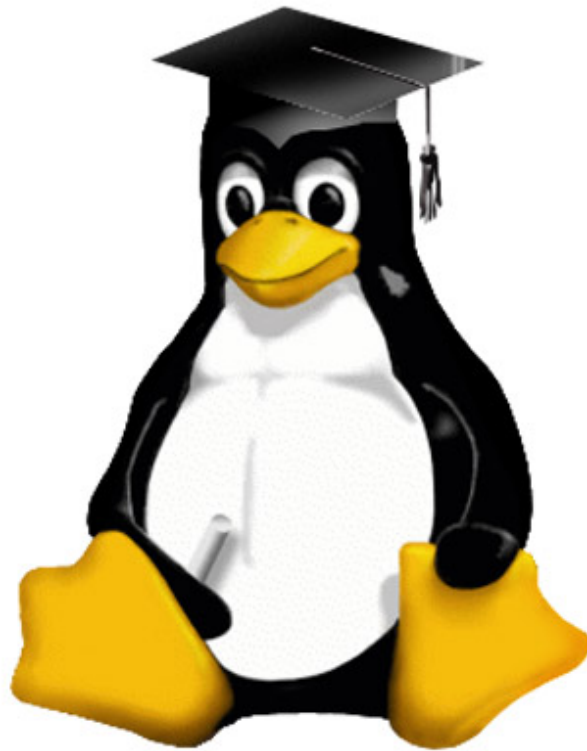


INTRODUCCIÓN

PRÁCTICA A GUADALINEX



**Apuntes complementarios de una
Acción de Formación en Centro
dependiente del CEP de Marbella-Coín**

Juan Rafael Fernández García
juanrafael dot fernandez at hispalinux dot es

Date : 2004/02/0623 : 56 : 56
Este es aún un documento en construcción

Índice general

1	¿Por qué software libre?	8
1.1	Nombres	8
1.1.1	El proyecto GNU	8
1.1.2	Y Linus creó Linux	10
1.1.3	Woody, Sarge y Sid se apellidan Debian	10
1.2	Software libre	11
1.3	Problemas	12
2	Bibliografía complementaria	14
2.1	En la misma distribución	14
2.2	Ayuda externa	14
I	Escritorio y exploradores de ficheros	15
3	Cuestiones previas	16
3.1	Arrancar una Guadalinex desde disquete	16
3.2	Inicio y fin de sesión	18
3.3	Chequeo inicial	19
3.4	Confesión	20
4	Trabajo con ficheros	21
4.1	Usuarios y permisos	22
4.1.1	¿Quién puede leer mis ficheros?	22
II	Ofimática	23
5	Trabajando con textos	24
5.1	OpenOffice Writer	24
5.1.1	¿Qué problema hay con el formato .doc?	24
5.1.2	Trabajo con textos	25
5.2	Más allá	27
5.2.1	LyX y L ^A T _E X	27

<i>ÍNDICE GENERAL</i>	3
5.3 Documentos en otros idiomas	29
5.4 Herramientas	33
5.4.1 Documentos colaborativos y control de revisiones	33
5.4.2 Corrección ortográfica	34
5.4.3 Diccionarios de consulta	35
6 Hojas de cálculo y gráficas	41
7 Clientes de Bases de datos	43
7.1 Ficheros .dbf	43
7.2 Ficheros de MS Access	44
7.3 Clientes y servidores	44
8 Presentaciones	47
9 Creación de páginas web	48
9.1 Más allá: <i>tidy</i>	48
9.2 Mas allá: matemáticas en la <i>web</i>	48
III Multimedia	49
10 Fotos y dibujos	50
10.1 Formatos	50
10.1.1 Imágenes de mapa de bits	50
10.1.2 Imágenes vectoriales	51
10.2 Visores de imágenes	51
10.3 Edición de bitmaps	51
10.4 Edición vectorial	52
11 Animaciones y videos	53
11.1 Animaciones	53
11.2 Video	53
12 Sonido	55
12.1 festival	55
IV Comunicaciones	56
13 Navegación Web	57
13.1 Mozilla	57
13.2 Los plugins	57
13.2.1 Plugins no libres (java, flash, acroread)	57

ÍNDICE GENERAL	4
14 Correo	58
14.1 Más allá	58
15 Comunicación en directo	59
15.1 IRC: comunicación en grupo	59
15.2 De uno a uno: Mensajería Instantánea	59
15.3 Redes <i>peer to peer</i>	59
15.4 Videoconferencia	59
V Aplicaciones didácticas	60
16 ¿Software educativo?	62
16.1 Aplicaciones incluidas en Debian	63
16.2 Enciclopedias libres	64
16.3 Enumeraciones de aplicaciones educativas	64
16.4 Herramientas de autor	64
16.4.1 jcllc	64
16.4.2 squeak	64
16.4.3 WIMS	65
16.4.4 Descartes	65
16.4.5 MALTED	65
VI Recursos materiales	66
17 Utilización de los dispositivos	67
17.1 ¿Cómo se <i>ripea</i> un CD de música?	67
17.2 ¿Cómo se utiliza la cámara de video digital?	67
VII Nociones fundamentales de UNIX	69
18 Redes	70
18.1 Samba. Convivencia con Windows	70
18.1.1 A pelo	70
18.2 Creación de un fichero de texto en remoto	70
18.2.1 Actividades previas	71
18.2.2 Creación local y subida con ftp	71
18.2.3 Uso de un cliente <i>ssh</i>	71

19	Nociones elementales de administración del sistema	72
19.1	¿Por qué este capítulo?	72
19.2	Actualizaciones e instalación de nuevo software	72
19.2.1	Actualizaciones	72
19.2.2	Listado de paquetes disponibles	73
19.2.3	Instalación de un programa	74
19.3	Control de usuarios y contraseñas	74
19.4	Demonios y configuración del sistema	74
19.5	Montando y desmontando	75
19.5.1	¿Cómo se utiliza un diskette?	75
19.5.2	¿Cómo se lee un cdrom?	75
19.6	Dispositivos usb	76
20	Instalación de un sistema GNU Linux	77
20.1	Particiones	77
20.2	Instalación	77
20.3	Localización	77
20.4	Creación de usuarios	77
20.5	Configuraciones generales: dispositivos...	77
20.5.1	Configurando la red	77

Índice de figuras

1.1	Categorías de software	8
4.1	Protocolo <i>file</i>	21
4.2	Permisos de un directorio	22
5.1	Ejemplo de uso de LyX	27
5.2	Ejemplo musical con L ^A T _E X	30
5.3	Griego en yudit	32
5.4	Árabe en yudit	32
5.5	Ejemplo de un wiki	34
6.1	Ejemplo de hoja de cálculo	42
7.1	.dbf en gnumeric	44
7.2	Texto delimitado en gnumeric	45
7.3	Un fichero de MS Access	46
7.4	Interfaz de consulta a postgresql	46
11.1	Captura de video MPEG1	54
11.2	Captura de DVD	54
19.1	Aptitude	73

Prefacio

El software libre se impone: ya ha llegado a Educación. El 12 de diciembre de 2002 el Parlamento de Andalucía aprueba una Proposición no de ley relativa al software libre en la administración; el 21 de marzo de 2003 se publica en BOJA el *Decreto 72/2003 de Medidas de Impulso a la Sociedad del Conocimiento en Andalucía*, seguido por varios desarrollos, como el de los Centros TIC y DIG (órdenes publicadas el 4 de abril de 2003).

Las razones económicas, políticas y técnicas, pero sobre todo filosóficas y éticas han hecho evidente a la administración andaluza la necesidad de adoptar soluciones relacionadas con el software de fuente abierta; es preciso ahora formar a los docentes para que se familiaricen con la nueva situación y puedan utilizar los recursos libres. Ha quedado demostrado que una solución triunfa si logra que los usuarios se familiaricen con ella, si es percibida como la forma natural de trabajar. Esta naturalidad es la que pretendemos conseguir: mostrando la calidad de los productos realizados con software libre (documentos, gráficos, presentaciones...) y su superioridad técnica como sistema operativo multiusuario y seguro.

Este material ha sido producido para ser utilizado en Acciones de Formación en Centro y Cursos de introducción a GuadaLinux en Institutos y Colegios de la región autónoma de Andalucía (España), dentro del plan de perfeccionamiento del profesorado desarrollado por el Centro de profesores de Marbella-Coín. Es un curso presencial, y por lo tanto más importante que los materiales documentales elaborados (documentos de introducción a GNU Linux hay muchos y muy buenos, entre los que destacaré los de TLDP-ES) será la práctica: los problemas que se presentan al utilizar por primera vez software libre, la puesta en común de experiencias colaborativas.

Documento elaborado enteramente con software libre. Podrá encontrar las fuentes \LaTeX de este documento en <http://cvs.hispalinux.es/cgi-bin/cvsweb/doc-curso-guadalinux-jr/>

y descargar la versión más actualizada de <http://es.tldp.org/Tutoriales/doc-curso-guadalinux-jr/Apuntes.pdf>.

Copyright © 2003-2004 Juan Rafael Fernández García <*juanrafael dot fernandez at hispalinux dot es*>.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation.

Capítulo 1

¿Por qué software libre?

1.1 El nombre y un poco de historia

1.1.1 El proyecto GNU

Hung Chao-Kuei ha creado un diagrama que explica las diferentes categorías de software¹.

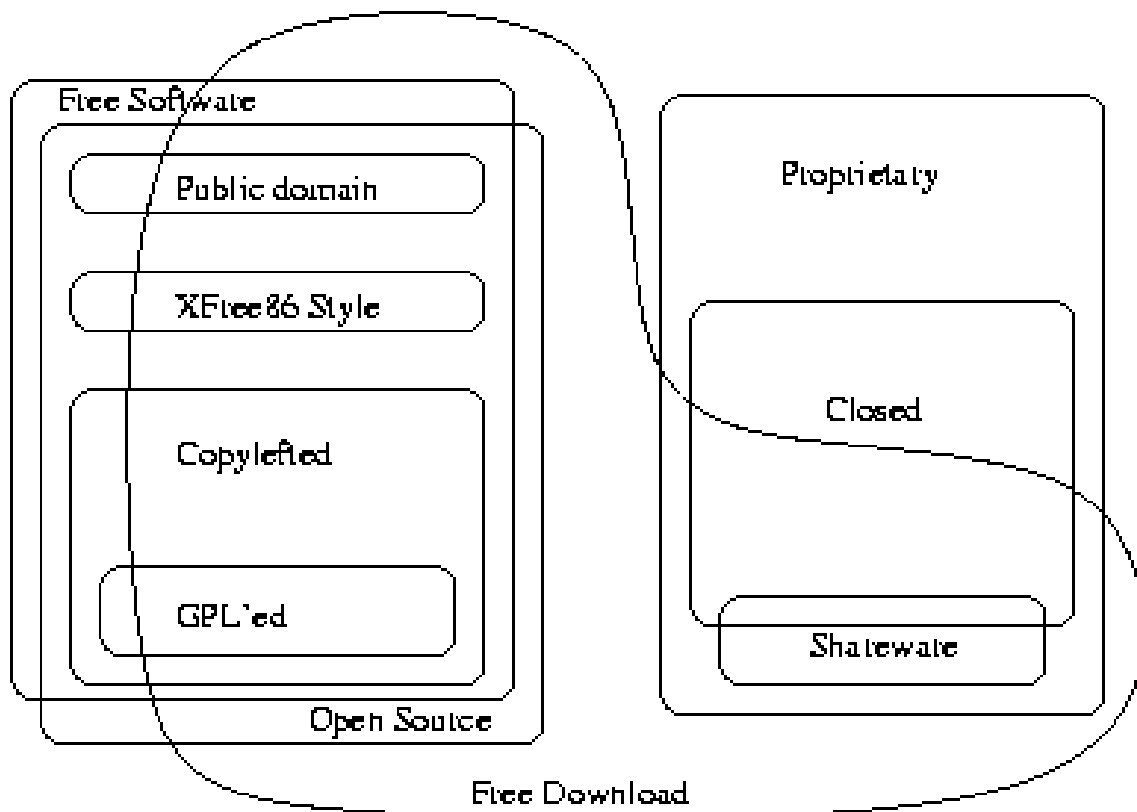


Figura 1.1: Categorías de software

Vemos que los términos extremos, *Software Libre* y *Software privativo*, están bien delimitados pero que hay una maraña de variantes intermedias. Entre ellas está el concepto del *Software Abierto* o *de Código Abierto*.

¹Está en <http://www.gnu.org/philosophy/category.fig>.

El concepto de software libre

A principios de los '80 Richard Stallman² decidió negarse a usar software privativo en su trabajo en el Laboratorio de Inteligencia Artificial del MIT y emprendió el desarrollo de un sistema completo de software libre³ llamado «GNU».⁴

El término «free software» [N. del T.: en inglés free = libre o gratis] se malinterpreta a veces — no tiene nada que ver con el precio. El tema es la libertad. Aquí, por lo tanto, está la definición de software libre: un programa es software libre, para usted, un usuario en particular, si:

- Usted tiene libertad para ejecutar el programa, con cualquier propósito.
- Usted tiene la libertad para modificar el programa para adaptarlo a sus necesidades (para que esta libertad sea efectiva en la práctica, usted debe tener acceso al código fuente, porque modificar un programa sin disponer del código fuente es extraordinariamente dificultoso).
- Usted tiene la libertad para redistribuir copias, tanto gratis como por un canon.
- Usted tiene la libertad para distribuir versiones modificadas del programa, de tal manera que la comunidad pueda beneficiarse con sus mejoras.

Como «free» [libre] se refiere a libertad y no a precio, no existe contradicción entre la venta de copias y el software libre. De hecho, la libertad para vender copias es crucial: las colecciones de software libre que se venden en CD-ROM son importantes para la comunidad, y la venta de las mismas es una manera importante de obtener fondos para el desarrollo de software libre. Por lo tanto, si la gente no puede incluir un programa en dichas colecciones, dicho programa no es software libre.

En <http://www.gnu.org/philosophy/free-sw.es.html> precisa

«Software Libre» se refiere a la libertad de los usuarios de ejecutar, copiar, distribuir, estudiar, cambiar y mejorar el software. Más precisamente, se refiere a las cuatro libertades de los usuarios de software:

- La libertad de correr el programa, con cualquier propósito (libertad 0).
- La libertad de estudiar cómo funciona el programa, y adaptarlo a sus necesidades (libertad 1). El acceso al código fuente es una precondition para esto.
- La libertad de distribuir copias de manera que se puede ayudar al vecino (libertad 2).
- La libertad de mejorar el programa, y liberar las mejoras al público de tal manera que toda la comunidad se beneficia (libertad 3). El acceso al código fuente es una precondition para esto.

Un programa es software libre si los usuarios tienen todas estas libertades.

¿Cuáles son las razones de Stallman? Él habla de razones éticas: de compartir conocimientos en la comunidad de software.

Las consecuencias son el movimiento por el Software Libre.

²La historia la cuenta él mismo en <http://www.gnu.org/gnu/thegnuproject.es.html>.

³Las ideas de Stallman están recogidas en <http://www.gnu.org/philosophy/philosophy.es.html>.

⁴Ver <http://www.gnu.org/gnu/gnu-history.es.html>. El significado del acrónimo es una típica broma *hacker*: “GNU is not Unix”.

Licencias y Copyleft

Una precisión: es común confundir los conceptos de licencia y copyright. El sentido del *copyleft*⁵ es *proteger* el código para que siga siendo libre. Hay licencias sin *derechos reservados*, que permiten a cualquiera hacer lo que quiera con ese código.

En dos palabras: el copyright (o copyleft) señala quién es el dueño o propietario intelectual del código, documento o traducción; la licencia señala los permisos que el propietario da respecto a su obra.

Unas cuantas pistas sobre el concepto de propiedad intelectual:

- <http://www.gnu.org/licenses/licenses.es.html>
- <http://www.gnu.org/licenses/license-list.html#DocumentationLicenses>
- <http://www.debian.org/doc/manuals/ddp-policy/ch-common.en.html#s2.2>
- <http://advogato.org/article/682.html>

1.1.2 1991: Y Linus creó Linux

Recorto y pego descaradamente de esta breve historia de Linux⁶:

«Linux es una versión libre de Unix desarrollada originalmente por Linus Torvalds en la Universidad de Helsinki en Finlandia, con la ayuda, a través de Internet, de numerosos programadores y expertos de Unix. Cualquiera que tenga el instinto y los conocimientos suficientes puede desarrollar y modificar el sistema. El núcleo de no utiliza código de AT&T o de cualquier otra fuente propietaria, y gran parte del software disponible para Linux ha sido desarrollado por el proyecto GNU de la Free Software Foundation. No obstante, también programadores de todo el mundo contribuyen a que aumente cada vez más el software disponible para Linux.

»Linux, el núcleo o kernel de GNU/Linux, se desarrolló originalmente como un proyecto que Linus Torvalds emprendió en su tiempo libre. Se inspiró en Minix, un sistema Unix básico desarrollado por Andy Tanenbaum. Las primeras discusiones acerca del núcleo Linux tuvieron lugar en el grupo de noticias de Usenet *comp.os.minix*. Estas discusiones se centraban sobre todo en el desarrollo de un sistema pequeño y académico de Unix para usuarios de Minix que querían algo más.»

El 5 de octubre de 1991 Linus anunció en *comp.os.minix* la primera versión pública del núcleo Linux, la versión 0.02, con las siguientes famosas palabras:

¿Suspiras por los fabulosos días de Minix-1.1, cuando los hombres eran hombres y escribían sus propios controladores de dispositivo? ¿Te encuentras sin un buen proyecto y te mueres por hincar los dientes a un sistema operativo que puedas intentar modificar para tus necesidades? ¿Encuentras frustrante que todo en Minix funcione? ¿Se acabaron las amanecidas para conseguir que funcione ese programa cañero? Entonces este mensaje puede que sea para ti.

Tal y como mencioné hace un mes, estoy trabajando en una versión libre de una especie de Minix para ordenadores AT-386. Por fin ha alcanzado el estado en el que incluso se puede usar (aunque a lo mejor no se puede, depende de para qué lo quieras), y deseo dejar el código fuente libre para que alcance mayor distribución. Sólo es la versión 0.02... , pero ya he ejecutado con éxito bash, gcc, gnu-make, gnu-sed, compress, etcétera, bajo este sistema.

1.1.3 Woody, Sarge y Sid se apellidan Debian

Páginas en español del proyecto Debian⁷

⁵Otra broma *hacker*, por oposición a *copyright*.

⁶<http://es.tldp.org/Manuales-LuCAS/LIPP2/lipp-2.0-beta-html/node16.html>.

⁷<http://www.es.debian.org/>.

1.2 ¿Por qué software libre?

Razones económicas

- Coste de licencias, sistemas operativos y utilidades
- Coste de mantenimiento y formación
Informes de la campaña pro software libre en la administración, de HispaLinux⁸
- El modelo *bazar* de desarrollo abierto⁹
- Oportunidades para la pequeña empresa andaluza
Proposición no de ley sobre software libre en Andalucía¹⁰

Razones éticas

El software es conocimiento: Filosofía del proyecto GNU¹¹

Razones técnicas

- código abierto
Muy interesante en un sentido polémico el artículo de Michael Jennings, "Windows XP muestra la dirección que Microsoft está tomando"¹².
Para muestra un botón.

