Las Preguntas Frecuentes (FAQ) sobre Debian GNU/Linux

Susan G. Kleinmann (*sgk@debian.org*),
Sven Rudolph (*sr1@inf.tu-dresden.de*), and
Joost Witteveen (*joostje@debian.org*)

8 de julio de 1999

Este documento responde Preguntas Frecuentes (FAQ) sobre Debian GNU/Linux. (Ver la sección 1.4 (copyrights) para los detalles del Copyright.)

Contents

1 Meta-preguntas

1.1 ¿En qué formato está escrito este documento?

Este documento se escribió utilizando el DTD de Linuxdoc-SGML. El sistema Linuxdoc-SGML (ahora llamado SGML-Tools) puede utilizarse para crear ficheros en una variedad de formatos distintos, por ejemplo info de GNU, HTML, LaTeX, el formato .dvi de TeX, y ficheros PostScript (TM). SGML-Tools está disponible como paquete Debian.

1.2 ¿Dónde puedo conseguir la última versión [en inglés] de este documento?

La última versión de este documento puede verse en formato HTML en la *Página principal de Debian* `<http://www.debian.org/doc/FAQ/>`.

Si lo desea en formato HTML, PostScript, .dvi y texto 'plano' puede encontrarlo en el *servidor FTP de Debian* `<ftp://ftp.debian.org/debian/>` y en cualquiera de sus *réplicas* `<http://www.debian.org/distrib/ftplist.html>` en el directorio *doc/FAQ/*.

Los ficheros originales SGML que se utilizan para crear este documento se encuentran disponibles en el código fuente del paquete `doc-debian`. El paquete `sgml-tools` contiene herramientas que permiten transformar este documento a otros formatos, a petición del usuario: ficheros info de GNU, HTML, LaTeX, formato .dvi de TeX, y PostScript.

1.3 ¿A dónde debo enviar preguntas, correcciones, etc. sobre [la versión inglesa de] este documento?

Los comentarios sobre este documento son bienvenidos. Envíelos por favor mediante correo electrónico a *debian-faq@debian.org*.

1.4 ¿(Cómo) Puedo redistribuir este fichero?

(Nota del traductor: La traducción de los párrafos de esta subsección no está aprobada por la Free Software Foundation, y por lo tanto carece de valor legal a todos sus efectos. Se ha incluído a título didáctico para beneficio de los lectores hispanoparlantes que no pueden leer el original en inglés. Para cualquier duda sobre el estado técnico legal en cuanto a Propiedad Intelectual la única fuente de consulta es el original en inglés, que se transcribe a continuación de la traducción.)

Este documento tiene copyright 1996, 1997, 1998, 1999 por parte de SPI (vea la sección 15.3.1 (Software in the Public Interest), software de interés público).

Se otorga el permiso de distribuir copias literales de este documento siempre que se preserve en todas las copias el aviso de copyright y este aviso de permiso.

Se otorga el permiso de copiar y distribuir versiones modificadas de este documento bajo las condiciones de las copias literales, si se asegura que el trabajo derivado resultante completo se distribuye bajo los términos de una nota de permiso idéntica a esta misma.

Se otorga el permiso de copiar y distribuir traducciones de este documento en otro idioma bajo las condiciones de las versiones modificadas, excepto que esta nota de permiso puede ser una de las traducciones aprobadas por la Free Software Foundation en lugar de sus originales en inglés.

This document is copyright 1996, 1997, 1998 by SPI (see section 15.3.1 (Software in the Public Interest)).

Permission is granted to make and distribute verbatim copies of this document provided the copyright notice and this permission notice are preserved on all copies.

Permission is granted to copy and distribute modified versions of this document under the conditions for verbatim copying, provided that the entire resulting derived work is distributed under the terms of a permission notice identical to this one.

Permission is granted to copy and distribute translations of this document into another language, under the above conditions for modified versions, except that this permission notice may be included in translations approved by the Free Software Foundation instead of in the original English.

1.5 ¿Cuáles son las fuentes de información que se utilizaron para obtener las preguntas y respuestas que se han presentado aquí?

Esta lista estaba mantenida por J.H.M. Dassen (*jdassen@wi.LeidenUniv.nl*) y Chuck Stickelman (*stick@richnet.net*).

Partes de la información provienen de

- El *anuncio de publicación de Debian-1.1* `<http://www.debian.org/1.1/release.html>` de Bruce Perens.
- La *FAQ de Linux* `<http://www.cl.cam.ac.uk/users/iwj10/linux-faq/>` de *Ian Jackson* `<http://www.cl.cam.ac.uk/users/iwj10/ijackson.html>`.
- Los *archivos de la lista de correo de Debian* `<ftp://ftp.debian.org/debian/debian-lists/>`

- 13.1 (El manual del programador de dpkg y el manual de normas de Debian)
- Los muchos desarrolladores, voluntarios y probadores beta.
- Los descascarados recuerdos de sus autores.

Los autores quieren agradecer a todos aquellos que hicieron posible este documento.

1.6 ¿Dónde puedo conseguir la última versión de esta traducción?

Esta traducción se distribuye en el paquete Debian denominado `doc-debian-es`, que se encuentra en la sección `doc`.

1.7 ¿A dónde debo enviar preguntas, correcciones, etc. sobre esta traducción?

Esta traducción ha sido realizada por el grupo de hispanización de Debian. Para ponerse en contacto con sus miembros, por favor utilice la lista de correo electrónico *debian-l10n-spanish@lists.debian.org*.

Si desea colaborar con nosotros, puede subscribirse a dicha lista de correo electrónico enviando un mensaje a *debian-l10n-spanish-REQUEST@lists.debian.org* con la palabra *subscribe* en el cuerpo del mensaje.

Esta traducción ha sido llevada a cabo por:

- César Ballardini (*cballard@santafe.com.ar*),
- Ricard Lluis Fuertes Mulet (*richy@millorsoft.es*),
- Luis Francisco González (*luisgh@cogs.susx.ac.uk*),
- Nicolás Lichtmaier (*nick@Feedback.com.ar*),
- Juan Ignacio Llona (*jillona@jet.es*),
- Carlos Martínez Txakartegi (*txakar@redestb.es*) y
- Enrique Zanardi (*ezanard@debian.org*).

2 Definiciones y un vistazo

2.1 ¿Qué es Debian GNU/Linux?

Debian GNU/Linux es una *distribución* en particular del sistema operativo Linux, junto a numerosos paquetes que funcionan sobre él. En principio, los usuarios pueden obtener el núcleo Linux a través de Internet o a partir de otras fuentes, y compilarlo ellos mismos. De la misma manera, pueden obtener el código fuente de muchas aplicaciones, compilar los programas y luego instalarlos en el sistema. En el caso de programas complicados, este proceso puede no sólo llevar un montón de tiempo, sino que además está sujeto a errores. Para evitarlo, los usuarios con frecuencia eligen una distribución de Linux, que les provea tanto el sistema operativo como los paquetes de aplicación. La diferencia que existe entre las distintas distribuciones de Linux radica en el software, protocolos, y prácticas que utilizan para empaquetar, instalar, y seguir la pista a los paquetes de aplicación cuando están instalados en el sistema del usuario; además de las herramientas de instalación y mantenimiento, documentación, y otros servicios.

Debian GNU/Linux es el resultado de un esfuerzo voluntario para crear un sistema operativo libre, de alta calidad, compatible con Unix, acompañado por una pléyade de aplicaciones. La idea de un sistema libre de estilo Unix se origina en el proyecto GNU, y muchas de las aplicaciones que hacen a Debian GNU/Linux tan útil fueron desarrolladas por dicho proyecto.

Para Debian, la palabra inglesa "free" tiene el significado de GNU (véanse las *Directrices sobre Software Libre de Debian (DFSG)*). Cuando hablamos de software libre, nos referimos a la libertad, no al precio. Software libre significa que usted tiene la libertad de distribuir copias de software libre, que usted recibe el código fuente o puede obtenerlo si lo desea, que puede cambiar el software o usar partes de él en nuevos programas libres; y que usted sabe que puede hacer todo esto.

Debian fue creado por Ian Murdock en 1993, inicialmente bajo el patrocinio del proyecto GNU de la Free Software Foundation. En la actualidad, los desarrolladores de Debian consideran su trabajo como un descendiente directo del proyecto GNU. Debian GNU/Linux es

- **Completo**: Debian incluye más de 2200 paquetes de software en este momento. Los usuarios pueden seleccionar qué paquetes instalar; Debian provee una herramienta para ese fin. Encontrará una lista con las descripciones de los paquetes actualmente disponibles con Debian en cualquiera de los *sitios réplica* de Debian.
- **libre para utilizar y redistribuir**: No se requiere ninguna clase de cuota para ser socio de ningún consorcio, ni pago solicitado para participar en su distribución y desarrollo. Todos los paquetes que formalmente son parte de Debian GNU/Linux son libres para ser redistribuidos, normalmente bajo los términos especificados por la Licencia Pública General de GNU.

  Los archivos FTP de Debian también tienen aproximadamente 400 paquetes de software (en los directorios `non-free` y `contrib` de los archivos FTP), los cuales se distribuyen bajo términos específicos que se incluyen con cada paquete.
- **dinámico**: Con alrededor de 300 voluntarios constantemente contribuyendo con código nuevo y mejorado, Debian evoluciona rápidamente. Se planea realizar nuevas entregas cada varios meses, y los archivos FTP se actualizan diariamente.

Aunque Debian en sí mismo es software libre, es una base sobre la cual se pueden construir distribuciones de Linux con valor añadido. Al proveer un sistema base completo y fiable, Debian proporciona a los usuarios de Linux un alto grado de compatibilidad y permite a los creadores de distribuciones de Linux eliminar la duplicación de esfuerzos y enfocar su trabajo en aquellas cosas que hacen especial su distribución.

2.2 Está bien, ahora sé qué es Debian...¿¡qué es Linux!?

Abreviando, Linux es el núcleo de un sistema operativo estilo Unix. Originalmente se diseñó para PCs con 386/486/Pentium; en estos momentos están en etapa de desarrollo las versiones para otros sistemas, incluyendo sistemas multiprocesador. Linux fue escrito por Linus Torvalds, junto con muchos científicos computacionales de todo el mundo.

Además del núcleo, usualmente un sistema "Linux" tiene:

- un sistema de ficheros que cumple con la Norma para la Jerarquía del Sistema de Ficheros de Linux (Linux Filesystem Hierarchy Standard) `<http://www.pathname.com/fhs/>`.
- una gran variedad de utilidades Unix, muchas de las cuales han sido desarrolladas por el proyecto GNU y la Free Software Foundation (FSF).

La combinación del núcleo Linux, el sistema de ficheros, las utilidades de GNU y FSF, y las otras utilidades, se han diseñado para cumplir la norma POSIX (IEEE 1003.1); véase 4.2 (compatibilidad de código fuente).

Si desea más información acerca de Linux, vea la *INFO-SHEET (cartilla informativa)* <ftp://sunsite.unc.edu/pub/Linux/docs/HOWTO/INFO-SHEET> de Michael K. Johnson y la *META-FAQ* <ftp://sunsite.unc.edu/pub/Linux/docs/HOWTO/META-FAQ>.

2.3 ¿Cuál es la diferencia entre Debian y las otras distribuciones de Linux?

Las siguientes características clave distinguen a Debian de las otras distribuciones de Linux:

El sistema de mantenimiento de paquetes de Debian:

Todo el sistema, o cualquier componente individual, puede actualizarse sin reformatear, sin perder los ficheros de configuración con las personalizaciones, y (en la mayoría de los casos) sin rearrancar el sistema. La mayoría de las distribuciones de Linux disponibles en la actualidad tienen alguna clase de sistema de mantenimiento de paquetes; el 7 (sistema de mantenimiento de paquetes de Debian) es único y particularmente robusto.

Desarrollo abierto:

Mientras que otras distribuciones son el producto del desarrollo de individuos, pequeños grupos cerrados, o vendedores comerciales, Debian es la única distribución de Linux que está siendo desarrollada cooperativamente gracias al trabajo de muchos individuos a través de Internet, en el mismo espíritu de Linux y otros paquetes de software libre.

Más de 300 voluntarios encargados de mantener paquetes trabajan en más de 2200 paquetes y mejoran Debian GNU/Linux. Los desarrolladores Debian no escriben nuevo software (en la mayoría de los casos), sino que contribuyen empaquetando software existente de acuerdo a las normas del proyecto, comunicando los informes de bugs a los desarrolladores originales, y suministrando soporte a los usuarios. Puede encontrar más información sobre 15.1 (cómo convertirse en colaborador).

El Sistema de Seguimiento de Bugs:

La dispersión geográfica de los desarrolladores Debian requiere de herramientas sofisticadas y de una comunicación rápida de los bugs y sus enmiendas para acelerar el desarrollo del sistema. Se pide encarecidamente a los usuarios que envíen los bugs que encuentren en un estilo formal, y los mismos se hacen fácilmente accesibles a través de archivos en la WWW o mediante correo electrónico. Si lo desea, puede leer más acerca de esto en la FAQ de la administración del 14 (Registro de Bugs).

Las normas de Debian:

Solamente Debian dispone de unas especificaciones extensivas de nuestros estándares de calidad, las normas de Debian (`Debian Policy`). Este documento define las calidades y estándares bajo las cuales se mantienen los paquetes Debian.

2.4 ¿Qué relación tiene Debian con el proyecto GNU de la Free Software Foundation?

El sistema Debian se edifica sobre los ideales del software libre, defendidos primero por la Free Software Foundation <http://www.gnu.org/> y en particular por Richard Stallman. Las poderosas herramientas de desarrollo de sistemas, las utilidades y las aplicaciones de la FSF también son una parte clave del sistema Debian.

El proyecto Debian es una entidad aparte de la FSF, sin embargo nos comunicamos regularmente y cooperamos en varios proyectos. La FSF solicitó explícitamente que llamásemos "Debian GNU/Linux" a nuestro sistema, y nos hace muy felices cumplir con dicha solicitud.

El objetivo a largo plazo de la FSF es el desarrollo de un nuevo sistema operativo denominado GNU, basado en el HURD (<http://www.gnu.org/software/hurd/hurd.html>),

2.5 ¿Cómo se pronuncia Debian, y qué significa esa palabra?

En español, el nombre del proyecto es una palabra llana, así que se pronuncia "Dé-bian", con acento en la e. La palabra es la contracción de los nombres Debra e Ian Murdock, quienes fundaron el proyecto.

3 Requisitos de Hardware

3.1 ¿En qué arquitecturas/sistemas funciona Debian GNU/Linux?

Debian GNU/Linux incluye el código fuente completo de todos los programas incluidos, así que debería funcionar en todos los sistemas soportados por el núcleo Linux; ver la *FAQ sobre Linux* `<http://www.cl.cam.ac.uk/users/iwj10/linux-faq/>` para más detalles.

La actual entrega de Debian GNU/Linux contiene una distribución completa de ejecutables para las siguientes arquitecturas:

i386: ésta cubre PCs basados en procesadores compatibles Intel, incluyendo los 386, 486, Pentium, Pentium Pro y Pentium II de Intel y procesadores compatibles de AMD, Cyrix y otros.

m68k: Esto cubre amigas y ataris con un procesador Motorola 680x0 para x>=2; con MMU.

Actualmente se está trabajando en el desarrollo de distribuciones de binarios de Debian para las arquitecturas Alpha, Sparc, Sparc64, PowerPC, Arm y Hurd-i386.

3.2 ¿Qué hardware presuponen los discos de arranque suministrados con Debian GNU/Linux?

