.1. Migración en un único paso

Esta migración es la más rápida de realizar, pero tiene muchos posibles inconvenientes. Se trata de realizar toda la migración a software libre de una sola vez.

Esta estrategia de migración depende, generalmente, del tamaño de la empresa (o el grupo seleccionado) y las aplicaciones que se utilicen y consiste en cambiar todo el software por sus equivalentes en software libre en los equipos de la empresa a la vez.

Dado que toda la migración se va a llevar a cabo en un solo paso, se recomienda tener muy bien planificadas las tareas a llevar a cabo, así como bien definida la lista de software a instalar en los equipos y las configuraciones a establecer para los diferentes servicios. No se debe dejar nada a la improvisación y hay que ser muy meticuloso a la hora de realizar el cambio.

Nota

Todos los usuarios cambiarán del viejo sistema al nuevo el mismo día. Es recomendable llevar a cabo el cambio durante un fin de semana o un día festivo.

Esta opción de migración suele ser la más adecuada para pequeñas empresas o administraciones en las cuales el número de equipos es muy reducido y en rara ocasión disponen de más de un servidor. Pero debido a la cautela que se ha de tener al planificar el cambio este camino de migración puede resultar complicado en empresas grandes, con más de 50 equipos y más de 1 ó 2 servidores.

Advertencia

Este camino de migración no es adecuado para empresas con un elevado número de equipos.

Una de las ventajas de optar por este camino de migración es que no se necesitará mantenimiento de dos sistemas diferentes (el viejo y el nuevo), porque el viejo sistema desaparece definitivamente. Las desventajas son que, de no haber planeado correctamente la migración, puede no terminarse a tiempo y además existe el peligro de que los usuarios rechacen la migración.

Una buena precondition para seguir este camino de migración es que el personal de TI ya posea el conocimiento necesario sobre software libre, ya sea porque lo utilizan a nivel privado o porque las aplicaciones o servicios individuales basados en software libre (como un servidor de e-mail bajo GNU/Linux) ya se utilizaban oficialmente en la empresa. Si además el personal de la empresa está abierto a nuevas tecnologías e interesado en el Software Libre las cosas serán más fáciles.

Ventajas

- No se tienen que mantener dos sistemas simultáneamente. El nuevo sistema reemplaza al anterior.
- Es muy práctico para empresas pequeñas.
- Es la estrategia de migración más económica.

Desventajas

- Se dispone de poco tiempo para realizar la migración.
- Los errores en la migración se pagan caros, no se puede utilizar el sistema antiguo mientras se arreglan estos errores.
- Los empleados descubren el nuevo sistema de forma abrupta. Pueden rechazar la migración.
- Requiere de una formación previa de los usuarios.

5.1.1.2.2. Migración Piloto e Implantación

Esta estrategia de migración suele ser la más adecuada para empresas con gran número de equipos y más de un servidor. Se procederá primero a la migración de las aplicaciones en un grupo reducido de equipos. Por ejemplo en una migración de servicios y clientes se puede utilizar un servidor y un equipo como piloto, aunque el número de equipos que formen parte de la migración piloto puede seleccionarse en función de las necesidades. De hecho, es habitual utilizar un departamento de la empresa como grupo piloto, para estudiar alternativas de migración posteriormente para el resto de la empresa.

Una vez instaladas las nuevas aplicaciones en los equipos piloto, se procederá a la comprobación de su correcto funcionamiento y a la verificación de que cumplen con los requisitos establecidos.

Advertencia

Si no se dispone de máquinas físicas suficientes (como suele suceder con los servidores) se tendrán que utilizar máquinas virtuales para simular los equipos piloto. Por ejemplo se puede instalar una distribución de GNU/Linux en una máquina virtual ejecutándose sobre Microsoft Windows o viceversa.

Después del período de evaluación se procederá a la instalación definitiva en el resto de máquinas. Este período de evaluación debe ser lo suficientemente largo para que

de tiempo a comprobar que todo funciona de la manera esperada. Esta es la razón principal por la que se utiliza uno o varios equipos pilotos: comprobar la corrección de funcionamiento. Si no evaluamos esta corrección adecuadamente, no sirve de nada utilizar esta estrategia de migración.

Como cualquier otro camino de migración se requiere una buena planificación de todas las tareas a llevar a cabo, la ventaja de este método es que se pueden corregir errores inesperados o incompatibilidades no contempladas sin perder la funcionalidad o la productividad actual del sistema de información actual de la empresa.

Evidentemente, una vez se compruebe que la prueba piloto funciona correctamente, pasaremos a implantar la migración el resto de equipos, pero con el conocimiento de conocer de antemano los problemas a los que nos vamos a enfrentar.

Esta es una de las estrategias más utilizadas por la empresas, sobre todo cuando existen una cantidad de aplicaciones o servicios no migrables a los que se debe dar soporte.

Ventajas

- La prueba piloto nos permite conocer los riesgos que se corren al realizar la migración.
- Permite comprobar cómo va a ser realizada la migración.
- Se identifica inequívocamente el software no migrable, con el consiguiente ahorro de tiempo posteriormente.
- Permite crear un proceso de migración que se aplicará posteriormente, cuando se implante en el resto de equipos.
- Permite formar a los usuarios antes de que se implante el sistema ya migrado.

Desventajas

- Se tiene que mantener el sistema piloto simultáneamente al sistema actual.
- Se necesitan más recursos para realizar la migración.
- Durante la prueba piloto, perderemos recursos humanos, ya que los implicados en esta prueba tengan una pérdida de productividad.

5.1.1.2.3. Transición por fases en grupos

Esta es una opción adecuada si se tienen identificados claramente grupos funcionales dentro de la empresa y se pretende ir integrando software libre paulatinamente. Los

grupos de usuarios migran del viejo sistema propietario al nuevo software libre consecutivamente. Esto tiene la ventaja de que a medida que se vayan realizando las migraciones de los grupos se irá ganando experiencia y se aprende de los errores cometidos. De esta manera si algo falla al migrar un grupo funcional se evitará que falle al migrar el siguiente grupo.

Consejo

Si los grupos no están ya establecidos, elegir un tamaño de grupo adecuado es esencial para contener los riesgos y gestionar los recursos.

El inconveniente de este camino de migración es que en ciertos escenarios el migrar grupo a grupo requiere duplicar recursos (mantener al mismo tiempo el sistema propietario antiguo y el nuevo sistema basado en software libre) o un alto grado de compatibilidad entre aplicaciones propietarias y aplicaciones de software libre. Para ilustrar esto se puede observar el siguiente ejemplo:

Ejemplo

Una empresa dispone de 2 servidores y 6 grupos funcionales. Uno de los servidores se utiliza para dar servicios de autenticación, almacenamiento compartido, acceso a una aplicación de gestión vía web todo con software propietario. El otro servidor se utiliza como apoyo en caso de fallo del servidor principal. En este entorno se pueden tomar varias decisiones, una de ellas puede ser implementar el nuevo servidor basado en software libre y comenzar a migrar los grupos, una vez finalizada la migración de todos los grupos el servidor que antes actuaba como servidor principal de software propietario ya no será necesario, así que se puede proceder a migrar dicho servidor. Si se está seguro de la compatibilidad de los clientes con el nuevo software libre se puede sustituir en una primera etapa los servidores y después proceder a migrar los grupos por fases.

Esta estrategia de migración es muy interesante, y mezcla la migración en un único caso junto a la prueba piloto. De esta manera, los usuarios se van adaptando paulatinamente al nuevo sistema, la transición se hace de forma progresiva, de manera que si algo falla, únicamente afectará al grupo sobre el que se ha decidido realizar la migración primaria. De esta manera, como ya hemos comentado antes, podemos subsanar esos errores para que el siguiente grupo sobre el que se realice la migración no padezca de este error y la migración sea más rápida.

Nota

Se puede aprovechar la migración para hacer cambio del hardware de los PC al mismo tiempo, reemplazando las máquinas en un grupo y luego instalando las sustituidas (si son mejores) en lugar de las viejas máquinas del siguiente grupo.

Además, este tipo de migración, al realizarse poco a poco dentro de la estructura empresarial, permite dosificar el esfuerzo de la empresa en adaptarse al nuevo software. De esta manera, cuando la migración esté a mitad, sólo habrá un grupo implicado activamente en la migración; el resto, o utilizarán el sistema antiguo (con lo cual no verán mermada su productividad) o llevarán utilizando el sistema migrado a software libre (por lo que la productividad será igual o mayor a los que aún no han migrado).

Ventajas

- La migración no afecta a todo el sistema.
- Se puede aprovechar la migración para realizar una renovación del hardware.
- Permite identificar posibles errores antes de que se produzcan en otros grupos.
- Si algo falla, solo afecta al grupo que está actualmente en transición.
- Como la migración se hace por grupos, sólo hay un grupo cada vez que pierda productividad.

Desventajas

- Conviven dos sistemas simultáneamente: doble trabajo para los administradores.
- Puede haber problemas de sincronismo entre ambos sistemas.
- Es más costoso que el resto de métodos, tanto económica como temporalmente.
- Si la empresa sobre la que se realiza la migración es grande, puede ser la única manera de realizar la migración.

5.1.1.2.4. Transición de usuario a usuario

Básicamente la misma opción de la transición en grupos, pero con un grupo compuesto por una sola persona.

Este método necesita escasos recursos, sin embargo resulta ineficaz para grandes administraciones. Además es muy lento, por lo que la migración puede extenderse durante un largo periodo de tiempo.

Siempre que se pueda, es preferible utilizar alguna de las otras estrategias, a no ser que se trate de migración de sistemas críticos, donde tengamos que realizar una migración muy poco a poco para que la transición afecte al sistema de forma muy progresiva.

5.1.2. Inventario

Una vez elegida una estrategia de migración, lo que debemos hacer es analizar cómo va a afectar la migración a las aplicaciones y servicios que actualmente se están empleando en la empresa.

5.1.2.1. Inventario de software

El inventario de software realizado en la etapa anterior permitirá identificar las aplicaciones que realmente se utilizan en la empresa y servirá como guía para establecer qué Software Libre se va a implantar.

Se espera que la mayoría, pero no todas, las aplicaciones tengan disponible su equivalente funcional que se ejecute nativamente en un cliente GNU/Linux. Y hay casos especiales (ideales) en los que los equivalentes funcionales son también lo que se conoce como “aplicaciones puente”. En general, una vez decidida la lista de aplicaciones utilizadas que requieren un equivalente funcional en software libre, se debe decidir que aplicaciones proporcionan las mismas funciones en software libre. Hay muchos recursos *on-line* que le pueden guiar en este proceso.

- Enlaces del proyecto SourcePYME
<http://www.sourcepyme.org/?q=node/13>
- Tabla de equivalencias/reemplazos/análogos de software Windows en GNU/Linux
<http://www.linuxrsp.ru/win-lin-soft/index-spanish.html>
- Linuxquestions.org: “Software equivalents to Windows Software”
http://wiki.linuxquestions.org/wiki/Linux_software_equivalent_to_Windows_software
- www.novell.com: “Novell Linux Desktop Equivalents of Windows Software”
<http://www.novell.com/cool solutions/feature/11684.html>

Si no se está familiarizado con el software libre, la tarea de decidir qué software va a sustituir al actual requerirá algo de trabajo de investigación por parte del personal encargado de migrar los sistemas. Aunque, independientemente de si se está familiarizado con el software libre o no, también se puede contratar los servicios de un consultor externo que ayude a la empresa en este punto.

Nota

En caso de no encontrar una aplicación en software libre equivalente a la aplicación propietaria utilizada actualmente, la mejor práctica sería que la misma empresa la desarrollara si tiene recursos suficientes, y la liberara bajo una licencia libre o que la empresa colabore en el desarrollo de la herramienta necesitada junto con el resto de la comunidad de software libre. “Caminante no hay camino, se hace camino al andar”.

5.1.2.2. Inventario hardware

Al tener inventariado el hardware se conoce en detalle de qué máquinas se dispone para la migración, incluyendo máquinas retiradas. Este inventario permitirá comprobar la compatibilidad del hardware con el nuevo software (tarjetas gráficas, impresoras, etc...). También permitirá planificar si se va a adquirir nuevo hardware o no.

Consejo

Se pueden recuperar máquinas ya retiradas para implementar nuevos servicios, como por ejemplo un servidor de impresión, un servidor de correo electrónico o incluso un servidor de almacenamiento compartido o servidor web. (En teoría cualquier GNU/Linux se puede ejecutar en una máquina con un procesador compatible con la arquitectura Intel 386, pero para obtener un rendimiento aceptable se recomienda que si se utilizan máquinas retiradas, éstas no tengan un procesador inferior a la categoría Pentium)

Como el inventario de software nos permite saber con precisión del hardware que disponemos, podremos clasificar el hardware en una de las siguientes categorías:

5.1.2.2.1. Hardware sin problemas conocidos con GNU/Linux

- Hardware soportado nativamente por el núcleo Linux

El núcleo Linux presente en la mayoría de distribuciones GNU/Linux actuales, incorpora de serie soporte para gran cantidad de hardware. De hecho, el 90% de los equipos funcionarán sin necesidad de instalar controladores adicionales al soportado por el núcleo.

Esta es una ventaja competitiva de GNU/Linux, ya que nos olvidamos, salvo en casos contados, de tener que buscar controladores para el hardware. Podemos consultar una lista del hardware soportado por el núcleo Linux en <http://hardware4linux.info/> y en <http://www.mandriva.com/en/hardware/>

- Hardware soportado por controladores libres

Para el núcleo Linux, existe gran cantidad de hardware que, aunque no está soportado directamente por drivers nativos, tienen soporte de la comunidad de software libre con drivers completamente libres. Normalmente, estos drivers acaban por incorporarse al núcleo del sistema.

Nota

Con los drivers libres, el 90% del hardware actual funciona sin problemas, simplemente basta con conectar el hardware y el núcleo de Linux se encarga de cargar los drivers adecuados para cada tipo de hardware.

- Hardware soportado por controladores privativos

Es posible, que cierto tipo de hardware, funcione completamente solo mediante drivers propietarios. Este problema va desapareciendo paulatinamente y atañe principalmente a las aceleradoras de gráficos 3D. Este tipo de hardware puede hacerse funcionar perfectamente con Linux, pero no se dispone de drivers libres que permitan el funcionamiento. Afortunadamente, hay multitud de proyectos de drivers libres que se encargan, poco a poco de conseguir drivers libres para estos dispositivos, con lo que a largo plazo, esto deja de ser un problema.

Nota

Actualmente, aceleradoras gráficas nVIDIA y algunas tarjetas inalámbricas tienen este problema.

- Hardware soportado por adaptadores de drivers privativos

Hay cierto tipo de hardware que, simplemente, no tiene ningún driver. En muchas ocasiones, podemos hacer funcionar ese hardware perfectamente mediante herramientas de adaptación de drivers. Generalmente, este problema aparece con los drivers de algunas tarjetas inalámbricas de última generación.

Para hacer funcionar este hardware, podemos utilizar herramientas como NDISwrapper¹⁴ para utilizar los drivers de otros sistemas operativos (en este caso de Microsoft Windows) con el núcleo de Linux hasta que tengamos un driver libre.

Importante

La solución de adaptar drivers de otros sistemas operativos privativos para hacerlos funcionar con el núcleo Linux no es una buena idea, y solo debemos utilizar esto como algo temporal hasta disponer de drivers libres.

Aún así, hay que tener en cuenta que la mayoría del hardware no necesita de este tipo de software para hacerlo funcionar. Esto es solo útil en hardware muy determinado y, generalmente para equipos portátiles que incorporen adaptadores de red inalámbricos poco comunes.

5.1.2.2.2. Hardware que funciona, pero solo en versiones recientes del kernel Linux

Gran parte del software que *no funciona*, se debe a que muchas de las distribuciones actuales tienen un proceso de publicación que se alarga desde los 3 meses hasta 1 año

¹⁴ <http://ndiswrapper.sourceforge.net>

(bien, Debian es una excepción, ya que sus versiones estables han tardado hasta 5 años en aparecer). Durante este proceso, los programadores y los empaquetadores de la distribución tienen que fijar un núcleo base, una versión idéntica para todos los desarrolladores. De esta manera se garantiza que todos los paquetes (unidades instalables de software libre donde se gestionan las dependencias entre ellas de forma automática, generalmente ficheros .deb o .rpm) que se desarrollan para esa distribución. Es decir, muchas de las distribuciones no disponen de la última versión del núcleo.