Windows XP se conecta a las computadoras de Microsoft de al menos 18 maneras. Microsoft Windows XP es dependiente para su operación de otras computadoras que el usuario no posee y no puede controlar.

Aquí está una lista (probablemente incompleta) de maneras en que Windows XP conecta la computadora de cada usuario a los servidores de Microsoft.

1. Servicio de Gateway para la Capa de Aplicaciones - (Requiere derechos de servidor)
2. Servicio de Fax
3. Verificación de las Firmas de Archivos
4. Procesamiento Genérico de Hosts para Servicios Win32 (Requiere derechos de servidor)
5. Reporte de Errores de Aplicaciones Microsoft
6. Analizador de Seguridad Básica de Microsoft
7. Test de Voz de DirectPlay Microsoft
8. Centro de Ayuda y Soporte de Microsoft
9. Servidor de Hosting para el Centro de Ayuda (solicita derechos de servidor)
10. Consola de Administración de Microsoft
11. Microsoft Media Player (le dice a Microsoft qué música y videos le gustan a usted.
12. Microsoft Network Availability Test
13. Microsoft Volume Shadow Copy Service
14. Utilidad de configuración de Windows Media (Setup_wm.exe, a veces corre cuando usted usa Media Player)

⁸<http://www.hispalinux.es/modules.php?op=modload&name=phpWiki&file=index&pagename=SLAdministracionInformes>.

⁹<http://es.tldp.org/Otros/catedral-bazar/>.

¹⁰<https://listas.hispalinux.es/pipermail/sl-administracion/2003-January/001898.html>.

¹¹<http://www.gnu.org/philosophy/philosophy.es.html>.

¹²<http://www.hevanet.com/peace/microsoft-es.htm>.

15. Programa de consola de MS DTC
16. Run DLL como aplicación (no hay indicación de qué dll o qué función en la DLL)
17. Aplicación de Servicios y Controladores
18. Servicio de Tiempo, configura la hora en su computadora desde un servidor de Microsoft. (Esto puede cambiarse para obtener la hora de otro servidor)

Hay otras maneras por las cuales Microsoft obtiene control:

1. Microsoft Office guarda un número en cada archivo que usted crea con macros de Visual Basic, que identifica a su computadora. Microsoft Office 97 guarda un número de identificación aunque no hayan macros.
2. El software de los mouses Microsoft tiene funcionalidad reducida hasta que se le permite conectarse con las computadoras de Microsoft.

Esta no es necesariamente una lista completa. Puede haber otras conexiones.

Otra cita de Michael Jennings no tiene desperdicio:

Usted no puede saber a qué contrato estará ligado en el futuro. Microsoft ha cambiado los términos del contrato al cual los usuarios están ligados incluyendo el nuevo contrato en algunas actualizaciones de seguridad y parches de bugs.

Parches de seguridad recientes requieren que el usuario acepte un contrato que le da a Microsoft privilegios de administrador¹³ sobre la computadora del usuario (los privilegios de administrador le dan control completo sobre la computadora y todos los datos incluidos en ella). El contrato dice que si el usuario quiere parchear su sistema contra un bug que permitiría un ataque por Internet, él o ella debe darle a Microsoft control legal sobre la computadora.

Este artículo explica el problema con mayor profundidad: Microsoft's Digital Rights Management – A Little Deeper¹⁴. Ayuda el pensar como abogado al momento de separar la frase crucial. La frase, "Estas actualizaciones relacionadas con la seguridad pueden eliminar su poder para copiar y/o ejecutar Contenidos Seguros y usar otro software en su computadora" legalmente incluye este significado: "Estas actualizaciones pueden eliminar su poder para usar otro software en su computadora." Nótese que el término "actualizaciones relacionadas con la seguridad" no tiene sentido, ya que algunas de las actualizaciones no tienen relación con la seguridad del usuario. Así, la frase significa efectivamente que Microsoft puede controlar la computadora del usuario sin aviso y cada vez que lo desee.

Dado que Microsoft puede cambiar el contrato en cualquier momento y sin control por parte del usuario, Microsoft puede ligar a los usuarios a contratos que invente en el futuro. Este es un nuevo desarrollo en la legislación contractual. Un usuario está ligado a un nuevo contrato si desea actualizaciones de seguridad o arreglos de bugs. Pero esto no le da al usuario control, ya que apenas las fallas de seguridad son ampliamente conocidas, cada computadora debe instalar los parches o quedar vulnerable. Los usuarios gastan considerable tiempo y dinero en sus computadoras, y no pueden evitar aceptar el nuevo contrato sin entregar toda su inversión y arriesgar su negocio y actividades personales.

- Es un Unix en rápido desarrollo

1.3 Los problemas del software libre

- no funciona todo el hardware
Linux Hardware Compatibility HOWTO¹⁵
Linux Hardware Database¹⁶

¹³<http://www.theregus.com/content/55/25435.html>.

¹⁴<http://bsdvault.net/article.php?sid=527&mode=\&order=0>.

¹⁵<http://192.168.1.4/doc/HOWTO/en-html/Hardware-HOWTO/index.html>.

¹⁶<http://lhd.zdnet.com/>.

- no tiene soporte ni está bien documentado y está todo en inglés
 - Es el código propietario el que no tiene soporte
 - Comunidad entusiasta
- Linux Málaga¹⁷
Lista de usuarios de Debian en español¹⁸
Asociación para la Difusión y el Avance del Software Libre de Andalucía¹⁹
Asociación de Usuarios Españoles de GNU/Linux²⁰
Las páginas de la distribución GuadaLinux²¹
- es más difícil
 - Es más difícil conducir un trailer que llevar un seiscientos
 - La curva de aprendizaje lleva hacia algún sitio
 - Con los nuevos desarrollos esta tesis ha dejado de ser cierta

Si no le convencen los argumentos de este capítulo, puede consultar

1. la siguiente página de Pedro Reina: <http://pedroreina.org/curso/slutiens.html>, que con seguridad será más convincente.
2. la introducción teórica al software libre de <http://curso-sobre.berlios.de/introsobre/>.

¹⁷<http://www.linux-malaga.org/>.

¹⁸<http://lists.debian.org/debian-user-spanish/>.

¹⁹<http://www.adala.org>.

²⁰<http://www.hispalinux.org>.

²¹<http://www.guadalinex.org>.

Capítulo 2

Bibliografía complementaria

2.1 En la misma distribución

Toda distribución GNU Linux contiene gran cantidad de documentación en diferentes formatos: los CÓMO, Preguntas más frecuentes (FAQ)... Basta medir el espacio que ocupa `/usr/share/doc/`¹ para comprobarlo, y es una parte del disco que es útil examinar.

A esta información debe añadirse la de los ficheros de manual que normalmente acompañan a cada aplicación y que se consultan con la orden `man APLICACION` y los ficheros de hipertexto del sistema de documentación del proyecto GNU (ejecute `pinfo` o si no está instalado `info` para explorarlos).

Tanta información debe ser organizada; hay varias tentativas de realizar una ordenación de la información contenida en la distribución, que pueden o no estar instaladas en la versión que usted usa: ejecute en una terminal `dhelp` o alternativamente `dwww`.

Finalmente, y no dudo de que estará instalada en la versión que usted utilice de GuadaLinux, el proyecto Gnome tiene su propia forma de facilitar el acceso a la documentación (Menu ->Ayuda).

2.2 Ayuda externa

- Sobre GNU Linux en general, el mejor fondo de documentación en castellano está en TLDP-ES².
- Como introducción general a GNU Linux podemos empezar por la de Vladimir Támara, Jaime Irving Dávila, Pablo Chamorro e Igor Támara (http://structio.sourceforge.net/guias/AA_Linux_colegio/)
- Listas de distribución. Destacaría mientras no exista la lista de usuarios de GuadaLinux las listas de Linex³ y de Debian⁴.
- Foros, como el de Linex, y más específicos los de GuadaLinux⁵ y GuadaLinux – CGA⁶.
- Hay ya en la red dos cursos libres sobre GuadaLinux

1. el de Grimaldos (descargable de las páginas oficiales de GuadaLinux) y
2. el elaborado por los profesores del I.E.S. Averroes⁷ de Córdoba.

Mi material no quiere ser otra cosa que un complemento a sus estupendas aportaciones.

¹La utilidad que deberemos usar es *du*.

²<http://es.tldp.org>.

³<http://www.linex.org>.

⁴<mailto:debian-user-spanish@lists.debian.org>. Para suscribirse hay que mandar un mensaje a <mailto:debian-user-spanish-request@lists.debian.org> con el asunto 'subscribe' (con la 'b', que a la mayoría de la gente se le pasa).

⁵<http://www.guadalinex.org>

⁶<http://www.juntadeandalucia.es/averroes/guadalinex>

⁷<http://es.tldp.org/Tutoriales/doc-curso-guadalinex-iesaverroes>.

Parte I

Escritorio y exploradores de ficheros

Capítulo 3

Cuestiones previas

Debemos, para que el curso tenga una aprovechamiento óptimo, establecer varias condiciones previas y dejar claros varios puntos. En primer lugar debemos señalar que la versión de Guadalinux existente en el momento de redactarse esta documentación sigue siendo una versión de prueba; en ocasiones vamos a referirnos a utilidades que se encuentran en la distribución *madre*, Debian, porque estamos convencidos de que acabarán entrando en una versión más madura.

Por otro lado el Guadalinux que conocemos constituye la parte «cliente» de un sistema que no se comprende sin el otro lado, sin el «servidor». No tienen sentido la mayor parte de las herramientas (clientes de bases de datos, ftp, mensajería instantánea, ssh, etc.) si no se dispone de los servicios para los que están destinados. En un Centro TIC estos servicios, más o menos limitados, los proporcionan en servidor de contenidos y la red corporativa; en el resto de los Centros, y para la formación al menos, deberemos utilizar una máquina configurada para que proporcione el mayor número posible de servicios de red: http (apache con php), ftp (proftpd), correo (pop e imap), mensajería instantánea (jabber), cvs, sshd, samba, nfs, base de datos (mysql o postgresql), ircd. También es muy deseable que los asistentes al curso dispongan de cuentas en este servidor (que, por razones de seguridad, puede y quizás deba estar aislado de las redes exteriores al Centro).

Para entendernos, el servidor se va a llamar `linux1` y se relacionará con una máquina Windows que se va a llamar `PROFESORES1`.

3.1 Arrancar una Guadalinux desde disquete

Cuando usted lea esta sección es probable que esté delante de una Guadalinux en funcionamiento, sea iniciada desde un CD sea instalada en el ordenador. Pero es posible que usted tenga un ordenador viejo, que quiere rejuvenecer dedicándolo a familiarizarse con el nuevo software; y es posible, como me pasaba a mí hasta hace poco, que el ordenador sea anterior a la posibilidad de arrancar desde el CD, y por tanto piensa que se encuentra en un callejón sin salida. No hay problema. Esta breve chuleta debe orientarle en la solución:

```
Como tengo un ordenador viejo, no tengo la opción de arrancar desde el
CD. Pensaba que
lo único que había que hacer es crear un diskette de arranque para
Guadalinux igual que se podía hacer en la Knoppix-es 3.1 que ya tenía.
Pero resulta que el CD de Guadalinux Edu 1.0rc2 que tengo no trae una
imagen de diskette, así que os contaré cómo me he apañado para poder
examinar el CD y probar la distribución.
```

```
Primero vamos a grabar la imagen contenida en KNOPPIX/boot-es.img en
un diskette.
```

```
¿Cómo se hace eso?
-----
```


1) Si utilizas MSDOS (o una ventana en Windows)

```
cd KNOPPIX
rawrite2 -f boot-es.img -d A
```

(a partir de aquí olvídate del DOS/Güindows)

ó 2) En GNU Linux, con el CD en su lector y un diskette vacío en la disquetera

```
su
mount /cdrom
cd /cdrom/KNOPPIX
dd if=boot-es.img of=/dev/fd0
```

Creía que ya había terminado porque ahora tengo un diskette vfat correcto (se puede montar: `mount -t vfat /dev/fd0 /mnt/disk_a`)

pero

el sistema no ha arrancado: la imagen de Knoppix es específica para Knoppix; tendremos que modificarla.

¿Qué hay en boot-es.img?

Se puede ver (y modificar)

a) Con mtools

Añade a `.mtoolsrc` una línea como esta (tendrás que copiar la imagen):

```
drive o: file="/root/test/boot-es.img"
```

Pruébalo ahora:

```
mdir o:
```

```
Volume in drive O has no label
Volume Serial Number is 3E13-8757
Directory for O:/

ldlinux  sys      7136 2003-01-02  1:27
syslinux cfg      2421 2003-01-04 20:50  syslinux.cfg
logo     16      19515 2003-01-02  1:27
boot     msg      134 2003-01-04 20:25
f2       1683 2003-01-05 12:15
miniroot gz      477104 2003-01-05 17:25
vmlinuz  921602 2003-01-02  1:27
spanish  kbd      256 2002-11-04 15:34  spanish.kbd
8 files  1 429 851 bytes
                36 864 bytes free
```

Vemos que los ficheros difieren de los contenidos en el subdirectorio `isolinux/` del CD de Guadalinux. Pero es que además el sistema de arranque ha variado: ya no usa `syslinux`, sino `isolinux`. Pero todos estos tecnicismos no nos interesan porque...

b) ¡Si ya tienes un diskette!

Monta el diskette, monta el CD de Guadalinux, y pisa los ficheros del mismo nombre, salvo `logo.16` (sencillamente porque no cabe, pero no importa porque no es más que la imagen que sale en la primera pantalla). Ahora tenemos un diskette de arranque por el viejo sistema (`syslinux`), pero que funciona.

Está claro que esta es una solución provisional, y que hay soluciones

mejores (ampliar la capacidad de la imagen de diskette, comprimir logo.16 ...), pero funciona y esta tarde puedo explorar la distribución con mi viejo Pentium-133.

Hay que decir, no obstante, que la evolución de Guadalinux es hacia exigencias de potencias de procesamiento y de memoria que harán difícil un uso no problemático (o al menos automático, sin necesidad de modificaciones y conocimientos avanzados) de hardware más o menos obsoleto. En mi modesta opinión se ha asumido el objetivo equivocado de que el usuario de windows WP note mínimamente el cambio. Como respuesta a este problema ha nacido el proyecto Guadalinux-mini.

3.2 Inicio y fin de sesión

La práctica y poco de sentido común nos lleva a plantearnos la necesidad de aclarar estos conceptos. Lo más urgente es aprender a cerrar correctamente la sesión y la máquina, porque un mal apagado puede dañar el sistema de ficheros y causar en el mejor de los casos complicaciones y pérdidas de tiempo.

Distinguiremos entre cerrar un programa que se ha colgado (en la jerga se llama matar un proceso), cerrar una sesión (el periodo de tiempo en que un usuario ha estado conectado) y apagar la máquina (todos los usuarios quedan desconectados y todos los procesos se detienen). No exponemos los métodos evidentes (pinchar en el icono, elegir opción en el menú).

1. Para cerrar una ventana se ensayará la combinación `Alt + F4` (es configurable, luego puede que funcione o que no).
2. Una solución alternativa es `Alt + F2`¹. Así visualizarás el lanzador de programas y los paneles de menú y podrás utilizar la 'bomba' para cerrar la ventana.
3. Una segunda forma de cerrar un programa consiste en 'matar el proceso'. Este método requiere poder abrir una terminal. Si el programa ocupa toda la pantalla `Alt + TAB` nos permitirá cambiar el foco entre aplicaciones.

```
[Usuario@Máquina]$ ps ax
```

nos devuelve el número de proceso de la aplicación, *Num*.

```
[Usuario@Máquina]$ kill Num
```

y normalmente la aplicación se cerrará. Si no es así,

```
[Usuario@Máquina]$ kill -9 Num
```

es mucho más drástico (no cierra los ficheros que tuviera abiertos...)²

Todo esto se puede hacer de modo gráfico, con aplicaciones como *gps* (Gnome) y *kpm* (KDE).

4. Para cerrar una sesión de X se pulsán simultáneamente `Ctl + Alt + Borrar_Caracter_Anterior` (la tecla con una flecha '<' que hay encima de Intro)
5. Para apagar el ordenador (*halt*) existe la orden siguiente:

```
[Usuario@Máquina]$ shutdown -h now
```

¹Esta solución me la ha recordado Rafael Bracho. Las combinaciones de teclas, que son configurables, están en Aplicaciones ->Preferencias de escritorio ->Combinaciones de teclas.

²Alternativamente existe la orden `killall Nombre_del_proceso`.

Existen otras órdenes equivalentes; la diferencia está en que *halt* y *poweroff* requieren permisos de superusuario.

6. Es posible que se hayan colgado el teclado o el ratón, pero no el sistema operativo. Si la máquina permite conexiones de red basta con conectarse desde otra máquina (con telnet o ssh) y matar (si tiene permiso) las X para que se reinicialicen.
7. Para reiniciar la máquina sin riesgo puede que funcione Ctl + Alt + Supr (depende de cómo esté configurado */etc/inittab*). Lo más probable es que haya que hacerlo desde una consola, por lo que tenemos que aprender a abrirlas. Es muy fácil, basta con la combinación de Ctl + Alt + Fx, donde x va del 1 al 6 (es decir, valen las teclas F1, F2... F6). Para volver a las X, Alt + F7.
8. Otra forma es utilizando la orden

```
[Usuario@Máquina]$ shutdown -r now
```

o, lo que es más corto (pero hay que ser el superusuario),

```
[root@Máquina]$ reboot
```

Realmente, cuando se tecllea Ctl + Alt + Supr lo que estamos haciendo es llamar a *shutdown*.

9. El método más drástico es la siguiente secuencia de combinaciones de teclas

```
Alt + PrintScreen + S
Alt + PrintScreen + U
Alt + PrintScreen + B
```

Para entendernos, lo que hemos hecho es mandar tres órdenes al ordenador: Sincronizar las caches con los discos, Umount (desmontar) los discos y Boot (reiniciar).

3.3 Chequeo inicial

El sistema operativo realiza un chequeo inicial cuando la máquina se ha apagado mal (no siempre, depende del sistema de ficheros) o después de un número configurable de arranques. El proceso es automático, pero si encuentra problemas que no puede resolver le pasa el control al usuario. ¿Qué hacer entonces? He aquí una pequeña chuleta, en la esperanza de que no se produzca ninguna complicación:

```
root@none# fsck -C /dev/hda1
```

suponiendo que el disco raíz esté en la primera partición del primer disco y esté montado como de sólo lectura; el sistema hará preguntas a las que responderemos pulsando Intro. Si todo ha ido bien nos dejará en una consola con una línea de comandos. Es el momento de teclear

```
root@none# reboot
```

para contener la respiración hasta comprobar que todo ha ido bien.