Los ficheros de configuración utilizados para construir el núcleo de la distribución estándar de Debian GNU/Linux presuponen una CPU 386, e incluyen soporte para tarjetas PCMCIA y SCSI para las cuales existan controladores en Linux. El soporte para tarjeta de red se consigue a través de módulos, por lo que no hay necesidad de compilar esos controladores en el núcleo.

3.3 ¿Cuánto espacio de disco es recomendable?

Una instalación generosa, suficiente para acomodar unos pocos usuarios, software para el sistema de ventanas X y varias aplicaciones grandes requeriría particiones de disco al menos de:

- 50 MBytes para el directorio raíz (/)
- 500 MBytes para /usr
- 50 MBytes para espacio de intercambio
- 50 MBytes para cada uno de los directorios de usuario
- 100 MBytes para /tmp
- 100 MBytes para /var

La cantidad óptima de espacio de disco utilizado para intercambio depende de forma crítica de la forma en la que el sistema será usado. Mucha gente pone el doble de espacio de disco del que tienen en RAM. Los sistemas con mucha RAM pueden no necesitar tanto espacio de intercambio, especialmente si sólo hay unos pocos usuarios. El proceso de instalación soporta sistemas sin espacio de intercambio.

3.3.1 Pero ¿cuánta RAM y espacio de disco son absolutamente esenciales?

Estos requisitos *mínimos* son suficientes para un sistema sin X11 y solamente 1 ó 2 usuarios:

- 22 MBytes de espacio de disco para el sistema `básico`; esto nos da un sistema Unix mínimamente funcional, pero no incluye aplicaciones y el soporte de red es mínimo.
- 140 MBytes de espacio de disco para un sistema estándar con aplicaciones que corran en la consola (esto es, sin interfaz X11).
- 190 MBytes de espacio de disco para un sistema estándar que corra aplicaciones con el sistema de ventanas X.
- 1250 MBytes o más para la totalidad de los programas opcionales (usted probablemente no necesite todos estos programas.)

Debian puede instalarse en sistemas con sólo 4 MBytes de RAM. Los últimos discos de instalación están especialmente organizados para dar un método simple de instalación en máquinas con poca memoria. Algunos usuarios nos han notificado sus éxitos usando Debian Linux para convertir PCs con RAM (y espacio de disco) limitada en terminales X. Se ha usado un sistema basado en un 386 con solamente 4MBytes de RAM y 40 MBytes de espacio de disco para correr Debian Linux de esta forma; esto es, tanto las funciones de red como de servidor X11 básico funcionaban satisfactoriamente. Este modo de operación funciona incluso si 1 MByte de la RAM se usa como un disco RAM cuando se arranca la máquina, lo que implica que sólo 3 MBytes de RAM son absolutamente esenciales para usar Debian Linux en un PC para usarlo como un servidor X. Este modo de operación requiere una partición de intercambio; sin ella, el sistema ni siquiera entrará en modo multiusuario.

3.4 ¿Cómo debo particionar mi disco?

Particionar un disco tiene la desventaja de que se tiene mucha menos flexibilidad para usar su espacio que en uno sin particionar. De cualquier forma, la mayoría de los usuarios encuentran que esa desventaja está más que superada por el hecho de que el daño a un sistema de ficheros en un disco particionado normalmente se limita a una sola partición. Aún más, las copias de seguridad de un disco particionado pueden ser gestionadas más fácilmente ya que los ficheros que cambian más a menudo es probable que estén localizados en una sola partición.

Un usuario con un disco de 1.6 GBytes concluyó tras una encuesta entre los usuarios de Debian que es razonable diseñar un esquema de particionado que siga de cerca el *Filesystem Hierarchy Standard (Estandar de Jeraquía de Sistema de Ficheros)* `<http://www.pathname.com/fhs/>`. Para su disco de 1.6 GByte, escogió estas particiones:

- 30 MBytes para el directorio raíz (/)
- 450 MBytes para `/usr`
- 50 MBytes para espacio de intercambio
- 1000 MBytes para directorios de usuario (parte de esto se puede usar para `/usr/local/`)
- 0 MBytes para `/tmp`; hizo `/tmp` un enlace simbólico a `/var/tmp`
- 40 MBytes para `/var`

Es posible usar un fichero de intercambio en lugar de una partición de intercambio. En cualquier caso, ésta suele ser una mala solución, por lo que recomendamos una partición de intercambio.

3.5 ¿Admite Debian discos muy grandes?

Hay un límite superior para el tamaño de la partición de disco que se usa para el arranque. Este límite se aplica a todos los sistemas operativos, no solamente a Linux. Básicamente, los BIOS típicamente disponibles en los PCs no pueden acceder a particiones mayores de 1024 cilindros o pistas. Por ello, *cualquier* sistema operativo usado en un PC no puede arrancarse desde una partición de disco mayor de 1 GByte. Merece la pena reseñar que esta restricción sólo se aplica a la partición desde la que Linux arranca. Las otras particiones pueden ser mayores. Una solución a esta limitación puede ser emplazar el directorio `/boot/` (y normalmente toda la partición raíz) en su propia (muy pequeña) partición, completamente dentro de los primeros 1024 bloques del disco.

El soporte para grandes particiones que no sean de arranque varía según el controlador. Se puede encontrar información detallada en el *Large-Disk mini-HOWTO*.

El núcleo Linux incluye el controlador de disco Multi-Dispositivo ('md'), que permite la concatenación simple de discos (llamada modo lineal) o soporte para troceado (también conocido como RAID 0) a través de software.

3.6 ¿(Cómo) Provee Debian soporte PCMCIA?

Las utilidades que proveen servicios para tarjetas PCMCIA han sido desarrolladas por David Hinds. Estas utilidades se suministran por Debian en el paquete `pcmcia-cs-KKK_VVV-RRR.deb`, donde los componentes 'VVV' y 'RRR' siguen las convenciones usuales de 7.3 (nombres de paquetes Debian), y el componente 'KKK' se refiere a la versión del núcleo para la cual el paquete `pcmcia-cs` ha sido construido.

El paquete `pcmcia-modules-KKK` debe ser reconstruido en aquellos sistemas que no usen el núcleo original de Debian. El paquete `pcmcia-source_VVV-RRR.deb` se provee para aquellos usuarios que necesiten recompilar los módulos o utilidades PCMCIA. Este paquete desempaqueta los ficheros fuente de las utilidades PCMCIA en el directorio `/usr/src/modules/pcmcia-cs/`. Ver el fichero `/usr/src/modules/pcmcia-cs/README.gz`; contiene instrucciones de cómo reconstruir los paquetes PCMCIA.

Las tarjetas PCMCIA que incluyen unidades IDE deben ser soportadas por el núcleo. La versión del núcleo distribuida con los discos de instalación de Debian incluye soporte para las citadas tarjetas PCMCIA. Esto es, su fichero `.config` incluye la línea: `CONFIG_BLK_DEV_IDE_PCMCIA=y`.

4 Cuestiones de Compatibilidad

4.1 ¿Cuán compatible es Debian con otras distribuciones de Linux?

Los desarrolladores de Debian se comunican con otros creadores de distribuciones de Linux, en un esfuerzo para mantener la compatibilidad binaria entre las distintas distribuciones. La mayoría de los productos comerciales corren tan bien sobre Debian como lo hacen sobre los sistemas en los cuales se compilaron.

Debian GNU/Linux se adhiere a la Estructura de Sistema de Ficheros Linux *(Linux File System Structure)* `<http://www.pathname.com/fhs/>` Sin embargo, hay espacio para la interpretación en algunas de las reglas que componen esta norma, por lo cual puede haber diferencias entre un sistema Debian y otros sistemas Linux. La versión más moderna de este estándar FSSTND se llama ahora FHS y tenemos pensado pasarnos a ella en breve.

4.2 ¿Cuán compatible es Debian en código fuente con otros sistemas Unix?

El código fuente Linux de la mayoría de las aplicaciones es compatible con otros sistemas Unix. Casi todo lo que está disponible en los sistemas Unix System V, y en los sistemas comerciales o libres derivados de BSD, también puede funcionar en Linux. Sin embargo, tal aserción no tiene ningún valor en el mercado comercial de Unix, pues no hay manera de probarla. En el ámbito del desarrollo de software, se requiere una compatibilidad completa, y no sólo en "casi todos" los casos. Por ese motivo, hace algunos años surgió la necesidad de normas, y en la actualidad la POSIX.1 (IEEE Standard 1003.1-1990) es una de las normas más importantes sobre compatibilidad de código fuente en los sistemas operativos al estilo Unix.

Linux está encaminado a adherirse a POSIX.1, pero las normas POSIX cuestan dinero, y la certificación POSIX.1 (y FIPS 151-2) es bastante cara; esto ha hecho más difícil el trabajo de los desarrolladores de Linux para obtener completa conformidad POSIX. Los costes de certificación hacen poco probable que Debian obtenga un certificado de conformidad oficial, aún cuando complete satisfactoriamente el conjunto de pruebas de validación. (El conjunto de pruebas de validación puede obtenerse libremente en la actualidad, así que se espera que más gente trabaje en los temas de POSIX.1.)

Unifix GmbH (Braunschweig, Alemania) desarrolló un sistema Linux que ha sido certificado como conforme a la norma FIPS 151-2 (que es un superconjunto de POSIX.1). Esta tecnología estuvo disponible en la propia distribución de Unifix, denominada Unifix Linux 2.0 y en la distribución Linux-FT de Lasermoon.

4.3 ¿Puedo utilizar los paquetes Debian (los ficheros ".deb") en mi sistema Linux RedHat/Slackware/...? ¿Puedo utilizar los paquetes RedHat (los ficheros ".rpm") en mi sistema Debian GNU/Linux?

Las diferentes distribuciones de Linux utilizan distintos formatos de paquete y distintos programas de administración de paquetes.

Usted probablemente puede:

Hay disponible un programa que permite desempaquetar un paquete Debian en una máquina que se ha instalado con una distribución 'extraña', y generalmente funciona, en el sentido en que desempaquetará los ficheros. Lo contrario posiblemente es cierto también, esto es, un programa que desempaquete un paquete RedHat o Slackware en una máquina basada en Debian Linux probablemente tendrá éxito desempaquetando el paquete y colocando la mayoría de los ficheros en los directorios deseados. La causa de este éxito puede encontrarse mayormente en la existencia de (y la amplia adhesión a) la Norma de Sistema de Ficheros Linux (Linux File System Standard).

Usted probablemente no quiera:

La mayoría de los administradores de paquetes escriben ficheros administrativos cuando se utilizan para desempaquetar un archivo. Dichos ficheros administrativos en general no están estandarizados. Por lo tanto, el efecto de desempaquetar un paquete Debian en un entorno 'extraño' puede tener efectos impredecibles (ciertamente no muy útiles) sobre el administrador de paquetes de dicho entorno. De la misma manera, las utilidades provenientes de otras distribuciones pueden tener éxito al desempaquetar sus archivos en un sistema Debian, pero probablemente causarán un fallo del sistema de administración de paquetes de Debian cuando llegue el momento de actualizar o borrar algunos paquetes, o incluso si se desea simplemente un informe exacto de los paquetes presentes en el sistema.

El mejor camino:

La Norma de Sistema de Ficheros Linux (y por lo tanto Debian GNU/Linux) requiere que los subdirectorios bajo `/usr/local/` se usen enteramente a discreción del usuario. Así que los usuarios pueden desempaquetar paquetes 'extraños' en este directorio, y luego administrar sus configuraciones, actualizarlos y borrarlos individualmente.

4.4 ¿Es capaz Debian de ejecutar mis viejos programas "a.out"?

¿Todavía tiene un programa así?

Para *ejecutar* un programa cuyo binario está en formato `a.out` (i.e., QMAGIC o ZMAGIC),

- Asegúrese de que su núcleo tenga soporte `a.out` incluido, ya sea directamente (CONFIG_BINFMT_AOUT=y) o como módulo (CONFIG_BINFMT_AOUT=m). (el paquete kernel-image de Debian contiene el módulo `binfmt_aout`.) Si su núcleo soporta binarios `a.out` mediante módulos, entonces debe asegurarse de que esté cargado el módulo `binfmt_aout`. Puede hacer esto en el momento del arranque si tiene una línea que diga `binfmt_aout` en el fichero `/etc/modules`. Puede también hacerlo desde la línea de órdenes si ejecuta `insmod NOMBREDIR/binfmt_aout.o` donde `NOMBREDIR` es el nombre del directorio donde se almacenan los módulos construídos para la versión del núcleo que está corriendo. En un sistema con la versión 2.0.36 del núcleo es probable que `NOMBREDIR` sea `/lib/modules/2.0.36/fs/`.
- Instale el paquete `libc4`. En el momento en el que se salga Debian 2.0, es muy probable que el paquete `libc4` haya sido eliminado de la distribución. Si este es el caso, usted podría querer buscar un CD-ROM de Debian antiguo (Debian 1.3.1 todavía tenía este paquete).
- Si el programa que desea ejecutar es un cliente X `a.out`, entonces instale también el paquete `xcompat`.

Si tiene una aplicación comercial en formato `a.out`, puede ser el momento de pedir que le envíen una actualización a formato `ELF`.

4.5 ¿Puede Debian ejecutar mis programas libc5 antiguos?

Sí. Simplemente, instale las bibliotecas necesarias de la sección `oldlibs`, que se incluyen en Debian 2.0 para conseguir compatibilidad con aplicaciones antiguas.

4.6 ¿Puede utilizarse Debian para compilar programas libc5?

Sí. En Debian, el sistema puede estar completamente actualizado a libc6 (incluyendo los paquetes de desarrollo), y aún así es posible desarrollar programas para libc5. Simplemente instale los paquetes `-altdev` necesarios. El conjunto mínimo de paquetes que necesitará es: `altgcc`, en la sección `devel` y `libc5-altdev` en la sección `oldlibs`.

Entonces tiene que colocar las herramientas `libc5` antes que las normales en el PATH. Esto es, ejecute la orden `export PATH=/usr/i486-linuxlibc1/bin:$PATH` (Esto no es esencial, sólo ventajoso). Si sólo lo va a utilizar una vez, sería suficiente con: `PATH=/usr/i486-linuxlibc1/bin:$PATH make [objetivo]`.

4.7 ¿Cómo debería instalar un paquete que no es Debian?

Los ficheros bajo el directorio `/usr/local/` no están bajo el control del sistema de administración de paquetes de Debian. Por lo tanto, es una buena práctica colocar el código fuente de su programa en `/usr/local/src/`. Por ejemplo, puede extraer los ficheros de un paquete denominado "fu.tar" dentro del directorio `/usr/local/src/fu`. Después de compilarlo, coloque los ejecutables en `/usr/local/bin/`, las bibliotecas en `/usr/local/lib/`, y los ficheros de configuración en `/usr/local/etc/`.

Si sus programas y/o ficheros realmente deben situarse en algún otro directorio, aún puede colocarlos bajo el directorio `/usr/local/`, y crear los enlaces simbólicos en los lugares necesarios que apunten al lugar de instalación en `/usr/local/`, por ejemplo, puede crear el enlace

```
ln -s /usr/local/bin/fu /usr/bin/fu
```

En cualquier caso, si el paquete que obtuvo permite la redistribución en su licencia, debería considerar la posibilidad de construir un paquete Debian de él, y enviarlo para que forme parte del sistema Debian. Las instrucciones para convertirse en desarrollador de paquetes se encuentran en el 13.1 (manual de normas de Debian).

4.8 ¿Por qué no puedo compilar programas que requieren libtermcap?

Debian usa la base de datos `terminfo` y la biblioteca `ncurses` para manejar la interfaz del terminal, en lugar de la base de datos `termcap` y la biblioteca `termcap`. Los usuarios que compilen programas que requieran conocimiento de la interfaz del terminal deben reemplazar las referencias a `libtermcap` con referencias a `libncurses`.