Hay que tener en cuenta que el desarrollo del núcleo de Linux es muy veloz, incluso algunas revisiones duran sólo horas. Por tanto, es muy posible que el hardware no esté soportado por la distribución GNU/Linux elegida en ese momento.

Importante

Hay que utilizar siempre la última versión de la distribución GNU/Linux elegida para realizar la migración. De esta manera no sólo accedemos a las últimas mejoras en el núcleo, sino que podremos utilizar hardware más moderno.

5.1.2.2.3. Hardware que funciona, pero con un controlador libre antiguo no mantenido

Aunque generalmente no se da el caso, hay hardware que por ser muy antiguo (cuando se dice “muy”, realmente es “muchísimo”, es decir, hardware de hace más de 15 años) simplemente no tiene soporte. Es un caso extremadamente extraño, ya que los usuarios GNU/Linux tienden a alargar al máximo la vida útil de sus equipos. En este caso, simplemente podremos instalar una versión de la distribución GNU/Linux algo antigua, de esta manera podremos seguir usando el hardware aunque no dispongamos de las últimas mejoras del núcleo. De todas maneras, estos casos son extremadamente raros, ya que el núcleo Linux muy adaptable y las últimas versiones pueden utilizarse incluso con equipos muy antiguos, ya que los drivers suelen adaptarse a las últimas versiones.

5.1.2.2.4. Hardware que funciona, pero con limitaciones

Hay cierto tipo de hardware que funciona con limitaciones. Es posible que adaptadores de pantalla con salida de televisión funcionen perfectamente, exceptuando esa salida. Lo mismo ocurre con algunos adaptadores de pantalla que, pese a disponer de aceleración 3D por hardware, sólo funcionan en modo 2D.

Generalmente, estos dispositivos disponen de un driver propietario y es la versión libre la que no consigue sacarle todo el partido al hardware. En la mayoría de los casos es porque los fabricantes de hardware no dan las especificaciones de sus dispositivos a los desarrolladores de controladores libres, por lo que a estos no les queda más remedio que investigar cómo funcionan estos dispositivos, por lo que sólo pueden dar soporte a las funciones que son capaces de comprender.

5.1.2.2.5. Hardware que no soporta GNU/Linux

Para finalizar con el inventario de hardware, hay que tener en cuenta el hardware que simplemente no funciona. Esto, como hemos comentado en anteriormente, solo ocurre en determinadas ocasiones:

- El hardware es demasiado nuevo, y aún no se ha incluido soporte en el núcleo.
- El hardware es extremadamente antiguo, y ya no funciona en versiones modernas del núcleo.
- El hardware depende de software específico para un sistema operativo concreto, con lo que al no funcionar en GNU/Linux este software, no podemos utilizarlo.

Nota

Es muy complicado encontrar hardware que no funcione en las versiones modernas de Linux. Si no lo hace, probablemente sí lo haga a corto plazo.

5.1.2.3. Conclusión

Es importante categorizar el hardware del que disponemos en las categorías de los puntos anteriores. De esta manera detectaremos el hardware que no podemos utilizar en la migración, con lo que podremos buscarle una alternativa (bien adquiriendo nuevo hardware o esperando a que haya soporte para incluir ese hardware en la migración).

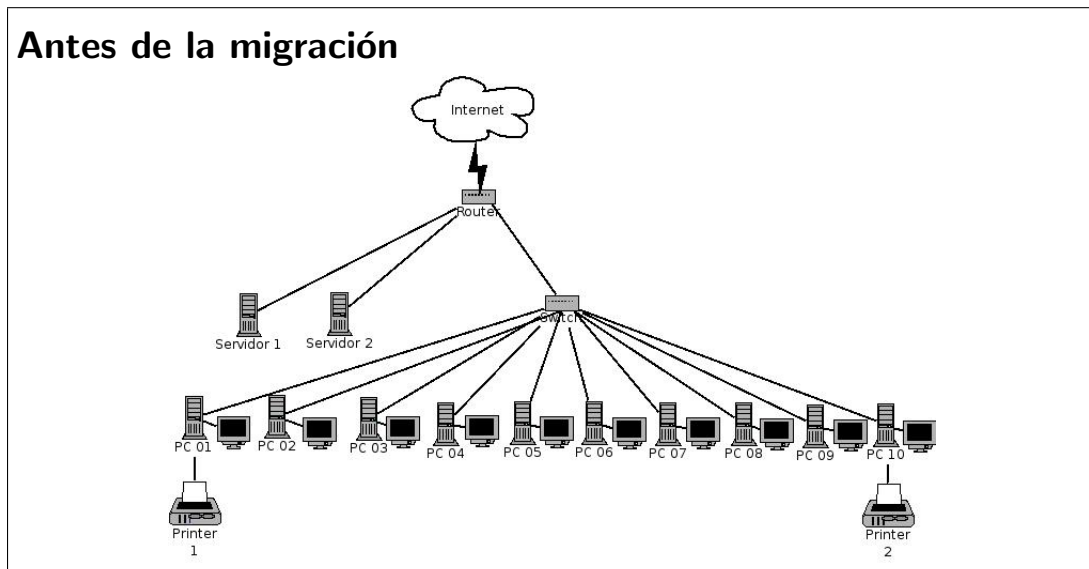
Aunque es posible encontrar hardware que no funcione con el sistema operativo GNU/Linux, generalmente no existen incompatibilidades que impidan por completo la migración. Si el hardware no funciona, generalmente no es culpa de los desarrolladores de GNU/Linux, sino de la empresa fabricante del hardware, que no muestra las especificaciones para poder desarrollar controladores libres.

5.1.3. Diagrama de red

Una vez se han estudiado los cambios en el software y en el hardware se procederá a reflejar dichos cambios en el diagrama de red de la empresa.

- ¿Se van a instalar nuevos servidores de servicios?
- ¿Se van a adquirir nuevos equipos de usuario?
- ¿Nuevas impresoras?
- ¿Se va a cambiar la conectividad entre equipos? Por ejemplo si se disponía de varias impresoras compartidas en distintos equipos de usuario y ahora se han centralizado en un servidor de impresión.

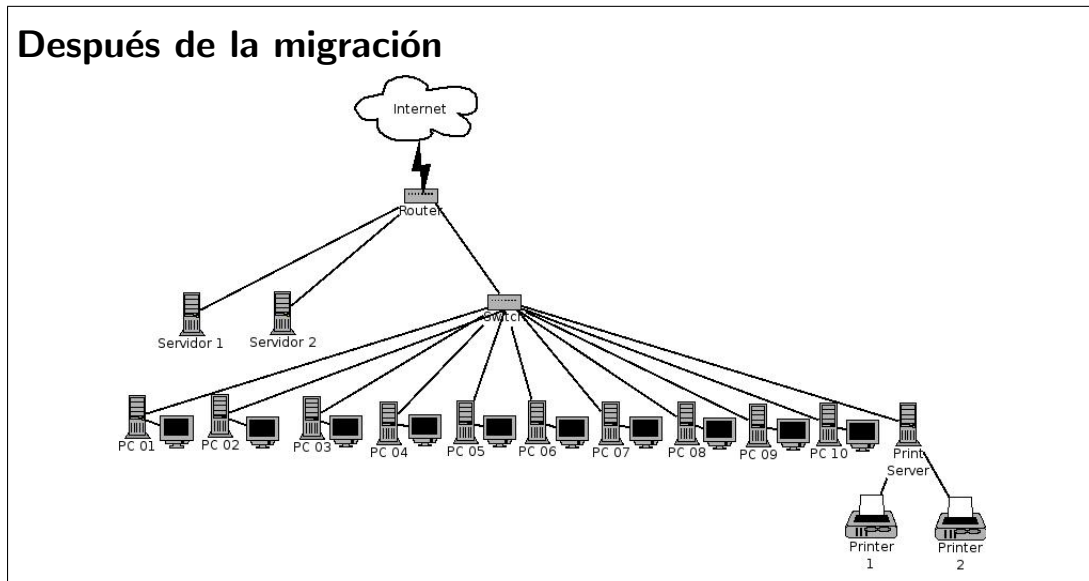
Es vital para futuros proyectos y futuros cambios tener bien documentado todo el proceso de migración. El diagrama de red es uno de los documentos más explicativos e importantes y se debe mantener actualizado.



Como podemos observar en el primer diagrama de red, tenemos un conjunto de 10 clientes que se conectan a través de un *switch* y un enrutador a Internet. En este caso, los clientes *PC01* y *PC02* disponen de impresoras que son compartidas en red.

Generalmente, cuando se hace una migración se aprovecha para actualizar la red o alguno de los equipos. Supongamos que se adquiere un equipo nuevo, para dotar al sistema de un entorno de red más estable, de manera que incluyamos un servidor de impresoras. De esta manera, si los usuarios en los equipos *PC01* o *PC02* apagan el equipo o reinician, no se interrumpe el servicio de estas impresoras.

Podemos incluso utilizar, como ya comentamos anteriormente, hardware antiguo para realizar estas tareas, ya que el hardware necesario para montar un servicio de impresión en red generalmente no tiene que ser muy potente.



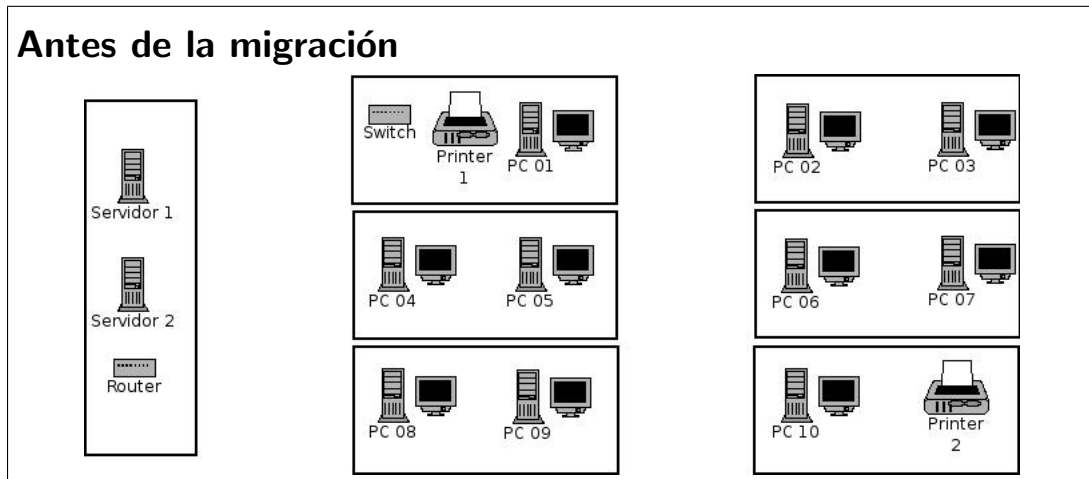
En el segundo diagrama de red podemos comprobar cómo al instalar el servidor de impresión, los equipos que realizaban esas tareas pueden utilizarse como el resto, sin interrumpir el servicio en ningún momento. Además, como el servicio de impresión está ahora localizado en un solo equipo, se puede optimizar este para gestionar de la manera más eficiente esa tarea especializada.

Importante

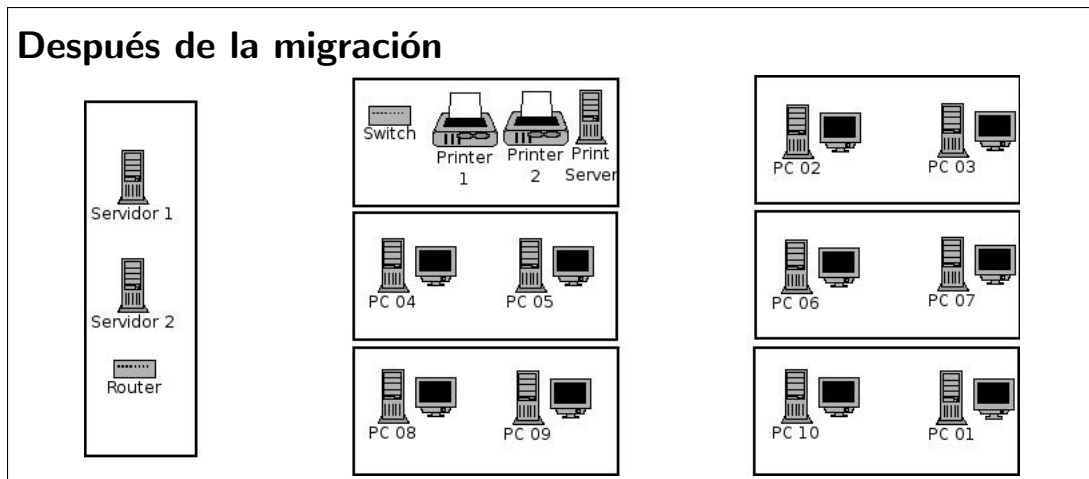
Aproveche la migración para realizar cambios estructurales en la red de equipos, así como cambios de hardware y relocalización de equipos dentro de sus instalaciones.

5.1.4. Diagrama de estructura

Al igual que con el diagrama de red, se debe contemplar cualquier cambio en el diagrama de estructura. Esto es, contemplar cualquier cambio en la posición física de los equipos de la empresa. Tener una buena documentación en este aspecto facilitará la tarea de los administradores de identificar dónde está cada equipo si la cantidad de máquinas de la empresa supera los 10 equipos.



Como podemos observar en el diagrama de estructura previo a la migración, los *servidores de impresión* anteriores estaban separados en lugares distintos. En cambio, en el diagrama de estructura posterior, al incluir un servidor de impresión dedicado, ahora se encuentran estructuralmente juntos. Esta es una manera de centralizar los recursos.



Siempre que sea posible, debemos tener servidores dedicados para los servicios que queramos ofrecer. No necesariamente un servidor por cada servicio, pero sí hay que tener bien definida las funciones de los equipos clientes de los servidores, para mantener los servicios que ofrecen estos servidores de forma independiente de los clientes que se conecten a ellos. Evidentemente, esto se debe realizar en la medida de lo posible. De todas formas, tenga en cuenta que la mayoría de las averías en los servicios de la empresa son provocados por los usuarios, de manera que si aislamos estos servicios en servidores dedicados, la fiabilidad será mucho mayor.

5.1.5. Elección de la estrategia de migración

Después de haber recopilado toda la información sobre la empresa y conocer cuáles son los tipos y estrategias de migración más comunes se debe decidir qué camino de migración

se va a seguir para planificar en detalle todas las tareas a llevar a cabo.

Debemos, por tanto, seleccionar el tipo y la estrategia de migración en función de las necesidades de su empresa.

En caso de disponer tanto de clientes como de servidores que proporcionen servicios, deberá ver si necesita migrar sólo una parte o ambos. Además, puede aprovechar la migración para relocalizar los equipos dentro de diferentes unidades estructurales (esto se plasma en el diagrama de estructura) y cómo se relacionan los equipos dentro de la red interna empresarial (en el diagrama de red).

Además, deberá tener en cuenta de qué hardware y software dispone actualmente, ya que puede que no sea posible realizar la migración de algunos equipos debido a esto.

Por tanto, tenemos que tener en cuenta y sopesar multitud de factores para poder realizar una migración a software libre con éxito.

Consejo

Para más información técnica en detalle consultar las guías mencionadas en el apartado de bibliografía.

5.2. Planificación de comunicaciones

La responsabilidad de comunicar y motivar al personal es una tarea, claramente definida, de los encargados de la gestión del cambio. Esta tarea comienza y debe ser llevada a cabo incluso antes de que el proyecto de migración empiece realmente. El liderazgo se alcanza mediante la comunicación, así pues el liderazgo y el estilo de comunicación están inseparablemente conectados, requiriendo un grado particularmente alto de habilidad social. Esto significa que los proyectos que se planeen deben ser transparentes para todos los miembros, de la empresa o de fuera de ella, involucrados en la migración. Se deben identificar tanto las áreas que se van a migrar como las que van a permanecer inalteradas.

Además, diferentes canales de comunicación deben ser utilizados para diseminar la información, como por ejemplo reuniones generales informativas, charlas con los empleados, seminarios o circulares internas, también se puede utilizar la intranet de la empresa (en caso de que se disponga).

Se deben establecer en una primera etapa los medios y las maneras de responder a las preguntas y las dudas, así como los miedos y preocupaciones del personal de la empresa, relacionados con el cambio.