3.4 Confesión

Soy un usuario veterano de GNU Linux. Lo que significa que soy un usuario anticuado. No conozco la mayoría de las interfaces que se han creado para facilitar el trabajo a los nuevos usuarios, por nuevas y porque no me gustan. Creo que se sigue un objetivo peligroso: *que los usuarios noten lo menos posible el cambio desde Windows*. Pero es que en Windows se trabaja mal: se crean vicios (es paradigmática la posibilidad de centrar un párrafo a golpes de la tecla espaciadora), en la búsqueda de la máxima facilidad no se informa de lo que está ocurriendo y cuando algo falla es muy difícil encontrar pistas, y los programas se inflaron y cargaron de opciones nunca utilizadas, necesitan máquinas cada vez más grandes y más rápidas para hacer lo mismo que se hacía. No es necesaria una máquina de la última generación para poder tener un Linux totalmente operativo, ágil, seguro y útil. La filosofía de Unix ha sido siempre la de programas pequeños, que hacen lo suyo bien, que colaboran entre ellos; el laconismo de Unix es legendario: el silencio es la señal del éxito de una orden. Ahora se pretende clonar los programas más usados del otro entorno: tenemos *oowriter*, que tarda milenios en arrancar y no corre en máquinas pequeñas, pero ¿por qué no enseñar a usar \LaTeX ? ¿Se han comparado los resultados?³

Por otro lado las interfaces son útiles porque en un principio ahorran tiempo y esfuerzo, pero hay que saber qué ocurre por debajo entre otras cosas porque suelen ser bastante imperfectas. Forman parte de la clase de programas que creen que son más listos que nosotros (por ejemplo los correctores ortográficos) y a mí personalmente me ofenden. Si debo usar el corrector lo haré cuando yo quiera. Si hay un error la interfaz no me va a ayudar; al contrario se habrá esforzado por ocultarme lo que ocurre realmente.

Mi intención es a enseñar qué pasa por debajo del capó del ferrari que estamos descubriendo, a manejar el volante aprende uno solo. O con otro manual.

³Matizaré: no es que esté en contra de que se presente a los nuevos usuarios la posibilidad de usar por ejemplo OpenOffice; pero estas aplicaciones siempre irán por detrás de las de Windows, porque *juegan en terreno contrario*. Lo que digo es que también deben mostrarse las alternativas, mucho más maduras, completas y ligeras y que los usuarios elijan. El esfuerzo será superior en un principio, porque en esto de la informática no llevamos más de dos días pero ya estamos llenos de inercias, sin embargo la curva de aprovechamiento es mucho más pronunciada.

Capítulo 4

Trabajo con ficheros

Una observación que tal vez sea necesaria. En cierto modo la programación que sigue este documento, su progresión, quizás sea inútil. Hay una serie de cosas, quizás las esenciales, que se aprenden 'de paso', con el uso y que no tienen día para aprenderse ni enseñarse. Como que en Unix se distinguen las mayúsculas de las minúsculas (no es lo mismo 'w' que 'W'). Que el tabulador (la tecla que hay encima de Bloq.Mayús) sirve para terminar las órdenes que se están introduciendo (escriba `naut` y pulse TAB; verá cómo el ordenador termina la orden de la línea de comandos `nautilus`. Que en una terminal la flecha hacia arriba devuelve la orden anterior y `Ctrl + Re`. Pág devuelve la pantalla anterior. Que se puede cambiar de consola virtual con `Ctrl + Alt + Fx`...

Arranquemos *nautilus*¹.

¿Qué se muestra escribiendo la siguiente orden en la línea de órdenes?

```
file:/home/usuario/
```

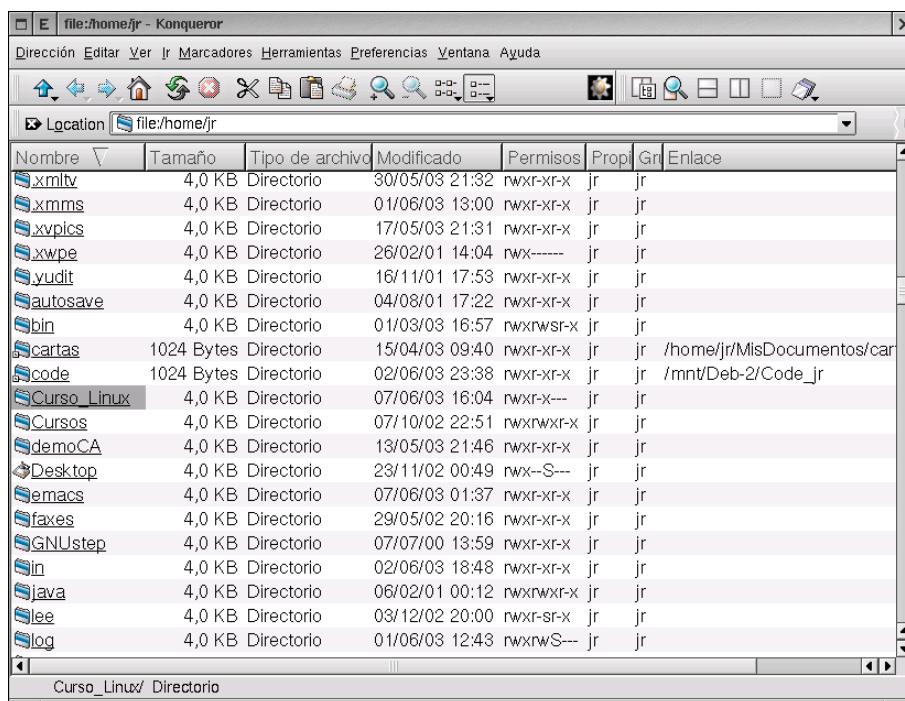


Figura 4.1: Protocolo *file*

¹Nos referiremos a *nautilus* porque es el navegador incluido en la distribución. Valdría perfectamente *konqueror* o incluso *mc*. De hecho prefiero estos programas porque nos informan de los permisos de los ficheros y en mi ignorancia me parece *konqueror* en particular muy superior al *nautilus* que estoy examinando.

El resultado debe ser algo parecido al de la figura 4.1.

¿Dónde están los programas? ¿dónde están nuestros datos y documentos?

4.1 Usuarios y permisos

4.1.1 ¿Quién puede leer mis ficheros?

Navegue por el árbol de ficheros y averigüe qué ficheros puede leer, a qué subdirectorios puede acceder.

¿Qué significan los datos de la figura 4.1.1?

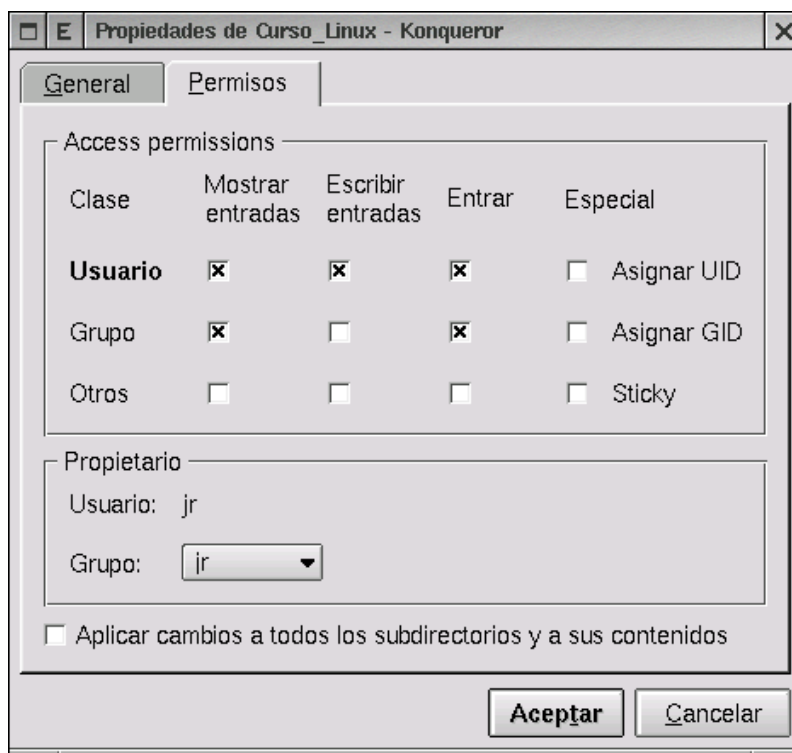


Figura 4.2: Permisos de un directorio

¿Quién es el dueño de este directorio? ¿a qué grupo pertenece? ¿quiénes pueden visitarlo?

Ejercicio. Váyase al directorio raíz e intente borrar el fichero *vmlinuz*. ¿Por qué no puede? Intente leer el contenido de */root*. ¿Se puede? ¿Por qué?

Parte II

Ofimática

Capítulo 5

Trabajando con textos

5.1 OpenOffice Writer


Ejercicio: Escriba una página con un título centrado, de 14 puntos y en negrita. Escriba el encabezamiento de una sección, algo de texto a continuación y finalmente inserte una imagen ajustada a la izquierda y una tabla de tres filas por tres columnas a la derecha de la imagen. Para terminar escriba el encabezamiento de una segunda sección, más un par de líneas de texto normal. Algo así:

Cabecera de sección

Esto es un poco de texto de relleno y no tiene significado ninguno. Es sólo que no se me ocurre qué escribir.

Subsección

Y ahora la primera subsección, que debe incluir la imagen y el cuadro. A la derecha de la figura escribimos la pequeña tabla.



La imagen

1	2	3
4	5	6
7	8	9

 es esta.

Un poco más de texto para terminar, y comprobar que el texto no se monta en la imagen (esto se hace con el paquete *picinpar* de L^AT_EX y es la primera vez que hago experimentos con él).

Segunda subsección

Con su texto.
Esta es la segunda y última línea.

Consejo: Use estilos. Es la única forma de mantener coherencia en textos extensos.

5.1.1 ¿Qué problema hay con el formato .doc?

En lo siguiente sigo un artículo de Jordi Mas¹ y las páginas donde Sun defiende el uso del software libre y de los estándares abiertos, en OpenOffice²

¹<http://enredando.com/cas/enredados/enredados266.html>.

²http://xml.openoffice.org/xml_advocacy.html.

1. Es el complicado resultado de una larga cadena de parches y actualizaciones de un formato original Dice Jordi Mas

Existen muchas variedades de versiones del formato .doc, incluso Microsoft ha tenido problemas de compatibilidad entre diferentes versiones de su propio producto. Cuando la empresa de Bill Gates introdujo una variante del formato .doc con la versión del Word 97 resultaba imposible compartir documentos con versiones de anteriores del mismo producto. De hecho, el formato .doc, como muchas de las tecnologías en Windows, arrastran una complejidad excesiva debido al número de versiones y ampliaciones con el que deben mantener compatibilidad. El .doc es un formato de intercambio de documentos difícil de interpretar por otras aplicaciones debido a su innecesaria complejidad, si lo comparamos, por ejemplo, con cualquier formato basado en el estándar XML como el de OpenOffice o AbiWord.

2. Serios problemas de portabilidad de los documentos: ¿cómo ver documentos en Beos, Mac, Solaris, Linux y resto de Unixes?
3. La presentación de los documentos varía de una máquina a otra y favorece errores de diseño de página/tipográficos en los usuarios; no respeta adecuadamente la distinción lógica entre forma, contenido y metainformación
4. Es propietario y Microsoft puede modificarlo cuando quiera que compremos una nueva versión.

Dice Sun

Open and free licensing guarantees that you are not at the mercy of a single company for improvements and fixes of the format or its supporting software, thus providing very strong protection for all investments and efforts you put into this format.

5. Es cerrado y no sabemos qué información contiene.

Jordi Mas:

(...) aún es más preocupante el hecho de que Microsoft Word almacena un identificador único que permite determinar con gran precisión el PC que ha creado el documento, éste es el sistema que por ejemplo se utilizó para determinar el autor del virus Melissa.

6. Es inseguro porque admite código ejecutable

Sigue Jordi

Otro problema importante del formato .doc es que su lenguaje de macros es una fuente inagotable para la transmisión de virus informáticos debido a la debilidad de su modelo de seguridad.

7. No hay analizadores (*parsers*) ni admite validación. Ante un error no hay manera de saber qué está pasando.
8. Ha sido superado por XML.

5.1.2 Trabajo con textos

Ejercicio: Los profesores abrirán con el editor predefinido (*OpenOffice Writer*, *oowriter*) un fichero cualquiera creado con MS Word y se examinará en común el nivel de éxito en la conversión.

Ejercicio: Cada profesor utilizará el editor para experimentar sus recursos relativos a la materia que imparte (tablas, ecuaciones... posible esbozo de un examen).

Seguidamente se lanzará (desde una terminal) *AbiWord*. Se comparará el tiempo que ha tardado en arrancar, las posibilidades y las limitaciones de la versión actual de la aplicación (rápidamente se descubren los problemas con las tablas, la edición de fórmulas matemáticas...)

Finalmente se examinará *gedit*.

Mi conclusión: Cada aplicación tiene su utilidad, dependiendo de las necesidades del momento.

Notas

- Apunte de Javier Sánchez: *Writer2LaTeX*, en <http://www.hj-gym.dk/hj/writer2latex/>.
- Otro apunte de Javier Sánchez: *dmaths para OpenOffice*, en <http://www.dmaths.com>.

5.2 Más allá

Debemos comprender la diferencia entre «editores de texto»³ y «procesadores de documentos» (o «procesadores de palabras», word processors). Quien mejor explica todo esto —lo siento, en inglés— es Christopher B. Brown, en <http://cbbrowne.com/info/wprant.html>. En concreto dice Brown que «el modo tradicional Unix de ocuparse de los documentos consiste en dividir el proceso de creación del documento en dos partes muy distintas, con conjuntos diferentes de herramientas. Por un lado se usa un editor de texto para crear un fichero de texto plano, y posteriormente se procede a generar el documento con alguna forma de “lenguaje de presentación”» (L^AT_EX, PostScript).

5.2.1 LyX y L^AT_EX

Jugando con LyX

La alternativa más sencilla de utilizar es LyX (véase 5.1).

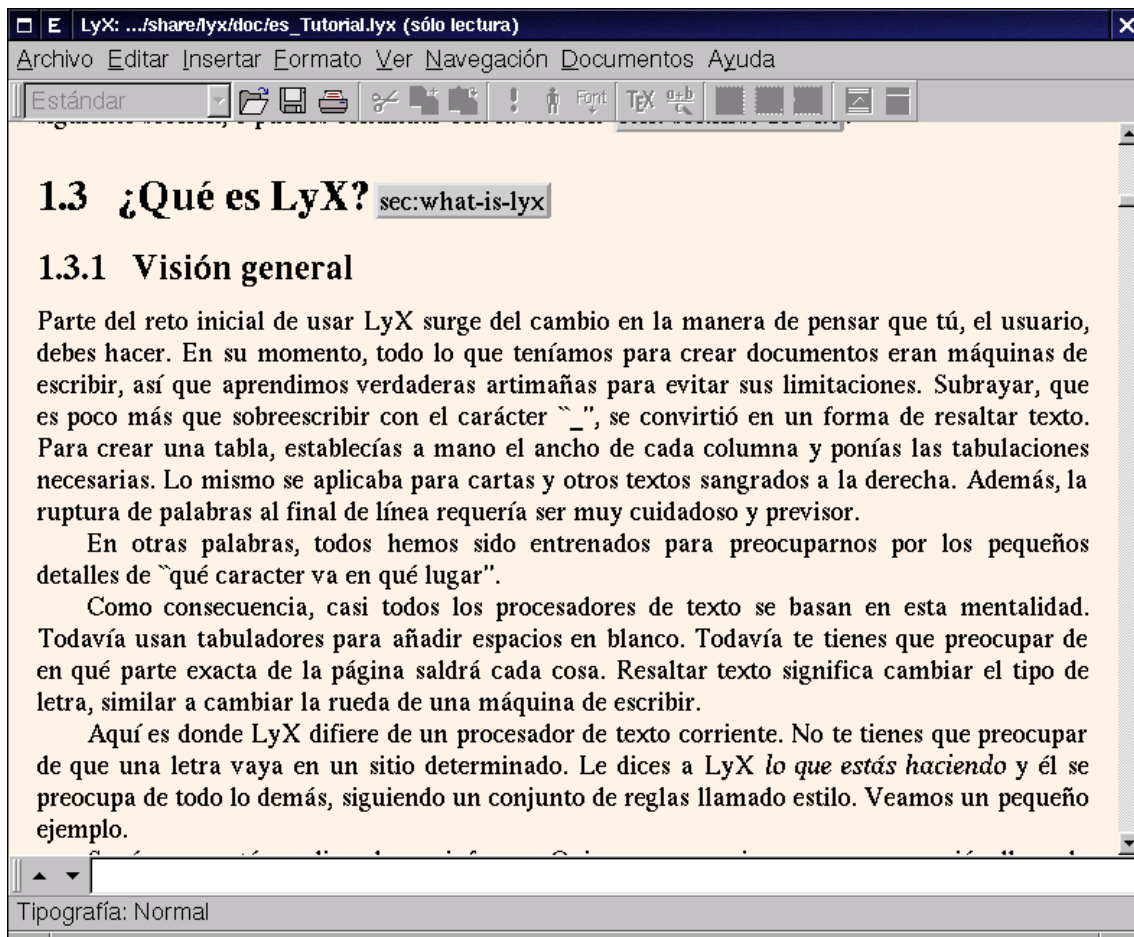


Figura 5.1: Ejemplo de uso de LyX

³vi, emacs, nano o gedit son editores de texto más o menos potentes. En este caso usaremos mcedit por su facilidad.

Cosas que pueden hacerse con \LaTeX

\LaTeX es el hermano mayor de LyX ⁴ y realmente es un programa de composición de documentos⁵ con resultados tipográficamente asombrosos, sobre todo en lo que refiere a expresiones matemáticas (y tengo que señalar que el fonética/fonología y cuando hay que escribir textos en alfabetos no latinos). Para escribir \LaTeX puede utilizarse cualquier editor de textos (yo suelo utilizar *mcedit*), pero la tarea resulta más fácil usando *emacs + auctex* y últimamente he descubierto lo fácil que es usar *kile*.

No podemos entrar aquí más que en mostrar unos cuantos ejemplos. En primer lugar señalaré que el documento que usted está leyendo, con sus índices automáticos e hipervínculos, está escrito en \LaTeX .

El resultado de escribir

```
\begin{Corolario}\label{cor:paracualquier}
Para cualquier familia finita  $\{M_i\}^{i=1}_n$  de  $A$ -módulos se tiene
la equivalencia de las dos afirmaciones siguientes:
\begin{enumerate}
\item Todos los  $A$ -módulos  $M_1, \dots, M_n$  son noetherianos (resp.
artinianos).
\item La suma directa  $\bigoplus_{i=1}^n M_i$  es noetheriana (resp.
artiniana).
\end{enumerate}
\end{Corolario}
```

Demostración. Argumentar por inducción sobre n , considerando en el caso de $n > 1$ la sucesión exacta corta siguiente

```
\begin{displaymath}
0 \longrightarrow M_n \xrightarrow{\iota} \bigoplus_{i=1}^n M_i \xrightarrow{\pi} \bigoplus_{i=1}^{n-1} M_i \longrightarrow 0,
\end{displaymath}
donde  $\iota$  y  $\pi$  son los homomorfismos dados por
\begin{displaymath}
\iota : u \mapsto (0, \dots, 0, u)
\end{displaymath}
y
\begin{displaymath}
\pi : (x_1, \dots, x_n) \mapsto (x_1, \dots, x_{n-1})
\end{displaymath}
para cualesquiera
 $u \in M_n$  y  $(x_1, \dots, x_n) \in \bigoplus_{i=1}^n M_i$ .
```

es

⁴He de reconocer que esto no es en absoluto cierto. LyX es un intento de facilitar el trabajo con \LaTeX , pero ha desembocado en otro formato de edición.