Para dar soporte a los ejecutables que hayan sido enlazados con la biblioteca `termcap`, y de los cuales no tenga el código fuente, Debian proporciona un paquete denominado `termcap-compat`. Éste provee tanto el fichero `libtermcap.so.2` como `/etc/termcap`. Instale este paquete si el programa falla al ejecutarse con el mensaje de error "can't load library 'libtermcap.so.2'", o se queja de que no encuentra el fichero `/etc/termcap`.

4.9 ¿Por qué no puedo instalar AccelX?

AccelX usa la biblioteca `termcap` durante su instalación. Vea 4.8 (acerca de termcap) más arriba.

4.10 ¿Por qué me da el mensaje de error "Can't find libX11.so.6" cuando trato de ejecutar `fu`?

Este mensaje de error puede significar que el programa está enlazado con la versión `libc5` de las bibliotecas X11. En este caso necesitará instalar el paquete `xlib6`, de la sección `oldlibs`.

4.11 ¿Puedo instalar y compilar un núcleo sin 'retoques' específicos para Debian?

Sí. Pero debe comprender antes las reglas que tiene Debian acerca de los ficheros de encabezado (headers).

Las bibliotecas C de Debian se construyen con las últimas entregas *estables* de los ficheros de encabezado de `gcc`. Por ejemplo, la distribución Debian-1.2 usó la versión 5.4.13 de los ficheros de encabezado. Esta práctica contrasta con los paquetes de fuentes del núcleo que se archivan en todos los sitios FTP de Linux, que utilizan versiones aún más nuevas de los ficheros de encabezado. Los encabezados del núcleo que se distribuyen junto con los fuentes de dicho núcleo se encuentran alojados en `/usr/include/linux/include/`.

Si necesitara compilar un programa con encabezados de núcleo más nuevos que los provistos por `libc6-dev`, entonces tendrá que agregar `-I/usr/src/linux/include/` a la línea de órdenes al compilar. Esto sucede en una situación, por ejemplo, con el paquete del demonio de automontaje (automounter daemon) (`amd`). Cuando los nuevos núcleos cambian las cuestiones internas que manejan el NFS, `amd` necesita enterarse de tales cambios. Para ello requiere la inclusión de las últimas versiones de los encabezados del núcleo.

5 Software disponible en el sistema Debian

5.1 ¿Qué tipos de aplicaciones y software de desarrollo hay disponible para Debian GNU/Linux?

Al igual que muchas distribuciones de Linux, Debian GNU/Linux ofrece:

- Las aplicaciones más importantes para desarrollo de software, manipulación de archivos, y procesamiento de texto, incluyendo gcc, libc, g++, make, texinfo, el shell Bash y numerosas utilidades de Unix mejoradas,
- Perl y sus bibliotecas,
- TeX, LaTeX, y dvips,
- el sistema de ventanas X, que proporciona un interfaz gráfico de usuario a través de red, y numerosas aplicaciones X,
- un conjunto completo de otras aplicaciones de red, incluyendo servidores WWW, navegadores, y herramientas de desarrollo.

Más de 2200 paquetes, desde servidores de noticias y lectores hasta soporte de sonido, programas de fax, hojas de cálculo y bases de datos, programas de procesamiento de imágenes, comunicaciones, utilidades de correo y de redes, servidores Web, e incluso programas de Ham-Radio están incluidos en la distribución. Otros 400 conjuntos de programas están disponibles como paquetes Debian, aunque no se incluyen formalmente en la distribución debido a restricciones de la licencia.

5.2 ¿Quién ha escrito todo este software?

- Los *autores* de cada programa aparecen en los créditos, en el archivo `/usr/doc/PAQUETE/copyright`, donde hay que substituir PAQUETE por el nombre del paquete.
- Una lista de los *Responsables* que empaquetan el software para el sistema Debian GNU/Linux aparece en el 7.4 (archivo de control de Debian) que viene con cada paquete

5.3 ¿Cómo puedo conseguir una lista de los programas que se han empaquetado para el proyecto Debian?

Está disponible una lista completa en cualquiera de los *mirrors* de Debian.

El archivo `indices/Packages-Master-i386.gz` contiene una lista que incluye descripciones cortas de todos los paquetes disponibles para el 386 (o chips mas avanzados). El archivo `indices/Packages-Master-m68k.gz` contiene una lista similar de paquetes que están disponibles para ordenadores con CPUs Motorola 680x0.

El *interfaz WWW a los paquetes Debian* resume los paquetes existentes en unas veinte "secciones" del archivo de Debian, así como los últimos 10 paquetes añadidos

5.4 ¿Qué falta en Debian GNU/Linux?

Johnie Ingram mantiene una lista de los paquetes que se necesitan más urgentemente para Debian bajo el nombre "Work-Needing and Prospective Packages" Para más detalles, ver la sección 15.1 (Contribuir al proyecto Debian).

5.5 ¿Dónde está 'which'?

Esta es una pregunta histórica, ya sin importancia, pero que se incluye en esta FAQ dado que aún conserva el récord de ser la pregunta hecha más frecuentemente en las listas de correo de Debian.

La funcionalidad de la utilidad de Berkeley `which` se proporciona en el shell Bash mediante la orden interna `type`. Para encontrar la ruta de directorios para un ejecutable "cualquiera", usar la orden `type cualquiera`. Para encontrar todas las rutas de directorios donde se puede encontrar a "cualquiera", usar `type -a cualquiera`.

Para compensar la falta de familiaridad de los usuarios con esta (en opinión del autor, oscura) característica del Bash, los responsables de Debian han construido un paquete llamado `debianutils`, que incluye un script de Bash llamado `which`. Este script incluye solo las siguientes líneas:

```
#!/bin/bash
for i in $*; do
  type -all -path $i | head -1
done
```

El paquete `debianutils` se almacena normalmente en la sección `base` del archivo de FTP, y por tanto forma parte de la instalación inicial de todo sistema Debian.

5.6 ¿(Cómo) Soporta Java Debian?

El Kit de Desarrollo de Java de Sun está disponible como paquete Debian (`jdk_VVV-RRR.deb`). El KDJ (JDK en inglés) permite ejecutar programas en Java y "applets", y escribirlos. Si el núcleo instalado está correctamente configurado (ver más abajo), el KDJ permitirá ejecutar programas en Java igual que otro tipo de ejecutables. El KDJ también incluye varios programas de demostración.

El núcleo de Debian está configurado con soporte para Java incluido como módulo (p.ej.,CONFIG_BINFMT_JAVA=m). Los usuarios que quieran construir su propio 11.1 (núcleo personalizado) pueden saltarse esto si lo prefieren. Una vez que el soporte para módulos está disponible en el núcleo, hay que asegurarse de que el módulo está cargado. Se puede hacer esto en tiempo de arranque insertando la línea `binfmt_java` en el fichero `/etc/modules`. Alternativamente, se puede instalar el módulo desde la línea de órdenes ejecutando la orden `insmod NOMBRE/binfmt_java.o`, donde `NOMBRE` es el nombre del directorio donde están almacenados los módulos compilados para la versión en ejecución del núcleo. En un sistema con la versión 2.0.36 del núcleo, `NOMBRE` será probablemente `/lib/modules/2.0.36/fs/`. Se puede comprobar si un módulo está cargado usando la orden `lsmod`.

Para ejecutar un "applet" en Java se necesita un navegador que sea capaz de reconocerlo y ejecutarlo. El navegador de Netscape que se puede instalar como paquete Debian puede usar "applets" en Java. (El código fuente del Netscape no es de acceso público. El paquete Debian del Netscape proporciona un asistente que ayuda en la instalación y gestión del Netscape en un sistema Debian. Este es un buen ejemplo de la integración de paquetes comerciales en el sistema Debian).

Una nota final con comentarios mixtos: Las políticas de licencias de Sun para el KDJ se están volviendo más restrictivas con el tiempo, así que este paquete puede no estar disponible como parte de Debian muy pronto. Es posible que esté disponible del mismo modo en que está disponible el Netscape para Debian GNU/Linux. Las buenas noticias son que hay actualmente varios programas en desarrollo que proporcionarán alternativas atractivas y de dominio público.

5.7 ¿Dónde está pine?

Debido a su restrictiva licencia, está en la sección non-free. Además, dado que la licencia ni siquiera permite que se distribuyan ejecutables modificados, hay que compilarlo uno mismo a partir del código fuente y los parches de Debian. Para la comodidad de los que usan dselect, el código fuente de la versión Debian de pine está disponible en los paquetes `pine396-src` y `pine396-diffs`. En el directorio source del árbol `project/experimental` puede haber disponibles versiones de pine 4.x.

5.8 ¿Cómo puedo comprobar si estoy usando un sistema Debian?

La existencia del programa `dpkg` demuestra que se pueden instalar paquetes Debian en el sistema.

Para asegurarse de que se ha instalado el sistema a partir de los auténticos discos base de Debian, se puede comprobar si existe `/etc/debian_version`.

5.9 ¿Cómo puedo saber la "versión" de Debian que estoy usando?

Hay un fichero, */etc/debian_version*, que contiene una única línea con el número de versión de la distribución, tal y como venía definido en el paquete `base-files`.

Los usuarios deben tener presente que el sistema Debian se compone de varias partes, las cuales pueden ser actualizadas de manera (casi) independiente. Cada "entrega" de Debian tiene contenidos inmutables y bien definidos. Las actualizaciones están disponibles de forma separada. Para una descripción de una sola línea sobre el estado de instalación del paquete `cualquiera`, se puede usar la orden `dpkg -list cualquiera`. (Sin argumentos, esta orden presenta las versiones de todos los paquetes instalados.) Para una descripción más exhaustiva, usar `dpkg -status cualquiera`.

5.10 ¿Cómo soporta Debian los idiomas no ingleses?

- Debian GNU/Linux se distribuye con asignaciones de teclado para casi dos docenas de teclados, y con utilidades (en el paquete `kbd`) para instalarlos, verlos y modificarlos. La instalación pide al usuario que especifique el teclado que va a usar.
- Se da soporte para las páginas de manual en francés, alemán, italiano y español a través de los paquetes `manpages-fr`, `manpages-de`, `manpages-it` y `manpages-es`. Para acceder a una de estas páginas de manual, el usuario debe cambiar el valor de LC_MESSAGES a la cadena de texto apropiada. En el caso de las páginas de manual en español, LC_MESSAGES ha de valer 'es_ES'. El programa `man` buscará entonces las páginas man en español en `/usr/man/es_ES/`.

5.11 ¿Qué hay sobre las limitaciones de exportación en los EE.UU.?

Las leyes de los EE.UU. plantean restricciones en la exportación de artículos de defensa, lo que incluye algunos tipos de software criptográfico. PGP y ssh se encuentran incluidos en esta categoría.

Para evitar que nadie tenga que correr riesgos legales innecesarios, ciertos paquetes Debian GNU/Linux solo están disponibles en un servidor fuera de los EE.UU. `<ftp://non-us.debian.org/pub/debian-non-US/>`, con una lista de espejos en `<ftp://non-us.debian.org/pub/debian-non-US/README.mirrors>`

6 Los archivos FTP de Debian

6.1 ¿Qué son todos esos directorios en los archivos de FTP de Debian?

El software que se ha empaquetado para Debian GNU/Linux está disponible en varios árboles de directorios de cada espejo de Debian.

El directorio `dists` contiene las "distribuciones", y es ahora la forma canónica de acceder a ellas.

6.2 ¿Cuántas distribuciones de Debian hay en el directorio `dists`?

Normalmente hay dos distribuciones, la distribución "stable" (estable) y la distribución "unstable" (inestable). A veces también hay una distribución "frozen" (congelada).

6.3 ¿Qué son todos esos nombres como slink, potato, etc.?

Son simplemente nombres clave. Cuando una distribución de Debian está en su fase de desarrollo, no tiene número de versión, solamente tiene un nombre clave. El objeto de estos nombres clave es hacer que las distribuciones de Debian sean más fáciles de replicar (si un directorio real tal y como `unstable` cambiara repentinamente de nombre a `stable`, habría que volver a traerse un montón de Megabytes de nuevo).

Actualmente, `stable` es un enlace simbólico a `slink` (o sea, Debian 2.1), y `unstable` es un enlace simbólico a `potato`, lo cual significa que `slink` es la distribución estable actual y `potato` es la distribución inestable actual.

6.4 ¿Y qué es eso de frozen?

Cuando la distribución unstable está suficientemente madura, se congela, y se crea una nueva unstable en el directorio `dists`, con un nuevo nombre clave. Pasados un mes o dos de pruebas, la distribución frozen se convierte en stable, es publicada, y la distribución stable anterior se convierte en obsoleta.

6.5 ¿Qué otros nombres clave se han utilizado ya?

Otros nombres clave que ya se han utilizado son: `buzz` para Debian 1.1, `rex` para Debian 1.2, `bo` para Debian 1.3.x, y `hamm` para Debian 2.0.

6.6 ¿De dónde provienen estos nombres clave?

Hasta ahora han sido personajes de la película Toy Story, de los estudios de animación Pixar.

6.7 ¿Y qué es "sid"?

Es una distribución especial para arquitecturas que todavía no han sido publicadas por primera vez.

Cuando sid no existía, la organización del FTP tenía un fallo: Había una presunción de que cuando se creaba una nueva arquitectura en unstable, sería publicada cuando esa distribución se convertía en la nueva stable. Para muchas arquitecturas este no es el caso, lo cual resultaba en que esos directorios tenían que ser cambiados de lugar en el momento de la publicación, desperdiciando un montón de ancho de banda.

Para esas arquitecturas todavía no publicadas, la primera vez que se publiquen habrá un enlace desde el stable actual hasta sid, y de ahí en adelante se crearán dentro del árbol unstable de forma normal. sid no será publicado nunca ni siquiera se accederá a él directamente, solamente a través de enlaces simbólicos en los árboles actuales stable, frozen y unstable. Será una mezcla de arquitecturas publicadas y no publicadas.

6.8 ¿De dónde procede el nombre de "sid"?

Sid era el chico de al lado que rompía los juguetes :-)

6.9 ¿Qué contiene el directorio stable?

- stable/main/: Este directorio contiene los paquetes que formalmente constituyen la distribución más reciente del sistema Linux de Debian.
- stable/non-free/: Este directorio contiene paquetes cuya distribución está restringida en un modo que obliga a que los distribuidores tengan muy presente ciertos requisitos de copyright. Por ejemplo, algunos paquetes tienen licencias que prohiben la distribución comercial. Otros pueden ser distribuidos, aunque de hecho son shareware, y no freeware. Las licencias de cada uno de estos paquetes debe ser estudiada, y posiblemente negociada, antes de que los paquetes se incluyan en cualquier redistribución (p.ej., en un CD-ROM).
- stable/contrib/: Este directorio contiene paquetes que son de *libre distribución*, pero que no cumplen con la política de requisitos de distribución de los paquetes del Proyecto Debian por alguna razón, p.ej., los paquetes tienen alguna modificación o restricción inusual, o sólo están disponibles en formato binario. Para estos paquetes, el proyecto no puede ofrecer al usuario ninguna forma de asegurarse de que están libres de Caballos de Troya, y no puede adaptarlos a otras arquitecturas. Paquetes sólo en binario que no sean de libre distribución se encuentran en el directorio `non-free`.

6.10 ¿Qué contiene el directorio unstable?

unstable contiene una muestra del sistema actual bajo desarrollo. Se invita a los usuarios a probar estos paquetes, aunque se les advierte sobre su estado incompleto.

También hay directorios main, contrib y non-free dentro de unstable.

6.11 ¿Qué son todos esos directorios dentro de `dists/stable/main`?