Aconsejamos seguir el siguiente plan de comunicación con los empleados:

- a) Para realizar esta tarea, se aconseja, antes de realizar la migración y cuando dispongamos de la planificación técnica realizada, notificar todos esos cambios al personal. De esta manera, el personal de la empresa sabe, antes de

que ocurran los cambios realmente, dónde van producirse. También debemos notificarles cuándo van a realizarse estos cambios.

- b) Establecer un sistema de comunicación de incidencias que puedan utilizar los empleados, para poder atender las dudas que les surjan. Este es un punto clave y no debe dejarse pasar, ya que si los empleados disponen de esta herramienta, la pérdida de productividad debida al cambio será menor, sobretodo durante las primeras semanas.
- c) Establezca una reunión general inmediatamente posterior al cambio, para notificar las posibles modificaciones al plan inicial de migración.
- d) Realice reuniones posteriores de control, para comprobar cómo los distintos usuarios se han adaptado al cambio y solucionar problemas de última hora.

Consejo

Es recomendable utilizar una herramienta de gestión de proyectos, como por ejemplo dotProject, para facilitar la tarea de gestionar el proyecto de migración y todas sus actividades.

5.3. Planificación de recursos humanos

Una vez dividida la migración en tareas detalladas y bien definidas se debe detallar también qué recursos humanos se van a asignar a cada tarea. Es decir, quién va a llevar a cabo qué tareas. En este punto es importante la buena comunicación entre todos los miembros del proyecto para que todos tengan claras las tareas que deben realizar.

Nota

En este tipo de proyectos es muy importante el buen desarrollo del plan de comunicaciones elaborado en el apartado anterior.

También se deben planificar los recursos que se van a dedicar a la formación, calendario de formación, etc. Así como los posibles incentivos a aplicar al personal participante en el proyecto de migración.

Estas directrices no pretenden ser una guía de Gestión de Recursos Humanos, y las empresas ya se habrán encontrado anteriormente con estas cuestiones en otras áreas. Tendrán capacidad de hacerles frente de manera amable y favorable y así el personal de Recursos Humanos debería implicarse desde el principio. La intención es simplemente resaltar el tipo de cuestiones que han surgido en otros sitios que han migrado a software libre.

Es muy importante que se consulte a todo el personal y que se le mantenga informado de lo que se va haciendo. Un modo de hacerlo es crear una intranet que se pueda mantener actualizada fácilmente y en la que haya una sección dedicada a las opiniones de los usuarios.

Hay ciertas reacciones típicas a los cambios en las prácticas laborales que habrá que afrontar:

5.3.1. Miedo a lo desconocido

El uso del software libre será completamente nuevo para la mayoría de los usuarios y el personal de sistemas. El miedo a lo desconocido hará que las personas se resistan al cambio porque es nuevo para ellas.

Habrán usuarios que son más curiosos por naturaleza, que pueden sentirse felices de conocer cosas nuevas y son ellos las que deberían probar el software libre en primer lugar. Hasta ahora la experiencia indica que una vez que la gente vence sus reservas encuentra que el software libre no es muy diferente en su uso en comparación con el software propietario y está encantada de usarlo. Por ello es probable que este grupo inicial de usuarios se pase al software libre con entusiasmo. En cualquier caso, esta gente sería también la que proporcione los comentarios y sugerencias más útiles.

El primer grupo de usuarios podría utilizarse en pruebas piloto y una vez que tengan cierta experiencia ya pueden convencer y enseñar a sus colegas. En cualquier caso, ya en la segunda fase, los usuarios que pudieran ser más reservados necesitarán disponer de mayores facilidades de apoyo en forma de ventanillas de atención, intranets y usuarios locales con experiencia.

El mismo proceso se puede usar con el personal de sistemas pero el esfuerzo de formación podría ser importante si el entorno propietario existente no es como UNIX. El personal de sistemas en particular necesita desterrar sus temores desde el comienzo. Serán un punto focal para todos los problemas que indefectiblemente van a aparecer y si no creen en el proyecto no podrán animar a los usuarios de manera positiva.

5.3.2. El temor de que el CV pierda importancia

Tanto el personal de sistemas como los usuarios pueden pensar que no usar el software “estándar industrial” perjudicará su capacidad para desarrollar su carrera. Este es un problema delicado que hay que tratar con mucho cuidado. La empresa no querrá verse muy implicada en este enfoque pero hasta que el software libre sea de uso generalizado las empresas se pueden encontrar con él con cierta frecuencia.

Importante

Es importante tratar de darle la vuelta a este punto, es decir, enfocar el hecho de conocer el software libre como un complemento a su carrera profesional. No sólo podrán decir que utilizan el software “estándar industrial”, sino que además saben utilizar software libre.

5.3.3. Saber es poder

La gente que conoce los sistemas y configuraciones existentes tiene un cierto poder y podrían sentirse bastante reacios a perderlo si el entorno de software libre es muy diferente del existente. Y otra vez aparece la necesidad de una gestión cuidadosa ya que esas personas tienen un papel fundamental en el funcionamiento de los sistemas existentes. Quizá sea necesario que estén entre los primeros en recibir formación sobre los nuevos sistemas para que su posición en la entidad se mantenga.

5.4. Plan de contingencia

Como en todo proyecto, durante la migración pueden surgir problemas. Por eso se debe estar bien preparado y organizado para poder asumir y resolver las posibles incidencias que aparezcan en el transcurso de la migración.

El conocimiento compartido es la base para poder ahorrar recursos cuando surgen problemas, por eso se debe establecer métodos para informar de errores para poder estudiar la causa, dar una solución y aprender del error cometido para evitar que vuelva a suceder.

Nota

Se recomienda la utilización de sistemas de gestión de incidencias. Para organizar, gestionar y priorizar los posibles imprevistos.

En un proyecto de este tipo, y sobre todo si no se tiene experiencia previa, se debe esperar lo inesperado, así que también se deben planificar las herramientas y recursos para realizar una vuelta atrás en el caso, muy desfavorable e improbable, de que la migración falle completamente.

5.5. Planificación temporal

La planificación temporal se usa para desmenuzar el proyecto en detalle. Esta planificación requiere que se establezcan las fechas y plazos de entrega de forma realista para cada paquete de trabajo. La planificación temporal del proyecto dependerá de la fecha límite

establecida para el proyecto de migración. Además, la planificación temporal del proyecto especificará las fechas de comienzo, hitos y fechas de finalización de cada paquete de trabajo. Este calendario también sirve como una de las bases para una monitorización y gestión eficiente del proyecto.

Se establecerá cual es la fecha más adecuada para llevar a cabo la migración, por ejemplo puede ser adecuado realizarla durante el cierre vacacional de la empresa (en caso de haberlo). Si es necesario se establecerán horarios especiales para el personal encargado de la migración, de manera que la migración se pueda llevar a cabo alterando lo mínimo posible el buen funcionamiento de la empresa.

5.5.1. Planificación de pruebas

Modificar el calendario de migración para que incluya la realización de pruebas y un periodo de soporte post-migración.

5.6. Plan de evaluación

Establecer criterios para evaluar el éxito de la migración. Básicamente se trata de verificar si se han alcanzado los objetivos establecidos para la migración.

- ¿El nuevo software cumple con los requisitos?
- ¿Se han mejorado los procesos e infraestructuras de la empresa?
- ¿Se ha conseguido reducir los costes en TI?
- ¿Los usuarios se encuentran satisfechos con el nuevo sistema?

5.7. Planificación económica

Una estimación de costes se debe llevar a cabo para establecer qué inversiones y recursos van a ser necesarios. Las inversiones (dependiendo del trabajo a realizar) y el tiempo (dependiendo de la intensidad de trabajo) deben ser diferenciadas en este contexto. Los siguientes tipos de costes deben ser considerados cuando se planifican cada una de las tareas:

- Costes de personal.
- Costes materiales consumibles (como por ejemplo, costes de papel e impresión).
- Costes de hardware (equipamiento a ser adquirido).

- Costes de adquisición y compra de licencias (en caso de haberlas).
- Costes de servicio, soporte y formación.
- Costes varios (costes de desplazamientos, servicios externos, etc.).

6. Implantación

Ha llegado el momento de poner en práctica todo lo que se ha estado planificando, cuantos más recursos se hayan dedicado a la planificación del proyecto, menos incidencias se encontrarán a la hora de ponerlo en marcha y realizarlo. En este punto se debe empezar a ejecutar paso a paso todas las tareas planificadas, formación e implantación técnica.

6.1. Formación

Se llevarán a cabo las acciones de formación establecidas durante la planificación, cabe recordar que este es uno de los puntos importantes de la migración. Haber establecido un buen plan de formación ayudará a que la migración sea un éxito en todos sus aspectos.

Nota

Se recomienda la utilización de las herramientas “puente” previa a la implantación definitiva del nuevo sistema para evitar pérdidas de productividad.

Posteriormente a la migración, hay que formar al personal para que pueda realizar las tareas de su puesto de trabajo con el nuevo software. Gran cantidad del software es similar al que ya utilizan (navegadores web, aplicaciones ofimáticas) con lo que para estos casos sólo es necesario realizar tareas de formación para adaptarse a este software, explicándoles cuales son las pequeñas diferencias que puedan encontrar.

Es posible que haya software que sea completamente nuevo, por lo que es necesario realizar clases de formación para que los empleados comiencen a utilizar este software. Tenga en cuenta que debe incentivar esta formación, porque los usuarios del sistema generalmente se niegan al cambio.

6.1.1. ¿Cómo realizar la formación?

Existen multitud de recursos para dar cursos de formación. Se pueden utilizar métodos tradicionales, como impartir clases dentro de la propia empresa o subcontratar esta formación externamente.

Estos métodos, aunque eficaces a la hora de formar a los empleados, suponen un gasto adicional en la empresa. Para minimizar este gasto, existen portales de formación que el propio empresario puede instalar en la intranet, de manera que los empleados no necesiten abandonar su puesto de trabajo para formarse.

Advertencia

Mientras dure la formación para habituarse a las nuevas herramientas de trabajo, es posible que haya una pérdida de rendimiento. Esto lo tenemos que tener en cuenta y es una de las consecuencias iniciales del cambio.

Entre el software disponible para realizar tareas de formación, destaca Moodle¹⁵.

El acceso a la formación es muy importante. Algunos sitios permiten a los usuarios decidir por sí mismos si quieren asistir mientras que otros exigen la asistencia. La elección dependerá de la cultura de la empresa y de qué trate el curso de formación. Los manuales y la documentación general suelen estar sólo en inglés y esto podría causar problemas en algunos empleados. La traducción al idioma local podría considerarse como gastos de migración pero entonces aparece el problema de la traducción continuada de las actualizaciones.

La interfaz de usuario de OSS, en concreto, Gnome y KDE, permite elegir los idiomas pero la traducción puede no ser completa en algunos puntos del menú y las pantallas de ayuda son siempre en inglés. Gnome en particular tiene buenas facilidades de acceso para las personas con discapacidad visual. Y además no todas las aplicaciones tendrán pleno soporte de localización. Aunque todo esto está cambiando con rapidez y la estructura que permita el uso de un idioma que no sea el inglés está ahí por si la empresa quiere utilizarla.

6.2. Implantación técnica

6.2.1. Instalando muchos equipos

Dado que se puede realizar la migración en un alto número de equipos, es posible utilizar herramientas de instalación automática de equipos. Para este propósito, se puede utilizar SystemImager, que también es software libre. Mediante esta aplicación y las modificaciones que se han realizado sobre ella, se puede instalar y configurar cualquier PC en un tiempo reducido (unos veinte minutos). El proceso creará las particiones del disco duro

¹⁵ Moodle es un sistema de gestión de cursos de libre distribución (course management system CMS) que ayuda a los educadores a crear comunidades de aprendizaje en línea. Visite Moodle en <http://moodle.org>

(independientemente del tamaño de este), configurará la red, se cargarán los módulos apropiados de hardware y se ejecutará el arranque. El ahorro de tiempo es considerable.

Para ello se realiza una imagen de un equipo modelo (golden client), que es el que se quiere volcar al resto de equipos. Si en algún momento se modifica esta imagen (por actualizaciones de software, cambios de versiones, etc.), mediante un sencillo comando se podrá actualizar cualquier PC con esta nueva modificación. De esta forma, los equipos siempre estarán actualizados. Si por error u otra causa se eliminasen o corrompiesen archivos de la estación del usuario, éstos se podrían recuperar muy fácilmente, simplemente actualizando la máquina cliente.

Importante

En caso de que el hardware no sea homogéneo, es probable que haya que realizar instalaciones manuales en parte de los equipos para configurar aquellos aspectos que el equipo modelo no contemple, como puede ser hardware nuevo, servicios específicos, etc.

Aunque esta guía no pretende ser muy técnica, consideramos que existen una serie de tareas básicas y delicadas que se darán en la mayoría de migraciones. Es por ello que vamos a mostrar algunos elementos técnicos clave y cómo podría plantearse la implantación en un entorno de software libre.

6.2.2. Migrando datos de usuarios a sistemas GNU/Linux

Proporcionan una forma rápida de consulta de nombres, direcciones y datos afines. El estándar más popular para servicios de directorios es el LDAP. Consiste en un protocolo abierto y se incluye en muchos productos como, por ejemplo, Evolution y OpenOffice.org.

El LDAP trabaja con definiciones de datos llamadas esquemas y las empresas pueden crear sus propios esquemas personalizados. Lamentablemente, los esquemas utilizados por las aplicaciones no siempre son compatibles entre sí, lo que significa, por ejemplo, que a OpenOffice.org le resulta difícil leer los datos de Evolution y viceversa.

La aplicación libre OpenLDAP se ajusta al estándar LDAP versión 3, y la versión 2.1 y posteriores se pueden configurar con toda una serie de gestores de bases de datos (como archivos planos o servidores SQL).

Existen herramientas que nos ayuden en este proceso, de manera que la información pueda migrarse a OpenLDAP, como puede ser Versora Progression Desktop¹⁶ o Windows To Linux Migration Tools¹⁷.

¹⁶ http://www.versora.com/products/progression_desktop_linux.php

La mayoría de los groupware proporcionan alguna forma de servicio de directorio, pero pocos son los que se integran de manera compatible en el LDAP. Con pocas facilidades para las técnicas de “cortar y pegar”, es difícil usar la base de datos de contactos que proporcionan en agentes de correo externo. La mayoría de ellos ofrecen sus propios agentes de correo, pero no dicen mucho en cuanto al nivel de integración disponible en el gestor de contactos incorporado.

OpenOffice.org, Evolution y Mozilla proporcionan funciones integrales de libreta de direcciones. Sin embargo, los formatos de almacenamiento utilizados no son intercambiables. Para poder interactuar es necesario utilizar algún software de sincronización.

Afortunadamente, esta tendencia está cambiando, y son cada vez más las aplicaciones que permiten intercambiar datos entre ellas. Actualmente, Evolution ha incorporado en EDS (Evolution Data Server), un conjunto de librerías que permiten desarrollar aplicaciones que trabajen con el mismo conjunto de datos. De esta manera, ya hay gestores de tareas y contactos como Pimlico¹⁸ que se integran y sincronizan la información con Evolution de manera transparente y flexible.

6.2.3. Realización de copias de seguridad

Es obvio el ahorro de tiempo que suponen las copias incrementales, quedando ahora limitado el pesado periodo de espera de casi un par de horas al primer backup. Luego, las copias sucesivas estarán listas en breves minutos (unos dos o tres minutos).

Consejo

Para realizar este tipo de copias de seguridad, podemos utilizar RSync¹⁹ o Amanda²⁰.

Un tema importante es la conveniencia de separar lo más posible la copia y el sistema original. Si se utiliza el mismo sistema de ficheros original para guardar la copia (aunque sea otra partición) es evidente que un fallo del disco o un pico de tensión pueden estropear a la vez sin original y sin copia. Una idea mejor es dirigir la copia a otra máquina de nuestra red o a un espacio de almacenamiento remoto aunque guardar la copia en otro disco, externo y diferente es también una buena opción.

Por último, también conviene que las copias no puedan ser sobrescritas -o borradas- por error, obteniendo permisos de escritura sólo durante el lapso que dura la realización de la copia, y volviendo otra vez a “sólo lectura” al finalizar ésta.

¹⁷ <http://sourceforge.net/projects/w2lmt/>

¹⁸ <http://www.pimlico-project.org>

¹⁹ <http://samba.anu.edu.au/rsync>

²⁰ <http://amanda.org>

6.2.4. Emulación de aplicaciones

Para aquellas aplicaciones que no se pueden ejecutar de forma nativa en Linux se puede utilizar el emulador llamado Wine²¹, que permite la ejecución de aplicaciones Windows dentro de GNU/Linux, y que pertenece también al mundo del software libre.