⁵Tampoco esto es cierto: es un conjunto de macros que trabajan por encima de TEX , que es el programa que realmente hace el trabajo.

Corolario 5.2.1 Para cualquier familia finita $\{M_i\}_{i=1}^n$ de A -módulos se tiene la equivalencia de las dos afirmaciones siguientes:

1. Todos los A -módulos M_1, \dots, M_n son noetherianos (resp. artinianos).
2. La suma directa $\bigoplus_{i=1}^n M_i$ es noetheriana (resp. artiniana).

Demostración. Argumentar por inducción sobre n , considerando en el caso de $n > 1$ la sucesión exacta corta siguiente

$$0 \longrightarrow M_n \xrightarrow{\iota} \bigoplus_{i=1}^n M_i \xrightarrow{\pi} \bigoplus_{i=1}^{n-1} M_i \longrightarrow 0,$$

donde ι y π son los homomorfismos dados por

$$\iota : u \mapsto (0, \dots, 0, u)$$

y

$$\pi : (x_1, \dots, x_n) \mapsto (x_1, \dots, x_{n-1})$$

para cualesquiera $u \in M_n$ y $(x_1, \dots, x_n) \in \bigoplus_{i=1}^n M_i$.

La figura 5.2 es un ejemplo musical.

5.3 Documentos en otros idiomas

¡No lo creerá usted, pero hay gente que hace preguntas! ¡Y espera que yo se las responda! El problema es que como esta distribución está destinada al andaluz medio, resulta que no está destinada a nadie real; los compañeros de griego me preguntan cómo pueden escribir griego usando los tipos de letra *truetype* que venían utilizando, e incluso una compañera quiere escribir en árabe. La distribución tal como la conocemos no lo permite, pero sí si podemos instalar unos cuantos paquetes, veamos cómo.

Antes hay que comprender un par de conceptos. La distribución está preconfigurada para usuarios españoles y textos en español. Estas son dos decisiones independientes: la primera se refiere al idioma en que la máquina (los programas, los mensajes de error, etc.) se dirigirá a nosotros (si existe la traducción correspondiente al mensaje; incluso se puede distinguir español de España, es_ES, de las restantes variedades del español), qué carácter mostrará en pantalla cuando pulsemos la tecla que tiene pintada una 'ñ' y cuestiones similares (como que el tipo de letra que sale por la pantalla contenga la 'ñ' y que el programa de impresión pueda enviársela a la impresora). La segunda se refiere a la codificación en que están los textos si no se indica lo contrario (recordemos los problemas que vivimos con los caracteres extraños cuando pasamos de guardar nuestros datos en DOS a guardarlos en Windows⁶).

Supongamos que queremos que la máquina se siga dirigiendo a nosotros en español⁷ pero queremos escribir en otro idioma y otro alfabeto. Y luego queremos poder imprimirlo. Vale, es cuestión de tres cosas: tipos de letra, codificaciones y configuración de la entrada por teclado.

1. Deberemos utilizar tipos de letra que contengan los caracteres de los idiomas que queremos escribir (nos vamos a centrar en griego y árabe). Más concretamente necesitamos tipos de letra *unicode, iso-10646-1*⁸.

⁶Esto tiene nombre: pasamos de usar cp437 a usar windows-1252. Ninguno de estas dos codificaciones se utilizan en GNU Linux, pero esta última es muy parecida a ISO-8859-1, la habitual, también conocida como *latin1*. Más modernas son ISO-8859-15, que incorpora en símbolo del euro, y UTF-8, que permite utilizar prácticamente todos los caracteres de las distintas escrituras del mundo. Es decir y resumiendo, para el castellano podrían utilizarse los '*locales*' es_ES.iso8859-1, es_ESeuro.iso8859-15 o es_ES.utf-8. Si esta explicación es demasiado breve y no acaba de comprenderla, puede consultar el «Debian Euro HOWTO», del paquete *euro-support*.

⁷No se ha pensado en los profesores de francés o inglés o incluso en los alumnos inmigrantes, para los que sería un gran alivio psicológico interactuar y poder escribir con su alfabeto y en su idioma.

⁸Disponemos en cualquier distribución derivada de Debian (puede que instalados o no) de los siguientes paquetes: *unifont, xfonts-100dpi, xfonts-base, xfonts-scalable, ttf-freefont, ttf-thryomanes* y *ttf-bitstream-vera*. Conviene también instalar los tipos *truetype* que proporciona gratuitamente Microsoft: *msttcorefonts*.

1

Petite Overture à danser

4

ERIK SATIE (1866-1925)

The image displays a musical score for 'Petite Overture à danser' by Erik Satie. The score is presented in four systems, each consisting of a grand staff (treble and bass clefs). The music is in 3/4 time and features a characteristic Satian style with simple, repetitive melodic lines and block chords. The first system starts with a key signature of one flat (B-flat) and a common time signature of 3/4. The second system begins at measure 7. The third system begins at measure 14 and includes a key signature change to two flats (B-flat and E-flat) and a time signature change to 3/4. The fourth system begins at measure 22 and includes a key signature change to two sharps (F# and C#) and a time signature change to 3/4. The score is written in a clean, minimalist style with clear notation for notes, rests, and dynamics.

Public Domain

Figura 5.2: Ejemplo musical con \LaTeX

Por supuesto, es posible utilizar tipos no empaquetados, pero la configuración de los distintos programas para que puedan utilizarlos no está unificada. Esta complicación es explicable por razones históricas: no hay una forma unificada de instalar y configurar nuevos tipos de letra, porque los programas que conforman la distribución tienen orígenes distintos y han implementado soluciones diferentes.

2. La codificación va a ser una de las tres posibles mencionadas antes, porque queremos seguir utilizando el español.
3. Puesto que ningún teclado contiene los miles de caracteres necesarios para escribir los miles de idiomas del mundo, es necesario alguna forma de «mapear» los diferentes alfabetos en nuestro teclado. Pero esto es también específico de los programas.

La primera solución que propongo es utilizar el editor de textos unicode (no procesador de textos) **yudit**, que tiene métodos de entrada para casi cualquier alfabeto. Debemos consultar en `/usr/share/yudit/src/` los mapas de teclado, en nuestro caso `GreekPolytonic.kmap` y `Arabic.kmap`. Sirvan las figuras 5.3 y 5.4 como ejemplos y pruebas.

yudit busca sus tipos de letra siguiendo las instrucciones del valor *yudit.fontpath*, en el fichero `/etc/yudit/yudit.properties`.

En el mismo **L^AT_EX** que estamos utilizando es posible escribir griego. Basta con haber comenzado el documento con la instrucción

```
\usepackage[polutonikogreek,spanish]{babel}
```

Y escribir ahora

```
\selectlanguage{polutonikogreek}
p\`antes >`anjrwpoi to\~u e>id\`enai >or\`egontai f\`usei. shme\~ion d' \\  
<h t\~wn a>isj'hsewn >ag'aphsis; ka'i g'ar qwr'is t\~hs gre'ias \\  
>agap\~wntai di' a<ut'as, ka'i m'alista t\~wn >'allwn <h di'a t\~wn \\  
>omm'atwn. o>u g'ar m'onon <'ina pr'attwmen >all'a ka'i mhj'en \\  
m'ellontes pr'attein t'o <or\~an a<iro'umeja >ant'i p'antwn <ws e>ipe\~in \\  
t\~wn >'allwn. a>'ition d' <'oti m'alista poie\~i gnwr'izein <hm\~as \\  
<'auth t\~wn a>isj'hsewn ka'i poll'as dhlo\~i diafor'as. f'usei \\  
m'en o>\~un a>'isjhsin >'eqonta g'ignetai t'a z\~w|a, >ek d'e ta'uths \\  
to\~is m'en a>ut\~wn o>uk >egg'ignetai mn'hmh, to\~is d' >egg'ignetai.  
\selectlanguage{spanish}
```

Este es el resultado:

πάντες ἄνθρωποι τοῦ εἰδέναι ὀρέγονται φύσει. σημεῖον δ' ἢ τῶν αἰσθήσεων ἀγάπησις· καὶ γὰρ χωρὶς τῆς χρείας ἀγαπῶνται δι' αὐτάς, καὶ μάλιστα τῶν ἄλλων ἢ διὰ τῶν ὀμμάτων. οὐ γὰρ μόνον ἵνα πράττωμεν ἀλλὰ καὶ μηθὲν μέλλοντες πράττειν τὸ ὄρα̅ν αἰρούμεθα ἀντὶ πάντων ὡς εἰπεῖν τῶν ἄλλων. αἴτιον δ' ὅτι μάλιστα ποιεῖ γνωρίζειν ἡμᾶς αὕτη τῶν αἰσθήσεων καὶ πολλὰς δηλοῖ διαφοράς. φύσει μὲν οὖν αἰσθησὶν ἔχοντα γίνεταὶ τὰ ζῶα, ἐκ δὲ ταύτης τοῖς μὲν αὐτῶν οὐκ ἐγγίγνεταὶ μνήμη, τοῖς δ' ἐγγίγνεταὶ.

Para escribir árabe en un derivado de T_EX hay que utilizar *arabtex*, u *omega*.

Pero los compañeros no quieren aprender *todavía* un entorno nuevo. Quieren utilizar el **OpenOffice**, porque es el entorno más parecido al que conocen. Bien, se puede hacer (y de camino aprendemos a añadir un nuevo tipo de letra). Esta solución va a ser válida para todos los programas de KDE3 y Gnome2 (incluido OpenOffice).

En primer lugar, debemos copiar el nuevo tipo a alguno de los subdirectorios recogidos en `/etc/fonts/fonts.conf` (o añadir otro en `/etc/fonts/local.conf`). Seguidamente reiniciaremos las *cachés* de tipos:

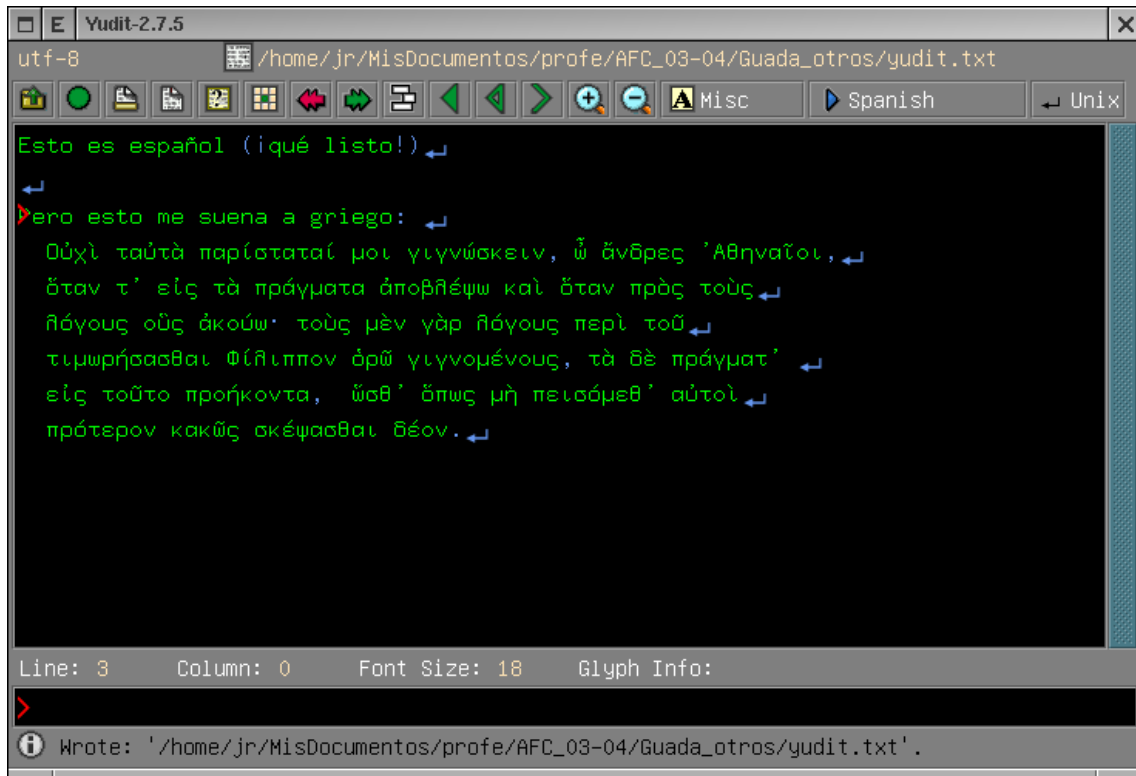


Figura 5.3: Griego en yudit

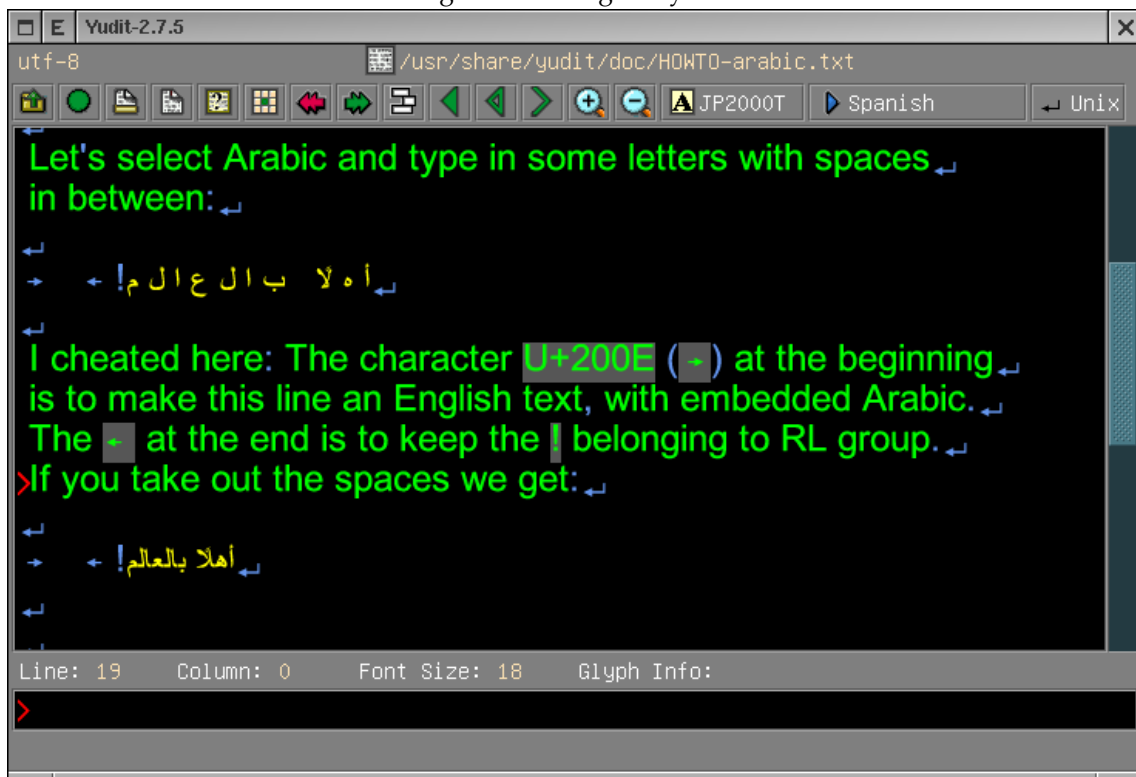


Figura 5.4: Árabe en yudit


```
[rooto@Máquina]# dpkg-reconfigure fontconfig
```

Abriendo en *oowriter* veremos que ya tenemos disponible la nueva letra. Otro tema es cómo se mapea el teclado.

5.4 Herramientas

5.4.1 Documentos colaborativos y control de revisiones

cvs *cvs* es un sistema para que varias personas puedan trabajar conjuntamente en un proyecto; guarda los cambios que se realizan a los ficheros, quién los ha hecho etc., de manera que puede recuperarse cualquier versión anterior si se ha cometido un error. Vamos a descargar para nuestro ejemplo datos del servidor *cvs* de TLDP-ES. Hay dos maneras de descargar los contenidos de un repositorio *cvs*:

(1) hacerse una cuenta anónima

(1.1) Escriba la siguiente cadena en una terminal

```
export CVSROOT=":pserver:anoncvs@cvs.hispalinux.es:/cvs/lucas"
```

Lo que estamos poniendo en la variable de entorno es qué método vamos a utilizar para conectarnos con la máquina, qué máquina y dónde está el ‘repositorio’ (¿por qué no empezamos a decir ‘fondo’ o ‘almacén’) de ficheros. Sólo vamos a poder leer, para poder escribir tendríamos que tener cuenta.

(1.2) Ejecute la siguiente orden (sólo hará falta la primera vez)

```
[Usuario@Máquina]$ cvs login
```

(1.3) Le preguntará contraseña: es ‘anoncvs’

Esto va a añadir una línea en el fichero *.cvspass* de su directorio personal.

(1.4) Ya puede descargar cualquier directorio o fichero. La primera vez deberá hacer

```
[Usuario@Máquina]$ cvs checkout doc-curso-guadalinex-iesaverros
```

En adelante bastará con situarse en el directorio que quiere actualizar y escribir

```
[Usuario@Máquina]$ cvs update
```

¿Fácil, verdad?

(2) mediante una interfaz web (esto es más fácil, pero no nos convierte en *hackers*, ¿verdad? :-)).

<http://cvs.hispalinux.es/cgi-bin/cvsweb>

En <http://es.tldp.org/cvs.html> hay más ayuda sobre todo esto⁹.

wiki La figura 5.5 es un ejemplo de un *wiki* recién instalado en mi máquina (paquete *phpwiki*). Una *Web WikiWiki* (*WikiWikiWeb*; por lo visto «*wiki wiki*» es hawaiano y significa ‘rápido’) es un sitio web en el que todo el mundo puede colaborar modificando y creando directamente su contenido. Es posible (y fácil) escribir colectivamente páginas. Todo lo que necesita aprender para poder editar páginas se reduce a hacer click en el botón “Editar Texto” que tiene al pie para modificar el texto que está viendo.

⁹Un autor no debe saber sólo descargar ficheros, sino que debe ser capaz de identificarse como un usuario con cuenta y no como anónimo, crear nuevos módulos (`cvs import -m "Creando el módulo" modulo v0 r0`), y subir ficheros al repositorio (`cvs ci modulo`).