Dentro de cada uno de los directorios principales (`dists/stable/main`, `dists/stable/contrib`, `dists/stable/non-free`, y `dists/unstable/main/`, etc. pero no `project/experimental/`, que es demasiado pequeño para ser subdividido), los paquetes binarios residen en subdirectorios cuyos nombres indican la arquitectura del chip para el que fueron compilados:

- binary-all, para paquetes que son independientes de la arquitectura. Esto incluye, por ejemplo, scripts en Perl.
- binary-i386, para paquetes que sólo se ejecutan en máquinas 80x86.
- binary-m68k, para paquetes que se ejecutan en máquinas basadas en uno de los procesadores Motorola 680x0. Actualmente se mantiene para computadoras Atari y Amiga, y para algunas placas industriales basadas en VME. No hay una implementación de Linux para los Macintosh basados en el antiguo m68k, porque Apple no suministró la información de hardware necesaria.

- binary-sparc/, para paquetes que se ejecutan en Sparcstations de Sun.
- binary-alpha/, para paquetes que se ejecutan en máquinas DEC de Alpha.
- binary-powerpc/, para paquetes que se ejecutan en máquinas PowerPC
- binary-arm/, para paquetes que se ejecutan en máquinas ARM.
- binary-hurd-i386/, para paquetes de GNU/Hurd que se ejecutan en máquinas 80x86.

6.12 ¿Dónde está el código fuente?

Se incluye código fuente para cualquier cosa en el sistema Debian. La mayor parte de los términos de licencia de los programas en el sistema *exigen* que se distribuya el código fuente junto con los programas, o que se incluya junto con los programas una oferta para entregar el código fuente.

Normalmente, el código fuente se distribuye en los directorios "source", que son paralelos a todos los directorios binary de cada arquitectura.

El código fuente de los paquetes en los directorios "contrib" y "non-free" puede estar disponible o no, ya que formalmente no son parte del sistema Debian.

6.13 Muy bien, ¿y los otros directorios que no están dentro de `dists`?

Existen los siguientes directorios suplementarios:

- "tools": Utilidades DOS para crear discos de arranque, particionar el disco duro, comprimir y descomprimir archivos, y arrancar Linux.
- "doc": Documentación e instrucciones sobre cómo enviar informes sobre bugs.

De interés particular para desarrolladores:

- "project/experimental/": Este directorio contiene paquetes y utilidades que se están desarrollando específicamente para el Proyecto Debian, y que aún están en fase de pruebas. Se invita a los usuarios a probar estos paquetes, aunque se les advierte sobre su estado incompleto.
- "private/project/Incoming/": Paquetes enviados por desarrolladores, pendientes de ser clasificados en la jerarquía de directorios por el responsable de la distribución.
- "indices/": Varias listas, incluyendo el archivo `Packages-Master`.

7 Conceptos básicos del Sistema de Gestión de Paquetes de Debian

7.1 ¿Qué es un paquete Debian?

Los paquetes normalmente contienen todos los archivos necesarios para implementar un conjunto de órdenes o características relacionadas. Hay dos tipos de paquetes Debian:

- *paquetes Binarios*, que contienen ejecutables, archivos de configuración, páginas man/info, información de copyright, y otra documentación. Estos paquetes se distribuyen en un 7.2 (formato de archivo) específico de Debian; se distinguen normalmente por tener una extensión de archivo '.deb'. Los paquetes binarios se pueden desempaquetar usando la utilidad Debian `dpkg`; se dan detalles en su página de manual.

- *Paquetes Fuente*, que consisten en un archivo `.dsc` que describe el paquete fuente (incluyendo los nombres de los siguientes archivos), un archivo `.orig.tar.gz` que contiene el código fuente original sin modificar en formato tar comprimido con gzip, y usualmente un archivo `.diff.gz` que contiene los cambios específicos de Debian al código fuente original. La utilidad `dpkg-source` empaqueta y desempaqueta los archivos fuente de Debian; se dan detalles en su página de manual.

La instalación de software por el sistema de paquetes usa "dependencias" que son cuidadosamente diseñadas por los encargados del paquete. Estas dependencias están documentadas en el archivo de `control` asociado con cada paquete. Por ejemplo, el paquete que contiene el compilador de C de GNU (`gcc`) "depende" del paquete `binutils` que incluye el enlazador y el ensamblador. Si un usuario intenta instalar `gcc` sin haber instalado antes `binutils`, el sistema de paquetes de Debian dará un mensaje de error avisando de que también necesita `binutils`, e instalará `gcc` sólo si el usuario acepta instalar `binutils` primero. (Sin embargo, esta opción puede ser anulada por el usuario persistente). Ver más 7.9 (detalles sobre dependencia de paquetes) abajo.

Las utilidades de paquetes de Debian se pueden usar para:

- manipular y gestionar paquetes o partes de paquetes,
- ayudar al usuario en la fragmentación de paquetes que deben ser distribuidos a través de un medio de tamaño limitado, como disquetes,
- ayudar a los desarrolladores en la construcción de archivos de paquetes, y
- ayudar a los usuarios en la instalación de paquetes que residen en un servidor de FTP remoto.

7.2 ¿Cuál es el formato de un paquete Debian?

Un "paquete" Debian, o un archivo Debian, contiene los ejecutables, bibliotecas y documentación asociada con una forma particular de un programa o conjunto de programas relacionados. Normalmente un archivo Debian tiene un nombre que acaba en `.deb`.

Los detalles del formato de los paquetes binarios Debian se describen en la página de manual `deb`(5). Este formato interno está sujeto a cambios, así que use siempre `dpkg-deb`(8) para manipular archivos `.deb`.

7.3 ¿Por qué los nombres de los paquetes Debian son tan largos?

Los nombres de los paquetes binarios siguen la siguiente convención: <nombre>_<NúmeroDeVersión>-<NúmeroDeRevisiónDebian>.deb

Nótese que `nombre` se supone que es el nombre del paquete. Como prueba, se puede sacar el nombre de paquete asociado con un archivo Debian concreto (archivo .deb) de una de las siguientes formas:

- inspeccionar el archivo "Packages" en el directorio donde se guardaba el paquete en un servidor de FTP de Debian. Este archivo contiene un párrafo describiendo cada paquete; el primer campo del párrafo es el nombre formal del paquete.
- usar la orden `dpkg -info nombre_VVV-RRR.deb`. Esto manda un mensaje a STDOUT que da, entre otras cosas, el nombre formal del paquete.

El componente `VVV` es el número de versión especificado por el desarrollador original. Aquí no hay estándares establecidos, así que el número de versión puede tener formatos tan distintos como "960428" y "2.7.2.l.3".

El componente `RRR` es el número de revisión Debian, y lo establece el desarrollador Debian (o un usuario individual, si decide construir el paquete él mismo). Este número corresponde al nivel de revisión del paquete debian (que incluye el Makefile específico de Debian, llamado `debian/rules`, así como el archivo de control Debian, normalmente llamado `debian/control`). De este modo, un nuevo nivel de revisión normalmente significa cambios en el Makefile de Debian, el archivo de control, los scripts de instalación o desinstalación, o en los archivos de configuración usados en el paquete.

7.4 ¿Qué es un archivo de control de Debian?

Se dan detalles sobre el contenido de un archivo de control en el *manual del programador de dpkg* `<ftp://ftp.debian.org/debian/doc/package-developer/programmer.ps.gz>`. Resumiendo, este es un ejemplo de archivo de control para el paquete hola:

```
Package: hola
Version: 1.3-13
Architecture: i386
Depends: libc5 (>= 5.2.18)
Installed-Size: 31
Maintainer: Ian Jackson <ian@chiark.greenend.org.uk>
Description: La clásica bienvenida, y un buen ejemplo
 El programa GNU hola produce saludo amistoso y familiar. Permite
 a los no programadores usar una herramienta clásica en la informática
 que de otro modo no tendrían disponible.
 .
 Hablando en serio; es un ejemplo de cómo se hace un paquete Debian.
 Es la versión Debian del programa 'hola mundo' del proyecto GNU (el
 cual es a su vez un ejemplo del proyecto GNU).
```

El campo Package informa del nombre del paquete. Este es el nombre mediante el cual el paquete se puede manipular con las utilidades de paquetes de Debian, y normalmente es parecido (aunque no necesariamente igual) que la primera cadena de caracteres del nombre del archivo Debian.

El campo version da a la vez el número de versión del desarrollador original y (como último componente) el nivel de revisión del paquete Debian de este programa, tal y como se describe en 7.3 ().

El campo Architecture especifica el procesador para el que fué compilado este binario en concreto.

El campo Depends da una lista de paquetes que tienen que estar instalados para poder instalar este paquete con éxito.

El campo Installed-Size indica cuanto espacio de disco ocupará el paquete instalado. Está pensado para que lo usen los programas de instalación para indicar si existe suficiente espacio de disco para instalar el programa.

El campo Maintainer da la dirección de correo electrónico de la persona actualmente responsable del mantenimiento del paquete.

El campo Description da un pequeño resumen de las características del paquete.

7.5 ¿Qué es un conffile de Debian?

Los conffiles son listas de archivos de configuración, normalmente guardadas en `/etc`, que el sistema de mantenimiento de paquetes no sobrescribe cuando se actualiza un paquete. Esto asegura que los valores locales de los contenidos de estos archivos se conserven, y es una característica imprescindible para permitir la actualización de paquetes en un sistema en funcionamiento.

Para determinar exactamente qué archivos se conservan durante una actualización, ejecute`dpkg -status paquete`.

7.6 ¿Qué son los scripts Debian preinst, postinst, prerm, y postrm?

Estos archivos son scripts ejecutables que se usan automáticamente antes o después de que se instale un paquete. Junto con el archivo `control`, todos estos archivos forman parte de la sección de "control" de un archivo Debian.

Los archivos individuales son:

preinst

Este script se ejecuta antes de que se desempaquete el contenido del archivo Debian (".deb"). Muchos scripts 'preinst' detienen los servicios de los paquetes que se actualizan hasta que su instalación o actualización se completa (después de la ejecución exitosa del script 'postinst').

postinst

Este script normalmente completa cualquier configuración necesaria en el paquete `nombre` una vez que `nombre` ha sido desempaquetado de su archivo Debian (".deb"). A veces, los scripts 'postinst' piden al usuario datos, y/o le advierten de que si acepta los valores por defecto, tendrá que acordarse de volver atrás y reconfigurar dicho paquete. Muchos scripts 'postinst' ejecutan las órdenes necesarias para empezar o reanudar un servicio una vez que el nuevo paquete ha sido instalado o actualizado. *Es una buena idea el repasar el contenido de un script 'postinst' en busca de consejos de configuración cuando se instale un paquete por primera vez.*

prerm

Este script normalmente detiene cualquier demonio asociado con un paquete. Se ejecuta antes de borrar los archivos asociados a un paquete.

postrm

Este script normalmente modifica enlaces u otros archivos asociados con `nombre`. (Ver notas sobre 7.8 (paquetes Virtuales).)

Todos los archivos de control se pueden encontrar en `/var/lib/dpkg/info`. Los archivos importantes para el paquete `nombre` empiezan con el nombre "nombre" y tienen extensiones "preinst", "postinst", etc., como corresponda. El archivo `nombre.list` en dicho directorio enumera todos los archivos que se instalaron con el paquete `nombre`. (Nótese que la localización de estos archivos es información interna de dpkg; no se debe confiar en ella.)

7.7 ¿Qué es un paquete Requerido/Importante/Estandar/Opcional/Extra?

Cada paquete Debian recibe una *prioridad* por los responsables de la distribución, como ayuda al sistema de mantenimiento de paquetes. Las prioridades son:

- **Requerido** Los paquetes requeridos son necesarios para el correcto funcionamiento del sistema. Esto incluye todas las herramientas necesarias para reparar defectos del sistema. No se deben eliminar estos paquetes o su sistema puede volverse totalmente inestable y probablemente ni siquiera será capaz de usar dpkg para volver a poner las cosas en su sitio. Sistemas con sólo los paquetes requeridos probablemente no son utilizables, pero tienen la suficiente funcionalidad para permitir al administrador del sistema arrancarlo e instalar más programas.

- Los paquetes **importantes** están presentes en cualquier sistema parecido a Unix. Aquí se encuentran los paquetes sin los que el sistema no funcionará correctamente. Esto *NO* incluye Emacs, X11, TeX o cualquier otra aplicación grande. Estos paquetes sólo constituyen la infraestructura base.
- Los paquetes **estándar** son los típicos en cualquier sistema Linux, incluyendo un sistema en modo carácter razonablemente pequeño, aunque no muy limitado. Se instalará por defecto si los usuarios no seleccionan nada más. No incluye demasiadas aplicaciones mayores, aunque incluye Emacs (que es más una pieza de la infraestructura que una aplicación) y una parte razonable de Tex y Latex (si es que es posible usarlos sin X).
- Los paquetes **opcionales** incluyen todos los que razonablemente se desearía instalar a no ser que no se sepa lo que son o exijan requisitos especiales. Esto incluye X11, una distribución completa de TeX, y montones de aplicaciones.
- Los paquetes **extra** tienen conflictos con otros de mayor prioridad, o es sólo probable que se usen si ya se sabe lo que son, o tienen requisitos especiales.

7.8 ¿Qué es un paquete virtual?

Un paquete virtual es un nombre genérico que se aplica a uno cualquiera de un conjunto de paquetes, los cuales proveen todos ellos de una funcionalidad básica similar. Por ejemplo, los programas `tin` y `trn` son lectores de noticias los dos, y por tanto satisfacen por igual cualquier dependencia de un programa que necesite un lector de noticias en el sistema para funcionar o ser útil. Se dice que ambos proveen el "paquete virtual" llamado `lector de noticias`.

Igualmente, `smail` y `sendmail` proveen ambos de la funcionalidad de un agente de transporte de correo. También se dice que ambos proporcionan el paquete virtual "agente de transporte de correo". Si cualquiera de ellos está instalado, cualquier programa que exija la instalación de un `agente de transporte de correo` se dará por satisfecho con la existencia de este paquete virtual.

Debian proporciona un mecanismo mediante el cual, si más de un paquete que permita el mismo paquete virtual se instala en un sistema, los administradores del sistema pueden marcar uno como el paquete preferido. La orden importante es `alternativas de actualización`, y se describe con detalle en la sección de 12.9 (diversions).

7.9 ¿A qué se refieren al decir que un archivo Depende/Recomienda/Sugiere/Entra en conflicto/Reemplaza/Proporciona otro paquete??

El sistema de paquetes Debian tiene un conjunto de "dependencias" diseñadas para indicar (mediante un único indicador) el nivel al que puede operar un Programa A independientemente de la existencia del Programa B en un sistema dado:

- El paquete A *depende* del paquete B si B tiene que estar instalado obligatoriamente para que funcione A. En ese caso, la versión de dependencia normalmente es un límite inferior, de modo que A depende de cualquier versión de B más reciente que alguna versión especificada.
- El paquete A *recomienda* al paquete B, si el responsable del paquete considera que la mayoría de usuarios no querrán A sin tener la funcionalidad permitida por B.
- El paquete A *sugiere* al paquete B si B contiene archivos que están relacionados con (y normalmente mejoran) la funcionalidad de A.

- El paquete A *Entra en conflicto* con el paquete B cuando A no funcionará si B está instalado en el sistema. La mayoría de las veces los conflictos son casos en los que A contiene archivos que son una mejora respecto a los que contiene B. Los "Conflictos" se combinan frecuentemente con "reemplazos".
- El paquete A *reemplaza* al paquete B cuando archivos instalados por B son borrados y (en algunos casos) sobrescritos por archivos en A.
- El paquete A *proporciona* al paquete B cuando todos los archivos y la funcionalidad de B se incorporan en A. Este mecanismo proporciona un modo de que los usuarios con espacio de disco reducido instalen la parte del paquete A que realmente necesitan.

