

# Todo lo que siempre quisiste saber sobre urpmi pero nunca te atreviste a preguntarlo

Traducido por Willy Walker de <http://mandrake.vmlinuz.ca/bin/view/Main/UsingUrpmi>

[Descargalo en PDF](#)

## Otros recursos para aprender sobre urpmi

Urpmi es una importante herramienta para todos los usuarios de Mandriva. Tomate tiempo para aprender utilizarlo. Esta página te da una descripción de las opciones más comúnmente usadas. Debajo están otros recursos con una información más detallada sobre urpmi:

- <http://www.urpmi.org/> : Página de buena documentación de urpmi en Francés y en Inglés.
- Páginas man: comprueba las páginas man para todas las opciones. Ésas son la fuente más actualizada de información.

Junto a una introducción muy básica, esta página intenta cubrir lo que no se cubre en las dos fuentes antedichas de información. Asumimos que sabes utilizar una página man y que has leído la página antedicha. Una vez que lo hayas hecho así, vuelve a esta página: hay más información sobre problemas no tan obvios que puede no funcionarte.

Usando urpmi

## Lista rápida de tareas comunes

Comando	Que te dice
<code>urpmq -i xxx.rpm</code>	Información del programa
<code>urpmq -il xxx.rpm</code>	Información y los archivos que instala
<code>urpmq --changelog xxx.rpm</code>	changelog (cambios)
<code>urpmq -R xxx.rpm</code>	Que requiere este rpm
<code>urpmf ruta/a/archivo</code>	Que rpm proporciona este archivo
<code>rpm -q --whatprovides ruta/a/archivo</code>	similar a urpmf, pero trabaja con ambos <code>hdlist.cz</code> y <code>synthesis.hdlist.cz</code>
<code>urpmi.update updates</code>	Actualizaciones disponibles desde sus fuentes de actualización
<code>urpmc</code>	Actualizaciones disponibles desde todas las fuentes urpmi (puede necesitar urpmi a urpmc primero)
<code>urpmq --list-media</code>	Lista los repositorios que tienes

Comando	Que hace
<code>urpme xxxx</code>	Elimina el rpm (y dependencias)
<code>urpmi -y xxx</code>	Muestra todos los rpms que coinciden con esta cadena. (cuidado: si hay solo una coincidencia trata de instalarlo)
<code>urpmi --keep xxx.rpm</code>	Instala rpm y sus dependencias, pero si algo ha sido eliminado parará ese hilo
<code>urpmi --update --auto-select</code>	Instala actualizaciones disponibles desde tu fuente de actualización
<code>urpmi --keep --auto --auto-select</code>	Actualiza todos los paquetes pero no borra nada, solo dice que no funciona
<code>urpmi --auto-select</code>	Mira que necesita ser actualizado y te avisa para que digas si o no

## urpmi.addmedia

Aquí es donde todo comienza. Añade un **repositorio urpmi usando el comando** `urpmi.addmedia` . Puedes obtener una lista de fuentes urpmi de la página [Easy Urpmi](#) . El formato para añadir un nuevo repositorio es como este:

```
urpmi.addmedia nombrerepositorio ftp://ftp.site.com/path/to/Mandriva/RPMS with ../base/hdlist.cz
```

Los espejos ftp de Mandriva-Linux contienen:

- main que contiene la mayoría de los programas.
- contrib que contiene algunos programas opcionales.
- update que se usa para corregir problemas de seguridad. Todo el mundo debería tener esta fuente configurada, incluso con una conexión lenta a internet. Al agregar una fuente de la actualización agrega **'--update'** así el urpmi puede distinguirlo de una fuente regular.

Puedes usar `synthesis.hdlist.cz` en vez de `hdlist.cz` si tu ancho de banda es pequeño, pero perderás un montón de funcionalidades... Por ejemplo, `urpmf` no puede buscar en paquetes no instalados si estás usando `synthesis.hdlist.cz`.

## Recuperando fuentes perdidas (cds)

Si ves que no tienes todos tus CDs listados como fuentes, prueba `urpmi.addmedia --distrib cdrom removable://mnt/cdrom` con el `cd1` en tu unidad

## Copiando CDs al disco duro y usándolo como fuente.

Si no te gusta hacer juegos malabares con CDs durante sesiones con urpmi y tienes un cierto sitio para guardar, haz un directorio (digamos: `/home/CDS/`), copia la `base/` y directorios de Mandriva/ (recurrentemente) del primer CD de instalación allí. Entonces copia el `RPMS2` a los directorios `RPMS8` en

el directorio previamente creado de Mandriva/. Después de quitar o de inhabilitar la instalación CDs como fuentes haz: `=urpmi.addmedia --distrib HD file:///home/CDS=`

Este no funciona en la versión 10.1 cuando la estructura de directorio en los CDs se ha cambiado. Los directorios de los cuales tienes que copiar todo los RPMS ahora son:

```
CD1: /media/main
CD2: /media/main2
CD3: /Applications
    /Drivers
    /media/commercial
    /media/dkms
    /media/drivers
    /main3
CD4: /kde3.3
    /media/main4
CD5: /media/main5
CD6: /Applications
    /media/main6
```

Después de que estas copias están completas, puedes hacer (como root)

```
cd /home/CDS (o el directorio que creaste )
genhdlist --dest `pwd`
```

Entonces usa el gestor de medios en el Centro de Control de Linux Mandriva para configurar tu directorio como fuente. Después de esto, deshabilita todas las entradas de cd.