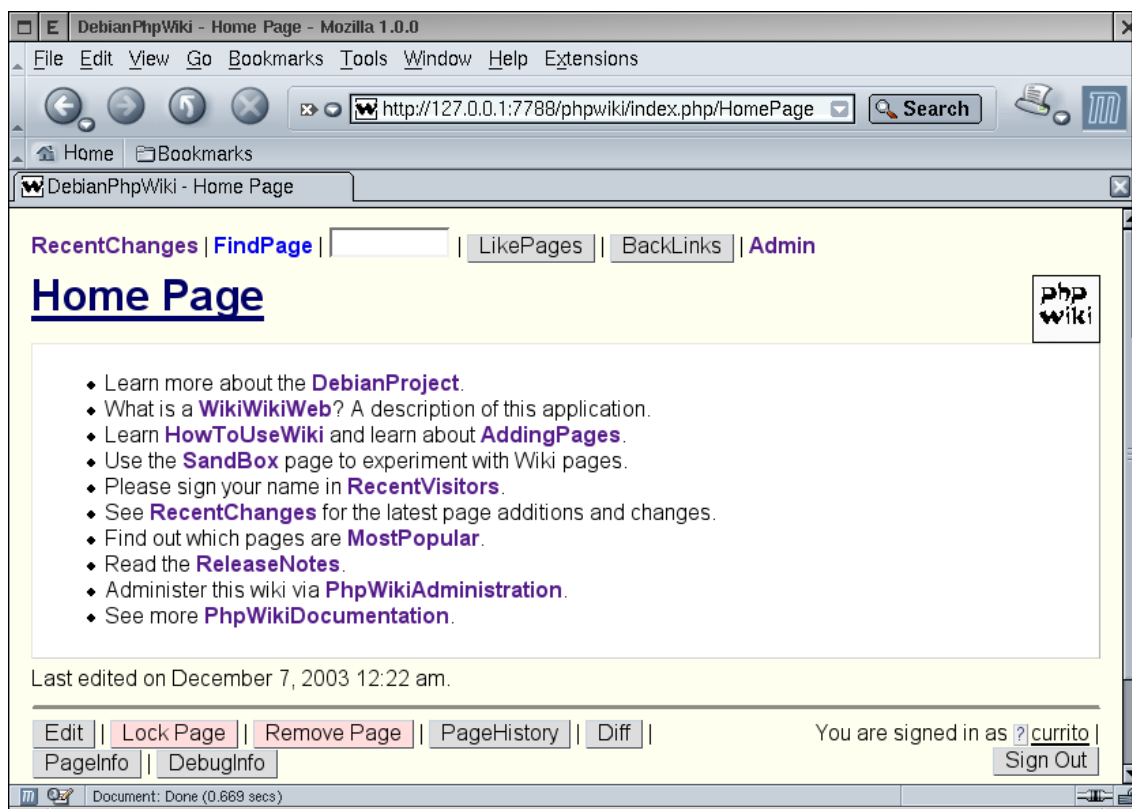


Figura 5.5: Ejemplo de un wiki

5.4.2 Corrección ortográfica

Quizás resulte paradójico para el que se acerca al tema por primera vez, pero para hablar de correctores ortográficos primero hay que hablar de diccionarios. En primer lugar evitemos una confusión generalizada, *de verba non est disputandum*. Diccionarios son listas de palabras de un idioma, entradas léxicas con su definición, tablas de equivalencias entre dos o más lenguas y cualesquiera otras variaciones que se nos planteen; nomenclatura, glosario, lexicón, lemario etc. son términos que se usan de manera no estable en la literatura y no deben impedir que nos entendamos¹⁰.

Este tipo de diccionarios están centralizados en Debian, y dependen en su totalidad del paquete *dictionaries-common*.

wordlists Siguiendo con las paradojas aparentes, uno de los usos principales de las listas de palabras ordenadas alfabéticamente es el del ataque o comprobación de la robustez de claves criptográficas y contraseñas (no estoy diciendo que sea su única utilidad, pero es significativo examinar los programas que dependen de su existencia). Se distribuyen en paquetes que empiezan por *w* seguida del nombre de un idioma: *wspanish*...

La lista que utilizará el sistema se decide mediante la orden

```
[root@Máquina]# /usr/sbin/select-default-wordlist
```

ispell, aspell, myspell Los diccionarios para los correctores ortográficos se nombran con *i* seguido del lenguaje: *ispanish* (porque se utilizaban originariamente por *ispell*, el corrector *internacional*).

Para elegir el español el superusuario utilizará

¹⁰Basta consultar la clasificación de dos páginas del término «diccionario» que hace Martínez De Sousa *Diccionario de lexicografía práctica* para llegar a la conclusión de que lo importante no es cómo se llama sino qué clase de diccionario queremos.

```
[root@Máquina]# /usr/sbin/select-default-ispell
```

Su origen está en el proyecto *COES*: Herramientas para Procesamiento de Lenguaje Natural en Español (<http://www.datsi.fi.upm.es/coes/>), de Santiago Rodríguez y Jesús Carretero, que se distribuye como software de libre disposición desde finales de 1994.

COES consta de un diccionario de unos 53.000 términos y un corrector ortográfico integrado en la utilidad Unix *ispell* y desarrollos derivados (*aspell*...). Hay que señalar que se ha ampliado el conjunto de herramientas lingüísticas con un diccionario de sinónimos/antónimos. Su particularidad es ser sensible a las reglas morfológicas de las palabras y no sólo a las raíces. Un corrector ortográfico «inteligente» tiene que hacer algo más que comparar las palabras del texto con una lista de palabras correctas; para distinguir «a» de «ha» tiene que tener reglas — acabaremos necesitando un mínimo análisis morfológico y sintáctico.

Un usuario puede elegir el fichero que contendrá su diccionario personal (`/ .default-ispell`) mediante la orden

```
[Usuario@Máquina]$ select-default-iwrap
```

5.4.3 Diccionarios de consulta

Existen varios diccionarios consultables en línea y hay varias interfaces de consulta disponibles. Por ejemplo cf. los paquetes *hdico* para consultar el diccionario Hachette de francés o *gnuod* para el diccionario Van Dale holandés. Por supuesto, el diccionario de la Real Academia Española de la Lengua también es consultable mediante un navegador. O puede utilizarse cualesquiera de los scripts disponibles, como el *xdrae*¹¹ de Xento Figal.

Nos vamos a centrar en los diccionarios que pueden ser descargados libremente, por dos motivos: no requieren conexión a internet para su consulta y esta puede hacerse desde por ejemplo un CD y la libertad de descarga y distribución nos garantizan su carácter libre.¹²

Un modelo: el tesoro informático de ORCA El objetivo de este glosario¹³ no es explicar el significado de los términos de informática usados en inglés, sino dar una lista de sugerencias para su traducción al español, para quien ya tenga suficientes conocimientos de informática en inglés. No obstante, Jaime Villate ha expresado su intención de que ORCA se convierta en un verdadero glosario, y ya aparecen algunas explicaciones.

Para muestra, un botón; consultamos «driver». Esta es la salida:

```
driver
  controlador, manejador, gestor, driver video
```

La principal fuente para este glosario ha sido la comunidad hispano-parlante que desarrolla y usa software libre, participando directamente en la edición del glosario a través de su interfaz web¹⁴, o indirectamente a través de sus discusiones en las listas de correo sobre el tema. Distingue entre «colaboradores» y «editor»: un colaborador no puede borrar lo que ya ha sido escrito por otro, pero puede escribir comentarios; el editor después va a leer esos comentarios, y altera la definición respectivamente. Los números de versiones terminados en **.0** quieren decir que el glosario acaba de ser revisado por el editor; si el último número no es cero, indica el número de contribuciones que han sido introducidas desde la última revisión. La última versión revisada es la 2.1.0, de mayo de 2002, que incluye 887 términos.

¹¹<http://xinfo.sourceforge.net/xdrae.php>. El recordatorio, y de que hay paquete Debian, es de Eloy Rafael Sanz.

¹²Hay bastantes más que los que vamos a examinar, de interés especializado: *pydict* de inglés y chino; *skk* y *edict* de japonés, *mueller* para inglés-alemán, *ksocrat* inglés-ruso...

¹³Proyecto ORCA - Herramientas de ayuda para los traductores y productores de software libre en español (programas y documentación), <http://quark.fe.up.pt/orca/index.es.html>; proyecto responsabilidad de Jaime E. Villate bajo los términos de la Licencia GNU para Documentación Libre. Realmente es un *tesauro*, como reconoce el propio Villate en correo a la lista `<debian-110n-spanish>` de fecha 29 de marzo de 2000: un «glosario» explica con detalle los términos, mientras que un «tesauro» sugiere simplemente sinónimos.

¹⁴<http://quark.fe.up.pt/orca>.

La red DICT El *Grupo de Desarrollo de DICT* pretende dar solución a un problema: ¿cómo estandarizar el acceso a los múltiples diccionarios disponibles?

La RFC 2229 describe el *protocolo DICT* como un protocolo de consulta/respuesta sobre TCP que permite a un cliente acceder a un diccionario de definiciones utilizando un conjunto de bases de datos de diccionarios de lenguajes naturales.

El grupo de desarrollo del proyecto de licencia GPL tiene su página en <http://www.dict.org>. El protocolo sigue el modelo cliente/servidor, con clientes de texto (`dict` e incluso una interfaz para `emacs`: `dictionary-el`) y con numerosas interfaces gráficas (`gsdict`, `kdick`, `gdick` (del paquete `gnome-utils`), `wordinspect` o el `wordtrans` de Ricardo Villalba¹⁵) y servidores que sirven información de bases de datos léxicas.

La gran ventaja es la existencia de diccionarios y tesoros que pueden instalarse localmente, en la máquina donde se va a realizar la consulta o en un servidor de área local. Examinemos los diccionarios de más interés de los ¡61! incluidos en la distribución que administro (Debian Sarge), para después pasar a comprender cómo se crea un diccionario nuevo (no traduzco la información proporcionada por los diccionarios en inglés porque de todos modos su utilidad requerirá la comprensión del inglés):

- *dict-gcide*: the GNU version of the Collaborative International Dictionary of English. The GCIDE contains the full text of the 1913 Webster's Unabridged Dictionary, supplemented by many definitions from WordNet, the Century Dictionary, 1906, and many additional definitions contributed by volunteers.
- *dict-wn*: este paquete es la adaptación al formato `.dict` de WordNet 2.0 (a Lexical Database for English from the Cognitive Science Laboratory at Princeton University). This dictionary contains only nouns, verbs, adverbs, and adjectives. It is not intended as a stand-alone dictionary, but rather as a supplement to the older but more complete dictionary found in `dict-gcide`.
- *dict-moby-thesaurus*: Moby Thesaurus is the largest and most comprehensive thesaurus data source in English available. This second edition has been thoroughly revised adding more than 5.000 root words (to total more than 30.000) with an additional *million* synonyms and related terms (to total more than 2.5 *million* synonyms and related terms).
- *dict-foldoc* y *dict-jargon*: the Free On-line Dictionary of Computing, and the Hacker's Jargon file. There is a great deal of overlap between the Jargon file and the FOLDOC. Although the FOLDOC is much larger than the Jargon file, there are numerous entries in the Jargon file that are not found in FOLDOC.
- *dict-vera*: Virtual Entity of Relevant Acronyms, a dictionary of acronyms used in the computer field.
- *dict-devil*: The Devil's Dictionary, a cynical and irreverent dictionary of common words.
- *dict-elements*: Jay Kominék's <jkominék@miranda.org> compilation of physical and chemical data about the elements.

No hay que ser muy perspicaces para percatarse de que un vez más el conjunto de conocimiento hecho público en la comunidad angloparlante es abrumadoramente superior al disponible en la cacareante hispanidad. No hay un diccionario español digno de ese nombre (sí decenas de esbozos de diccionarios informáticos, pero no es ese el tema que nos concierne ahora). En el tema de los diccionarios bilingües libres destaca el proyecto `Freedict`¹⁶, con 42 diccionarios de y a las principales lenguas. Destacaremos la disponibilidad de

- `dict-freedict-eng-spa`, `dict-freedict-spa-eng`
- `dict-freedict-fra-eng`, `dict-freedict-eng-fra`

¹⁵<http://www.escomposlinux.org/rvm/wordtrans/index.php>.

¹⁶<http://www.freedict.de/Info.html>.

- `dict-freedict-eng-ita`, `dict-freedict-lat-eng` (sí, latín-inglés)

Para el alemán disponemos, además de `dict-freedict-deu-eng` y `dict-freedict-eng-deu`, de las aproximadamente 124.000 entradas de *dict-de-en*.

Además, he añadido localmente

- *dict-i2e*: diccionario inglés-español, empaquetado por Ricardo Villalba¹⁷, con unas 24.500 entradas. Por lo que recuerdo fue creado por Alfredo Casademunt, a su vez basándose en el trabajo de José Luis Triviño.
- la versión `.dict` del *Glosario de Orca*

Un ejemplo nos mostrará el uso del protocolo:

```
[Usuario@Máquina]$ dict font
```

da la siguiente salida

```
8 definitions found
```

```
From WordNet (r) 2.0 (August 2003) [wn]:
```

```
font
```

- ```
n 1: a specific size and style of type within a type family [syn:
 {fount}, {typeface}, {face}]
n 2: bowl for baptismal water [syn: {baptismal font}, {baptistry},
 {baptistery}]
```

```
From The Collaborative International Dictionary of English v.0.48 [gcide]:
```

```
Font \Font\, n. [F. fonte, fr. fondre to melt or cast. See
 {Fount} to cast, and cf. {Fount} a font.] (Print.)
 A complete assortment of printing type of one size, including
 a due proportion of all the letters in the alphabet, large
 and small, points, accents, and whatever else is necessary
 for printing with that variety of types; a fount.
 [1913 Webster]
```

```
From The Collaborative International Dictionary of English v.0.48 [gcide]:
```

```
Font \Font\, n. [AS. font, fant, fr. L. fons, fontis, spring,
 fountain; cf. OF. font, funt, F. fonts, fonts baptismaux, pl.
 See {Fount}.]
 1. A fountain; a spring; a source.
 [1913 Webster]

 Bathing forever in the font of bliss. --Young.
 [1913 Webster]

 2. A basin or stone vessel in which water is contained for
 baptizing.
 [1913 Webster]

 That name was given me at the font. --Shak.
 [1913 Webster]
```

---

<sup>17</sup><http://wordtrans.sourceforge.net>. No podemos pasar por alto el diccionario español/alemán de más de 23.000 entradas que se está desarrollando por su iniciativa y la de Cristina Embid en <http://diccionario.escomposlinux.org>.

From The Free On-line Dictionary of Computing (19 Sep 2003) [foldoc]:

font

```
<text> A set of {glyphs} ({images}) representing the
{characters} from some particular {character set} in a
particular size and {typeface}. The image of each character
may be encoded either as a {bitmap} (in a {bitmap font}) or by
a higher-level description in terms of lines and areas (an
{outline font}).
```

```
There are several different computer representations for
fonts, the most widely known are {Adobe Systems, Inc.}'s
{PostScript} font definitions and {Apple}'s {TrueType}.
{Window systems} can display different fonts on the screen and
print them.
```

```
[Other types of font?]
```

```
(2001-04-27)
```

From i2e [i2e]:

```
font : tipo (de letra)
```

From i2e [i2e]:

```
font : fuente
```

From ORCA - Glosario de Informática Inglés-Español [glosario]:

font

```
fuelle, tipo de letra, (TIPO DE LETRA, fuente, en español, tiene que
ver con agua, no con tipografía)
```

La creación de un diccionario depende de los paquetes *dictfmt*, *dictzip* y claro, su utilidad depende de la presencia de *dictd*.<sup>18</sup>

1) se convierte el fichero fuente al formato *.dict* con *dictfmt* (en el manual se explican distintos tipos de ficheros fuente) a la vez se crean los ficheros índices. Quizás convenga explicarlo con un ejemplo. Vamos a crear el diccionario *test*, con el siguiente texto:

```
Este texto debe aparecer
aunque sean varias líneas. Aquí irá
la licencia, etc.
```

```
%h Esta línea desaparece
%h aaaaa
%d
la letra a
```

```
%h eeeee
```

---

<sup>18</sup>Allá por año 1999 se utilizaba un paquete ruso llamado *dicttools*. Se puede encontrar documentación alternativa sobre este tema en los manuales de *dictfmt* y *dictzip*, en las RFC pertinentes y en [http://www.freedict.de/How\\_make\\_dictionaries.html](http://www.freedict.de/How_make_dictionaries.html), donde se menciona un enfoque muy interesante en el que no podemos entrar aquí (<http://www.physik.fu-berlin.de/~tburnus/freedict/>).

```
%d
la letra e
```

```
%h iiii
%d
la letra i
```

```
%h ooooo
%d
la letra o
```

```
%h uuuuu
%d
la letra u
```

¿Este texto dónde aparece?

Ahora hacemos (recuerde: en una sola línea)

```
[Usuario@Máquina]$ cat test | dictfmt -p -u http://localhost \
-s "Bonita Prueba" --locale es_ES test
```

que nos ha generado test.dict

```
00-database-8bit
```

```
00-database-url
 http://localhost
```

```
00-database-short
 Bonita Prueba
```

```
00-database-info
```

```
This file was converted from the original database on:
 Wed Jan 7 00:47:20 2004
```

```
The original data is available from:
 http://localhost
```

The original data was distributed with the notice shown below.  
No additional restrictions are claimed. Please redistribute this  
changed version under the same conditions and restriction that apply  
to the original version.

```
Este texto debe aparecer
aunque sean varias líneas. Aquí irá
la licencia, etc.
```

```
aaaaa
la letra a
```

```
eeee
la letra e
```

```
iiii
la letra i
```

```
oooo
la letra o
```

```
uuuuu
la letra u
```

¿Este texto dónde aparece?

y `test.index`. Es conveniente comprimir el fichero `.dict` resultante con `dictzip`.

2) copie los `*.dict.dz` y `*.index` a `/usr/local/share/dictd`

3) informe al servidor de la presencia de los nuevos diccionarios en `/etc/dictd/dictd.order`

4) en `/etc/dictd/dictd.conf` se fija el fichero que listará los diccionarios del sistema, en mi caso la línea que nos interesa dice (en Debian es `dictdconfig` el que crea el fichero `db.list`)

```
include /var/lib/dictd/db.list
```

5) `/usr/sbin/dictdconfig -write`

6) `/etc/init.d/dictd restart`



## Capítulo 6

# Hojas de cálculo y gráficas

La aplicación que permite generar y trabajar con hojas de cálculo es *OpenOffice Calc* (`oocalc`)<sup>1</sup>.

**Ejercicio.** Intente generar la hoja de cálculo de la figura 6.1.

**Ejercicio.** Cree una hoja con los resultados ficticios de un grupo de alumnos, y que la aplicación calcule medias y otras estadísticas y las refleje en gráficas.

---

<sup>1</sup>Esta afirmación es verdadera respecto a Guadalinex-edu, no a la Guadalinex Ciudadano disponible cuando escribo esta documentación, donde también podemos utilizar *gnumeric*.