Información más detallada sobre estos temas se puede encontrar en el *Manual del Programador de Debian*.

7.10 ¿Qué significa Pre-Dependencia?

"Pre-Dependencia" es una dependencia especial.

En el caso de la mayoría de paquetes, `dpkg` extraerá el contenido del archivo correspondiente (p.ej., su archivo `.deb`) tanto si los archivos de los que depende existen en el sistema como si no. Desempaquetar significa que `dpkg` extraerá los archivos del paquete que se espera que queden instalados en el sistema de ficheros, y los pondrá en su localización correspondiente. Si el paquete *depende* de la existencia de otro paquete no instalado en el sistema, `dpkg` se negará a completar la instalación ejecutando su orden "configure" hasta que los otros paquetes se instalen.

Pero para algunos paquetes, `dpkg` se negará incluso a desempaquetar los archivos hasta que se resuelvan las dependencias. Se dice de esos paquetes que "Pre-dependen" de la presencia de algún otro paquete. El proyecto Debian proporcionó este mecanismo para permitir la actualización segura de sistemas en formato `a.out` a formato `ELF`, donde el orden en que se instalan los paquetes era crítico.

Información más detallada sobre el uso de estos términos se puede encontrar en el *Manual del Programador de Debian*.

7.11 ¿Qué significa desconocido/instalar/quitar/borrar/conservar en el estado del paquete?

Estas marcas indican lo que el usuario desea hacer con un paquete (ya sea mediante las acciones del usuario en la sección "Select" de `dselect`, o por las llamadas directas del usuario a `dpkg`). Sus significados son:

- desconocido - el usuario no ha dicho si quiere el paquete
- instalar - el usuario ha dicho que quiere instalar o actualizar el paquete
- quitar - el usuario quiere desinstalar el paquete, pero sin borrar cualquier fichero de configuración que exista.
- Borrar - el usuario quiere eliminar completamente el paquete, incluyendo ficheros de configuración.
- conservar - el usuario no quiere procesar este paquete, quiere mantener la versión actual con su estado actual, cualquiera que sea.

7.12 ¿Dónde puedo encontrar información detallada sobre la creación de paquetes Debian?

Ver 13.1 (el Manual del Programador de dpkg y el Manual de Normas de Debian)

8 Las Herramientas de Gestión de Paquetes de Debian

8.1 ¿Qué programa(s) tiene Debian para la gestión de sus paquetes?

8.1.1 dselect

Este programa es una interfaz basada en menús al sistema de gestión de paquetes de Debian. En particular es útil para primeras instalaciones y para actualizaciones a gran escala. `dselect` puede

- guiar al usuario al elegir qué paquetes instalar o borrar, asegurándose de que ningún paquete esté en conflicto con otro, y que todos los paquetes necesarios para que cada uno de los elegidos funcione estén adecuadamente instalados.
- avisar al usuario de inconsistencias e incompatibilidades en las elecciones
- determinar el orden en el que los paquetes han de ser instalados.
- realizar la instalación o borrado de forma automática y
- guiar al usuario a través de cualquier proceso de configuración requerido para cada paquete.

`dselect` comienza presentando al usuario un menú de 7 entradas, cada una de las cuales corresponde a una acción concreta. El usuario puede seleccionar una acción utilizando las flechas del teclado que mueven la barra iluminada y presionando INTRO para seleccionar la acción iluminada.

Lo que el usuario ve a continuación depende de la acción elegida. Si elige una opción diferente de `Access` o `Select`, `dselect` simplemente procederá a ejecutar la opción seleccionada: p.ej., si el usuario eligió `Remove`, dselect procederá a borrar todos los ficheros marcados para ser borrados la última vez que el usuario ejecutó la opción `Select` para elegirlos.

Las dos entradas `Access` y `Select` llevan a menús adicionales. En ambos casos, los menús se presentan en una pantalla dividida en dos; la parte superior presenta una lista de opciones navegable, mientras que la parte inferior presenta una explicación ("información") sobre cada una de las opciones.

Una extensa ayuda en línea está disponible: Utilice la tecla '?' para acceder a la pantalla de ayuda y luego '.' para sucesivamente ver cada una de las páginas disponibles, una a una.

Algunos usuarios encuentran más fácil navegar por `dselect` cuando usa colores en la salida por pantalla. Para ver los colores, asegúrese de que ha ejecutado: `export TERM=linux` antes de ejecutar `dselect`.

El orden en el que las acciones se presentan en el menú inicial de `dselect` refleja el orden en el que el usuario ejecutaría normalmente `dselect` para instalar paquetes. Sin embargo, un usuario puede elegir cualquiera de las opciones tantas veces como sea necesario (incluyendo no usar ninguna opción en absoluto, dependiendo de lo que quiera hacer).

- Primeramente elija un "Método de Acceso" (`Access Method`). Este será el método mediante el cual el usuario planea acceder a los paquetes de Debian; p.ej., algunos usuarios pueden acceder a los paquetes de Debian mediante un CD-ROM mientras que otros planean acceder a ellos mediante ftp anónimo. El "Médodo de Acceso" es almacenado al salir de `dselect`, de forma que si no cambia, no será necesario volver a usar esta opción más.
- Luego, "Actualice" (`Update`) la lista de paquetes disponibles. Para ello, `dselect` lee el fichero "Packages.gz" que debería estar incluido en el directorio superior de la estructura que contiene los paquetes Debian que se van a instalar. (Aunque si no lo encuentra allí, `dselect` le permite la opción de crearlo por usted.)

- Elija (`Select`) para seleccionar los paquetes que desea instalar en el sistema. Tras seleccionar esta opción del menú, el sistema presenta al usuario una pantalla de ayuda; se puede salir de la ayuda (de esta y de cualquiera) presionando la barra espaciadora. Lo mejor (si es la primera vez que utiliza dselect) es que lea *toda* la ayuda, pero presionando '.' repetidamente irá presentando todas las páginas una tras otra. Una vez que el usuario sale de la pantalla de ayuda, aparece el menú en dos secciones para la elección de los paquetes a instalar (o borrar). La parte superior es una ventana relativamente estrecha que presenta una parte de la lista de los 2200 paquetes; la parte inferior de la pantalla es una ventana con "información" sobre los paquetes o grupos de paquetes seleccionados en la parte superior.

  Muchos usuarios noveles suelen mostrar confusión en relación a los siguientes aspectos de la pantalla de la opción `Select`:

  - "Marcar para borrado" de paquetes: Se puede especificar qué paquetes deben ser borrados marcando el nombre del paquete o la etiqueta de un grupo de paquetes, p.ej. "Todos" (`All`) y presionando:
    - la tecla '-'. Esto borra la mayoría de los ficheros asociados con el paquete, pero preserva los ficheros listados como 7.5 (ficheros de configuración) y demás información de configuración.
    - la tecla '_'. En este caso, se borra *cualquier* fichero que pertenezca a este paquete.

    Observe que si "marca para borrar" "Todos los Paquetes" (`All Packages`) su sistema quedará reducido a los paquetes base instalados inicialmente. Esto es problablemente lo que usted quería.
  - Si pone un paquete "en espera" (presionando '='): De forma efectiva, esto indica a `dselect` que no actualice un paquete incluso en el caso de que la versión presente en el sistema sea menos reciente que la disponible en la fuente de la distribución que esté usando (la que fue especificada cuando estableció el Método de Acceso con la opción `Access Method`). (La versión disponible en la distribución viene en el fichero `Packages.gz` que se lee al activar la opción de menú "Update") Sacar un paquete del estado de espera (presionando ':'): Se trata de la opción por defecto y significa que el paquete será actualizado siempre y cuando esté disponible una versión más reciente.
  - Orden de presentación de los paquetes: El orden de presentación predeterminado es por Prioridad; para cada valor de prioridad, los paquetes aparecen ordenados según el directorio (alias sección) del archivo en el que están almacenados. De esa forma, algunos paquetes de (digamos) la sección A pueden aparecer primero, seguidos de paquetes de la sección B, seguidos de otros paquetes (de prioridad más baja) de la sección A. Se puede cambiar el orden de presentación utilizando la tecla 'o' que va ciclicamente, cambiando de orden de presentación.
  - Significado de las etiquetas en la parte superior de la pantalla: Las etiquetas de la parte superior pueden expandirse utilizando la tecla 'v' (del inglés verbose, detallado). Al hacer esto, gran parte del texto que originalmente cabía en la pantalla se desplaza a la derecha. Para poder verlo es necesario usar la flecha derecha; para volver a la izquierda, utilice la flecha izquierda.
  - Qué hacer en la pantalla de conflicto/dependencia: Si un usuario elige (tanto para instalar como para borrar) un paquete, digamos `fu.deb` que depende de o recomienda otro, digamos, `blurf.deb`, entonces `dselect` presentará un extracto de la pantalla principal de selección. El proceso comienza presentando la ayuda a pantalla completa, de la que se puede salir usando la barra espaciadora. Después, puede elegir los paquetes relacionados, aceptando las acciones sugeridas (de instalar o no), o rechazándolas. Para conseguir esto último, basta presionar Mayúsculas-D; para volver a las sugerencias puede usar Mayúsculas-U. En cualquier caso, puede salvar su elección y volver al menú principal de instalación mediante Mayúsculas-Q.

- Una vez de vuelta al menú principal, se puede proceder a la opción de "Instalación" (`Install`) para desempaquetar y configurar los paquetes elegidos. Alternativamente, si lo que quiere es eliminar ficheros, puede elegir la opción de "Borrado" (`Remove`). Puede salir del programa en cualquier momento mediante la opción `Quit`; cualquier elección que haya hecho será preservada por `dselect`.

8.1.2 dpkg

Se trata del programa principal de gestión de paquetes. `dpkg` puede ejecutarse con muchas opciones. Algunas de las más comunes son:

- Averiguar cuáles son las opciones existentes: `dpkg -help`.
- Imprimir el fichero de control (y demás información) de un paquete específico: `dpkg -info fu_VVV-RRR.deb`
- Instalar un paquete (incluyendo el desempaquetado y la configuración) en el sistema de ficheros del disco duro: `dpkg -install fu_VVV-RRR.deb`.
- Desempaquetar (pero no configurar) un paquete Debian en el sistema de ficheros del disco duro: `dpkg -unpack fu_VVV-RRR.deb`. Esta operación *no* deja al paquete necesariamente operativo; algunos ficheros, pueden necesitar algo más de configuración para funcionar correctamente. Esta orden borra cualquier versión del programa instalada previamente, y ejecuta el script 7.6 (preinst) asociado al paquete.
- Configurar un paquete que ha sido previamente desempaquetado: `dpkg -configure fu`. Entre otras cosas, esta acción ejecuta el script 7.6 (postinst) asociado al paquete. También actualiza los ficheros de configuración del paquete especificados en `conffiles`. Debe notar que el argumento de la operación 'configurar' es el nombre del paquete (p.ej. fu), *no* el del archivo del paquete Debian (p.ej. fu_VVV-RRR.deb).
- Extraer un único fichero llamado digamos "blurf" (o un grupo de ficheros llamados "blurf*" de un archivo de paquete Debian: `dpkg -fsys-tarfile fu_VVV-RRR.deb | tar -xf - blurf*`
- Borrar un paquete (pero no sus ficheros de configuración): `dpkg -remove fu`.
- Borrar un paquete (incluidos los ficheros de configuración): `dpkg -purge fu`.
- Listar el estado de los paquetes que contienen la cadena "fu*": `dpkg -list 'fu*'`.

8.1.3 dpkg-deb

Este programa manipula un archivo de paquete Debian(`.deb`). Algunos de sus usos comunes son:

- Averiguar cuáles son las opciones existentes: `dpkg-deb -help`.
- Determinar qué ficheros están contenidos en un archivo de paquete Debian: `dpkg-deb -contents fu_VVV-RRR.deb`)
- Extraer los ficheros contenidos en un archivo de paquete Debian dado en un directorio especificado: `dpkg-deb -extract fu_VVV-RRR.deb tmp` extraerá los ficheros de `fu_VVV-RRR.deb` en el directorio `tmp/`. Esta operación es conveniente para examinar el contenido de un paquete en un directorio concreto, sin necesidad de instalar el paquete en el sistema de ficheros raíz.

Puede obtener más información en la página de manual de `dpkg-deb`(1).

8.1.4 dpkg-split

Este programa divide un paquete grande en ficheros más pequeños (p.ej. cara a copiarlo mediante un grupo de disquetes) y puede usarse para unir de nuevo los ficheros en uno solo. Sólo se puede usar este programa en un sistema Debian pues se vale de `dpkg-deb` para desglosar el archivo de paquete Debian en sus componentes. Así, por ejemplo, si quisiera dividir un fichero .deb grande en N partes,

- Ejecute la orden `dpkg-split -split fu.deb`. Esto producirá N ficheros de aproximandamente 460 KBytes cada uno en el directorio actual.
- Copie los N ficheros en disquetes.
- Copie el contenido de los disquetes en el disco duro que haya elegido en la otra máquina.
- Una los ficheros parciales mediante `dpkg-split -join "fu*"`.

8.2 Debian asegura ser capaz de actualizar programas que están siendo ejecutados; ¿Cómo se puede hacer esto?

Debian GNU/Linux incluye un programa llamado `start-stop-daemon` que es usado por los scripts de instalación para arrancar demonios durante el proceso de arranque de la máquina o para pararlos cuando cambia el nivel de ejecución del núcleo (p.ej. si se cambia de multi-usuario a usuario-único o a parada). El programa `start-stop-daemon` también se utiliza cuando se instala un paquete nuevo que contiene un demonio, para parar demonios en ejecución y rearrancarlos según sea necesario, p.ej. cuando se instala un paquete que contiene un script de configuración actualizado.

8.3 ¿Cómo se puede comprobar qué paquetes han sido instalados en un sistema Debian?

Para averiguar cuál es el estado de todos los paquetes instalados en un sistema Debian, debe ejecutar la orden `dpkg -list`. Esto muestra un resumen de una línea por cada paquete, que contiene un símbolo de estado de 2 letras, el nombre del paquete correspondiente, la versión que está *instalada*, y una descripción muy breve del mismo.

Para averiguar el estado de los paquetes que encajan con un patrón que empiece por "fu" puede ejecutar la orden `dpkg -list 'fu*'`

Para obtener información más detallada de un paquete en particular puede usar la orden `dpkg -status fu`.

8.4 ¿Cómo se puede averiguar qué paquete produjo un fichero en particular?

Para indentificar el paquete que produjo un fichero llamado `fu` puede usar cualquiera de las siguientes opciones:

- `dpkg -search nombredefichero`. En este caso se busca el fichero `nombredefichero` entre los paquetes instalados. (Esto equivale (actualmente) a buscar todos los ficheros de extensión `.list` contenidos en el directorio `/var/lib/dpkg/info/`.)
- `grep fu Contents`, o `zgrep fu Contents.gz`. Esto realiza una búsqueda de ficheros que contengan la subcadena `fu` en cualquier parte de sus caminos completos. Los ficheros `Contents` y `Contents.gz` residen en los directorios principales (stable, non-free, contrib, development) de un sitio FTP de Debian. Cada fichero `Contents` se refiere únicamente a los paquetes que se encuentran bajo el árbol de directorios en el que se encuentra. Por ello, es posible que tenga que buscar más de un fichero `Contents` para encontrar cual es el paquete que tiene el fichero `fu`.

  Este método tiene como ventaja sobre `dpkg -search` que es capaz de encontrar ficheros contenidos en paquetes que no han sido instalados previamente en el sistema.

9 Conseguir e instalar Debian GNU/Linux

9.1 ¿Dónde/cómo puedo conseguir los discos de instalación de Debian?