Wine no necesita tener instalada ninguna partición Windows en la estación de trabajo GNU/Linux. Desde las aplicaciones nativas de Windows emuladas con WINE se puede acceder a los archivos, impresoras y bases de datos en red. Aunque se pueden emular un gran número de aplicaciones de forma óptima, hay ciertas aplicaciones que por su complejidad o su interacción con otros programas, plantean dificultades, y se necesitará ayuda de los desarrolladores para modificar pequeñas partes del código de las aplicaciones corporativas (si se dispone del código fuente o son aplicaciones internas de la empresa).

6.2.5. Servidores de archivos

Los servidores de archivos en red permiten a las máquinas que funcionan en la red su acceso al almacenamiento de datos en una máquina remota como si fuera local.

6.2.5.1. Sistema de archivos NFS

Este es el estándar *de facto* y ha estado en uso durante muchos años. El subconjunto que se implanta normalmente no proporciona una gran seguridad, aunque hay una variante segura que se ha puesto en marcha en algunas variantes comerciales de Unix.

NFS consiste en un servidor que exporta archivos desde la máquina en la que está funcionando a clientes que funcionan en otras máquinas adjuntas a la red. Se controla lo que otras máquinas puedan adjuntar a esos archivos pero una vez que se adjunta uno entonces el tráfico en la red queda fuera de toda sospecha. En la versión de Linux hay una mínima autenticación de los usuarios de Linux.

6.2.5.2. Samba

Samba es un producto que pone en marcha el protocolo SMB de Microsoft. Es fundamental para la integración de los sistemas basados en Windows y software libre, y viene en la mayoría de las distribuciones GNU/Linux.

²¹ <http://www.winehq.org/>

6.2.5.3. Appletalk

En el caso de los que tienen máquinas Apple de Macintosh, Netatalk²². proporciona la ejecución del protocolo AppleTalk

6.2.6. Bases de datos

Muchos proyectos pequeños de bases de datos en Windows usan Access. Este resulta un producto atractivo para mucha gente pues es bastante sencillo para comenzar, y tiene una interfaz de usuario familiar. Pero Access tiene fuertes limitaciones, al no haber sido diseñado para un entorno multiusuario pesado y no puede hacer frente a grandes conjuntos de datos.

Las bases de datos más grandes podrían usar SQL Server o una de las bien conocidas bases de datos relacionales: Oracle, Sybase, DB2, etc. En el caso de estos grandes sistemas, puede que lo mejor sea dejar que las bases de datos funcionen en la plataforma existente y sólo transferir las aplicaciones de cliente a plataformas software libre.

Hay modos estándar de conectarse a bases de datos relacionales en red, así que la elección de plataforma puede ser diferente para la base de datos y las aplicaciones de cliente. Además, la mayoría de las bases de datos propietarias que no son de Microsoft están disponibles en plataformas GNU/Linux y Unix, por lo que es posible cambiar el sistema operativo sin tener que aprenderse una base de datos totalmente nueva.

Por otra parte, las bases de datos propietarias pueden resultar muy caras por lo que vale la pena considerar si un producto libre podría hacer el trabajo con eficacia.

Las dos bases de datos más conocidas son MySQL y PostgreSQL. Las dos son productos maduros y cuentan con una buena base de instalaciones y equipos creativos muy activos. Las dos tienen buen soporte para el SQL estándar y pueden dar un magnífico rendimiento. También vale la pena recordar que las bases de datos no tienen que ser relacionales.

Algunas tareas van mejor con otros modelos, y el uso directo de software libre como la Berkeley DB de Sleepycat puede ser extremadamente eficaz. Igualmente, el modelo LDAP de bases de datos jerárquicas en red es muy adecuado para algunos tipos de aplicaciones distribuidas.

A veces transferir los datos es la parte más fácil del trabajo, aunque si el acceso a los mismos es por la red como tablas SQL entonces hay mucho más que hacer.

Lo más probable es que los problemas vengan de las utilidades secundarias y lenguajes scripting que rodean a cualquier base de datos práctica. El propio SQL está estandarizado, aunque casi todos los vendedores de bases de datos lo amplían y animan a la gente

²² <http://netatalk.sourceforge.net/>

a usar sus aplicaciones no estándar. También suele haber otros modos de conseguir un resultado dado en SQL, y elegir cuál es el más eficaz, varía de una base de datos a otra.

Muchas aplicaciones para bases de datos se construyen con generadores de aplicaciones o constructores de formularios y puede que no funcionen con otra base de datos que no sea aquella con la que venían.

Tanto MySQL como PostgreSQL han evolucionado muchísimo en los últimos años, por lo que es importante que se asegure de leer revisiones recientes al estudiar cuál usar y si hacer la migración o no.

6.2.6.1. Migración de las bases de datos de Access

Access sólo está disponible en plataformas Windows. Por ello, esas bases de datos deben ser transferidas a algún otro paquete si se proyecta un entorno completamente de software libre. Un interesante y útil escenario intermedio implica la migración de los datos a una base de datos libre, pero sin dejar de usar Access como entorno frontal. Esto tiene la deseable propiedad de eliminar muchas de las restricciones y problemas del almacén de datos Access.

Existen varios programas y scripts que exportarán una base de datos Access que tienen toda la información necesaria para recrear las tablas en otro gestor de bases de datos. Algunos producen archivos para ser copiados en una nueva plataforma, mientras otros se conectan directamente en la red y hacen los cambios de inmediato. Un ejemplo de los scripts para escritor de archivos es `exportsql2`²³, que produce archivos con cláusulas DROP TABLE, CREATE TABLE e INSERT que reproducirán la base de datos Access en MySQL.

En el informe de Paul DuBois titulado *Migrating from Microsoft Access to MySQL*²⁴ se describen otras herramientas de migración.

Una vez que los datos han sido transferidos ya se puede seguir usando Access como entorno cliente borrando las tablas localmente y enlazando con las tablas recién creadas en el servidor MySQL.

6.2.6.2. Migración de las bases de datos de SQL Server

El proceso aquí es en gran medida igual al descrito anteriormente; en el caso de las bases de datos sencillas suele bastar con exportar los datos a un formato común (normalmente CSV) y luego importarlos a la nueva base de datos. Las bases de datos más complejas que incluyen procedimientos almacenados y disparadores necesitarán más esfuerzo, y

²³ <http://www.cynergi.net/exportsql>

²⁴ <http://www.kitebird.com/articles/access-migrate.html>

en estos casos vale la pena mirar la gama de herramientas disponibles para ayudar al proceso de migración. Algunas son software libre y otras comerciales. He aquí algunos ejemplos:

- PGAdmin²⁵ es software libre para administrar bases de datos PostgreSQL. Hay utilidades modulares para ellas que manejan la migración de datos de otros gestores de bases de datos.
- SQLPorter²⁶ de Realssoftstudio: un producto comercial disponible en diversas variantes según los gestores de bases de datos inicial y final.
- SQLWays²⁶ de Ispirer: un producto comercial que da soporte a una amplia gama de gestores de bases de datos.
- SQLyog²⁸ es otra herramienta comercial que gestiona en MySQL y también maneja la migración de datos desde otras bases de datos de tipo ODBC.
- El sitio web de MySQL da una lista amplia de otras herramientas de conversión²⁹.

6.2.7. Sistemas de monitorización y administración

Para realizar un mantenimiento del equipo, es aconsejable instalar un sistema de monitorización y administración en todos los equipos, de manera que puedan ser accedidos de forma remota por los administradores de sistemas.

se aconseja el uso de OpenSSH³⁰ para realizar conectar por terminal a los equipos de forma remota, aunque otra opción atractiva es Webmin³¹, un sistema de administración vía web que permite configurar todos los aspectos de un equipo a través del navegador web de forma cómoda.

OpenSSH es una implementación libre de las herramientas de conectividad SSH que utilizan habitualmente los administradores remotos. Los usuarios de *telnet*, *rlogin* y *FTP* transmiten su password sin encriptar, en cambio OpenSSH encripta todo el tráfico (incluyendo los passwords) para eliminar efectivamente los riesgos de seguridad. Además,

²⁵ <http://www.pgadmin.org>

²⁶ <http://www.realssoftstudio.com>

²⁷ <http://www.ispirer.com/productos>

²⁸ <http://www.webyog.com/sqlyog>

²⁹ <http://mysql.com/portal/software/convertors>

³⁰ <http://www.openssh.org/>

³¹ <http://www.webmin.com/>

OpenSSH permite realizar conexiones seguras tunelizadas mediante varios métodos de autenticación, y soporta todas las versiones del protocolo SSH. OpenSSH está desarrollado por el proyecto OpenBSD (otra variante libre de Unix). Es software libre bajo la licencia BSD.

Webmin es una interfaz para la administración de sistemas Unix. Usando cualquier navegador web moderno, se pueden crear cuentas de usuario, configurar el servidor web Apache, realizar cambios del sistema de red, manejar el subsistema de compartición de archivos, etc. Webmin elimina la necesidad de editar manualmente los ficheros de configuración, y permite manejar el sistema desde la consola o de forma remota.

6.2.8. Otros elementos a migrar

Para obtener información más detallada de qué utilidades y aplicaciones hay disponibles para realizar migración de otros subsistemas, como puede servicios de red (compartición de impresoras, Web, FTP, SAMBA) o aplicaciones específicas (gestores de bases de datos personales, transformación de documentos a formatos libres, aplicaciones de ofimática) o técnicas (utilización de herramientas CAD, dibujo vectorial, etc), le recomendamos que consulte la biografía que se recoge al final de este documento.

6.3. Consejos de implantación

Hay ciertas circunstancias que pueden hacer que la introducción del software libre sea más fácil.

6.3.1. Introducir nuevas aplicaciones en un entorno familiar

Muchas de las aplicaciones libres funcionarán con sistemas operativos propietarios y esto nos brinda la oportunidad de introducir estas aplicaciones sin tener que cambiar totalmente el entorno. Por ejemplo OpenOffice.org, Mozilla y Apache funcionarán con Windows y así pueden utilizarse en sustitución de Office, Internet Explorer e ISS respectivamente. Aparte de ser menos rupturista, este enfoque permite que la reacción del usuario pueda ser calibrada a pequeña escala y que los planes para la formación de los usuarios puedan hacerse sobre la base de la experiencia real. Además, problemas como la conversión de formatos de archivos, macros y plantillas se pueden facilitar si la antigua aplicación se mantiene disponible durante algún tiempo.

Este enfoque significa que la elección de la aplicación en el entorno final se va a ver limitada a las que trabajan en el actual. Por ejemplo, el navegador final puede ser otro, pero Mozilla es el único que funcionará tanto con Windows como con GNU/Linux.

6.3.2. Lo fácil primero

Los primeros cambios serán los que no afecten a la comunidad de usuarios. Eso quiere decir que los primeros cambios se harán en el servidor. Estos cambios van a proporcionar la plataforma para la posterior introducción de los cambios en el lado del cliente. Muchos de los cambios relativos al servidor serán compatibles con el entorno actual, con lo que se podrá minimizar el efecto de ruptura. Por ejemplo, los servidores de nombres DNS, los servidores DHCP y los servidores de bases de datos principales con bases de datos propietarias como Oracle podrían ser todos ellos candidatos a ser reemplazados por un sistema en software libre equivalente y seguir interactuando con el resto de los sistemas actuales como antes. Más adelante se hablará de esto en detalle.

Hay aplicaciones como Samba que no se usarían en un entorno de software libre puro, pero que permiten la coexistencia de los antiguos sistemas propietarios y el software libre. El uso temprano de éstas puede ser muy eficaz en la división de los entornos en partes manejables.

6.3.3. Mirar hacia adelante

6.3.3.1. Desarrollos web basados en estándares

Insistir en que los desarrollos web hechos tanto internamente como por contratistas produzca un contenido que se pueda visualizar en todos los navegadores actuales de la web, en particular los navegadores libres. Esta sería una buena práctica en cualquier caso ya que las empresas no deberían requerir software específico para visualizar su contenido. Hay herramientas web para validar si una web sigue perfectamente el estándar, como el W3C Validator³² para comprobar la compatibilidad de las páginas web.

6.3.3.2. Evitar las macros y los scripts

No fomentar el uso indiscriminado de macros y scripts en documentos y hojas de cálculo; encontrar otros modos de proporcionar la necesaria funcionalidad. Ésta también es una buena práctica ya que de forma habitual los virus se valen de las macros y los scripts para infectar los sistemas. Además, las macros se pueden usar fácilmente para robar datos y corromper documentos: por ejemplo, podrían hacer que el documento diga cosas diferentes dependiendo de quien lo esté visionando y que se imprima otra cosa.

³² <http://validator.w3.org/>

6.3.3.3. Uso de formatos abiertos y estándar

Insistir en el uso de formatos de archivos abiertos y estándar, como PostScript y PDF. Hay cierta discusión sobre si PostScript y PDF son estándares abiertos o no. Es más una discusión sobre definiciones estrictas y en concreto sobre quién controla el estándar. En realidad, estos son los únicos formatos de archivos estándar que tienen un amplio uso en este momento, especificaciones públicamente accesibles y que se se pueden usar sin grandes restricciones.

Se están haciendo intentos para crear formatos de archivos estándar basados en XML y el Open Document Format (ODT) es un ejemplo. Sin embargo, sólo porque un archivo esté basado en XML ello no significa que vaya a ser abierto. En particular, no se deben usar formatos de archivos propietarios para archivos que son solamente para lectura y que el receptor no los va a editar. También en este caso sería una buena práctica, pues dichos archivos son una forma corriente de difundir virus. Usar esos formatos propietarios significa que la empresa se verá atrapada por el vendedor del software propietario durante bastante tiempo. Esos formatos propietarios también pueden incluir grandes cantidades de metadatos como, por ejemplo, texto previamente borrado, que si otros pueden visionar sería embarazoso para la empresa. Visualizar estos metadatos no es nada difícil.

Al escribir documentos en colaboración con otros, usar el formato que sea mínimo común denominador. Por ejemplo, hacer uso del formato Word 97 en lugar de Word 2000. Esto aumentará la posibilidad de que las aplicaciones libres puedan interactuar con el documento.

6.3.3.4. Usar protocolos abiertos y estándar

Utilizar protocolos abiertos estándar. Los protocolos abiertos estándar se definen como los que están libres patentes y cuentan con una implantación de software libre.

6.3.3.5. Desarrollar aplicaciones en 3 capas

Desarrollar sistemas basados en por lo menos un modelo de tres niveles donde el código de aplicación es independiente de la interfaz humana y de los métodos de acceso a los datos. Por ejemplo, si es posible, tener una interfaz de navegador que se pueda usar en un navegador web libre. Construir aplicaciones de esta forma modular facilitará hacer la migración bit a bit. Esto no sólo reducirá la escala de cualquier fase de migración sino que también reducirá el riesgo de fallo. Las tradicionales aplicaciones monolíticas de cliente son notablemente difíciles de manejar.

6.3.3.6. Utilizar tecnologías multiplataforma

Insistir en que las nuevas aplicaciones que necesite la empresa se escriban de manera que se sean portables. Esto incluye el usar lenguajes estandarizados portables como ANSI C,

Java, Python y Perl, y usar sólo librerías multiplataforma y librerías de construcción de interfaces (GUI Toolkits) portables. Evitar lenguajes y APIs de arquitecturas específicas. Evitar la construcción de aplicaciones que requieran la presencia de otras aplicaciones propietarias.

7. Evaluación

Ejecutar el plan de evaluación y continuar monitorizando el sistema en el tiempo identificando carencias o mejoras para incrementar paulatinamente la calidad del sistema de información de la empresa.

Para evaluar si la migración ha tenido éxito, podemos valorar los siguientes puntos:

- ¿Se ha migrado el Sistema Operativo de manera satisfactoria?

Es decir, el nuevo sistema operativo funciona, da al menos los mismos servicios que el sistema anterior, y además lo hace de forma correcta. En caso de detectarse algún error en alguno de estos puntos, debemos revisar qué ha fallado en la migración.

Evidentemente, habrá casos en los que técnicamente no ha podido darse la migración del sistema, debido, generalmente a problemas con aplicaciones que sólo funcionen en el sistema anterior a la migración y que no puedan hacerse funcionar con los emuladores y las máquinas virtuales, como ya comentamos recientemente.