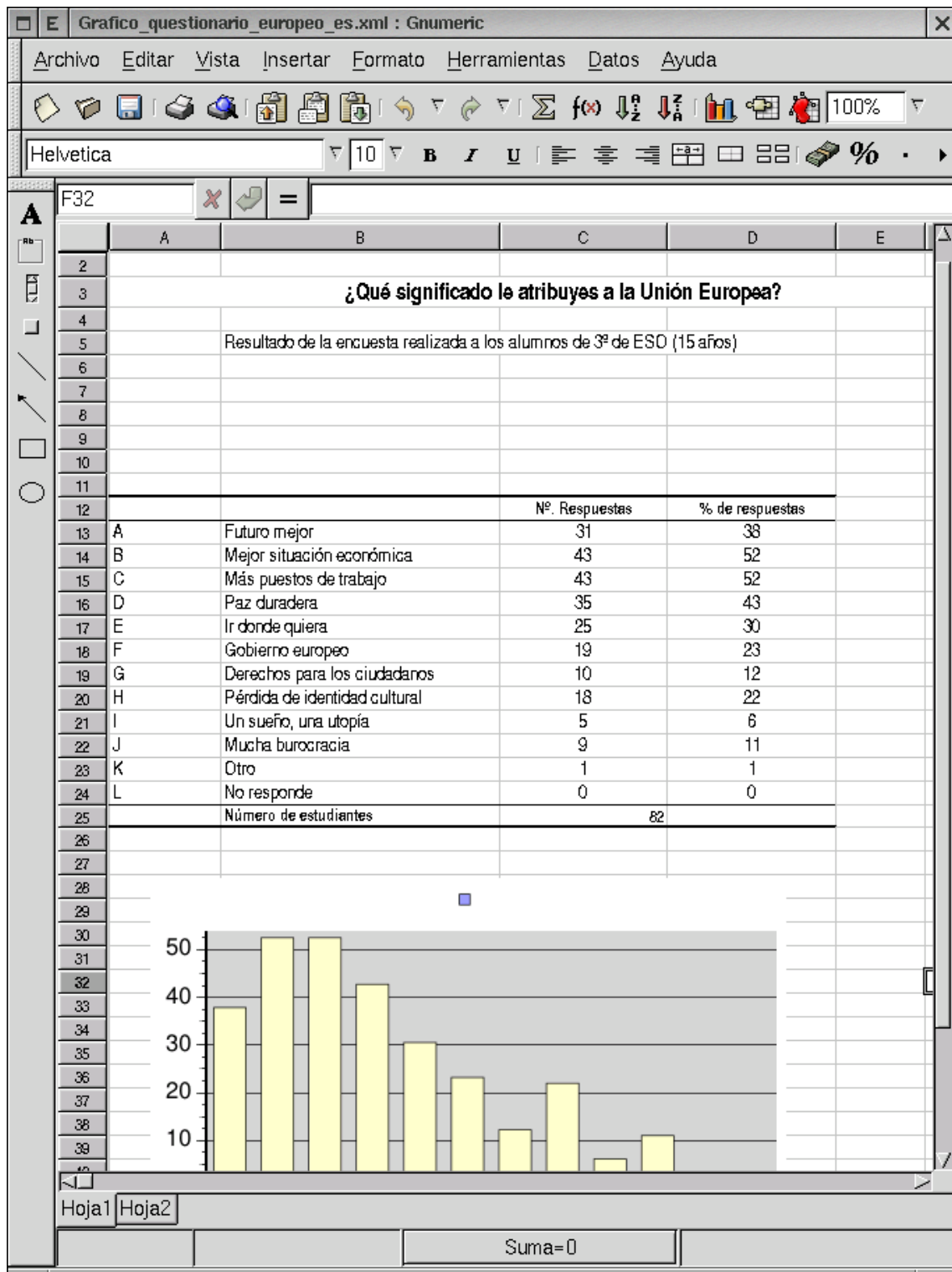


Figura 6.1: Ejemplo de hoja de cálculo

# Capítulo 7

## Cientes de Bases de datos

Nuestra distribución está pensada para una máquina cliente. El problema a la hora de explicar los clientes de bases de datos es que no disponemos de ninguna base de datos que consultar, ni se ha planteado que se vaya a instalar un servidor de bases de datos (sea *mysql* o *postgresql*). Como siempre, la cosa tiene solución; primero aprenderemos a leer los datos de bases de datos elementales, cuyas tablas se limitan a ficheros y después crearemos una base de datos de prueba en mi máquina.

### 7.1 Ficheros .dbf

Los ficheros clásicos creados por *dbase*, *clipper*<sup>1</sup>, etc. pueden leerse directamente con las herramientas que nos proporciona la distribución.

Hay que señalar que programas como *gnumeric* pueden abrir directamente ficheros **.dbf**, con mayor o menor éxito, como puede comprobar en la figura 7.1 (vemos que no importa los campos memo y que no reconvierte la codificación de caracteres, por eso 'Buñuel' se queda en 'Bu').

Ahora usaremos *dbfdump*, del paquete *libxbase-perl*, utilizado sobre nuestra vieja tabla de directores de cine.

```
[Usuario@Máquina]$ /usr/bin/dbfdump --info director.dbf
Filename: director.dbf
Version: 0x83 (ver. 3 with DBT file)
Num of records: 14
Header length: 162
Record length: 51
Last change: 1993/6/14
Num fields: 4
Field info:
Num Name Type Len Decimal
1. CDDIREC C 3 0
2. DIRECTOR C 35 0
3. NACION C 2 0
4. NOTAS M 10 0
```

Y ahora viene la magia de la línea de comandos: volcamos los datos separados por comas a un fichero de texto, eliminamos el campo memo que no tenía ningún interés, y además reconvertimos la codificación de cp850 (la del DOS) a latin1, la que nos va a mostrar correctamente los caracteres en GNU Linux:

---

<sup>1</sup>A propósito, y no hay en esta documentación un lugar mejor para apuntarlo, hay un proyecto bastante completo de una versión para GNU Linux del lenguaje *clipper* que se llama *Harbour* (<http://www.harbour-project.org/>).

|    | A       | B                   | C      | D                          |
|----|---------|---------------------|--------|----------------------------|
| 1  | CDDIREC | DIRECTOR            | NACION | NOTAS                      |
| 2  | 002     | ALFRED HITCHCOCK    | UK     | Field type 'M' unsupported |
| 3  | 013     | BERNARDO BERTOLUCCI | ES     | Field type 'M' unsupported |
| 4  | 008     | CECIL B. DE MILE    | US     | Field type 'M' unsupported |
| 5  | 006     | FERNANDO COLOMO     | ES     | Field type 'M' unsupported |
| 6  | 010     | FERNANDO TRUEBA     | ES     | Field type 'M' unsupported |
| 7  | 007     | JOHN FORD           | US     | Field type 'M' unsupported |
| 8  | 003     | JOHN HOUSTON        | US     | Field type 'M' unsupported |
| 9  | 005     | LOUIS MALLE         | FR     | Field type 'M' unsupported |
| 10 | 004     | LUCHINO VISCONTI    | IT     | Field type 'M' unsupported |
| 11 | 012     | LUIS BU             | ES     | Field type 'M' unsupported |
| 12 | 014     | NIKITA MIHALKOV     | RU     | Field type 'M' unsupported |
| 13 | 001     | PEDRO ALMODOVAR     | ES     | Field type 'M' unsupported |
| 14 | 011     | RIDLEY SCOTT        | US     | Field type 'M' unsupported |
| 15 | 009     | WERNER HERZOG       | D      | Field type 'M' unsupported |
| 16 |         |                     |        |                            |

Figura 7.1: .dbf en gnumeric

```
[Usuario@Máquina]$ dbfdump --fs "," --nomemo director.dbf |recode pc/ >director.csv
```

Ahora utilizamos la función 'Obtener datos externos' de *gnumeric* (mejor que abrirlo directamente, porque nos permitirá definir los separadores de campos, etc.). La prueba de que ha ido mejor, en la figura 7.2.

## 7.2 Ficheros de MS Access

*mdbtools* (<http://mdbtools.sourceforge.net/>, existe paquete en Debian) nos sacará de apuros.

Quizás usted esté buscando *gmdb* (figura 7.3).

## 7.3 Clientes y servidores

Que las bases de datos se rijan por la arquitectura cliente/servidor no significa que de cara a un usuario la forma de trabajar sea radicalmente distinta. Lo único que cambia es la necesidad de un administrador del sistema, que haya instalado y configurado la base de datos y le dé a usted permiso para acceder a ella. Una vez configurada, la forma de trabajo le resultará familiar y si tiene permiso podrá crear tablas, vistas...

La figura 7.4 debería tranquilizarle.

- mergeant/gnomedb, phppgadmin o phpmyadmin

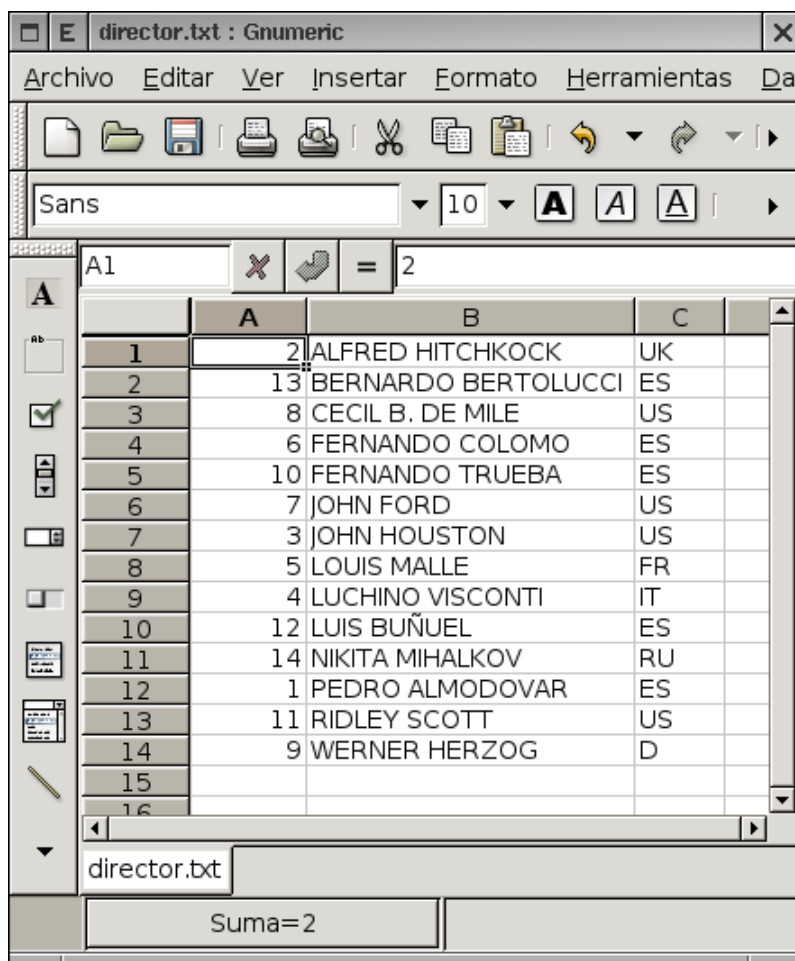


Figura 7.2: Texto delimitado en gnumeric

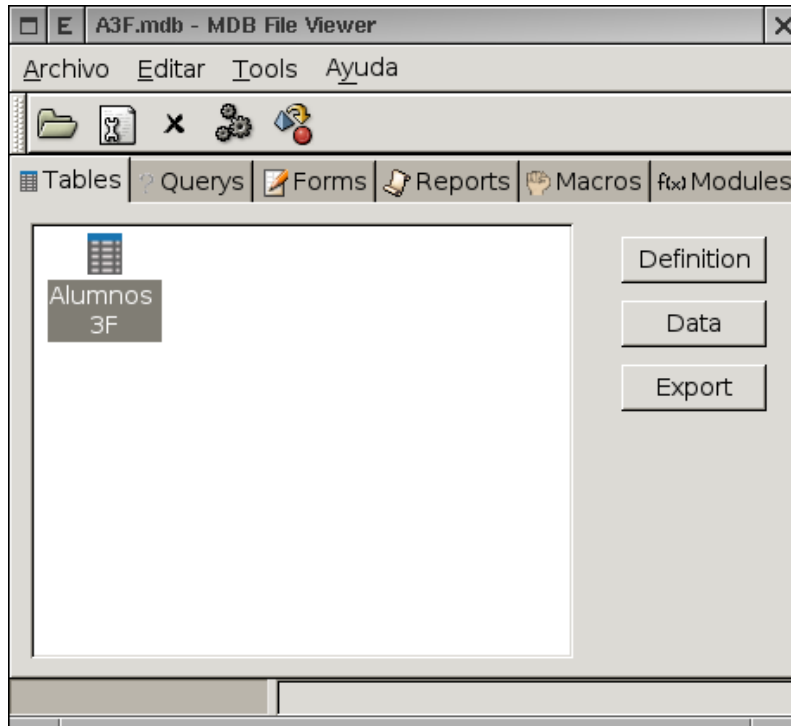


Figura 7.3: Un fichero de MS Access

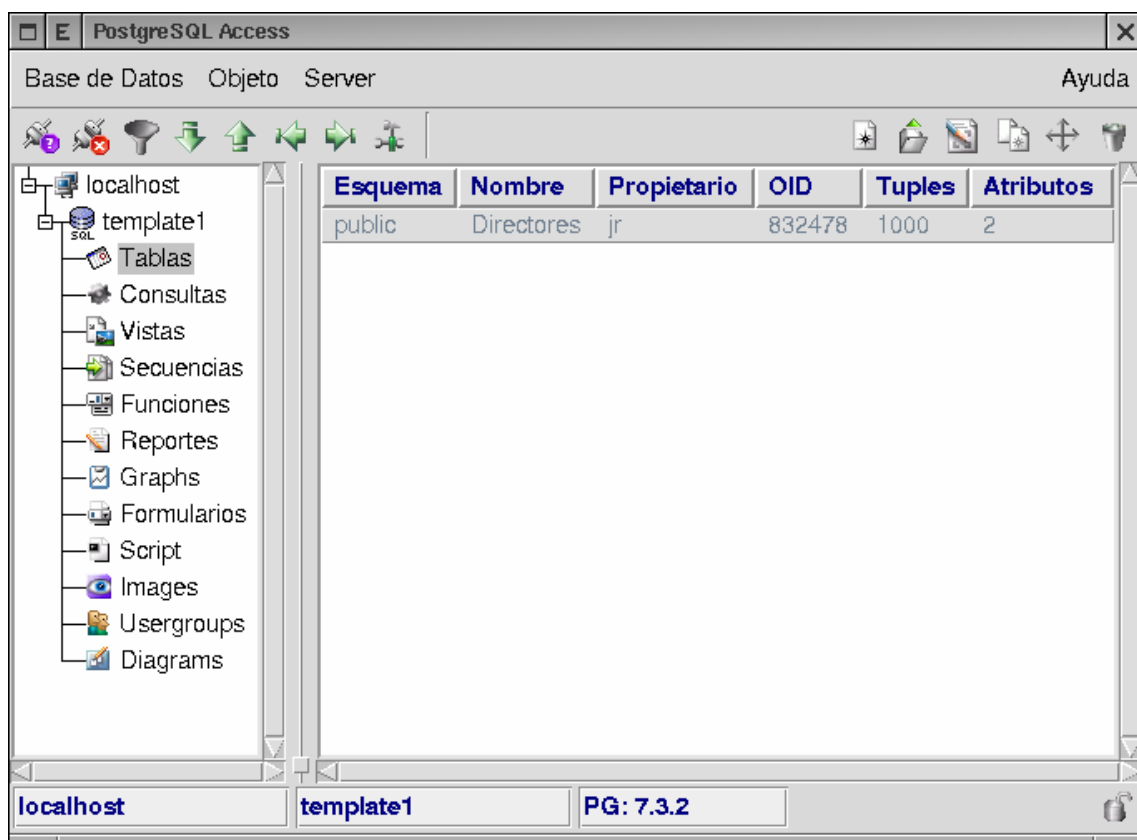


Figura 7.4: Interfaz de consulta a postgresql

## Capítulo 8

# Presentaciones

- *OpenOffice Impress* es el equivalente libre de Microsoft PowerPoint.
- `pdfscreen`, `prosper`, `texpower` (clases LaTeX)

## Capítulo 9

# Creación de páginas *web*

Con OpenOffice Writer

Con mozilla Composer

Más allá: editores .html (*quanta; bluefish, amaya*)

Trabajo con hojas de estilo

### 9.1 Más allá: *tidy*

Para analizar la corrección de un fichero:

```
[Usuario@Máquina]$ tidy -clean -latin1 FICHERO.html > /dev/null
```

Para corregirlo:

```
[Usuario@Máquina]$ tidy -clean -latin1 FICHERO.html > F_CORREGIDO.html
```

### 9.2 Mas allá: matemáticas en la *web*

MathML. cf. amaya, docbook-mathml, libgtkmathview-bin



## **Parte III**

# **Multimedia**

# Capítulo 10

## Fotos y dibujos

### 10.1 Formatos

Hay dos clases de ficheros de imágenes: los de mapa de bits (donde en el fichero están contenidos los niveles de color de cada punto de la imagen) y los vectoriales (que en el fondo son especies de programas que lo que contienen son las instrucciones para dibujar la imagen). Cada tipo tiene sus ventajas y sus inconvenientes: para almacenar una foto evidentemente será mucho más conveniente guardar su mapa de bits; por otro lado una imagen vectorial puede ser estirada o modificada en su tamaño, etc. sin pérdida de calidad.

La conversión de una imagen vectorial a mapa de bits es trivial: sólo hay que guardar la representación en pantalla. El proceso inverso no lo es, aunque hay programas que lo realizan con más o menos éxito: *autotrace* y *potrace* serían intentos de crear alternativas libres a Corel Autotrace.

Sobre los formatos conviene leer si quiere saberse más la documentación contenida en el paquete *imagemagick-doc* 'Image Formats', donde habla de que reconoce y muestra hasta 88 formatos. Si quiere saber qué ficheros puede leer y/o escribir su versión de *imagemagick*, ejecute la siguiente orden:

```
[Usuario@Máquina]$ convert -list format |less
```

Si quiere averiguar a qué formatos vectoriales puede exportar con *autotrace*, teclee

```
[Usuario@Máquina]$ autotrace -list-output-formats |less
```

La utilidad *identify*, del paquete *imagemagick*, nos da información interesante de cualquier fichero gráfico.

#### 10.1.1 Imágenes de mapa de bits

Los ficheros bitmap más comunes son *.bmp* (de Microsoft), *.pcx*, *.tiff*, *.gif*, *.png* y *.jpeg*. Ficheros vectoriales son los *.cdr* (de Corel Draw), *.fig*, *.wmf* (Windows Metafile) y *.emf* (Microsoft Enhanced Metafile), *.cgm* (Computer Graphics Metafile) y *.svg*<sup>1</sup>.

El algoritmo de compresión del formato *.gif* (CompuServe Graphics Interchange Format), LZW, tiene problemas de licencia por lo que desaconsejo el uso de este formato; es además totalmente prescindible.

La diferencia entre *.png* (Portable Network Graphics) y *.jpeg* (Joint Photographic Experts Group JFIF Format) es que en este último formato la compresión se produce con pérdida. Está indicado

---

<sup>1</sup>No vamos a hablar aquí de los *.dxf*, AutoCAD Drawing Exchange format, que sí pueden leerse y escribirse en el mundo Linux.

principalmente para imágenes procedentes de fotografías, donde el original es muy complejo y con muchos cambios de color de punto a punto; por otro lado su compresión es muy eficaz y resulta muy útil cuando la calidad no es prioritaria, para por ejemplo preparar imágenes que se van a poner en la *web*.

Para la conversión de un formato a otro podemos abrir con cualquier visor o editor de imágenes el fichero y salvarlo en el nuevo formato, o bien —lo que resultará más eficaz— utilizar *convert*, que además nos permitirá modificar desde la línea de comandos las imágenes. Para convertir simultáneamente varios ficheros no conozco otro recurso que *mogrify*.