Puede conseguir los discos de instalación copiando los ficheros adecuados del servidor FTP de Debian: `<ftp://ftp.debian.org/pub/debian/>` y sus *réplicas* `<http://www.debian.org/distrib/ftplist.html>`. Por favor, siga las instrucciones dadas en la Guía de Instalación. Está disponible (en inglés) en los ficheros

install.txt (Texto) y *install.html* (HTML).

Algunos casos especiales se detallan a continuación.

9.2 ¿Cómo consigo e instalo Debian desde un CD-ROM?

Linux soporta el sistema de ficheros ISO 9660 (CD-ROM) con las extensiones Rock Ridge (conocidas anteriormente como "High Sierra"). Diversos *distribuidores* `<http://www.debian.org/distrib/vendors>` ofrecen Debian en este formato.

Atención: Cuando vaya a instalar a partir de CD-ROM, normalmente no es una buena idea escoger el método de acceso `cdrom` de dselect. Este método normalmente es muy lento. Los métodos de acceso `mountable` y `apt`, por ejemplo, son mucho mejores para la instalación a partir de CD-ROM.

9.3 Tengo mi propia grabadora de CDs, ¿hay imágenes de CD disponibles en algún sitio?

Sí. Para que a los distribuidores de CD les resulte más fácil proporcionar discos de alta calidad, existen imágenes de CD oficiales *aquí* `<http://cdimage.debian.org/>`.

9.4 ¿Cómo puedo obtener/instalar Debian a partir de un conjunto de disquetes?

Copie los paquetes de Debian en disquetes formateados. Da igual que el formato sea DOS, el nativo de Linux "ext2", o el de "minix" ; sólo tiene que usar la orden mount apropiada al formato que esté usando

Usar disquetes tiene los siguientes problemas:

- Nombres cortos de MS-DOS: Si pretende poner ficheros de paquetes de Debian en disquetes formateados para MS-DOS, se dará cuenta de que sus nombres son demasiado largos, y no se ajustan a la limitación 8.3 de los nombres de ficheros en MS-DOS. Para resolverlo, los desarrolladores de Debian ponen todos sus paquetes disponibles con nombres en formato 8.3 en subdirectorios "msdos" aparte (`stable/msdos-i386/`, `non-free/msdos-i386/`, `contrib/msdos-i386/`, y `development/msdos-i386/`). Los ficheros de estos subdirectorios son simplemente enlaces simbólicos hacia los ficheros de archivo de Debian; sólo se diferencian de los ficheros que hay en `binary-i386/`, y el resto de directorios, en que sus nombres son más cortos.
- Tamaños de fichero grandes: Algunos paquetes tienen más de 1.44 MBytes, y no cabrán en un solo disquete. Para resolverlo, use la utilidad 8.1.4 (dpkg-split), disponible en el directorio `tools` en `<ftp://ftp.debian.org/debian/>` y sus *réplicas* `<http://www.debian.org/distrib/ftplist.html>`.

Tiene que tener soporte de disquetes en el núcleo para poder leer y escribir en disquetes; la mayoría de los núcleos vienen con soporte para disquetes incluido.

Para montar un disquete bajo el nombre `/floppy` (un directorio que se debe haber creado durante la instalación), use:

- `mount -t msdos /dev/fd0 /floppy/` si el disquete está en la unidad A: y tiene formato MS-DOS,
- `mount -t msdos /dev/fd1 /floppy/` si el disquete está en la unidad B: y tiene formato MS-DOS,
- `mount -t ext2 /dev/fd0 /floppy/` si el disquete está en la unidad A: y tiene formato ext2 (el normal de Linux).

9.5 ¿Cómo puedo conseguir e instalar Debian directamente desde un servidor de FTP?

Instale la herramienta de Debian 10.2 (dpkg-ftp).

Para detalles de cómo instalar un paquete, vea las notas sobre 8.1.2 (dpkg).

Luego llame al programa `dselect`, que llamará a `dpkg-ftp` por usted, le guiará a través de la selección de paquetes y los instalará, sin que tenga que copiarlos previamente en su sistema. Este método está pensado para ahorrar al usuario espacio de disco y tiempo. Fíjese en que no se necesita una configuración especial del núcleo para acceder a paquetes de Debian e instalarlos mediante este método.

Para usar este servicio de `dselect`, tendrá que saber:

- el nombre de dominio del servidor de FTP anónimo que pretenda usar.
- el directorio que contiene los ficheros que pretende instalar, o los subdirectorios que contienen dichos ficheros. Este directorio debe contener un fichero llamado "Packages" (o su version comprimida, "Packages.gz").

9.6 ¿Cómo puedo conseguir e instalar Debian desde cinta de streamer?

De momento, instalar paquetes directamente desde cinta no es posible. Pero se puede usar `tar`, `cpio`, o `afio` para copiar los ficheros de archivo de Debian a cinta, y luego copiarlos al disco local para instalarlos. En la misma línea, se pueden usar disquetes que contengan ficheros "tar" que serán copiados a disco local antes de ser gestionados por las utilidades de paquetes de Debian.

9.7 ¿Cuál es la última version de Debian?

Actualmente existen dos versiones de Debian:

- "2.1":, alias "stable". Este es software estable, pero puede cambiar cuando se incorporen modificaciones importantes.
- la versión de "unstable" (de desarrollo); . Esta es la versión actualmente en desarrollo; se actualiza continuamente. Puede recuperar paquetes del archivo "unstable" en cualquier servidor de FTP de Debian y usarlos para actualizar su sistema en cualquier momento. Está planeado que ésta se convierta en una nueva distribución de Debian varios meses después de la última publicación.

10 Actualización del sistema Debian

10.1 ¿Cómo puedo actualizar mi distribución Debian 1.3.1 (o anterior), basada en lib5, a la versión 2.0 (o posterior), basada en libc6?

Hay varias formas de actualizarse:

- Utilizando un sencillo script de shell llamado `autoup.sh` que actualiza los paquetes más importantes. Después de que `autoup.sh` ha hecho su trabajo, puede utilizar dselect para instalar los paquetes que falten *en masa*. Este es probablemente el método recomendado, pero no el único. Actualmente, la última versión de `autoup.sh` puede encontrarse en los siguientes lugares:

  – `<http://www.taz.net.au/autoup/>`

  – `<http://csanders.vicnet.net.au/autoup/>`

  Se espera que esté en los archivos FTP de Debian algún día.

- Siguiendo de cerca el *libc5-libc6-Mini-HOWTO* `<ftp://ftp.debian.org/pub/debian/doc/libc5-libc6-Mini-HOWTO.txt>` y actualizar los paquetes más importantes a mano. `autoup.sh` está basado en este Mini-HOWTO, así que este método debería funcionar más o menos como usar `autoup.sh`.

- Usando una versión libc5 de APT. APT quiere decir "A Package Tool" (una herramienta de paquetes), y podría sustituir a dselect algún día. De momento, funciona simplemente como un interfaz de línea de órdenes, o como método de acceso de dselect. Encontrará una versión libc5 en el directorio `dists/slink/main/upgrade-older-i386` en los archivos de Debian.

- Utilizando únicamente dselect, sin actualizar primero ningún paquete a mano. Se recomienda encarecidamente que NO utilice este método si puede evitarlo, porque dselect, por sí solo, actualmente no instala los paquetes en el orden óptimo. APT funciona mucho mejor y es más seguro.

10.2 ¿Cómo puedo mantener mi Sistema Debian actualizado?

Uno puede simplemente realizar un ftp anónimo a un archivo de Debian, buscar por los directorios hasta encontrar el fichero deseado, traérselo y finalmente instalarlo usando `dpkg`. Observe que `dpkg` instalará los ficheros actualizados en el momento, incluso en un sistema ya en marcha, sin tener que detenerlo. A veces, un paquete revisado requerirá la instalación de la versión actualizada de otro paquete, en cuyo caso la instalación fallará hasta que/a menos que el otro paquete esté instalado.

Mucha gente encuentra que esta forma de actualizar lleva mucho tiempo debido a que Debian evoluciona tan rápidamente—semanalmente se suelen añadir una docena de paquetes nuevos, o más. Este número es mayor cuando se aproxima la publicación de una nueva versión. Para poder manejarse bien con tal avalancha, mucha gente prefiere usar un programa automatizado. Hay disponibles tres paquetes diferentes para este propósito:

- `dpkg-ftp`. Éste es un método de acceso para `dselect`. Se puede invocar desde el propio `dselect`, permitiendo así al usuario obtener ficheros e instalarlos en un sólo paso. Para hacer ésto, arranque el programa `dselect`, elija la opción "0" ("Choose the access method to use"), marque la opción "ftp" y luego especifique la máquina y directorio remotos. `dpkg-ftp` transferirá entonces los ficheros seleccionados a su máquina (ya sea en esta sesión de `dselect` o en posteriores). Observe que, al contrario que el programa `mirror`, `dpkg-ftp` no recoge todos los ficheros del sitio réplica, sino que transfiere sólo los ficheros que haya seleccionado, y que necesiten ser actualizados.

  `dpkg-ftp` está disponible en el directorio `stable/binary-all/net/` en cualquier archivo de Debian.

- `mirror`. Este script Perl, y su programa gestor (opcional) denominado `mirror-master`, pueden usarse para obtener partes especificadas del árbol de directorios de un servidor especificado, *via* ftp anónimo.
- `dftp`. Este script Perl puede usarse para traerse de un servidor determinado paquetes Debian especificados por el usuario. Empieza obteniendo los ficheros Packages.gz de los directorios especificados por el usuario (ej. stable, contrib, non-free) y le presenta una lista de paquetes. Éstos están localizados en varias secciones: new upgrades (nuevas actualizaciones), downgrades ("des-actualizaciones"), new packages (nuevos paquetes), ignored upgrades (actualizaciones ignoradas) e ignored packages (paquetes ignorados). Entonces el usuario selecciona los paquetes que quiera y dftp los instala. Ésto hace muy fácil mantener su sistema Debian 100% actualizado todo el tiempo sin tener que bajarse paquetes que no va a instalar. `dftp` puede usarse para llamar a 8.1.1 (dselect), proporcionando de esta forma un modo integrado para obtener y actualizar los paquetes Debian de nuestro sistema. Cuando finalice la instalación, se puede usar otra orden `dftp` para eliminar los ficheros de los paquetes (".deb"). Los cambios en el sistema de ficheros remoto son tratados automáticamente por `dftp`, el cuál compara el fichero Packages.gz local con los ficheros del sistema remoto.

  El programa dftp está disponible como un script en el directorio `project/misc` en cualquier archivo FTP de Debian.

10.3 ¿Debo cambiar a modo monousuario para actualizar un paquete?

No. Los paquetes pueden actualizarse en cualquier momento, incluso en sistemas en funcionamiento. Debian tiene un programa llamado `start-stop-daemon` que se invoca para parar y luego rearrancar un proceso en ejecución si es necesario durante la actualización de un paquete.

10.4 ¿Tengo que guardar todos esos ficheros .deb en mi disco?

No. Si se ha bajado los ficheros a su disco (lo que no es absolutamente necesario (ver 10.2 (dpkg-ftp) o 10.2 (dftp))), después de que los haya instalado, puede borrarlos de su sistema.

10.5 ¿Cómo puedo mantener un registro de los paquetes que he ido añadiendo al sistema?

`dpkg` mantiene un registro de los paquetes que han sido desempaquetados, configurados, borrados, y/o purgados, pero no mantiene un registro (de momento) de los mensajes que envía al terminal cuando el paquete se está manipulando. Algunos usuarios realizan esta tarea simplemente usando `tee`:

```
dpkg -iGOEB -R stable/binary non-free/binary contrib/binary | \
    tee -a /root/dpkg.log
```

La misma orden usando los parámetros largos quedaría:

```
dpkg --install --refuse-downgrade --selected-only \
  --skip-same-version --auto-deconfigure \
  --recursiv estable/binary non-free/binary contrib/binary | \
    tee -a /root/dpkg.log
```

11 Debian y el núcleo

11.1 ¿Qué herramientas proporciona Debian para crear núcleos personalizados?

Se recomienda a todos aquellos usuarios que deseen (o necesiten) compilar un núcleo personalizado que obtengan el paquete `kernel-package` (que se encuentra en la sección `misc` en los sitios FTP de Debian). Este paquete contiene el script necesario para compilar el núcleo, y permite crear un paquete Debian kernel-image sólo con la ejecución de la orden `make-kpkg kernel_image` en el directorio principal de los fuentes del núcleo. Hay disponible una ayuda ejecutando la orden `make-kpkg -help`, y en la página manual de make-kpkg(8).

Los usuarios deberán obtener separadamente el código fuente del último núcleo (o del núcleo que quieran) de su servidor de Linux favorito.

Para construir un núcleo personalizado, se deberán tener los siguientes paquetes instalados: `gcc`, `libc6-dev`, `bin86`, `binutils`, y `make`.

Ejecutando la orden `dpkg -install kernel-package_VVV_all.deb` se creará el directorio `/usr/src/linux-VVV/`, y establecerá el enlace `/usr/src/linux` para que apunte al directorio `/usr/src/linux-VVV/` que contendrá los fuentes del núcleo.

En el fichero `/usr/doc/kernel-package/README` se dan instrucciones detalladas del uso del paquete. Resumidamente se debe:

- Descomprimir los fuentes del núcleo, y hacer un `cd` al directorio recién creado.
- Modificar la configuración del núcleo usando una de las siguientes órdenes:
  - `make config` (para terminales, con un interfaz línea-a-línea).
  - `make menuconfig` (para un interfaz por menús basado en ncurses). Observe que para usar esta opción debe estar instalado el paquete `ncurses3.0-dev`.
  - `make xconfig` (para un interfaz X11). Para usar esta opción se requiere tener instalados los paquetes relevantes de las X.

  Cualquiera de los pasos anteriores genera un nuevo `.config` en el directorio principal de los fuentes del núcleo.
- Ejecutar la orden: `make-kpkg -rev Custom.N kernel_image`, donde N es un número de revisión asignado por el usuario. El nuevo archivo Debian así creado tendrá la revisión Custom.N, por ejemplo, `kernel-image-2.0.36_Custom.1_i386.deb` para el núcleo 2.0.36.
- Instalar el paquete creado.
  - `Ejecute dpkg -install /usr/src/kernel-image-VVV_Custom.N.deb` para instalar el núcleo. El script de instalación,
    - ejecutará el cargador de arranque, LILO (si está instalado),
    - instalará el núcleo personalizado en /boot/vmlinuz_VVV-Custom.N, y establecerá los enlaces simbólicos apropiados a la nueva versión del núcleo.
    - preguntará al usuario si quiere hacer un disquete de arranque. Este disco de arranque contendrá sólo el núcleo. Mire las notas adicionales sobre la creación de un 11.2 (disco de arranque personalizado).
  - Para usar otros cargadores de arranque (como `loadlin`), puede copiar esta imagen a otros lugares (por ejemplo, a una partición `MS-DOS`).

11.2 ¿Cómo puedo crear un disco de arranque personalizado?

Esta tarea se facilita mucho con el paquete `boot-floppies`, que se encuentra normalmente en la sección `devel` de los archivos FTP de Debian. Los scripts de este paquete producen discos de arranque en el formato `SYSLINUX`. Son discos con formato `MS-DOS` cuyo master boot record (registro maestro de arranque) ha sido alterado para que arranque Linux (o cualquier otro sistema operativo que se haya definido en el fichero syslinux.cfg en el disquete) directamente. Otros scripts de este paquete sirven para producir discos root de emergencia y pueden incluso reproducir los discos base.