- ¿Se han migrado las aplicaciones?

Se tiene que valorar si se utilizan las mismas aplicaciones o alguna de sus alternativas, si no hay limitación de características por usar software libre, es decir, que al menos las aplicaciones ofrezcan la misma (o parecida) funcionalidad a las que se utilizaban anteriormente. En caso de que la aplicación cumpla con este requisito, se puede considerar un éxito.

Desafortunadamente, existen multitud de aplicaciones propietarias o desarrolladas por las mismas empresas que no van a poder ser migradas al nuevo sistema. Para ello tendremos que adaptar el software al nuevo sistema, o adaptar el sistema al software (como en el punto anterior, hay que realizar pruebas con emuladores y virtualizadores).

- ¿Se han adaptado los usuarios?

¿Cómo trabajan los usuarios? ¿En qué medida han mejorado/disminuido su productividad? Este es un punto clave, ya que pese a que normalmente la productividad decaiga inicialmente, esta variable debe estudiarse a más largo plazo. Como siempre, no sólo dependerá de lo fácil que sea la transición al nuevo software, sino también de la predisposición al cambio que los empleados tengan.

- ¿Se ha mejorado con el cambio?

Hay que valorar también si el cambio ha mejorado en algún aspecto respecto al anterior sistema, es decir, si ahora se pueden ofrecer servicios que antes no se podía o aplicaciones a las que no se podía acceder.

Generalmente, los administradores de servicios ganarán con el cambio, ya que el sistema de acceso remoto y administración de un sistema GNU/Linux es mucho más potente y flexible que uno en un entorno Microsoft Windows. Los costes de mantenimiento se reducen, los costes de licencia también, e incluso los costes de actualización del hardware.

8. Casos de éxito

8.1. Proyecto SourcePYME

8.1.1. EBIR Iluminación

La empresa EBIR Iluminación, S.L., situada en Torrente, se dedica principalmente al diseño, fabricación y comercialización de luminarias para cuarto de baño entre los que se encuentran: Viseras de luz halógena e incandescente, apliques de luz halógena e incandescente, focos de luz halógena e incandescente,....

Se constituye en 1995, dedicándose desde sus inicios a la fabricación de luminarias de cuarto de baño, teniendo un crecimiento muy importante en los últimos años.

Hoy en día la empresa está formada por 48 personas y cuenta con más de 500 clientes, tanto a nivel nacional como internacional.

8.1.1.1. Descripción de hardware

En los servidores utilizan actualmente Windows 2000 Server. En los equipos de escritorio utilizan Windows XP en su mayoría aunque tienen algunos con Windows 2000 y Windows 98. En total su parque de hardware consta principalmente de 2 servidores y de 32 equipos de escritorio.

8.1.1.2. Descripción de software

En cuanto al software utilizado este es de ofimática en los equipos de escritorio principalmente: Office Xp, Office 2000 y Acrobat Reader.

Actualmente no hay software desarrollado por la empresa en el sentido estricto de la palabra, solo tienen definidas algunas macros en Excel.

Además utilizan una aplicación Access para realizar consultas a la base de datos del ERP a través de ODBC.

También tienen software de gestión que en este caso consta de el ERP GEINPROD que utiliza el motor de bases de datos Pervasive 9.5.

En cuanto a herramientas de diseño principalmente utilizan Solid Edge.

Y por último mencionar el software de seguridad informática utilizado, en concreto el Antivirus McAfee Viruscan 7.0 junto con un FireWall Cisco (Hardware).

8.1.1.3. Objetivo

El objetivo que se propuso fue realizar una migración parcial de los equipos de escritorio. En concreto el ofimático: Office, navegador de internet y correo electrónico.

Además se pretende complementar las herramientas de software disponible en la empresa con utilidades estadísticas que permitan el manejo de datos para realizar estadísticas sencillas, histogramas, etc.

En último lugar también se busca aplicaciones de retoque fotográfico y software para grabación de Cds.

8.1.2. Resimol

La actividad de Resimol va dirigida al sector del calzado. La empresa está ubicada en Crevillente, Alicante, y su principal misión es investigar en el uso de nuevos materiales y la creación de nuevas formas para las suelas de zapatos. El resultado de este trabajo de investigación ha sido el desarrollo de productos innovadores bajo dos nuevas marcas: softsole y proxim.

Los materiales que utiliza Resimol en sus procesos de fabricación son el PVC, TPR y EVA.

8.1.2.1. Descripción de hardware

Hay 4 Pcs y un ordenador portátil. Más los 2 Servidores.

Disponen de una impresora multifunción, una impresora matricial, una PDA y una cámara de fotos digital. Debería darse soporte a la sincronización de datos con el PDA y la descarga de imágenes de la cámara digital, así como las impresoras.

Los formatos de datos que se utilizan son los usuales en entornos de oficina, como los tipos de datos de Microsoft Office, PDF y archivos Zip.

8.1.2.2. Descripción de software

Se dispone de dos servidores uno de ellos ejecuta Debian y se utiliza para la distribución de correo electrónico mediante Sendmail y otro con Ubuntu y se utiliza para el ERP. El resto de equipos ejecutan Windows XP y Microsoft Office. También se utiliza una aplicación para realizar operaciones bancarias a través de internet proporcionada por la entidad bancaria con la que trabaja la empresa.

8.1.2.3. Objetivo

Para una migración menos intrusiva se buscaron aplicaciones de software libre equivalentes a las aplicaciones de software propietario utilizadas en la empresa. Tras su estudio y aceptación se implantaron en los equipos de trabajo.

También se implementó un terminal con el sistema operativo GNU Linux y las aplicaciones libres apropiadas para estudiar su aceptación por parte de los usuarios.

Posteriormente, una vez se haya comprobado el correcto funcionamiento de las nuevas aplicaciones, en los equipos que sólo utilicen aplicaciones de software libre se migrará el sistema operativo.

Se implantará un servidor de impresión en el servidor Ubuntu.

8.1.3. Mipesa Mecanizados, S.L.

Mipesa Mecanizados, S.L., empresa valenciana fundada en 1983 cuya actividad productiva se fundamenta en la fabricación y ensamblaje de piezas y conjuntos máquinas para los siguientes sectores: Automoción, quirúrgico, textil, químico, hostelería, juguetes, agrícola, muebles, fabricantes de maquinaria en general.

8.1.3.1. Descripción de hardware

Servidor HP el cual se planea sustituir por uno nuevo en un corto plazo de tiempo. Unas 30 terminales. Tienen dos terminales tontos EPIA Nano a 800 Mhz con Linux (elux) y un emulador de Terminal Server instalado a modo de test para la posible sustitución de todos los equipos por este tipo de terminal.

8.1.3.2. Descripción de software

Todas las aplicaciones se ejecutan en el servidor, incluida la ofimática. El servidor tiene instalado Windows 2000 server, ERP GEINPROD 9.7 y motor de base de datos PERVASIVE.

En el apartado de ofimática se utiliza OpenOffice, PDFCreator, FileZip, Microsoft Outlook Express e Internet Explorer.

Los equipos de escritorio ejecutan Windows 2000 Profesional y Windows XP (los más nuevos). Todos ellos acceden a las aplicaciones del servidor mediante Terminal Server.

Se utiliza Autocad para visualizar los diseños de los clientes (sólo visualizar, no modificar).

Estantería robotizada que se maneja a través del puerto paralelo de un PC.

8.1.3.3. Objetivo

Sustituir el software ofimático propietario por software libre (Office, Outlook, Iexplorer).

Dotar de herramientas de software libre para la importación de planos realizados en Autocad.

Se detecta que toda la LAN está conectada directamente a Internet aunque sin especificar puerta de enlace a excepción del PC con antivirus que se utiliza para verificar el correo. Además de este hay 5 equipos más completamente configurados para acceder a Internet. Todo lo anterior se resume en que los equipos no están aislados del tráfico procedente de internet con el riesgo consiguiente.

MIPESA ha desarrollado una estantería robotizada que se controla mediante un programa hecho a medida. Ese programa se ejecuta en un viejo PC con Windows 98 y temen que le pueda quedar poco tiempo de vida. Una posible solución puede ser crear una imagen virtual de ese PC y replicarla en otro equipo.

Sustitución gradual de los PC por terminales tontas EPIA Nano con Linux instalado. Sin embargo se deberá valorar si la facilidad de mantenimiento compensa el coste asociado a las licencias de terminal server que serán necesarias instalar en el servidor (nuevo) Windows 2003 Server.

Sistema de videovigilancia con cámaras IP.

8.1.4. Instituto Valenciano de la Edificación (I.V.E.)

El Instituto Valenciano de la Edificación (IVE), es una fundación constituida en octubre de 1986 con carácter particular y naturaleza permanente, siendo de interés público y estando sometida al derecho privado.

En los 20 años de su existencia, el INSTITUTO VALENCIANO DE LA EDIFICACION, ha desarrollado diversas actuaciones de investigación y desarrollo en el campo de la edificación.

8.1.4.1. Descripción de hardware

Disponen de más de 25 PCs clónicos con diferentes versiones de Windows instalados, un servidor y algunos portátiles.

También tienen algunas memorias USB, PALM, disco duro externo, cámara de fotos y grabadora de voz, además de unas 10 impresoras aproximadamente.

8.1.4.2. Descripción de software

La mayoría de PCs tienen Windows XP Professional, algunos tienen Windows XP Home y Windows 98.

Utilizan el siguiente software de terceras partes: AutoCAD LT 2007, Brics CAD v6, Corel Draw 12, MS Office 2003, Adobe Acrobat 7.0 Professional, Adobe Photoshop CS2, Total Commander 6.55, Menfis 5.7, Delphi 2006, Report Builder 10, Nero 7, Partition Magic 7, Norton Ghost 10, Outlook 2003, Dragon NaturallySpeaking 8.0 y BorgCHAT 0.9.

Además tienen desarrollados por la propia empresa un programa de facturación y un programa de tratamiento de archivos BC3.

Como software de seguridad, utilizan Antivirus Trend Micro Office Scan.

8.1.4.3. Objetivo

Están interesados en hacer una migración parcial en equipos y total en las aplicaciones que tengan un buen equivalente en software libre.

También desearían disponer de algunos terminales completamente Linux.

Las funcionalidades necesarias de las aplicaciones son procesador de texto, hoja de cálculo, presentación de diapositivas, correo electrónico, acceso a internet, agenda calendario de tareas, gestión de proyectos, CAD dibujo técnico, diseño vectorial, generación y modificación de pdf, programación de aplicaciones, contabilidad, facturación, herramienta CASE y reconocimiento de voz.

Actualmente, han instalado un servidor con Linux y Samba que utilizan como servidor de archivos.

8.2. Casos de éxito en España

8.2.1. INES: Iniciativa Española de Software y Servicios

La información sobre INES está extraída de una presentación de Jesús M. González Barahona.

8.2.1.1. ¿Que es INES?

INES es la Iniciativa Española de Software y Servicios, una plataforma tecnológica española en el área los sistemas y servicios de software. Forma una red de cooperación científico-tecnológica integrada por los agentes tecnológicos relevantes del área, y su fin último es incrementar la competitividad de la industria española, posicionándola entre los líderes europeos.

Se puede acceder la web de INES en la siguiente dirección: <http://www.ines.org.es/>

8.2.1.2. Software libre en INES

El software libre figura destacadamente en la agenda estratégica de investigación de INES, en la que sus acciones van encaminadas a promover comunidades de desarrollo basadas en software libre.

Para ello se ha creado dentro de INES un grupo de trabajo con Software Libre, donde se desarrollan las siguientes actividades en I+D:

- Calidad y gestión del software libre (modelos de calidad, métricas de gestión y calidad, etc.)
- Métodos y herramientas para software libre (trabajo en colaboración, forjas, etc.)
- Transferencia a otros ámbitos
- Desarrollo de software libre en áreas clave (sociedad de la información, implementación de estándares, etc.)

El grupo de trabajo de Software Libre está coordinado por Juan A. Cáceres (Telefónica I+D) y Jesús M. González Barahona (Universidad Rey Juan Carlos).

En este grupo participan varias universidades, empresas (incluyendo PYMES) y centros tecnológicos.

8.2.2. Proyecto MORFEO

Los datos del proyecto MORFEO están sacados de su web.

8.2.2.1. Objetivos de MORFEO

Acelerar el desarrollo de estándares software relacionados con Arquitecturas Orientadas a Servicios que son clave tanto en la integración de sistemas como en la evolución de la red como ecosistema en el que proliferen los servicios.

Crear oportunidades de negocio en el ámbito e integración de soluciones dirigidas a empresas y a la Administración que estén basadas en plataformas y aplicaciones estándares desarrolladas en el marco de la comunidad.

Mejorar la productividad y garantizar la calidad de los desarrollos ligados a proyectos software de código abierto que puedan integrarse dentro de la infraestructura de desarrollo software estándar en este tipo de proyectos (Gforge).

Catalizar proyectos de I+D+i en el ámbito software que integren de manera natural a diversos agentes científicos y tecnológicos, contribuyendo al impulso de actividades de I+D+i y al desarrollo de un tejido industrial fuerte en los países donde los miembros del consorcio operan.

8.2.2.2. ¿Qué proporciona la comunidad MORFEO?

La comunidad proporcionará una base de conocimiento en torno al modelo asociado al software de código abierto y facilitará asesoría a empresas en temas relacionados con el software de código abierto. En este sentido, la comunidad también contemplará el desarrollo de:

- Una base de información acerca de cuestiones generales sobre software libre o de código abierto, donde se desarrollará un discurso no excluyente frente al software comercial tradicional.
- Guías que asesoren a las empresas acerca de las implicaciones derivadas del uso de software de código abierto, de los pasos que han de dar en el proceso de liberación de software y de las posibilidades con las que cuentan en la definición de modelos de negocio alrededor de software de código abierto.

La comunidad MORFEO proporciona un marco abierto de colaboración entre empresas (grandes empresas y PYMEs), universidades, centros de investigación y la Administración que confían en (o desean experimentar) las posibilidades que el software de código abierto ofrece de cara a la consecución de los objetivos antes indicados. Desde su creación, ha tenido una amplia repercusión en medios y cuenta con el respaldo explícito de diversas Administraciones Públicas en España: Junta de Andalucía, Gobierno de Aragón, Junta de Comunidades de Castilla la Mancha, Junta de Extremadura, Generalitat de Catalunya y Generalitat Valenciana.

La comunidad se alimentará de la contribución de sus miembros pero Telefónica I+D actúa como “motor” de la comunidad liberando componentes software propios y sumando recursos a la comunidad.

Puede ver los miembros de la comunidad MORFEO visitando www.morfeo-project.org

8.3. Distribuciones españolas de GNU/Linux

8.3.1. Lliurex

LliureX es el proyecto de la Conselleria de Educación de la Generalitat Valenciana que tiene como objetivo principal la introducción de las nuevas Tecnologías de la Información

y la Comunicación basadas en software libre en el Sistema Educativo de la Comunidad Valenciana.

Para ello se ha creado LliureX, una distribución GNU/Linux creada por la Conselleria de Educación de la Generalitat Valenciana, orientada al Sistema Educativo de nuestra Comunidad.

LliureX está basado completamente en Software Libre y es gratuito.

La Conselleria de Educación de la Generalitat Valenciana apuesta por impulsar el proceso de modernización tecnológica a partir del proceso formativo.

La meta es formar profesionales que usen la informática como medio para el desempeño de sus tareas cotidianas y que sean capaces de crear software adaptado a las necesidades específicas de la Comunidad Valenciana.

La Conselleria de Educación de la Generalitat Valenciana apuesta por el Software Libre como modelo de desarrollo sobre el que cimentar la introducción de las nuevas tecnologías en el entorno educativo de la Comunidad Valenciana.

Y ¿por qué usar Software Libre para lograr estos objetivos?. Los motivos son los siguientes:

- Formación Tecnológica Genérica

Los estudiantes deben ser formados para usar la tecnología de la información en base a su funcionalidad y no a la empresa que la crea y distribuye, como sucede con el software propietario.

Si los alumnos son capaces de buscar una palabra en cualquier diccionario de papel, con independencia de la editorial que lo publica, también deben ser capaces de redactar un texto en cualquier editor de texto, con independencia de la empresa que lo haya creado. De lo contrario, no se enseña o se transmite conocimiento, sino que se adiestra para el consumo de los productos de una determinada empresa.