### 10.1.2 Imágenes vectoriales

Tradicionalmente en el mundo Unix/Linux el formato vectorial más extendido ha sido *.fig*, utilizado por la aplicación *xfig* acompañado por la utilidad *fig2dev* del paquete *transfig*, que permite la conversión a otros formatos vectoriales y de bitmaps.

*.svg* (Scalable Vector Graphics) es un estándar propuesto por la W3C.

## 10.2 Visores de imágenes

— Visores: *display* (*imagemagick*), *eog* (Eye Of Gnome)

— Galerías de fotos: *gthumb*, *(x)zgv*, *gallery*

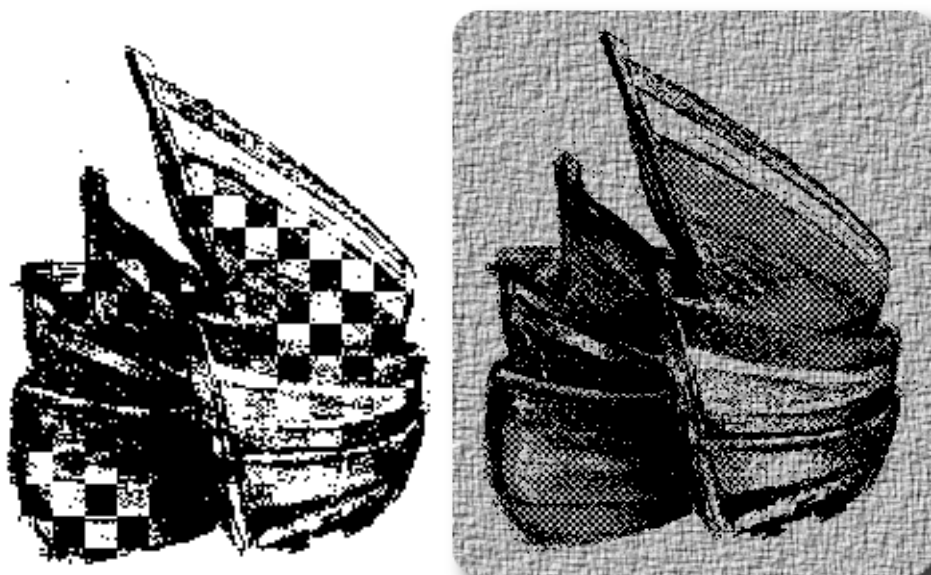
Relacionado con las colecciones de fotos, quizás sea interesante saber cómo convertir los nombres que les dan a las capturas las cámaras digitales a un nombre más significativo. Yo utilizo

```
[Usuario@Máquina]$ jhead -n *
```

que crea los nombres de todas las fotos de un subdirectorio a partir de la fecha y hora en que se tomaron, almacenada en la cabecera EXIF. Evidentemente *1225-022901.jpg* (25 de diciembre, casi a las 2:30) da más información que *img\_0221.jpg*.

## 10.3 Edición de bitmaps

De entre la multitud de aplicaciones diferentes vamos a explorar someramente las posibilidades de *The Gimp*.



Si piensa utilizar *The Gimp* no olvide instalar 'Grokking the Gimp' (paquete `grokking-the-gimp`), un completo curso de uso de este potente programa.

## 10.4 Edición vectorial

Las dos aplicaciones que generan gráficos vectoriales incluidas en Guadalinex son *sodipodi* y (mucho más completa) *OpenOffice Draw*.

# Capítulo 11

## Animaciones y videos

### 11.1 Animaciones

**Formatos:** `.gif` animados y `.png` animados (`.mng`),  
`.swf` (Flash)

### 11.2 Video

Aquí hay que distinguir entre *formatos* (que vienen a ser los vehículos en que se distribuyen los datos) y *codecs*, el algoritmo en que se almacena y habitualmente se comprime la información gráfica y sonora.

Los formatos más conocidos y usados son `.avi` (Microsoft Audio/Visual Interleaved), `.mov` (Apple QuickTime), y `.mpeg` (Motion Picture Experts Group file interchange format version 1). En cuanto a codecs el de más actualidad es el **Divx** en alguna de sus versiones. Para un listado más o menos exhaustivo, la salida de

```
[Usuario@Máquina]$ mplayer -ac help
```

y de

```
[Usuario@Máquina]$ mplayer -vc help
```

es apabullante:

```
61 audio & 157 video codecs
```

**Reproducción:** *xine* y sus variaciones *gxine*, *totem* (y *mplayer*, *ogle*, *vlc*)

Estos programas pueden reproducir ficheros localizados en el disco o en internet, VCD, SVCD y DVD (por supuesto si disponemos del hardware que lo haga posible).

La figura 11.1 es un ejemplo de un fotograma MPEG-PS de 320x240 puntos.

Y la figura 11.2, una captura de un DVD con versión original en inglés y subtítulos en castellano, lanzado con la siguiente orden:

```
[Usuario@Máquina]$ mplayer -vo xv dvd:///1 -alang en -slang es
```

**Edición:** *kino*

Para aprender cómo se importa video de una cámara digital, consulte la página 67.



Figura 11.1: Captura de video MPEG1



Figura 11.2: Captura de DVD

# Capítulo 12

## Sonido

- Formatos de sonido: wav, mp3, ogg
- el conversor universal: *sox*.
- *xmms*, el reproductor.

Para obtener información sobre cómo extraer digitalmente el contenido de un CD de música, consulte 17.1 en la página 67.

### 12.1 festival

*festival* es un programa que convierte texto escrito en voz, es decir que lee un texto 'en voz alta' (imagine usted las posibilidad didácticas que tiene esto).

```
[Usuario@Máquina]$ festival --language spanish --tts Texto_en_español
```

Los idiomas para los que tiene diccionarios (con distintas voces, masculinas y femeninas, y distintas calidades) son inglés, gaélico y español.

Hay varios interfaces gráficos que usan *festival*: *kong-speaker*, *gnopernicus* y *screader*.

**Parte IV**

**Comunicaciones**



# Capítulo 13

## Navegación Web

### 13.1 Mozilla

**Truco.** Puede librarse de las ventanas emergentes (*popups*) no deseadas, esas en las que se le cuele la publicidad y mucho más, configurando adecuadamente su *mozilla*: Editar ->Preferencias ->Privacidad y seguridad ->Ventanas emergentes.

### 13.2 Los plugins

#### 13.2.1 Plugins no libres (java, flash, acroread)

El plugin de flash no es libre. El problema es que los usuarios están habituados a él. ¿Puede ser sustituido?

Pues resulta que hay software libre para generar y reproducir animaciones flash, y plugins libres y formas de que cualquier aplicación se pueda utilizar como un plugin de mozilla. . .

- *OpenOffice 1.1* permite exportar a flash (creo: no lo he comprobado)
- hay un conjunto de bibliotecas que pueden utilizarse por varios lenguajes de programación (perl, python, c, php) para crear directamente animaciones desde los programas: libming
- *swf-player* es un reproductor de animaciones flash e incorpora un plugin para navegadores basados en mozilla (mozilla, firebird, galeon. . .)
- *tubsock* es otro reproductor libre
- *mozplugger* permite que cualquier aplicación actúe como un plugin (así cualquier reproductor vale)

# Capítulo 14

## Correo

- mutt

- Mozilla Mail, evolution

Para comprobar que funciona el correo nunca nunca mande un mensaje de prueba a una lista de distribución. Lo único que conseguirá es mala fama eterna. El truco está en mandárselo a <mailto:echo@rediris.es>, que se lo devolverá. Así podrá ver que su sistema de correo funciona.

### 14.1 Más allá

Firmas y encriptación con *gpg*

# Capítulo 15

## Comunicación en directo

### 15.1 IRC: comunicación en grupo

- *Mozilla Chatzilla, xchat*

### 15.2 De uno a uno: Mensajería Instantánea

Vamos a utilizar *gaim* para conectarnos a un servidor *jabber*.

### 15.3 Redes *peer to peer*

*mldonkey*

### 15.4 Videoconferencia

*gnomemeeting*

## **Parte V**

# **Aplicaciones didácticas**

Este tema da para un libro y no para una sección de un capítulo de una introducción general. De todas formas puede ser interesante esbozar un mapa del territorio, para que todos podamos explorarlo orientados desde el principio.

## Capítulo 16

# ¿Software educativo?

Lo primero que tenemos que preguntarnos es qué es software educativo. Por supuesto que un programa de dibujo es software educativo, o un procesador de textos con corrector ortográfico; probablemente éstos se usarán mucho más que otras aplicaciones que se conocen por tal denominación porque *sólo son* software educativo. Intentemos una clasificación:

- (a) Aplicaciones para que funcione la escuela (sistemas operativos, protocolos de red, programas de comunicaciones, de elaboración de horarios, de consulta y administración de datos...)
- (b) Aplicaciones para que una clase funcione mejor (sistema de seguimiento de asistencia y partes de incidencias, registro de la marcha del proceso docente/discente, sistemas de control y monitorización remotos del trabajo de los alumnos)
- (c) Programas de comunicaciones y entornos de trabajo colaborativo (wikis, cvs, pizarras electrónicas, chat, correo electrónico)
- (d) Programas y estándares de búsqueda, consulta y elaboración de la información
- (e) Programas y recursos para informar al resto de la comunidad educativa (padres, administración) de la marcha del proceso educativo
- (f) Documentos de texto libres (= distribuibles, modificables, copiables y reutilizables) que puedan utilizarse como cursos, unidades didácticas, fragmentos de unidades temáticas (objetivos, ejercicios, exposición...) o ayuda a la confección de unidades didácticas en alguno de los niveles de algún sistema educativo; aplicaciones para su publicación y difusión
- (g) Relación de experiencias y conclusiones de proyectos de uso de materiales libres; exposiciones sobre la naturaleza educativa del software libre en sí mismo y sobre sus modelos de desarrollo cooperativo<sup>1</sup>
- (h) Recursos audiovisuales libres en formatos libres que puedan utilizarse como apoyo a la práctica docente (bancos de imágenes, de sonidos, videos...)
- (i) Recursos documentales o aportaciones parciales a recursos documentales (diccionarios, enciclopedias)
- (j) Guías, consejos, sugerencias, ejercicios o trucos para la aplicación didáctica de las aplicaciones ofimáticas (editores, hojas de cálculo...) y de comunicación (navegación guiada — WebQuests...)
- (k) Aplicaciones específicas de ayuda al aprendizaje (simuladores, tutoriales, sistemas de ejercitación)
- (l) Herramientas de autor para la creación de aplicaciones didácticas (*jclick, squeak, malted, wims...*)

---

<sup>1</sup>Aportación de Pepe Grimaldos.

- (m) Lenguajes de programación en tanto que pueden utilizarse para crear aplicaciones didácticas
- (n) Juegos y guías de utilización didáctica de juegos (para idiomas, infantil, necesidades educativas especiales, etc.)
- (o) Entornos integrados de trabajo en el aula o a distancia (e-learning, nukes)

**Nota** No siendo fundamentalistas, quizás sea conveniente dar a conocer estas recopilaciones de software libre para Microsoft Windows:

- GNUWin II<sup>2</sup>
- [ftp://ftp.softwarelibre.ulpgc.es/SL\\_Win32/](ftp://ftp.softwarelibre.ulpgc.es/SL_Win32/)<sup>3</sup>

## 16.1 Aplicaciones incluidas en Debian

Sin *aptitude* la tarea de averiguar las aplicaciones incluidas en la distribución<sup>4</sup> es excesivamente ardua (cf. 19.2.2). *aptitude* sirve para instalar nuevas aplicaciones. Esta tarea es lógico que sólo pueda realizarla el administrador (¿qué pasaría si todo el mundo pudiera instalar y desinstalar paquetes?). Pero también sirve para que un usuario sin privilegios pueda examinar el listado de paquetes disponibles, clasificados por secciones (matemáticas, ciencias...)

**Ejercicio** Divididos por afinidades de asignaturas, examinar brevemente las aplicaciones disponibles que parece que podrían ser utilizadas en cada materia.

Si no tenemos acceso a *aptitude* tenemos como siempre otros recursos. Supongamos que queremos buscar programas que se refieran al sistema solar:

```
[Usuario@Máquina]$ apt-cache search "solar system"
space-orbit - A 3D space combat simulator
ssystem - 3D solar system simulator
xephem - An interactive astronomical ephemeris for X.
```

Me temo que vamos a tener que aprender inglés y a afinar las búsquedas. Si queremos tener información más completa sobre uno de estos paquetes ejecutaremos esta orden

```
[Usuario@Máquina]$ apt-cache show space-orbit
```

No podemos olvidarnos de hablar del proyecto Debian-edu<sup>5</sup>. Será de mucho interés y bastante exhaustivo examinar las dependencias de los paquetes junior-toys (lletters, xhanglider, xplanet), junior-gnome (drgenius y mathwar) y junior-kde (kgeo, kstars y kworldclock), junior-art (gimp, tuxpaint y xpaint), junior-games-sim (con juegos tan educativos como freeciv y lincity), junior-games-text (que llaman por ejemplo a juegos como el ahorcado en inglés), junior-typing (tipptrainer, tuxtype, typespeed, gtypist).

Otro proyecto relevante es el de KDE, incorporado al paquete kdedu, que en su versión actual aporta flashkard, kalzium, keduca, kgeo, khangman, kiten, klettres, kmessedwords, kmplot, kpercentage, kstars, ktouch, kverbos y kvotrain.

<sup>2</sup><http://gnuwin.epfl.ch/es/index.html>.

<sup>3</sup>[ftp://ftp.softwarelibre.ulpgc.es/SL\\_Win32/](ftp://ftp.softwarelibre.ulpgc.es/SL_Win32/).

<sup>4</sup>Evidentemente no me refiero a la subsección 'Aplicaciones educativas' del menú.

<sup>5</sup><http://www.debian.org/devel/debian-jr/index.es.html>.

## 16.2 Enciclopedias libres

<http://enciclopedia.us.es/>

<http://www.lencarta.org/>

<http://www.lin-carta.org/>

<http://es.wikipedia.org/wiki/Portada>

Recuerde que son herramientas cooperativas, con sus ventajas e inconvenientes. Los alumnos también pueden enviar aportaciones.

## 16.3 Enumeraciones de aplicaciones educativas

Existen en internet varios sitios que se dedican a la enumeración y en ocasiones a la evaluación del software educativo.

- En castellano
  - LinuxForKids ([http://www.linuxforkids.org/Spanish\\_index.html](http://www.linuxforkids.org/Spanish_index.html))
  - SoftwareLibreEducativo de GIEducAr (<http://www.lugli.org.ar/wiki/bin/view/Gleduwiki/SoftwareLibreEducativo>)
  - Linux Educativo en Internenes (<http://www.internenes.com/programas/categoria.php3?c=Linux+Educativo>)
- En inglés
  - Quizás la relación más importante de aplicaciones sea el Seul/Edu Educational Application Index (<http://richtech.ca/seul>)
  - la categoría 'Education' dentro del *FSF/UNESCO Free Software Directory* (<http://www.gnu.org/directory/education/>)
  - La recopilación del proyecto francés OfSet (Organization for Free Software in Education and Teaching, <http://www.ofset.org/freeduc/>)

## 16.4 Herramientas de autor

### 16.4.1 jclíc

*JClíc* ([http://www.xtec.es/recursos/clic/jclíc/index\\_esp.htm](http://www.xtec.es/recursos/clic/jclíc/index_esp.htm)), de Ramón Busquets.

Está a punto de terminarse y ya hay versiones de prueba. Al estar hecho en *java* debe de funcionar correctamente en GNU Linux.

### 16.4.2 squeak

*Squeak* es una herramienta multimedia que permite desarrollar contenidos.

Su sitio *web* es <http://www.squeak.org>, pero existe un grupo activo de adaptación al español<sup>6</sup>. Existe ya un paquete para Linux, que puede encontrarse en su repositorio (<http://cvs.linex.org/debian/>).

Se ha informado públicamente de que se está estudiando la incorporación de la versión en español a Guadalinex (mensaje del 5 de diciembre de 2003 a la lista [andared.averroes@juntadeandalucia.es](mailto:andared.averroes@juntadeandalucia.es)).

<sup>6</sup><http://squeak.linex.org/>.



### 16.4.3 WIMS

El WWW Interactive Mathematics Server<sup>7</sup> permite trabajar con matemáticas de forma interactiva en Internet. Pueden realizarse dos clases principales de actividades: ejercicios interactivos y herramientas computacionales o gráficas de ayuda a la resolución de problemas.

### 16.4.4 Descartes

Descartes<sup>8</sup> es una herramienta de trabajo para los profesores de matemáticas que deseen crear lecciones interactivas en el formato de páginas Web, ya sea para ser colocadas en un servidor de Internet o en el disco de un ordenador.

### 16.4.5 MALTED

MALTED (Multimedia Authoring for Language Tutors and Educational Development) es una herramienta de autor con licencia GPL que permite la creación de unidades interactivas, encaminada a la enseñanza de lenguas extranjeras.

Su página principal es <http://malted.cnice.mecd.es/> y sus responsables se están preocupando seriamente porque funcione correctamente en Linux (<http://sauce.cnice.mecd.es/cprc0001/malted/>). La primera beta fue presentada a la comunidad de software libre mediante mensaje de Carlos J. Medina a <[sl-educacionlistas.hispalinux.es](mailto:sl-educacionlistas.hispalinux.es)> de fecha 1 de agosto 2003 y en esta lista se ha informado de su desarrollo.

El 21 de enero de 2004 Carlos J. Medina ha anunciado una nueva versión

(...) en ella ya se incluye el Editor MALTED (dve) para modificar o crear unidades didácticas (...) Así mismo el Navegador MALTED (RTS) ha sido mejorado con un nuevo inicio que da acceso a las unidades didácticas del PC, de la red o de Internet

---

<sup>7</sup><http://wims.unice.fr/>.

<sup>8</sup><http://descartes.cnice.mecd.es>.

## **Parte VI**

# **Recursos materiales**

## Capítulo 17

# Utilización de los dispositivos

### 17.1 ¿Cómo se *ripea* un CD de música?

*Ripear* es en la jerga sacar la pista del CD de forma digital (porque siempre nos queda el recurso de grabarlo de la tarjeta de sonido), normalmente para volver a grabarla en otro CD con nuestra selección de canciones, o comprimirla a *.mp3* (o a su equivalente libre, porque *.mp3* tiene también problemas de licencia, que es *.ogg*). Por lo tanto vamos a pasar de *.cdda* (el contenido digital de una pista del CD) a *.wav* (sonido digital) y finalmente a *.ogg* (el sonido comprimido)<sup>1</sup>.

El programa que trae la Guadalinex es *grip*, y en mi casa (donde soy también superusuario) funciona perfectamente. El problema es que se necesitan permisos de superusuario (para ser más exactos, el problema es que *grip* accede al dispositivo `/dev/sr0`, y éste sólo puede ser leído por `root`).

¿Soluciones? O cambiar los permisos de `/dev/sr0` (se podría crear un grupo al que perteneciera por ejemplo el coordinador TIC y que sí pudiera leer y escribir en este dispositivo) o hacer que *grip* tenga el `id` de usuario a `root` (lo que es un peligro, pero desde luego ridículo si pensamos que por ejemplo las máquinas arrancan desde un CD); ambas cosas sólo puede hacerlas el superusuario.