Cuando se instala el paquete boot-floppies, se instalan los ficheros en `/usr/src/boot-floppies/`, incluyendo los tres scripts, denominados `bootdisk.sh`, `rootdisk.sh`, y `basedisk.sh`. Se recomienda a los usuarios de estos scripts revisarlos y personalizarlos apropiadamente. En particular, uno debe incluir el núcleo seleccionado cuando invoque el script `bootdisk.sh`. *Tenga en cuenta que este núcleo no se puede usar para cargar el disco raíz (root disk) en un RAMDISK salvo que se haya compilado con `CONFIG_RAMDISK=y` y `CONFIG_INITRD=y`.*

Para copiar el paquete kernel-image-2.0.36 en un disco de 1.44 MBytes colocado en A: ejecute la siguiente orden:

```
./bootdisk.sh /usr/src/kernel-image-2.0.36_1.00.deb /dev/fd0 1440
```

Asegúrese de incluir un editor y cualquier otra herramienta que seleccione cuando cree su disco raíz (root disk) personalizado con `./rootdisk.sh`.

Mucha gente encuentra que los discos de arranque y raíz se comportan como unos buenos discos de emergencia.

11.3 ¿Qué medios especiales proporciona Debian para el uso de módulos?

El paquete `modconf` de Debian proporciona un shell script (`/usr/sbin/modconf`) que puede usarse para personalizar la configuración de los módulos. Este script presenta un interfaz basado en menús, preguntando al usuario acerca de las particularidades de los controladores de dispositivos de su sistema. Las respuestas se utilizan para personalizar el fichero `/etc/conf.modules` (que lista alias y otros argumentos que deben usarse conjuntamente con algunos módulos), y `/etc/modules` (que lista los módulos que deben cargarse en el arranque).

Al igual que los (nuevos) ficheros Configure.help que hay ahora disponibles para ayudar en la construcción de núcleos personalizados, el paquete modconf viene con una serie de ficheros de ayuda (en `/usr/lib/modules_help/`) que proporcionan información detallada sobre los argumentos apropiados para cada uno de los módulos.

11.4 ¿Puedo desinstalar sin peligro un núcleo antiguo y, si es así, cómo?

Sí. El script `kernel-image-NNN.prerm` comprueba antes si el núcleo que está utilizando actualmente es el mismo que quiere desinstalar. Por lo tanto puede borrar paquetes de núcleos no deseados ejecutando la siguiente orden:

```
dpkg -purge -force-remove-essential kernel-image-NNN
```

12 Personalización de la instalación de Debian GNU/Linux

12.1 ¿Cómo puedo asegurarme de que todos los programas usen el mismo tamaño de papel?

El fichero `/etc/papersize` contiene el nombre del tamaño de hoja usado por omisión en todo el sistema (por ej. letter o A4). Puede especificarse también mediante la variable de entorno `PAPERSIZE`. Para más detalles consulte la página de manual `papersize(5)`.

12.2 ¿Cómo puedo proporcionar acceso a periféricos de hardware sin comprometer la seguridad?

Muchos ficheros de dispositivos del directorio `/dev` pertenecen a ciertos grupos predefinidos. Por ejemplo `/dev/fd0` pertenece al grupo `floppy`, y `/dev/dsp` pertenece al grupo `audio`.

Para que un usuario determinado tenga acceso a uno de dichos dispositivos, añada al usuario al grupo al que pertenece el dispositivo. Esto evita tener que ejecutar chmod sobre el dispositivo.

12.3 ¿Cómo cargo una fuente de consola en el arranque al estilo Debian?

El paquete `kbd` ahora admite esto, mire en `/etc/kbd/config`.

12.4 ¿Cómo puedo configurar los valores por omisión de una aplicación X11?

La instalación Debian de X11 espera que usted mantenga los ficheros en `/usr/X11R6/lib/X11/app-defaults/` sin cambios. Si usted desea configurar aplicaciones X globalmente, añada sus preferencias en `/etc/X11/Xresources`. Este archivo está marcado como un archivo de configuración, por lo tanto su contenido se mantendrá entre actualizaciones.

12.5 Cada distribución parece tener un método diferente para 'arrancar'. Cuénteme acerca del usado por Debian.

Como todo UNIX, Debian arranca ejecutando el programa `init`. El fichero de configuración para `init` (que es `/etc/inittab`) especifica que el primer script que debe ejecutarse será `/etc/init.d/rcS`. Este script verifica y monta los sistemas de ficheros, carga módulos del núcleo, comienza los servicios de red (llamando al script `/etc/init.d/network`), programa el reloj, inicializa alguna otra cosa, y luego ejecuta todos los scripts (excepto aquellos con un '.' en el nombre) en `/etc/rc.boot/`. Estos scripts especifican el teclado a usarse, recuperan ficheros perdidos estando en un editor, y configuran los puertos serie.

Después de completar el arranque, `init` ejecuta todos los scripts de inicio de un directorio indicado por el runlevel predeterminado (este valor se especifica por la entrada `id` en `/etc/inittab`). Como la mayoría de los UNIX compatibles con System V, Linux tiene 7 runlevels:

0 (detiene el sistema), 1 (modo único-usuario), de 2 a 5 (varios modos multi-usuario), y 6 (reinicializar el sistema).

Los sistemas Debian vienen configurados con id=2, lo que indica que el runlevel será de '2' al entrar al estado multi-usuario, y que se ejecutarán los scripts en `/etc/rc2.d/`.

De hecho, los scripts en cualquiera de los directorios `/etc/rcN.d/` son sólo enlaces simbólicos de vuelta a los scripts en `/etc/init.d/`. Sin embargo, los *nombres* de los ficheros en cada uno de los directorios

`/etc/rcN.d/` están elegidos para indicar la *manera* en que los scripts en `/etc/init.d/` serán ejecutados. Específicamente, antes de entrar a cualquier runlevel, se ejecutan todos los scripts cuyo nombre comienza con 'K'; estos scripts detienen servicios. Luego se ejecutan todos los scripts cuyo nombre comienza con 'S'; estos scripts inician servicios. El número de dos dígitos que sigue a la 'K' o 'S' indica el orden en que los scripts se ejecutarán. Los de números más bajos se ejecutarán primero.

Esta estrategia funciona porque los scripts en `/etc/init.d/` llevan todos un argumento que puede ser 'start' (comenzar), 'stop' (terminar), o 'reload' (reiniciar), y llevarán a cabo la tarea indicada por éste. Por ejemplo, con el argumento 'reload' la orden `/etc/init.d/sendmail reload` envía al daemon sendmail un señal que le hace releer su fichero de configuración. Estos scripts se pueden usar para controlar varios procesos incluso después de que el sistema haya sido iniciado.

12.6 Pareciera ser que Debian no usa `rc.local` para personalizar el proceso de inicialización, ¿qué facilidades provee Debian para esta tarea?

Suponga que un sistema necesita ejecutar el script `fu` al inicializar, o al entrar en un runlevel en particular. Entonces el administrador del sistema debería:

- Colocar el script `fu` en el directorio `/etc/init.d/`.
- Ejecutar la orden `update-rc.d` con argumentos apropiados para preparar enlaces entre los directorios rc?.d (especificados desde la línea de órdenes) y `/etc/init.d/fu`. Aquí, '?' es un número de 0 a 6 y coresponde a un runlevel estilo System V.
- Reinicializar el sistema.

La orden `update-rc.d` creará enlaces entre ficheros en los directorios rc?.d y el script en `/etc/init.d/`. Cada enlace comenzará con una 'S' o una 'K', seguida de un número, seguido por el nombre del script. Los scripts que comiencen con 'S' en `/etc/rcN.d/` serán ejecutados al entrar al runlevel `N`. Los que lo hagan con con una 'K' serán ejecutados al dejar el runlevel `N`.

Uno podría, por ejemplo, obligar al script `fu` a ejecutarse en el arranque, poniéndolo en `/etc/init.d/` e instalando los enlaces con `update-rc.d fu defaults 19`. El argumento 'defaults' se refiere a los runlevels predeterminados, que son los que van del 2 al 5. El argumento '19' se asegura de que `fu` sea llamado antes que cualquier otro script que contenga el número 20 o un número mayor.

12.7 ¿Cómo puedo reemplazar un fichero instalado por un paquete con otro?

Suponga que el administrador o un usuario local desea usar el programa "login-local" en lugar del "login" provisto por el paquete Debian `login`. No:

- Sobreescriba `/bin/login` con `login-local`.

El sistema administrador de paquetes no tendrá conocimiento acerca de este cambio, y simplemente reescribira su `/bin/login` cuando `login` (o cualquier paquete que provea `/bin/login`) sea instalado o actualizado.

En lugar de eso,

- Ejecute `dpkg-divert -divert /bin/login.debian /bin/login` para indicar que todas las futuras instalaciones de paquetes `login` escriban el fichero `/bin/login` en `/bin/login.debian`.
- Luego ejecute `cp login-local /bin/login` para mover su versión localmente compilada a su lugar.

El mensaje acerca del uso de `dpkg-divert` provee más detalles (los cuales se pueden ver usando la orden `dpkg-divert -help`).

Hay más información disponible en el *Manual del Programador Debian*.

12.8 ¿Cómo puedo incluir el paquete Debian que creé localmente en la lista de paquetes disponibles usada por el sistema de administración de paquetes?

Puede hacer esto de cualquiera de las siguientes dos maneras:

- Use `dselect`, y vuelva a seleccionar el método de acceso. Se le preguntará por un directorio en donde sus paquetes locales residen.
- Ejecute la orden `dpkg-scanpackages DIR_BIN FICHERO_DE_REEMPLAZOS [PREFIJO] > Packages.nuevo`

  donde:

  - DIR-BIN es un directorio donde se hayan almacenados archivos Debian (que normalmente tendrán la extensión ".deb").
  - FICHERO_DE_REEMPLAZOS es un fichero que es editado por los que mantienen la distribución y normalmente se almacena en un archivo ftp Debian en `indices/override.main.gz` para la distribución principal ("main").
  - PREFIJO es una secuencia de caracteres opcional que puede añadirse en el fichero Packages.new que está siendo producido.

  Una vez que haya armado el fichero `Packages.nuevo`, avise al sistema administrador de paquetes sobre él con la orden `dpkg -update-avail Packages.nuevo`.

12.9 A algunos usuarios les gusta mawk, a otros les gusta gawk; algunos prefieren vim mientras que otros prefieren elvis; algunos quieren usar trn, a otros les gusta tin; ¿cómo soporta Debian la diversidad?

Hay muchos casos en donde dos paquetes proveen dos versiones diferentes de un programa, los cuales proveen la misma funcionalidad básica. Los usuarios podrían preferir uno sobre el otro por hábito, o porque la interfaz de uno es algo más agradable que la de otro. Otros usuarios en el mismo sistema pueden elegir diferente.

Debian usa un sistema de paquetes "virtual" para permitir a los administradores elegir (o dejar que los usuarios elijan) sus herramientas favoritas cuando hay dos o más que proveen la misma funcionalidad básica, y al mismo tiempo permitir que se satisfagan las dependencias sin necesidad de especificar un paquete en particular.

Por ejemplo, podrían existir dos versiones diferentes de lectores de news en un sistema. El paquete servidor de news podría 'recomendar' que exista *algún* lector en el sistema, pero la elección de `tin` o `trn` se puede dejar a cada usuario. Esto se logra haciendo que ambos paquetes provean el paquete virtual `news-reader`. *Cuál* de los programas es invocado será determinado por un enlace apuntando desde un fichero con el nombre del paquete virtual `/etc/alternatives/news-reader` al fichero seleccionado, p. ej. `/usr/bin/trn`.

Un simple enlace es insuficiente para soportar el uso completo de un programa alternativo, normalmente páginas de manual y posiblemente otros ficheros de soporte pueden ser seleccionados también. El script Perl `update-alternatives` provee una manera de asegurar que todos los ficheros asociados con un paquete específico sean seleccionados como valor por omisión en el sistema.

13 Cómo obtener soporte para Debian GNU/Linux

13.1 ¿Qué otra documentación existe sobre y para un sistema Debian?

- Instrucciones de instalación para la versión actual: ver `<ftp://ftp.debian.org/pub/debian/dists/stable/main/disks-i386/current/install.html>`.
- Manual del programador de dpkg y manual de normas de Debian
  - El *dpkg programmer's manual* `<ftp://ftp.debian.org/debian/doc/package-developer/programmer.ps.gz>` es la documentación primaria sobre los aspectos técnicos de la creación de paquetes Debian de ejecutables y fuentes. El *Debian policy manual* `<ftp://ftp.debian.org/debian/doc/package-developer/policy.ps.gz>` documenta las normas que un paquete debe cumplir para que sea incluido en Debian.

    Estos documentos también están incluidos en el paquete `dpkg-dev` más actual.
- Documentación sobre paquetes Debian instalados: La mayoría de los paquetes tienen ficheros que son desempaquetados en `/usr/doc/PAQUETE`.
- Documentación sobre el proyecto Linux: El paquete Debian `doc-linux` instala las versiones más recientes de todos los HOWTOs y mini-HOWTOs procedentes del *Linux Documentation Project*. Este paquete está en la sección `doc` de los archivos FTP.
- Páginas 'man' estilo Unix: La mayoría de las órdenes tienen páginas de manual escritas en el estilo de los ficheros 'man' originales de Unix. Están referenciados por la sección del directorio 'man' donde residen: p.ej., fu(3) se refiere a una página de manual que reside en /usr/man/man3/, y puede ser llamada usando la orden: `man 3 fu`, o simplemente `man fu` si la sección 3 es la única que contiene una página sobre `fu`. Se puede saber qué directorio de `/usr/man/` contiene una cierta página de manual ejecutando `man -w fu`. Los nuevos usuarios de Debian deben tener en cuenta que las páginas 'man' de muchas órdenes generales del sistema no estarán disponibles hasta que instalen estos paquetes (ambos están almacenados en el directorio `stable/binary-all/doc/` del archivo FTP de Debian:
  - `manpages` (ver 5.10 (soporte para idiomas distintos del inglés)).
  - `man-db`, el cual contiene el propio programa `man`, así como otros programas para manipular las páginas de manual.
- Páginas 'info' estilo GNU: La documentación de usuario para muchas órdenes, particularmente herramientas GNU, no está disponible en páginas 'man', sino en ficheros 'info' que pueden leerse con la herramienta GNU `info` o ejecutando `M-x info` desde GNU Emacs. Su principal ventaja sobre las páginas 'man' originales es que componen un sistema hipertexto. *No* requiere la WWW; `info` puede ejecutarse desde una consola de texto. Fue diseñado por Richard Stallman y precedió la WWW. El programa que manipula ficheros 'info' se instala como un paquete aparte `info`, normalmente almacenado en la sección `doc` del archivo FTP.

13.2 ¿Existe algún recurso en línea para hablar sobre Debian?

Hay varias listas de correo relacionadas con Debian:

- **debian-announce@lists.debian.org (moderada)**. (Normalmente) anuncios importantes sobre el sistema. Como media, se envía menos de un mensaje a la semana por aquí.
- **debian-changes@lists.debian.org**. Anuncios de nuevos paquetes para el sistema Debian. Esta lista puede llevar varios anuncios al día.

- **debian-user@lists.debian.org** . Una lista de correo donde los usuarios pueden preguntar y recibir consejos sobre el uso y configuración de paquetes de Debian u otros aspectos de Debian Linux. Esta lista es un valioso medio de aprendizaje; Puede haber 50 mensajes al día o más en esta lista.
- **debian-sparc@lists.debian.org** . Esta lista es para aquellos que están envueltos en la adaptación de software Debian a la plataforma SPARC.
- **debian-alpha@lists.debian.org** . Esta lista es para aquellos que están envueltos en la adaptación de software Debian a plataformas DEC alpha.
- **debian-68k@lists.debian.org** . Esta lista es para aquellos que están envueltos en la adaptación de software Debian a plataformas Motorola 680x0; actualmente sólo Atari y Amiga.
- **debian-talk@lists.debian.org** .