La formación debe ser en tecnologías genéricas sin obligar, ni a los alumnos, ni a los centros educativos, a adquirir costosas licencias para entrar en contacto con las nuevas tecnologías.

Esto es especialmente cierto cuando se habla de software cuya función es escribir textos, manejar una base de datos, navegación por internet y correo electrónico, etc.

Por tanto, y desde esta óptica, el uso del Software Libre es básico y fundamental.

- La Brecha Tecnológica

El abaratamiento progresivo que ha sufrido el hardware hace que cada vez sea más asequible la compra de un ordenador, suficientemente potente, para desempeñar tareas útiles.

Sin embargo, el problema ha ido derivando hacia la adquisición de Software Propietario original. Y aquí es donde se marca la diferencia entre alumnos que pueden

adquirir licencias del Software Propietario original, y aquellos que, no disponiendo de recursos económicos para adquirir software, se ven abocados a la copia ilegal del mismo.

De la misma forma que los alumnos se llevan a casa los libros de texto para seguir formándose y aprendiendo, es necesario que puedan utilizar en sus hogares el mismo software que utilizan en las aulas de informática de sus centros educativos. Y esto es realmente difícil de llevar a la práctica cuando se trata de Software Propietario, ya que sería necesario adquirir una licencia por cada ordenador en el que se vaya a ejecutar dicho Software.

- Bilingüismo

La Comunidad Valenciana es una comunidad bilingüe y las tecnologías no pueden ser ajenas a esta realidad.

La posibilidad de que los alumnos y profesores puedan, en las aulas informáticas, trabajar con su propia lengua e incluso crear nuevos programas o contenidos curriculares, no tiene el atractivo suficiente para que las multinacionales desarrollen software específico, básicamente por motivos económicos o políticos. Sin embargo para la Consellería de Educación es un punto fundamental y que la utilización de Software Libre permite llevar a cabo.

- Optimización de Recursos. Economía.

Implantar Software Libre en el sistema educativo implica liberar recursos que actualmente se destinan a la adquisición de licencias de Software Propietario y reparaciones de hardware y que podrían ser destinados a la construcción de nuevos colegios, dotación de material docente, o disminución de la ratio ordenador/alumno.

Desde la Conselleria de Educación es inadmisibles pagar por algo que se puede obtener de forma completamente gratuita.

En resumen, el Proyecto Lliurex tiene como principal objetivo implantar el uso de las nuevas tecnologías, principalmente a través del uso de Software Libre, en el Sistema Educativo Público de la Comunidad Valenciana.

8.3.2. LiNEX

GnuLinEx es una distribución de software libre que incluye el kernel Linux y está basada en Debian GNU/Linux y GNOME, contando con OpenOffice.org como Suite Ofimática, entre otras aplicaciones.

Está impulsado por la Consejería de Infraestructuras y Desarrollo Tecnológico de la Comunidad Autónoma de Extremadura (España), siendo pionero y secundado por otros organismos públicos y privados del resto de España. Durante un periodo considerable de tiempo, la comunidad extremeña ofreció también apoyo a la de Andalucía (la cual

se inspiró en GnuLinux para desarrollar Guadalinux) en la implantación de soluciones abiertas en colegios, administración, etc.

Esta distribución ha conseguido los siguientes premios:

- Premio que concede la Asociación de Usuarios de las Telecomunicaciones (AUTEL) al desarrollo de servicios avanzados de Tecnologías de la Información y la Comunicación (octubre de 2002)
- Premio de la Asociación de Usuarios Españoles de Linux (Hispalinux) a la Junta de Extremadura, por el desarrollo de gnuLinEx (noviembre de 2002)
- Premio «Computerworld 2003» a la Consejería de Educación, Ciencia y Tecnología de la Junta de Extremadura por la creación de gnuLinEx (marzo de 2003)
- Premio «Vettonia, voz Castúa» al consejero de Educación, Ciencia y Tecnología, concedido por la Agrupación Extremeña de Alcorcón, por el desarrollo de gnuLinEx (junio de 2003)
- Premio Europeo de la Innovación Regional, Categoría de Sociedad de la Información, otorgado por la Comisión Europea en el marco del Programa Regional Europeo de Acciones Innovadoras y entregado en el pleno del Comité de las Regiones, en Bruselas, el 22 de abril de 2004.

8.4. OpenBravo

Openbravo, S.L. es una compañía española de software que se está expandiendo en todo el mundo. La historia de Openbravo está basada en proporcionar al segmento de las pymes de un sistema de gestión empresarial (ERP) adaptado a sus necesidades.

Como primer paso para el cumplimiento del objetivo, dos de los tres cofundadores, Nicolás Serrano e Ismael Ciordia, desarrollaron un nuevo enfoque tecnológico que permitía el desarrollo de aplicaciones web de manera mucho más eficiente que como se venía haciendo habitualmente hasta el momento. En agosto de 2001 fundaron, junto con Moncho Aguinaga, Tecnicia (actualmente Openbravo) con el objetivo de aplicar esta innovadora arquitectura tecnológica al ámbito de los sistemas de gestión, iniciativa que en 2002 se materializó con la creación y lanzamiento del actual ERP.

En 2005, dos emprendedores con una experiencia muy relevante en la creación de empresas en diferentes zonas del mundo, deciden analizar la oportunidad de expandir las operaciones de Openbravo basándose en el modelo innovador de negocio que proporciona la filosofía de software libre. En 2006 estos dos emprendedores deciden incorporarse a Openbravo con el objetivo de hacer realidad el plan de negocio y como reza el logo, ¡abrir el futuro de los ERPs!

En paralelo, Sodena, un organismo español de desarrollo empresarial de la región de Navarra (una sociedad de capital riesgo), decide incorporarse al proyecto como accionista e inversor de referencia. Como resultado de la incorporación de los nuevos socios y la inyección financiera por parte de Sodena, el código de Openbravo se publica bajo licencia libre desde abril de 2006.

El sistema ya se ha implementado con éxito en pymes de distinto tamaño, que operan en diferentes sectores como la producción, la logística y distribución y los servicios. Openbravo ofrece a la red de partners una completa gama de servicios y herramientas profesionales para ayudarlos a desarrollar un negocio sólido y rentable basado en Openbravo.

8.5. Ikusnet

El grupo Ikusnet es un conjunto de empresas especializadas en la promoción y adopción de soluciones de código abierto a lo largo de toda España. Este grupo, que cuenta con 10 empresas, está especializado en:

- La implantación de redes de cableado estructurado, redes privadas virtuales, wireless, etc.
- Cubrir las necesidades formativas de sus clientes además de fomentar el uso del software libre.
- la aplicación soluciones para la optimización de la gestión interna de la empresa y de la relación con sus clientes a la vanguardia del desarrollo tecnológico.
- El equipamiento integral para la empresa actual: servidores, sistemas de alimentación ininterrumpida, back-up, equipos informáticos, periféricos, PDAs, componentes, multimedia, software, etc.

El grupo Ikusnet ha participado en numerosos proyectos, entre los que podemos destacar el desarrollo de la plataforma de comunicación externa en internet e intranet para la comunicación interna, con un gestor de documentos y aplicaciones de la gestión documental de la biblioteca del Museo Oteiza: fondo documental, fondo bibliográfico, fondo de prensa y catálogo de obras.

9. Promotores de Software Libre




















A continuación presentamos una lista de promotores de software libre por comunidades autónomas.

CCAA	Entidad (Sitio web)
Andalucía	Junta Andalucía (http://www.guadalinex.org)
Aragón	ITA (http://www.ita.es)
Asturias	CTIC (http://www.fundacionctic.org)
Baleares	IBIT (http://www.ibit.org)
Canarias	OSL (http://www.softwarelibre.ulpgc.es)
Cantabria	LinuxGlobal (http://www.linuxglobal.org)
Castilla la Mancha	CESLCAM (http://www.ceslcam.com)
Castilla León	CARTIF (http://www.cartif.es)
Cataluña	CATPL (http://www.catpl.org)
Extremadura	LiNEX (http://www.gnulinex.net) Vivernet (http://www.vivernet.com)
Galicia	CESGA (http://www.cesga.es)
La Rioja	AULIR (http://www.aulir.com)
Madrid	IMDEA (http://www.educa.madrid.org/web/madrid_linux/)
Murcia	CENTIC (http://www.centic.es)
Navarra	ATANA (http://www.atana.org)
País Vasco	ESI (http://www.esi.es) ESLE (http://www.esle-elkartea.org)
Valencia	COIICV (http://www.coicv.org) LLIUREX (http://www.lliurex.net) SourcePYME (http://www.sourcepyme.org)

10. Software recomendado por SourcePYME












Dentro del Proyecto SourcePYME, hemos seleccionado aquel software que mejor se adapta a cada una de las necesidades en función de la categoría a la que pertenece, de esta manera, esta lista puede servir como referencia para la selección del mejor software en cada una de las áreas.

10.1. Software de servicios y administración

Servidor Web	
★★★★★ Apache 2	http://www.apache.org     (Apache v2)
Servidor DNS	
★★★★★ BIND	http://www.isc.org/index.pl?/sw/bind/     (BSD)
Servidor FTP	
★★★★★ ProFTPD	http://www.proftpd.org/     (GPL)
Servidor DHCP	
★★★★★ ICS DHCP	http://www.isc.org/index.pl?/sw/dhcp/     (BSD)
Proxy	
★★★★★ Squid	http://www.squid-cache.org   = = (GPL)
Cortafuegos	
★★★★★ Shorewall	http://www.shorewall.net/ =  = = (GPL)
Antivirus	








































★★★★☆ ClamAV Antivirus	http://clamav.net =     (GPL)
Servidores de Impresión	
★★★★★ CUPS	http://www.cups.org/ =    (GPL-LGPL)
★★★★★ SAMBA	http://www.samba.org/ =    (GPL)
Interconectividad con Microsoft Windows	
★★★★★ SAMBA	http://www.samba.org/ =    (GPL)
Sistema de autenticación y directorio	
★★★★☆ OpenLDAP	http://openldap.org/ =  = = (OpenLDAP PL)
Servidor de correo electrónico	
★★★★★ PostFix	http://www.postfix.org =  = = (IBM Public License)
Servidore de mensajería instantánea	
★★★★★ Openfire	http://www.igniterealtime.org/projects/openfire/     (GPL)
Sistemas de sincronización y copias de seguridad	
★★★★★ rSync	http://rsync.samba.org/     (GPL)
★★★★★ Bacula	http://www.bacula.org/     (GPL-LGPL)
Sniffers	
★★★★★ WireShark	http://www.wireshark.org/     (GPL)
★★★★★ Kismet	http://www.kismetwireless.net/     (GPL)
Monitorización	
★★★★★ OCS Inventory NG	http://www.ocsinventory-ng.org/     (GPL)
★★★★★ Zenoss	http://www.zenoss.com/product/core     (GPL)
★★★★★ Pandora	http://pandora.sourceforge.net/     (GPL)

Virtualización y emulación	
★★★★★ VirtualBox	http://www.virtualbox.org     (GPL)
★★★★★ DosBox	http://dosbox.sourceforge.net/     (GPL)
★★★★☆ Wine	http://www.winehq.org/    (LGPL)
Servidores RDBMS	
★★★★★ PostgreSQL	http://www.postgresql.org/     (GPL)
Administración y control remoto	
★★★★★ TightVNC	http://www.tightvnc.com/    (GPL)
★★★★★ Putty	http://www.chiark.greenend.org.uk/~sgtatham/putty/    (MIT)
★★★★☆ Xming	http://www.straightrunning.com/XmingNotes/    (GPL-LGPL-ZLib-BSD-MIT)
★★★★☆ RDesktop	http://www.rdesktop.org/    (GPL)
Gestión de negocios (ERP y CRM)	
★★★★★ Openbravo	http://www.openbravo.com     (MPL)
★★★★★ SugarCRM	http://www.sugarforge.org/content/open-source/     (GPL)
Portales e intranets	
★★★★★ Drupal	http://drupal.org/     (GPL)
★★★★★ Zope	http://www.zope.org     (ZPL-GPL)
Gestión de proyectos y Groupware	
★★★★★ Zimbra	http://www.zimbra.com    (Yahoo Public License)
★★★★★ eGroupWare	http://www.egroupware.org/     (GPL)
Formación	

★★★★★ Moodle	http://www.moodle.org =    (GPL)
Control de versiones	
★★★★★ GIT	http://git.or.cz/     (GPL)
Control de incidencias	
★★★★★ TRAC	http://trac.edgewall.org/     (BSD)

10.2. Aplicaciones de escritorio

Procesador de texto	
★★★★★ OpenOffice.org Writer	http://openoffice.org     (LGPL)
Hoja de cálculo	
★★★★★ OpenOffice.org Calc	http://openoffice.org     (LGPL)
Bases de datos texto	
★★★★★ OpenOffice.org Base	http://openoffice.org     (LGPL)
Presentaciones	
★★★★★ OpenOffice.org Impress	http://openoffice.org     (LGPL)
Organizadores de información personal (PIM)	
★★★★★ Evolution	http://www.gnome.org/projects/evolution/ =    (GPL)
Visores y generadores de documentos electrónicos	
★★★★★ PDFCreator	http://www.pdfforge.org/products/pdfcreator  = = = (GPL)
★★★★★ Evince	http://www.gnome.org/projects/evince/ =    (GPL)
★★★★★ KPDF	http://kpdf.kde.org/ =    (GPL)
★★★★☆ SumatraPDF	http://blog.kowalczyk.info/software/sumatrapdf/  = = = (GPL)
Sincronización de dispositivos	

★★★★★ Opensync	http://www.opensync.org/ =    (GPL)
Maquetación	
★★★★★ Scribus	http://www.scribus.net/     (GPL)
Diagramas	
★★★★★ Kivio	http://www.koffice.org/kivio/ =    (GPL)
Gestión de proyectos	
★★★★★ Openproj	http://openproj.org     (CPAL)
Compresores y descompresores	
★★★★★ 7 Zip	http://www.7-zip.org/  = = = (LGPL)
★★★★★ PeaZip	http://peazip.sourceforge.net/ =    (GPL)
Grabación de CD y DVD	
★★★★★ K3b	http://k3b.plainblack.com/ =  = = (GPL)
★★★★★ InfraRecorder	http://infrarecorder.sourceforge.net/  = = = (GPL)
Navegador web	
★★★★★ Mozilla FireFox	www.mozilla.com/firefox/     (MPL)
Correo electrónico	
★★★★★ Mozilla ThunderBird	http://www.mozilla.com/thunderbird/     (MPL)
★★★★★ GNOME Evolution	http://www.gnome.org/projects/evolution/ =    (GPL)
Telefonía IP	
★★★★★ OpenWengo	http://www.openwengo.com/     (GPL)
Lectores de noticias	
★★★★☆ RSSOwl	http://www.rssowl.org/     (GPL)
Mensajería instantánea	

★★★★★ Pidgin (antes GAIM)	http://www.pidgin.im/     (GPL)
Reproducción de video	
★★★★★ MPlayer	http://www.mplayerhq.hu/     (GPL)
★★★★★ VLC Media Player	http://www.videolan.org/vlc/    (GPL)
Reproducción de audio	
★★★★★ BMP	beep-media-player.org/    (GPL)
Edición de vídeo	
★★★★★ Jahshaka	http://www.jahshaka.org     (GPL)
★★★★★ Kdenlive	http://kdenlive.org/    (GPL)
Edición de audio	
★★★★★ Ardour	http://ardour.org/    (GPL)
★★★★☆ Audacity	http://audacity.sourceforge.net/     (GPL)
Contabilidad y facturación	
★★★★☆ Keme	http://keme.sourceforge.net/    (GPL)
Edición y retoque fotográfico	
★★★★★ Gimp	http://www.gimp.org/     (GPL)
★★★★★ Paint.NET	http://www.getpaint.net/     (MIT)
Edición CAD/CAM	
★★★★★ FreeCAD	http://juergen-riegel.net/FreeCAD/Docu/     (GPL-LGPL)
Edición de imágenes vectoriales	
★★★★★ Inkscape	http://www.inkscape.org/     (GPL)
Edición 3D	

★★★★★ Blender	http://www.blender.org/     (GPL)
Entornos Integrados de Desarrollo de Aplicaciones	
★★★★★ Eclipse	http://www.eclipse.org     (Eclipse Public License)
★★★★★ SharpDevelop	http://www.icsharpcode.net/OpenSource/SD/     (GPL)
★★★★★ KDevelop	http://www.kdevelop.org/     (GPL)

11. Directorio de software

Una de las necesidades a la hora de realizar una migración, es disponer de una serie de alternativas basadas en software libre. En función del tipo de migración que vayamos a realizar nos interesará poder tener una lista de software, o *directorio de software* que nos sirva de referencia para la búsqueda de aplicaciones alternativas de calidad.