### 17.2 ¿Cómo se utiliza la cámara de video digital?

Los Centros TIC disponen de una cámara de video digital. Es posible controlarla directamente desde el ordenador, y transferir los datos grabados utilizando el puerto Firewire (también llamado iLink y `ieee1394`) sin pérdida ninguna de calidad (recordemos que los datos están grabados de forma digital en la cinta DV).<sup>2</sup> Este proceso es sencillo si se ha configurado correctamente antes; vamos a aprender a averiguar qué pasa si algo no funciona.

Primero hay que comprobar que existen los dispositivos que controlan la cámara (supondremos en lo que sigue que tenemos un núcleo suficientemente actualizado).

```
[Usuario@Máquina]$ ls -l /dev/raw1394
[Usuario@Máquina]$ ls -l /dev/video1394
```

Si el núcleo reconoce la existencia de los dos dispositivos (el primero sirve para comunicarse con la cámara y el segundo para utilizar la entrada DV-IN de la misma; por supuesto este segundo dispositivo sólo será necesario en caso de que la cámara tenga entrada digital), podemos seguir adelante. En caso contrario necesitamos permisos de superusuario para crearlos:

<sup>1</sup>Existe la posibilidad de comprimir la muestra original sin pérdida de información y calidad pero con menor compresión; la aplicación se llama *flac* y la verdad es que no es muy usada.

<sup>2</sup>Por supuesto que también podemos pasar al ordenador las grabaciones de una cámara analógica: basta una entrada de video compuesto o *usb*, de las que proporcionan las sintonizadoras de televisión y algunas tarjetas de video.

```
[root@Máquina]# cd /dev
[root@Máquina]# ./MAKEDEV raw1394
[root@Máquina]# ./MAKEDEV video1394
```

¿Están cargados los módulos de control de la cámara? Veámoslo:

```
[usuario@Máquina]$ /sbin/lsmmod | grep 1394
```

En mi máquina devuelve lo siguiente<sup>3</sup>:

```
video1394 12632 0 (unused)
ohci1394 23016 0 [video1394]
ieee1394 42724 0 [video1394 ohci1394]
```

Si no estuvieran cargados (deberían cargarse al arrancar el ordenador, incluyendo en */etc/modules* *ohci1394* y eventualmente *video1394*), habrá que recurrir al superusuario:

```
[root@Máquina]# modprobe ohci1394
[root@Máquina]# modprobe video1394
```

El demonio *hotplug* nos informará de que el sistema ha reconocido la cámara al encenderla (claro, con el cable previamente conectado):

```
raw1394: /dev/raw1394 device initialized
```

Pero podemos tener problemas de permisos.

```
[Usuario@Máquina]# ls -l /dev/ raw1394
crw-rw---- 1 root disk 171, 0 Oct 9 00:24 /dev/raw1394
```

Bueno, pues si queremos poder trabajar con la cámara tendremos que ser el superusuario o pertenecer al grupo *disk*.

Ahora es cuando podemos utilizar *kino* o *dvgrab* para importar un video al ordenador.

---

<sup>3</sup>No repetiré más que *video1394* sólo será necesario si existe y se va a usar la entrada DV-IN de la cámara.

## **Parte VII**

# **Nociones fundamentales de UNIX**

# Capítulo 18

## Redes

### 18.1 Samba. Convivencia con Windows

Samba es la versión para GNU Linux del protocolo de redes de Microsoft.

Es probable que el Centro donde se imparta este curso haga un uso intensivo de redes Windows y por tanto del cliente y servidor Samba.

#### 18.1.1 A pelo

Para entender las cosas lo mejor es ver como funcionan por debajo. Vamos montar un ordenador Windows, supuestamente de la sala de profesores (supondremos que además no sabemos cómo se llama). Empezamos preguntando al que sabemos que está disponible por las máquinas de la red

```
[Usuario@Máquina]$ smbclient linux1
```

No devolverá entre otros datos el nombre de la máquina que nos interesa: 'PROFESORES1'. Ahora tenemos que ver qué recursos se han exportado en PROFESORES1:

```
[Usuario@Máquina]$ smbclient PROFESORES1
```

Vemos que está exportado el disco C bajo el nombre de "DISCO C" (dése cuenta del espacio y de cómo se protege en el ejemplo). ¿Cómo podemos montarlo? (lo siguiente en una línea; /mnt/profesores1 es un directorio arbitrario pero que debe existir)

```
[root@Máquina]# mount -t smbfs //PROFESORES1/DISCO\ C /mnt/profesores1
```

Ahora ya podemos utilizar el disco del ordenador de la sala de profesores como si fuera un disco más de nuestro ordenador: lo hemos incorporado a nuestro árbol de directorios.

Cuando hayamos acabado de trabajar, a desmontar el recurso:

```
[root@Máquina]# umount /mnt/profesores1
```

### 18.2 Creación de un fichero de texto en remoto

Vamos a crear en el subdirectorio Curso\_Linux el fichero s1.txt, donde usted deberá anotar en primer lugar para qué utiliza habitualmente el ordenador (escribir exámenes, navegar, jugar...) o bien cómo espera utilizarlo; y en segundo lugar resumirá brevemente sus impresiones sobre el curso y la necesidad y/o viabilidad del software libre.

### 18.2.1 Actividades previas

Hay que avisar de que se va a trabajar en dos ordenadores a la vez, y que hay que pasar ficheros de un ordenador a otro; que esto se hace mediante una especie de ‘lenguaje’ que tienen que hablar los dos ordenadores para entenderse y que se llama ‘protocolo’. Hay que explicar un poco las nociones más básicas de redes, los protocolos ssh, ftp y http.

Para lograr el objetivo, enseñar a los profesores a crear un fichero en su subdirectorio personal del servidor, se examinarán los dos métodos siguientes:

### 18.2.2 Creación local y subida con ftp

Podemos crear el fichero en local y subirlo al subdirectorio mediante un cliente de ftp.

Telee las siguientes instrucciones e intente comprender los mensajes que devuelve el ordenador. ¿Todas las instrucciones transcritas son necesarias?

```
[Máquina_local]$ mcedit s1.txt
[Máquina_local]$ ftp linux1
linux1 ftp> mkdir Curso_Linux
linux1 ftp> cd Curso_Linux
linux1 ftp> pwd
linux1 ftp> Curso_Linux
linux1 ftp> put s1.txt
linux1 ftp> dir
linux1 ftp> quit
```

### 18.2.3 Uso de un cliente ssh

La alternativa es usar un cliente ssh<sup>1</sup> para crear en el servidor el subdirectorio Curso\_Linux, colgando del espacio propio de cada alumno y trabajar directamente allí en la redacción del fichero.

```
[Usuario@Máquina_local]$ ssh -l {Usuario} linux1
linux1$ mkdir Curso_Linux
linux1$ cd Curso_Linux
linux1/Curso_Linux$ touch s1.txt
linux1/Curso_Linux$ mcedit s1.txt
```

Recuerde: para cerrar una conexión ssh se usará la orden `exit`.

---

<sup>1</sup>Recordemos que estamos utilizando una distribución sin configurar, en la que todos los usuarios se llaman ‘usuario’.

## Capítulo 19

# Nociones elementales de administración del sistema

### 19.1 ¿Por qué este capítulo?

En mi modesta opinión no se puede trabajar con una máquina de un sistema operativo avanzado (GNU Linux, \*BSD, Solaris, Windows NT) si no se distingue lo que es trabajar como administrador o con una cuenta de usuario, si no se saben montar nuevos dispositivos, matar procesos fallidos o si un fichero puede ser modificado por personas no autorizadas.

Todo esto nos lleva a la discusión de qué debe saber un usuario. Mi respuesta es que debe saber cuanto quiera saber (es absurdo poner techos por arriba), a partir de un mínimo de conocimientos que deben exigirse tras el proceso de formación: me parece irresponsable pensar que se van a utilizar los ordenadores para trabajar en clase y en las tareas burocráticas/administrativas y que luego no se exija un control de quién puede acceder a qué datos. Trabajamos con personas menores de edad; en tanto que personas y en tanto que menores es exigible la confidencialidad. Y es imprescindible una política de seguridad y de respaldos (backups) si se van a utilizar los ordenadores de una forma seria, para confeccionar y almacenar exámenes, para la elaboración de materiales, etc.

**Mi conclusión** Un usuario debe saber aquello que le permita asumir su responsabilidad en el uso de los ordenadores.

Estoy hablando de saber utilizar el ordenador de forma eficaz. El ordenador es un medio de comunicación, como el teléfono, como el coche; como ellos debe saberse utilizar, sin revantar el capó, sin saberse de memoria decenas de números de teléfono o cómo funcionan.

Pero iría un paso más. Actúo como activista del software libre y mi objetivo es su difusión. Pienso que no puede conseguirse la consolidación con el nuevo sistema operativo si los usuarios no lo instalan en su casa, en sus ordenadores personales; pienso también que los profesores no estaremos cómodos hasta que nos hayamos familiarizado en casa con las aplicaciones y los recursos. Y esto nos convierte en administradores de nuestro propio sistema, y nos obliga a comprender cuestiones de seguridad (¿cómo averiguar si están escaneando mis puertos?, ¿estoy actualizado en actualizaciones de seguridad?) y de configuración (¿qué protocolo usa el modem que me ha proporcionado mi proveedor de internet?)...

### 19.2 Actualizaciones e instalación de nuevo software

#### 19.2.1 Actualizaciones

Woody se instala desde 7 CDs. Ahora bien, esos CDs se crearon en agosto de 2002. ¿No se han descubierto errores ni fallos de seguridad entre los miles de programas que componen la distri-



bución?

Una distribución estable pasa por una fase de congelación, en la que no se añaden nuevas versiones ni características, y por una fase de corrección de errores; no sale una versión estable mientras ninguno de los programas incluidos tenga un error grave.

Pero por supuesto que sí se descubren fallos de seguridad. La forma que tiene Debian de corregir estos fallos es poner los paquetes actualizados en su servidor de seguridad, y el administrador no tiene más que seguir los siguientes pasos:

- Añadir esta línea  

```
deb http://security.debian.org/ woody
updates/main updates/contrib updates/non-free
```

(todo en una línea) a `/etc/apt/sources.list`.
- Ejecutar `apt-get update` para actualizar la base de datos
- Ejecutar `apt-get upgrade` para instalar las nuevas versiones.

De esta forma tendremos un sistema operativo, si no al día, seguro y estable.

## 19.2.2 Listado de paquetes disponibles

`aptitude` es tu amigo seas o no seas *root*. Te mostrará nos cosas: los paquetes disponibles y los paquetes instalados.

```

Actions Undo Options Views Help
f10: Menu ? : Help q: Quit u: Update g: Download/Install Pkgs
aptitude 0.2.11.1 Will use 22,2MB of disk space DL Size: 69,0MB
--- electronics - Programs for working with circuits and electronics
--- games - Games, toys, and fun programs
--- gnome
--- graphics - Utilities to create, view, and edit graphics files
--- hamradio - Software for ham radio operators
--- interpreters - Interpreters for interpreted languages
--- kde
--- libdevel
--- libs - Collections of software routines
--- mail - Programs to write, send, and route email messages
--- math - Numeric analysis and other mathematics-related software
--\ misc - Miscellaneous software
--\ contrib - Programs which depend on software not in Debian
c daemontools-installer <none> 0.76-8
p libtrain-bin <none> 0.9b-6
p med-bio-contrib <none> 0.3-1
p mtink <none> 0.9.55-1
p seti-applet <none> 0.4.1-3
Debian Med bioinformatics packages (contrib and non-free)
This meta package will install bioinformatics related Debian packages
for use in medical research.

These packages which will be installed by this package reside in
contrib or non-free.
```

Figura 19.1: Aptitude

Pero resulta que no está instalado en la versión que manejo, o sea que tenemos que recurrir a trucos.

Hay varias maneras de averiguar los paquetes disponibles, otras de averiguar las utilidades (incluidas las que todavía no han sido empaquetadas) disponibles. Puede ser útil para empezar una

exploración de

`http://linuxshop.ru:10000/linuxbegin/win-lin-soft-spanish/`

que nos informará de las utilidades que podemos buscar (en google, slashdot...)

Si seguimos sin tener disponible ninguna utilidad similar a *aptitude* tendremos que recurrir a las páginas de Debian<sup>1</sup> para conseguir información sobre cada paquete. Aconsejo explorar con detalle las secciones *Mathematics*, *Science* y *Text*.

¿Cómo averiguamos los paquetes instalados en nuestra máquina?

```
[Usuario@Máquina]$ dpkg --get-selections | less
```

nos mostrará la lista completa (para terminar, basta con pulsar `q`).<sup>2</sup>

Otro truco interesante es buscar si está instalada una utilidad determinada de la que sólo recordamos parte del nombre; buscaremos por ejemplo si está instalada *nautilus* supuniendo que sólo recordábamos la primera sílaba del nombre:

```
[Usuario@Máquina]$ dpkg --get-selections | grep nau
```

*grep* es una aplicación muy útil. Nos permite por ejemplo buscar una cadena de texto en todos los ficheros de un directorio:

```
[Usuario@Máquina]$ grep CADENA_QUE_BUSCA *
```

Devuelve el nombre del fichero y la línea donde aparece.

### 19.2.3 Instalación de un programa

Si el programa está empaquetado la cosa es sumamente fácil:

```
apt-get install PAQUETE.
```

El sistema se encarga de las dependencias y conflictos entre paquetes.

Desinstalar es igual de fácil:

`apt-get remove PAQUETE` ó `apt-get -purge remove PAQUETE`, si queremos que se borren los ficheros de configuración y datos que ha dejado el paquete. Lógicamente sólo *root* puede realizar estas operaciones. ¿Qué pasaría si cualquier usuario pudiera desinstalar cualquier paquete?

## 19.3 Control de usuarios y contraseñas

`adduser`, `deluser`

`passwd`

## 19.4 Demonios y configuración del sistema

```
[Usuario@Máquina]$ ps axww | less
```

Un vistazo a `/etc/init.d`.

Un vistazo a `/proc`.

<sup>1</sup><http://www.debian.org/Packages/unstable/>.

<sup>2</sup>El carácter que indica las *tuberías* se teclea con `AltGr + 1`.

## 19.5 Montando y desmontando dispositivos y sistemas de ficheros

Quién puede montar un dispositivo es configurable. Depende de una instrucción que se añade al fichero */etc/fstab*; si la línea que se refiere al dispositivo contiene la opción *user* cualquier usuario podrá montarlo. En caso contrario sólo *root* podrá hacerlo. Por ejemplo véase la salida pertinente al cdrom de la tabla *fstab*:

```
[Usuario@Máquina]$ cat /etc/fstab | grep cdrom
/dev/cdrom /cdrom iso9660 ro,user,noauto 0 0
```

Lo que nos informa de que el cdrom se va a montar en */cdrom*, que va a ser de solo lectura (*ro*) y podrá ser montado por los usuarios.

El demonio *automount* permite montar de forma semiautomática a los usuarios los recursos que aparecen en la citada tabla. La distribución facilita la operación en caso de que esté prevista por el sistema: a golpe de ratón se montan discos duros, CDs, etc. Pero queremos conocer qué se esconde debajo del capó, ¿no? Porque es posible que a alguien se le pase un dispositivo, ¿o esto no puede ocurrir?

### 19.5.1 ¿Cómo se utiliza un diskette?

Depende del administrador del sistema que usted pueda montar o desmontar diskettes. Vamos a suponer que se le permita montarlos (lo que conlleva problemas de seguridad) y que no se realice automáticamente.

Montar se refiere a añadir los datos del nuevo disco al árbol de directorios del sistema operativo; desde el momento en que un diskette se monta sus datos pueden consultarse y/o modificarse por cualquiera con permiso para ello; además se guarda una imagen en cache para acelerar las lecturas y escrituras. Nunca debe sacarse un diskette sin haberlo desmontado previamente, porque es posible que los últimos datos todavía no se hayan escrito en el mismo.

Supongamos que queremos utilizar un diskette formateado para Windows. La orden es (existen formas más simples de hacerlo pero dependen de que el administrador las haya preparado)

```
[root@Máquina]# mount -t vfat /dev/fd0 /mnt/dosa
```

La orden simplificada y al alcance de un usuario es (suponemos que */mnt/dosa* o el directorio donde se quiera montar el diskette existe)

```
[Usuario@Máquina]$ mount /mnt/dosa
```

El proceso inverso se conoce como desmontar el diskette. Se hace así

```
[Usuario@Máquina]$ umount /mnt/dosa
```

### 19.5.2 ¿Cómo se lee un cdrom?

Supongamos un CD normal, de datos; no un (S)VCD ni DVD (tienen otros sistemas de ficheros)

```
[root@Máquina]# mount -t iso9660 /dev/cdrom /cdrom
```

o bien (simplificado y si se le permite a los usuarios)

```
[Usuario@Máquina]$ mount /cdrom
```

Cuando acabe de usarlo, recuerde:

```
[Usuario@Máquina]$ umount /cdrom
```

## 19.6 Dispositivos usb

```
[root@Máquina]# mount -t vfat /dev/sda1 /mnt/usb/
```

o bien (simplificado, etc.)

```
[Usuario@Máquina]$ mount /mnt/usb/
```

## Capítulo 20

# Instalación de un sistema GNU Linux

### 20.1 Particiones

Como usted ya sabrá, Guadalinex se distribuye en discos vivos (*Live-CDs*), pero da la oportunidad de instalar el nuevo sistema operativo en su disco duro sin interferir en el funcionamiento de su sistema operativo anterior.

La herramienta que realiza este proceso es *qtparted*. *qtparted* cambia con éxito el tamaño de particiones NTFS (de Windows XP) y FAT32 (versiones anteriores de Windows), dejando espacio para formatear con un sistema de ficheros propio de Linux (aconsejo que sea ext3) la partición que se le va a dedicar.

### 20.2 Instalación

### 20.3 Localización

### 20.4 Creación de usuarios

*adduser*

### 20.5 Configuraciones generales: dispositivos...

#### 20.5.1 Configurando la red

En caso de que nuestra máquina no esté en una red que reciba las ip's de un servidor DHCP, deberemos asignarla manualmente. La distribución cuenta con herramientas gráficas que realizan esta tarea durante la instalación (y siempre disponemos de *etherconf*), pero si hemos cometido algún error deberemos saber cómo remediarlo. Un ejemplo y un poco de sentido común nos ayudará a hacerlo con éxito. Este es el contenido del fichero `/etc/network/interfaces`

```
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet static
address 192.168.0.4
netmask 255.255.255.0
```

```
network 192.168.0.0
broadcast 192.168.0.255
gateway 192.168.0.1
```

Seguidamente procederemos a reiniciar los servicios de red

```
[root@Máquina]# /etc/init.d/networking restart
```

Ahora, la orden

```
[Usuario@Máquina]$ /sbin/ifconfig
```

nos devolverá en la segunda línea nuestra ip. Ya tenemos red. Es el momento de aprovecharla para actualizar el sistema, para entre otras cosas cerrar los problemas de seguridad que se hayan podido presentar desde que se grabó el CD. Repase la sección 19.2.