Para suscribirse a debian-X (siendo X announce, changes, user, talk), envíe un mensaje a debian-X-request@list.debian.org con la palabra "subscribe" en el campo Subject:. Si tiene un navegador de World Wide Web que maneje los formularios, puede suscribirse a debian-announce y debian-user usando el *formulario de WWW* `<http://www.debian.org/MailingLists/subscribe.html>`. También puede de-suscribirse usando el citado formulario.

La dirección de correo del responsable de la lista es *listmaster@lists.debian.org*

Los archivos de las listas de correo de Debian están disponibles a través de la WWW en `<http://www.debian.org/Lists-Archives/>` y a través de FTP en `<ftp://ftp.debian.org/debian/debian-lists/>`.

Los usuarios pueden hacer preguntas a los mantenedores de los paquetes, ya que sus direcciones de correo se proporcionan en los ficheros de control de Debian (ver sección 7.4 (Debian control file)) que se incluyen con cada paquete. También se pueden conocer los nombres y direcciones de correo de los mantenedores buscándolo en el fichero "Packages", ya que este fichero es simplemente una concatenación de todos los ficheros de control de paquete disponibles en un árbol de directorios en particular. Para extraer el fichero de control de un paquete Debian en concreto, use la orden,

```
dpkg --info packageName_VVV-RRR.deb
```

Otra lista de correo relacionada, **debiangame**, es mantenida por Gary Moore (`<mailto:gary@ssc.com>`) en la Universidad de Washington. Como el nombre sugiere, está dedicada a hablar sobre juegos que han sido (o serán) empaquetados para Debian. Para suscribirse, mande un mensaje a `listproc@u.washington.edu`, poniendo en el cuerpo del mensaje:

```
subscribe debiangame Nombre Apellido
```

ListProc requiere tanto el Nombre como el Apellido.

Los usuarios deberán enviar preguntas no específicas de Debian a uno de los grupos de noticias de Linux, que se llaman comp.os.linux.* o linux.*. Specialized Systems Consultants (también conocido como SSC) mantiene una *lista de grupos de noticias sobre Linux, Unix, X, y redes* `<http://www.ssc.com/linux/news.html>` en su sitio WWW.

13.3 ¿Cuál es el código de conducta en las listas de correo?

Cuando use una de las listas de correo de Debian, por favor siga estas reglas:

- No ofenda; no es cortés. Además, la gente que está desarrollando Debian son todos voluntarios que donan su tiempo, energía y dinero en un intento de agrupar los esfuerzos del proyecto Debian.

- No use lenguaje obsceno; hay quien recibe las listas via packet radio, donde las palabras malsonantes son ilegales.
- Asegúrese de que está usando la lista adecuada.
- No intente enviar un mensaje hasta que no esté suscrito. Mensajes procedentes de no-subscriptores son ignorados para hacer las listas de correo usables. Por ello, solamente los mensajes procedentes de direcciones de correo que se correspondan con la de un subscriptor son aceptados. Si manda a menudo mensajes desde otras direcciones que no sean aquella desde la que se suscribió, envíe un mensaje al responsable de las listas *listmaster@lists.debian.org* con el asunto "please add subscriber alias".
- Ver sección 14.2 (reporting bugs) sobre cómo comunicar bugs.

13.4 ¿Hay alguna forma rápida de buscar información sobre Debian GNU/Linux?

Hay toda una variedad de motores de búsqueda que sirven documentación relacionada con Debian.

- *Verism's search site*. Para obtener información sobre cómo remitir un informe de bug, introduzca las palabras `debian bug submit` y busque por "all of these words".
- *DejaNews news search service*. Para encontrar las experiencias que la gente ha tenido en la búsqueda de manejadores para las controladoras Western Digital, pruebe la búsqueda de esta frase:

  ```
  linux & WD
  ```

  Esto le dice a DejaNews que dé cuenta de cualquier mensaje conteniendo la cadena "linux" Y la cadena "WD". Cuando lo usé, descubrí que mi tarjeta WD (la cual tenía desde hacía solo 6 meses) ha sido declarada obsoleta por Adaptec, ahora que han comprado WD. Así que no hay manejadores disponibles. (Bendito sea Adaptec.)
- El *AltaVista Search Engine* también se puede usar para hacer búsquedas en Usenet (aunque parece que no está tan al día como DejaNews). Por ejemplo, buscar la cadena "cgi-perl" da una explicación más detallada de este paquete que el breve campo de descripción de su fichero de control.

14 El Sistema de Notificación de Bugs de Debian

14.1 ¿Existen registros de bugs conocidos?

La distribución de Linux Debian tiene un sistema de seguimiento de bugs que almacena detalles de bugs notificados por usuarios y desarrolladores. Se asigna un número a cada bug y se mantiene almacenado hasta que es marcado como resuelto.

Hay disponibles copias de esta información en `<http://www.debian.org/Bugs/>`.

Un servidor de correo proporciona acceso a través de correo electrónico a la base de datos del sistema de seguimiento. Para obtener las instrucciones envíe un mensaje a request@bugs.debian.org con la palabra "help" en el cuerpo del mensaje.

14.2 ¿Cómo informo de un bug en Debian?

Si ha encontrado un bug en Debian, por favor lea las instrucciones para informar de un bug en Debian. Estas instrucciones pueden obtenerse de cualquiera de las siguientes formas:

- Por ftp anónimo. Los sitios réplica de Debian contienen las instrucciones en el fichero `doc/bug-reporting.txt`.
- Desde la WWW. Hay una copia de las instrucciones en `<http://www.debian.org/Bugs/Reporting.html>`
- En cualquier sistema Debian con el paquete `doc-debian` instalado. Las instrucciones se encuentran en el fichero */usr/doc/debian/bug-reporting.txt*.

Use estas direcciones de correo para enviar informes de bugs:

- submit@bugs.debian.org: para informes de bugs generales. Deberá recibir una confirmación automática de su notificación. También se le asignará automáticamente un número de seguimiento de bug, entrará en el registro de bugs y será enviado a la lista de correo debian-bugs-dist.
- maintonly@bugs.debian.org: para enviar informes de bugs sólo al que mantiene el paquete. No será enviado a la lista de correo debian-bugs-dist.
- quiet@bugs.debian.org: para enviar informes de bugs sólo al registro de bugs, sin enviarlos a debian-bugs-dist o al que mantiene el paquete.

Por favor, tenga en cuenta que existe un comprobador de paquetes Debian llamado *Lintian* `<http://www.debian.org/lintian/>`, que está diseñado para comprobar de forma automática violaciones de las normas y otros errores de empaquetado comunes de los paquetes Debian. De esta forma, si usted detecta un bug en un paquete que sea probable que aparezca también en otros paquetes, podría ser mejor ponerse en contacto con los *encargados de Lintian* `<mailto:lintian-maint@debian.org>` de forma que se escriba una nueva comprobación para Lintian en lugar de informar del bug directamente. Esto evitará que el bug aparezca de nuevo en versiones posteriores del paquete, o de cualquier otro paquete de la distribución.

15 Contribuir al proyecto Debian

Las donaciones de tiempo (para desarrollar nuevos paquetes, mantener paquetes existentes, o proporcionar soporte a usuarios), recursos (para mantener réplicas de los archivos de FTP y WWW), y dinero (para pagos de nuevos "bancos de prueba" así como de hardware para los archivos) pueden ayudar al proyecto.

15.1 ¿Cómo puedo convertirme en un desarrollador de software para Debian?

El desarrollo de Debian está abierto a todos, y se necesitan nuevos usuarios con la habilidad suficiente y/o el deseo de aprender, para mantener paquetes existentes que hayan quedado "huérfanos" de sus responsables anteriores, para desarrollar nuevos paquetes, y para proporcionar soporte a usuarios.

Todos los detalles acerca de convertirse en un desarrollador Debian se pueden encontrar ahora en el paquete `developers-reference`. Debe por tanto instalar este paquete y leerlo con atención.

Existe una versión en línea de este documento *aquí* `<http://www.debian.org/doc/packaging-manuals/developers-reference/>`.

15.2 ¿Cómo puedo contribuir con recursos al proyecto Debian?

Como el proyecto aspira a formar un gran conjunto de software accesible rápida y fácilmente a través del globo, se necesitan urgentemente réplicas. Es deseable pero no absolutamente necesario tener una réplica de todo el archivo. Los componentes actuales del proyecto suman ya más de 6 GBytes, repartidos más o menos así:

- slink/main: 3100 MBytes.
- slink/contrib: 210 MBytes.
- slink/non-free: 570 MBytes.
- potato: 2100 MBytes (seguramente crecerá cuando vaya a publicarse).
- sid: 1120 MBytes.

La mayor parte del proceso de mantenimiento de una réplica se realiza automáticamente mediante scripts, sin necesidad de intervención humana. De todas formas, ocasionalmente puede ocurrir un fallo o cambio en el sistema que sí requiera la intervención humana.

Si tiene una conexión de alta velocidad a Internet, los recursos para mantener una réplica de toda o parte de la distribución, y está deseando utilizar su tiempo en proporcionar un mantenimiento regular al sistema (o en encontrar a alguien que lo haga), por favor póngase en contacto con `debian-admin@debian.org`.

15.3 ¿Cómo puedo contribuir económicamente al proyecto Debian?

Uno puede realizar donaciones individuales a una de las dos organizaciones que son críticas para el desarrollo del proyecto Debian.

15.3.1 Software in the Public Interest

Software in the Public Interest (SPI) es una organización sin ánimo de lucro formada cuando FSF retiró su patrocinio sobre Debian. Actualmente estamos en el proceso de incorporación como una organización IRS 501(c)(3) sin ánimo de lucro. El propósito de la organización es desarrollar y distribuir software libre. Nuestros objetivos son muy similares a los de la FSF, y animamos a los programadores a usar la Licencia Publica General de GNU (GNU General Public License) en sus programas. De todos modos, nosotros tenemos un objetivo ligeramente distinto, en el sentido de que estamos construyendo y distribuyendo un sistema Linux que diverge en muchos detalles técnicos del sistema GNU planeado por la FSF. Seguimos en comunicación con la FSF, y cooperamos enviándoles cambios a software GNU y pidiendo a nuestros usuarios donaciones para la FSF y el proyecto GNU.

Para dirigirse a SPI: `<http://www.spi-inc.org/>`

15.3.2 Free Software Foundation

En este momento no hay ninguna relación formal entre Debian y la Free Software Foundation. De todos modos, la Free Software Foundation es responsable de algunos de los componentes de software más importantes en Debian, incluyendo el compilador de C de GNU, GNU Emacs, y gran parte de la biblioteca C que es usada por todos los programas del sistema en tiempo de ejecución. FSF es pionera de gran parte de lo que es hoy en día el software libre: ellos escribieron la Licencia Pública General que se usa en gran parte del software Debian, e inventaron el proyecto "GNU" para crear un sistema Unix completamente libre. Debian debe considerarse un descendiente del sistema GNU.

Al hacer una donación a la FSF, asegúrese de comentarles que es un usuario de Debian. Por favor, contacte con ellos en:

- E-mail: *gnu@gnu.org*
- Dirección Postal:

```
Free Software Foundation
59 Temple Place - Suite 330
Boston, MA  02111-1307
EE. UU.
```

- Número de teléfono: +1-617-542-5942
- Fax (incluyendo Japón): +1-617-542-2652
- Free Dial Fax (en Japón): 0031-13-2473 (KDD), 0066-3382-0158 (IDC)
- En la WWW : `<http://www.gnu.org/>`

16 Redistribución de Debian GNU/Linux en un producto comercial

16.1 ¿Puedo producir y vender CDs de Debian?

Adelante. No necesita permiso para distribuir nada que hayamos *publicado*, así que puede crear las copias maestras de su CD tan pronto como finalicen las pruebas beta. No tiene que pagarnos nada a nosotros. De todos modos, publicaremos una lista de fabricantes de CD que donen dinero, software y tiempo al proyecto Debian, y recomendaremos a los usuarios para que compren de los fabricantes que hayan donado, así que es una buena propaganda hacer donaciones. Por supuesto todos los fabricantes de CD deben cumplir con las licencias de los programas que se incluyen en Debian. Por ejemplo, muchos de los programas están cubiertos por la GPL, lo que obliga a distribuir su código fuente.

16.2 ¿Puede incluirse Debian en un paquete junto con software no libre?

Sí. Aunque todos los componentes principales de Debian son software libre, ofrecemos un directorio non-free para programas que no sean libremente redistribuibles. Los fabricantes de CD *tal vez* tengan permiso para distribuir programas que hayamos puesto en ese directorio, dependiendo de los términos de cada licencia o de sus acuerdos privados con los autores de esos paquetes de software. Los fabricantes de CD pueden también distribuir en el mismo CD software no libre que obtengan de otras fuentes. Esto no es nada nuevo: hoy en día muchos fabricantes distribuyen software libre y comercial en el mismo CD. Por supuesto, seguimos recomendando a los autores de software a que publiquen como software libre los programas que escriben.

16.3 Estoy creando una distribución de Linux especial para un "mercado vertical". ¿Puedo usar Debian GNU/Linux como base del sistema Linux y añadir mis propias aplicaciones sobre él?

Sí. Por ejemplo, una persona está construyendo una distribución "Linux para Hams", con programas especializados para Radioaficionados. Ha empezado con Debian como "sistema base", y añadido programas para controlar el transmisor, seguir satélites, etc. Todos los programas que añade están empaquetados con el sistema de paquetes de Debian para que sus usuarios puedan actualizarlos fácilmente cuando publique los siguientes CDs.

Debian también proporciona un mecanismo que permite a los desarrolladores y administradores de sistema instalar versiones locales de ficheros seleccionados de modo tal que no sean sobrescritos cuando otros paquetes sean actualizados. Esto se discute más a fondo en la pregunta sobre 12.7 (desvíos).

16.4 ¿Puedo hacer un "paquete" Debian de mi programa comercial, para que se instale sin esfuerzo en cualquier sistema Debian?

Adelante. La utilidad de empaquetado es software libre.

17 ¿Qué tipo de cambios puedo esperar que aparezcan en la próxima versión de Debian GNU/Linux?

17.1 Mayor seguridad

Debian ahora dispone de contraseñas "Shadow".

Además, también estará disponible la biblioteca Linux de módulos de autentificación "enchufables" (Pluggable Authentication Modules `<http://parc.power.net/morgan/Linux-PAM/>`; también conocida como `libpam`) que permite a los administradores de sistemas escoger el modo de autorización para cada aplicación. Inicialmente estará configurada para autentificación a través de contraseñas shadow.

17.2 Más soporte para usuarios no angloparlantes

Debian ya tiene algún 5.10 (soporte para usuarios no angloparlantes). Esperamos encontrar gente que proporcione soporte para más idiomas.

Algunos programas ya soportan internacionalización, así que necesitamos catálogos de mensajes. Muchos programas aún deben ser internacionalizados.

17.3 Más arquitecturas

Actualmente sólo hay disponible una versión completa de Debian para las arquitecturas i386 y m68k. Se espera disponer pronto de un sistema Debian completo en otras arquitecturas.

17.4 Linux 2.2

Debian 2.1 será la última versión de Debian basada en Linux 2.0.x. La versión siguiente, de nombre clave `potato` probablemente incluirá el núcleo Linux estable más reciente, versión 2.2.

17.5 glibc 2.1

Debian 2.1 será la última versión de Debian basada en GNU libc 2.0.x. `potato` probablemente incluirá la biblioteca GNU libc estable más reciente, versión 2.1.x (que ya se está usando en `slink` en alguna de las arquitecturas no Intel).