Pasamos, a continuación, a enumerar la lista de aplicaciones basadas en software libre por categorías.

11.1. Servicios y administración































A continuación mostramos una lista de software dedicado a ofrecer servicios para que consuman los clientes, generalmente de forma remota. también se presentas herramientas típicas que se encuentran en servidores, así como algunas herramientas administrativas.

11.1.1. Servidores Web

Software	Enlace y características
★★★★★ Apache 2	http://www.apache.org     (Apache v2)
★★★★★ thttpd	http://www.acme.com/software/thttpd/ =    (BSD)
★★★★★ LightHTTPD	http://www.lighttpd.net =    (BSD)
★★★★☆ Mathopd	http://www.mathopd.org =    (BSD)
★★★★☆ IBM Http Server	http://www.ibm.com/   =  (Apache 2)
★★★★☆ AOLServer	http://aolserver.sourceforge.net     (MPL)

































Software	Enlace y características
★★★★☆ Roxen Webserver	http://www.roxen.com/products/cms/webserver/     (GPL)
★★★★☆ Boa Webserver	http://www.boa.org =    (GPL)
★★★★☆ Jigsaw W3C	http://www.w3.org/Jigsaw/     (W3CIRNL)
★★★★☆ Savant	http://savant.sourceforge.net  = = = (GPL)
★★★★☆ Jetty	http://www.mortbay.org     (Apache v2)

11.1.2. Servidores DNS

Software	Enlace y características
★★★★★ BIND	http://www.isc.org/index.pl?/sw/bind/     (BSD)
★★★★★ PowerDNS	http://www.powerdns.com/     (GPL)
★★★★★ NSD	http://www.nlnetlabs.nl/nsd/ =  =  (BSD)
★★★★★ djbdns	http://cr.yp.to/djbdns/ =  =  (Sin licencia)
★★★★☆ DHCP-DNS Dual Server	http://sourceforge.net/projects/dhcp-dns-server/     (GPL)
★★★★☆ MaraDNS	http://www.maradns.org/ =    (MIT)
★★★★☆ pDNSd	http://www.phys.uu.nl/~rombouts/pdnsd/idex.html =    (GPL)
★★★★☆ Oak DNS Server	http://www.digitallumber.com/oak/     (LGPL)
★★★★☆ MyDNS	http://mydns.bboy.net/ =  = = (GPL)
★★★★☆ Posadis	http://posadis.sourceforge.net/ =  = = (GPL)
★★★★☆ Xyria:DNS	http://www.doomed-reality.org/site/projekte/xyria/ =  =  (BSD)

Software	Enlace y características
★★★★★ SheerDNS	http://threading.2038bug.com/sheerdns/ =     (GPL)



11.1.3. Servidores FTP

Software	Enlace y características
★★★★★ ProFTPD	http://www.proftpd.org/     (GPL)
★★★★★ GNU InetUtils FTPd	http://www.gnu.org/software/inetutils =    (GPL)
★★★★★ wzdFTPd	http://www.wzdftpd.net/trac =  =  (BSD)
★★★★★ ncftpd	http://www.ncftp.com/ =    (BSD)
★★★★★ WinSCP	http://sourceforge.net/projects/winscp/  = = = (GPL)
★★★★★ PureFTPd	http://www.pureftpd.org/project/pure-ftpd =    (BSD)
★★★★★ WU-FTPd	http://www.wu-ftp.org/ =    (BSD)
★★★★★ acFTP	http://sourceforge.net/projects/acftp/  = = = (GPL)
★★★★★ Apache FTP Server	http://incubator.apache.org/ftpserver/     (Apache v2)
★★★★★ CrossFTP Server	http://www.crossftp.com     (GPL)
★★★★★ Hermes FTP Server	http://hermesftp.sourceforge.net/     (GPL)








11.1.4. Servidores DHCP

Software	Enlace y características
★★★★★ ICS DHCP	http://www.isc.org/index.pl?/sw/dhcp/     (BSD)
★★★★★ DHCP-DNS Dual Server	http://sourceforge.net/projects/dhcp-dns-server/     (GPL)





11.1.5. Proxys

Software	Enlace y características
★★★★★ Squid	http://www.squid-cache.org   = = (GPL)


11.1.6. Cortafuegos (firewalls)

Software	Enlace y características
★★★★★ Shorewall	http://www.shorewall.net/ =  = = (GPL)
★★★★☆ IPCop	http://ipcop.org/ =    (GPL)
★★★★☆ Guarddog	http://www.simonzone.com/software/guarddog/ =  = = (GPL)
★★★★☆ Ebox Platform	http://ebox-platform.com/ =  = = (GPL)
★★★★☆ Firestarter	http://www.fs-security.com/ =  = = (GPL)

11.1.7. Antivirus

Software	Enlace y características
★★★★☆ ClamAV Antivirus	http://clamav.net =    (GPL)
★★★★☆ ClamWin	http://www.clamwin.com/  = = = (GPL)



11.1.8. Servidores de impresión

Software	Enlace y características
★★★★★ CUPS	http://www.cups.org/ =    (GPL-LGPL)
★★★★★ SAMBA	http://www.samba.org/ =    (GPL)




11.1.9. Interconectividad con Microsoft Windows

Software	Enlace y características
★★★★★ SAMBA	http://www.samba.org/ =    (GPL)

11.1.10. Sistemas de autenticación y directorio





Software	Enlace y características
★★★★☆ FreeRadius	http://freeradius.org/ =  = = (GPL-LGPL)
★★★★☆ OpenLDAP	http://openldap.org/ =  = = (OpenLDAP PL)

11.1.11. Servidores de correo electrónico


Software	Enlace y características
★★★★★ PostFix	http://www.postfix.org =  = = (IBM Public License)
★★★★★ hMailServer	http://www.hmailserver.com/  = = = (GPL)
★★★★★ QMail	Qmail http://www.qmail.org/ =  = = (Sin Licencia)
★★★★☆ Exim	http://www.exim.org/ =  = = (GPL)
★★★★☆ Courier MTA	http://www.courier-mta.org/ =  = = (GPL)
★★★★☆ SendMail	http://www.sendmail.org/ =  = = (GPL)

11.1.12. Servidores de mensajería instantánea

Software	Enlace y características
★★★★★ Openfire	http://www.igniterealtime.org/projects/openfire/     (GPL)
★★★★★ ejabberd	http://www.process-one.net/en/ejabberd/     (GPL)




Software	Enlace y características
★★★★☆ jabberd	http://jabberd.jabberstudio.org/  (GPL)
★★★★☆ Tigase	http://www.tigase.org/  (GPL)
★★★★☆ psyced	http://www.psyced.org/  (GPL)
★★★★☆ OpenIM	http://www.open-im.net/en/  (BSD)

11.1.13. Sistemas de sincronización y copias de seguridad

Software	Enlace y características
★★★★★ rSync	http://rsync.samba.org/  (GPL)
★★★★★ Bacula	http://www.bacula.org/  (GPL-LGPL)
★★★★☆ Amanda	http://www.amanda.org/ =  (MIT)
★★★★☆ BackupPC	http://backuppc.sourceforge.net/ =  (GPL)
★★★★☆ rdiff-backup	http://www.nongnu.org/rdiff-backup/ =  = = (GPL)

11.1.14. Sistemas de redes, monitorización y alarmas

11.1.14.1. Sniffers

Software	Enlace y características
★★★★★ WireShark	http://www.wireshark.org/  (GPL)
★★★★★ Kismet	http://www.kismetwireless.net/  (GPL)
★★★★☆ Aircrack-ng	http://www.aircrack-ng.org/  (GPL)

11.1.14.2. Monitorización

Software	Enlace y características
★★★★★ OCS Inventory NG	http://www.ocsinventory-ng.org/  (GPL)
★★★★★ Zenoss	http://www.zenoss.com/product/core  (GPL)
★★★★★ Pandora	http://pandora.sourceforge.net/  (GPL)
★★★★★ Zabbix	http://www.zabbix.com/  (GPL)
★★★★★ Nagios	http://www.nagios.org/  (GPL)
★★★★☆ OpenNMS	http://www.opennms.org  (GPL)
★★★★☆ MRTG	http://oss.oetiker.ch/mrtg/  (GPL)
★★★★☆ SNM	http://snm.sourceforge.net/  (GPL)
★★★★☆ NTop	http://www.ntop.org  (GPL)

11.1.15. Software de virtualización y emulación

Software	Enlace y características
★★★★★ VirtualBox	http://www.virtualbox.org  (GPL)
★★★★★ DosBox	http://dosbox.sourceforge.net/  (GPL)
★★★★☆ Wine	http://www.winehq.org/ =  (LGPL)
★★★★☆ QEmu	http://fabrice.bellard.free.fr/qemu/ =  = = (GPL-LGPL-BSD)
★★★★☆ Xen	http://www.xen.org =  = = (GPL-LGPL-BSD)
★★★★☆ Linux VServer	http://linux-vserver.org/ =  = = (GPL)


11.1.16. Servidores RDBMS

Software	Enlace y características
★★★★★ PostgreSQL	http://www.postgresql.org/  (GPL)
★★★★★ MySQL	http://www.mysql.com  (GPL)
★★★★☆ FireBird	http://www.firebirdsql.org/  (MPL)

11.1.17. Software de administración y control remoto











Software	Enlace y características
★★★★★ TightVNC	http://www.tightvnc.com/  (GPL)
★★★★★ Putty	http://www.chiark.greenend.org.uk/~sgtatham/putty/  (MIT)
★★★★☆ Xming	http://www.straightrunning.com/XmingNotes/  (GPL-LGPL-ZLib-BSD-MIT)
★★★★☆ RDesktop	http://www.rdesktop.org/  (GPL)
★★★★☆ UltraVNC	http://www.uvnc.com/  (GPL)

11.1.18. Software de gestión de negocios (ERP y CRM)



Software	Enlace y características
★★★★★ Openbravo	http://www.openbravo.com  (MPL)
★★★★★ SugarCRM	http://www.sugarforge.org/content/open-source/  (GPL)
★★★★☆ OpenXpertia	http://sourceforge.net/projects/openxpertya/  (MPL)
★★★★☆ Pentaho	http://www.pentaho.org  (MPL)
★★★★☆ Adempiere	http://www.adempiere.com/  (GPL)

Software	Enlace y características
★★★★☆ TinyERP	http://www.tinyerp.org/     (GPL)
★★★★☆ Compiere	http://www.compiere.com/     (GPL)
★★★★☆ OpenTabs	http://www.opentaps.org/     (GPI)
★★★★☆ GNU Enterprise	http://www.gnuenterprise.org/    (GPL)
★★★★☆ VTiger	http://www.vtiger.com/     (MPL)
★★★★☆ Abanq	http://www.infosial.com/     (GPL)

11.1.19. Sistemas de creación de portales e intranets

Software	Enlace y características
★★★★★ Drupal	http://drupal.org/     (GPL)
★★★★★ Zope	http://www.zope.org     (ZPL-GPL)
★★★★★ Joomla	http://www.joomla.org/     (GPL)
★★★★☆ Plone	http://www.plone.org     (GPL)
★★★★☆ Xoops	http://www.xoops.org     (GPL)
★★★★☆ OpenCMS	http://www.opencms.org/     (GPL)
★★★★☆ Postnuke	http://www.postnuke.com/     (GPL)

11.1.20. Software de gestión de proyectos y groupware

Software	Enlace y características
★★★★★ Zimbra	http://www.zimbra.com =    (Yahoo Public License)
★★★★★ eGroupWare	http://www.egroupware.org/     (GPL)
★★★★★ dotProject	http://www.dotproject.net/     (GPL)
★★★★☆ openGroupWare	http://www.opengroupware.org     (GPL)

11.1.21. Sistemas de formación (web)

















Software	Enlace y características
★★★★★ Moodle	http://www.moodle.org =    (GPL)
★★★★★ Sakai Project	http://www.sakaiproject.org/     (Educational Community License)
★★★★☆ ILIAS	http://www.ilias.de/     (GPL)
★★★★☆ ATutor	http://www.atutor.ca/     (GPL)
★★★★☆ Claroline	http://www.claroline.net/     (GPL)
★★★★☆ Dokeos	http://www.dokeos.com/     (GPL)
★★★★☆ LogiCampus	http://logicampus.sourceforge.net/     (LogiCreate Public License)
★★★★☆ OLAT	http://www.olat.org/     (Apache v2)

11.1.22. Servidores de control de versiones

Software	Enlace y características
★★★★★ GIT	http://git.or.cz/     (GPL)

Software	Enlace y características
★★★★★ Subversion	http://www.nongnu.org/cvs/     (GPL)
★★★★★ Bazaar	http://bazaar-vcs.org/     (GPL)
★★★★☆ CVS	http://www.nongnu.org/cvs/     (GPL)

















11.1.23. Sistemas de control de incidencias

Software	Enlace y características
★★★★★ TRAC	http://trac.edgewall.org/     (BSD)
★★★★☆ Mantis	http://www.mantisbt.org/     (GPL)
★★★★☆ BugGenie	http://www.thebuggenie.com/     (MPL)
★★★★☆ GNATS	http://www.gnu.org/software/gnats/     (GPL)













11.2. Aplicaciones de escritorio

11.2.1. Ofimática










11.2.1.1. Procesadores de texto

Software	Enlace y características
★★★★★ OpenOffice.org Writer	http://openoffice.org     (LGPL)
★★★★☆ Abiword	http://www.abisource.com/     (GPL)
★★★★☆ KWord	http://www.koffice.org/kword/     (GPL)
★★★★☆ Lyx	http://www.lyx.org     (GPL)









11.2.1.2. Hojas de cálculo

Software	Enlace y características
★★★★★ OpenOffice.org Calc	http://openoffice.org     (LGPL)
★★★★☆ Gnumeric	http://www.gnome.org/projects/gnumeric/     (GPL)
★★★★☆ KSpread	http://www.koffice.org/kspread/     (GPL)







11.2.1.3. Bases de datos

Software	Enlace y características
★★★★★ OpenOffice.org Base	http://openoffice.org     (LGPL)
★★★★☆ Glom	http://www.gnom.org/   (GPL)
★★★★☆ Kexi	http://www.koffice.org/kexi/    (GPL)

11.2.1.4. Presentaciones

Software	Enlace y características
★★★★★ OpenOffice.org Impress	http://openoffice.org     (LGPL)
★★★★☆ KPresenter	http://www.koffice.org/kpresenter/     (GPL)

11.2.1.5. Organizadores de información personal (PIM)













Software	Enlace y características
★★★★★ Evolution	http://www.gnome.org/projects/evolution/    (GPL)
★★★★★ Kontact	http://kontact.org    (GPL)
★★★★☆ Mozilla Sunbird	www.mozilla.org/projects/calendar/sunbird/     (MPL)

Software	Enlace y características
★★★★☆ Chandler Project	http://chandlerproject.org/     (GPL)
★★★★☆ Pimlico Suite	http://www.pimlico-project.org/    (GPL)








11.2.1.6. Visores y generadores de documentos electrónicos (PDF, PS, etc.)

Software	Enlace y características
★★★★★ PDFCreator	http://www.pdfforge.org/products/pdfcreator     (GPL)
★★★★★ Evince	http://www.gnome.org/projects/evince/    (GPL)
★★★★★ KPDF	http://kpdf.kde.org/    (GPL)
★★★★☆ SumatraPDF	http://blog.kowalczyk.info/software/sumatrapdf/     (GPL)
★★★★☆ xCHM	http://xchm.sourceforge.net/     (GPL)
★★★★☆ gsView	http://pages.cs.wisc.edu/~ghost/gsview/     (GPL)
★★★★☆ PDFEdit	http://pdfedit.petricek.net/    (GPL)
★★★★☆ xPDF	http://www.foolabs.com/xpdf/    (GPL)








11.2.1.7. Sincronización de dispositivos

Software	Enlace y características
★★★★★ Opensync	http://www.opensync.org/    (GPL)
★★★★☆ Multisync (ver OpenSync)	http://multisync.sourceforge.net/    (GPL)
★★★★☆ GNOME Pilot	http://live.gnome.org/GnomePilot    (GPL)
★★★★☆ Wammu	http://wammu.eu/    (GPL)

11.2.1.8. Maquetación

Software	Enlace y características
★★★★★ Scribus	http://www.scribus.net/     (GPL)
★★★★☆ Passepartout	http://www.stacken.kth.se/project/pptout/    (BSD)








11.2.1.9. Diagramas







Software	Enlace y características
★★★★★ Kivio	http://www.koffice.org/kivio/    (GPL)
★★★★☆ Dia	http://live.gnome.org/Dia     (GPL)

11.2.1.10. Gestión de proyectos






Software	Enlace y características
★★★★★ Openproj	http://openproj.org     (CPAL)
★★★★☆ Gantt Project	http://ganttproject.biz     (GPL)
★★★★☆ KPlato	http://www.koffice.org/kplato/    (GPL)
★★★★☆ GanttPV	http://www.pureviolet.net/ganttpv/     (GPL)
★★★★☆ Planner	http://live.gnome.org/Planner    (GPL)

11.2.1.11. Compresores y descompresores

Software	Enlace y características
★★★★★ 7 Zip	http://www.7-zip.org/     (LGPL)
★★★★★ PeaZip	http://peazip.sourceforge.net/    (GPL)

Software	Enlace y características
★★★★☆ Ark	http://kde.org =    (GPL)
★★★★☆ File Roller	http://fileroller.sourceforge.net/ =    (GPL)










11.2.1.12. Grabación de CD y DVD

Software	Enlace y características
★★★★★ K3b	http://k3b.plainblack.com/ =  = = (GPL)
★★★★★ InfraRecorder	http://infrarecorder.sourceforge.net/  = = = (GPL)
★★★★★ Brasero	http://www.gnome.org/projects/brasero/ =  = = (GPL)
★★★★★ SimplyBurns	http://simplyburns.berlios.de = =  = (GPL)
★★★★☆ GnomeBaker	http://sourceforge.net/projects/gnomebaker/ =  = = (GPL)

11.2.2. Internet

11.2.2.1. Acceso a la Web: navegadores






Software	Enlace y características
★★★★★ Mozilla FireFox	www.mozilla.com/firefox/     (MPL)
★★★★★ Konqueror	http://www.konqueror.org/ =  =  (GPL)
★★★★★ K3b	http://kde.org =  = = (GPL)
★★★★☆ Epiphany	http://www.gnome.org/projects/epiphany/ =  = = (GPL)
★★★★☆ K-Meleon	http://kde.org  = = = (GPL)
★★★★☆ Minimo	http://www.mozilla.org/projects/minimo/ = = =  (MPL)

Software	Enlace y características
★★★★☆ Galeon	http://galeon.sourceforge.net/ =  = = (GPL)
★★★★☆ Flock	http://www.flock.com/     (MPL)
★★★★☆ Amaya	http://www.w3.org/Amaya     (GPL)

11.2.2.2. Correo electrónico






Software	Enlace y características
★★★★★ Mozilla ThunderBird	http://www.mozilla.com/thunderbird/     (MPL)
★★★★★ GNOME Evolution	http://www.gnome.org/projects/evolution/ =    (GPL)
★★★★★ KMail	http://kontakt.kde.org/kmail/ =    (GPL)
★★★★☆ MyOE	http://myoedev.blogspot.com/  = = = (GPL)
★★★★☆ Sylpheed	http://sylpheed.sraoss.jp/en/ =  =  (GPL)
★★★★☆ Claws Mail	http://www.claws-mail.org/ =  =  (GPL)

11.2.2.3. Telefonía IP

Software	Enlace y características
★★★★★ OpenWengo	http://www.openwengo.com/     (GPL)
★★★★★ Ekiga	http://ekiga.org/ =  = = (GPL)

11.2.2.4. Lectores de noticias

Software	Enlace y características
★★★★☆ RSSOwl	http://www.rssowl.org/     (GPL)













Software	Enlace y características
★★★★☆ Agregator	http://akregator.kde.org/ =    (GPL)
★★★★☆ Liferea	http://liferea.sourceforge.net/ =  =  (GPL)










11.2.2.5. Mensajería instantánea

Software	Enlace y características
★★★★★ Pidgin (antes GAIM)	http://www.pidgin.im/     (GPL)
★★★★★ Miranda	http://www.miranda-im.org/  = = = (GPL)
★★★★★ Adium	http://www.adiumx.com/ = =  = (GPL)
★★★★★ Kopete	http://kopete.kde.org/ =    (GPL)
★★★★☆ aMSN	http://www.amsn-project.net/     (GPL)
★★★★☆ Psi	http://psi-im.org/     (GPL)
★★★★☆ KPopup	http://www.henschelsoft.de/kpopup_en.html =  = = (GPL)



11.2.3. Reproducción Multimedia

11.2.3.1. Vídeo

Software	Enlace y características
★★★★★ MPlayer	http://www.mplayerhq.hu/     (GPL)
★★★★★ VLC Media Player	http://www.videolan.org/vlc/ =    (GPL)
★★★★☆ Xine	http://www.xinehq.de/     (GPL)
★★★★☆ Media Player Classic	http://sourceforge.net/projects/guliverkli/     (GPL)









Software	Enlace y características
★★★★☆ Noatun	http://noatun.kde.org/ =    (GPL)
★★★★☆ Totem	http://www.gnome.org/projects/totem/ =    (GPL)
★★★★☆ Ogle	http://www.dtek.chalmers.se/groups/dvd/ =    (GPL)




11.2.3.2. Audio

Software	Enlace y características
★★★★★ BMP	beep-media-player.org/ =  =  (GPL)
★★★★★ Listen	http://www.listen-project.org/ =  =  (GPL)
★★★★★ Rhythmbox	http://www.gnome.org/projects/rhythmbox/ =    (GPL)
★★★★★ Amarok	http://amarok.kde.org/ =    (GPL)
★★★★☆ Audacious	http://audacious-media-player.org/ =  = = (GPL)
★★★★☆ Banshee	http://banshee-project.org/ =  =  (GPL)
★★★★☆ XMMS	http://www.xmms.org =    (GPL)











11.2.4. Edición multimedia

11.2.4.1. Vídeo

Software	Enlace y características
★★★★★ Jahshaka	http://www.jahshaka.org     (GPL)
★★★★★ Kdenlive	http://kdenlive.org/ =  = = (GPL)
★★★★★ LiVES	http://lives.sourceforge.net/ =    (GPL)




Software	Enlace y características
★★★★☆ Cinelerra	http://www.xmms.org =  = = (GPL)
★★★★☆ Kino	http://www.kinodv.org/ =  =  (GPL)

11.2.4.2. Audio

Software	Enlace y características
★★★★★ Ardour	http://ardour.org/ =   = (GPL)
★★★★☆ Audacity	http://audacity.sourceforge.net/     (GPL)
★★★★☆ CSound	http://csound.sourceforge.net/     (LGPL)






11.2.5. Gestión

11.2.5.1. Contabilidad y facturación

Software	Enlace y características
★★★★☆ Keme	http://keme.sourceforge.net/ =  = = (GPL)
★★★★☆ Contalinux	http://www.gnulinux.net/softliberado/contalinux.php =  = = (GPL)
★★★★☆ BulmaGES	http://developer.berlios.de/projects/bulmages =  = = (GPL)











11.2.6. Diseño e imagen

11.2.6.1. Fotografía


Software	Enlace y características
★★★★★ Gimp	http://www.gimp.org/     (GPL)
★★★★★ Paint.NET	http://www.getpaint.net/  = = = (MIT)

Software	Enlace y características
★★★★☆ Krita	www.koffice.org/krita/ =   = (GPL)

11.2.6.2. CAD/CAM

Software	Enlace y características
★★★★★ FreeCAD	http://juergen-riegel.net/FreeCAD/Docu/   = = (GPL-LGPL)
★★★★☆ QCAD	http://www.qcad.org     (GPL)
★★★★☆ BRL-CAD	http://sourceforge.net/projects/brlcad/     (GPL-LGPL-BSD)

11.2.6.3. Vectorial (2D)

Software	Enlace y características
★★★★★ Inkscape	http://www.inkscape.org/   = = (GPL)
★★★★☆ Karbon14	http://www.koffice.org/karbon/ =    (GPL)
★★★★☆ SodiPodi	http://www.sodipodi.com/index.php3?section=home =    (GPL)





































11.2.6.4. 3D

Software	Enlace y características
★★★★★ Blender	http://www.blender.org/     (GPL)

11.2.7. Desarrollo de aplicaciones

11.2.7.1. Entornos Integrados de Desarrollo (IDE)

Software	Enlace y características
★★★★★ Eclipse	http://www.eclipse.org     (Eclipse Public License)

Software	Enlace y características
★★★★★ SharpDevelop	http://www.icsharpcode.net/OpenSource/SD/     (GPL)
★★★★★ KDevelop	http://www.kdevelop.org/ =  = = (GPL)
★★★★★ Anjuta DevStudio	http://anjuta.sourceforge.net/ =  = = (GPL)
★★★★★ MonoDevelop	http://www.monodevelop.com/ =    (MIT-GPL)
★★★★☆ Netbeans	http://www.blender.org/     (CDDL)
★★★★☆ Dev-C++	http://www.bloodshed.net/devcpp.html     (GPL)
★★★★☆ Bluefish	http://bluefish.openoffice.nl/ =    (GPL)
★★★★☆ Kompozer	http://kompozer.net/     (MPL-GPL-LGPL)
★★★★☆ Lazarus	http://www.lazarus.freepascal.org/     (GPL)
★★★★☆ RADRails	http://www.radrails.org/ =    (APL/GPL)
★★★★☆ Gambas	http://gambas.sourceforge.net/ =  =  (GPL)
★★★★☆ wxDev-C++	http://wxdsgn.sourceforge.net/  = = = (GPL)
★★★★☆ JVantage Community Edition	http://www.jvantage.com/     (Apache v2)
★★★★☆ DataWeb Project	http://dwprj.sourceforge.net/     (GPL)

11.3. ¿Dónde encontrar más aplicaciones?

Diariamente se crean nuevas aplicaciones y herramientas basadas en software libre, por lo que tenemos que estar atentos y tener recursos para realizar búsquedas que nos permitan encontrar aplicaciones que, probablemente, solo tienen unos meses de vida.

Para poder realizar esta búsqueda, vamos a utilizar las *Forjas de software*, los *directorios de software* y los *repositorios de software*.

11.3.1. Forjas de software

Las *forjas de software* son lugares en línea preparados específicamente para desarrolladores, de manera que puedan crear nuevos proyectos de software libre y disponer de toda la infraestructura necesaria para llevar a término dichos proyectos.

Generalmente, las forjas de software disponen de un repositorio de código con control de versiones (indispensable para desarrollar software en condiciones), sistemas de control de fallos del software, foros y listas de correo para usuarios y desarrolladores, espacio web para la página del proyecto y algunos servicios más complejos, como las granjas de compilación (necesarias para los proyectos grandes, que necesitan mucho tiempo de compilación).

Por otra parte, las forjas de software generalmente disponen de un servicio de descargas con múltiples *mirrors* por todo el mundo, de manera que la difusión de los proyectos que contienen se puedan realizar de manera ininterrumpida.

Otro de los servicios interesantes de las forjas de software es que permiten la comunicación entre desarrolladores y usuarios, de manera que puedan resolverse mejor los problemas relacionados con ese software.

Actualmente, las forjas de software también están incluyendo servicios para la venta de soporte asociada a proyectos de software libre, de manera que los desarrolladores puedan cobrar por los servicios de soporte asociados, si se presta la ocasión, de manera muy sencilla tanto para el desarrollador como para el usuario que hace una petición de soporte.

Algunas de las forjas más conocidas son:

- Savannah (<http://savannah.gnu.org/>)
- SourceForge (<http://sourceforge.net/>)
- Codeplex (<http://www.codeplex.com/>)
- Tigris (<http://www.tigris.org/>)
- PgFoundry (<http://pgfoundry.org/>)
- Freshmeat (<http://freshmeat.net/>)
- FSF Directory (<http://directory.fsf.org/>)
- OSDir (<http://osdir.com/>)
- GNOME Files (<http://www.gnomefiles.org/>)
- KDE-Apps (<http://www.kde-apps.org/>)
- Rediris (<https://forja.rediris.es/>)

11.3.2. Directorios de software en línea

Los *directorios de software* no son más que listados ordenados por categorías con referencias al sitio original desde donde podemos descargar el software. Actualmente, también ofrecen sistemas de comentarios por parte de los usuarios y sistemas de votación para las distintas versiones, de manera que podemos comprobar si el software tiene o no aceptación, o preguntar a otros usuarios dudas en cuanto a funcionalidad de dicho software.

- BerliOS (<http://www.berlios.de/>)
- Freshmeat (<http://freshmeat.net/>)
- Open Suorce Windows (<http://www.opensourcewindows.org>)
- OSSWin Project (<http://osswin.sourceforge.net>)
- Alternativas Libres (<http://alts.homelinux.net>)
- CDLibre (<http://www.cdlibre.org>)

11.3.3. Repositorios de software

Generalmente, con las distribuciones de GNU/Linux suelen venir preinstalados unos *repositorios de software*. Estos repositorios no son más que colecciones de software ya empaquetadas y preparadas para la distribución GNU/Linux que utilizemos.

Los repositorios de software son los lugares desde lo que instalamos generalmente software en sistemas operativos abiertos, ya que los programas empaquetados que contienen han sido testeados por los desarrolladores de la distribución. De esta manera, nos ahorramos tiempo en mantenimiento, ya que podemos instalar el software cómodamente (generalmente, con unos pocos clicks, ya que las distribuciones actuales tienen gestores de repositorios de software con una interfaz gráfica muy sencilla e intuitiva).

Además, los repositorios de software, junto con el software de gestión de repositorios instalado en cada versión, nos libra de tener que compilar, buscar, descargar e instalar las dependencias de ese software, de manera que siempre que podamos, es preferible utilizar el software preempaquetado de estos repositorios.

Existen tres grandes tipos de repositorios: repositorios Debian, respositorios RPM (Redhat) y metarepositorios.

Tanto los repositorios Debian como los RPM son repositorios de software preempaquetado, precompilado y preconfigurado, de manera que instalar un nuevo programa solo implica indicar qué tiene que instalarse, y el gestor de paquetes adecuado (APT para los repositorios Debian o YUM para los repositorios RPM) se encarga del resto. Solo en el software específico para administradores tendremos que realizar configuraciones a mano de ese software.

Por otra parte, los metarepositorios tienen un métodos de funcionamiento heredado de BSD. Estos metarepositorios, solo contienen especificaciones de cómo descargar y compilar el software pero no el software preempaquetado. Dentro de este tipo de repositorios podemos contemplar el sistema que utiliza Gentoo (usando su gestor de metapaquetes Emerge). La ventaja de este tipo de repositorios es que podemos modificar la manera en que se compilan los programas, optimizándolos para nuestra arquitectura específica o sólo incluyendo cierta características de los programas para, por ejemplo, ahorrar espacio. Desgraciadamente, no es oro todo lo que reluce, ya que en el caso de tener problemas en la compilación, las soluciones pueden ser más costosas de encontrar. Además, hay que tener en cuenta que la compilación de un software “grande”, como pueda ser el sistema de ventanas gráfico X.org o el entorno de escritorio GNOME puede llevar mucho tiempo, con lo que tenemos que tenerlo en cuenta, ya que los requisitos para compilar dicho software suele ser mayor que el requisito para usarlo.

De todas maneras, el hecho que nos interesa es poder realizar búsquedas en el repositorio. Por ejemplo, en los repositorios Debian podemos utilizar la utilidad *apt-cache* para realizar búsquedas, de manera que podamos encontrar software ya empaquetado con la versión soportada por nuestra distribución.

12. Bibliografía

A continuación citamos otros documentos, en los cuales se basa esta guía, de utilidad que describen mas en profundidad los procesos de migración a Software Libre

- Directrices IDA de migración a software de fuentes abiertas
- Linux Client Migration Cookbook, Version 2. IBM
- Migration Guide. A guide to migrating the basic software components on server and workstation computers. Bundesministerium des Innern. KBSt Publication Series.
- Wikipedia <http://es.wikipedia.org>
- Aspectos legales y de explotación del software libre. Parte I. Malcolm Bain, Manuel Gallego Rodríguez, Manuel Martínez Ribas, Judit Rius Sanjuán.