Un método alternativo de copiar y montar las imágenes ISO - que también funcionará en versiones más antiguas - lo puedes ver en [CdToHD](#)

## Copiando rpms (incluyendo CDs de instalación) a un solo directorio y como fuente .

Crear un directorio, por ejemplo/RPMS, para mantener todos los rpms

Copiar un montón de rpms (de instalar cds, por ejemplo):

```
find /INSTALL_CDS/ -name *.rpm -print -exec cp {} /RPMS \;
```

Si has copiado los CDs instalación al disco duro, o tienes sus imágenes ISO en la unidad de disco (ver [CdToHD](#) para ver como montarlos), esto puede hacerse en un paso.

En este ejemplo, los CDs fueron montados como /INSTALL\_CDS/CD1, /INSTALL\_CDS/CD2, etc.

```
cd /RPMS
```

```
genhdlist
```

Esto creará entonces un archivo `hdlist.cz` y un archivo `sythesis.hdlist.cz` basado en lo que encuentra en el directorio en el que está.

Busca pubkeys para los rpms y cópialos en `/RPMS/pubkey`. Para ML LE2005, están en `CD1/media/media_info` :

```
mkdir /RPMS/pubkey
```

```
cp /INSTALL_CDS/CD1/media/media_info/pubkey* /RPMS/pubkey/
```

entonces añade la fuente:

```
urpmi.addmedia local_rpms file://RPMS/ with hdlist.cz
```

Se ha conocido que esto funciona bien para unos, y no tan bien para otros. Un problema que pasa a unos es que mirando instalables rpms usando `rpm-drake`, la información bajo el sumario está perdida (Es decir `filelist`...).

## urpmi.update

El comando `urpmi.update` actualiza la lista de paquetes de un repositorio `urpmi`. La lista cambia cuando cambia en el servidor, necesitas así actualizar siempre que desees instalar un nuevo paquete de un repositorio cambiante, como [¿¿MandrivaCooker??](#). Si estás utilizando un repositorio para un lanzamiento, eso normalmente no cambia, no necesitas actualizar la lista siempre que instales algo. Funciona así:

```
urpmi.update nombre_repositorio
```

o

```
urpmi.update -a
```

El argumento `-a` significa "todos los media".

## urpmi.removemedias

Si quieres librarte de un repositorio, este es el comando a usar. Se usa así:

```
urpmi.removemedias nombre_repositorio
```

## urpmmf

`urpmmf` es, en mi experiencia, lo mejor de las partes desconocidas del sistema de manejo de paquetes `urpmi`. Con él puedes encontrar cualquier archivo que desees, conocer el paquete al que pertenece, ver una descripción del paquete y un montón de cosas más.

Supón, que quieres compilar un visor de imágenes en tu PC. Descomprimes el tarball, y ejecutas el `./configure`. Ahora, el script de configuración se queja de que falta `jpeglib.h`. (\*NOTA: el script de configuración usualmente dice "jpeg library missing", así este ejemplo no es tan relevante) Todo lo que tienes que hacer es `urpmmf jpeglib.h`.

Esto responderá:

```
libjpeg62-devel:/usr/include/jpeglib.h
```

```
mozilla-devel:/usr/include/mozilla-1.4a/jpeg/jpeglib.h
```

Así sabes que la cabecera jpeglib es parte de libjpeg62-devel, y ahora puedes instalarlo usando urpmi libjpeg62-devel. Eso te ahorra un montón de tiempo :-)

También, puedes querer instalar un cliente de correo-e, pero no conoces un cliente para Linux. Así que quieres realizar una búsqueda en el sumario de los paquetes, con el término 'mail' y 'client'. El comando `urpmf --summary 'mail' -a client` devuelve

```
evolution:Integrated GNOME mail client, calendar and address book.
```

```
squirrelmail:Squirrelmail is a webmail client for PHP4.
```

```
sylpheed-claws:Enhanced version of the Sylpheed e-mail client
```

```
comsat:A mail checker client and comsat mail checking server.
```

```
cscmail:CSCMail is a GTK email client written in Perl
```

```
sylpheed:A GTK+ based, lightweight, and fast e-mail client
```

```
tradeclient:Email Client with PIM features for X
```

La opción `--summary` se usa para realizar una búsqueda en el sumario, y el `-a` traduce 'y'.

## urpmi

Éste es el comando principal de la administración del paquete, que utilizas para instalar el paquete que deseas y toda lo que dependa de él, así que no tendrás que ir buscándolo todo. Se usa así

```
urpmi nombrepaquete
```

Por ejemplo, crees que Sylpheed es un buen cliente de correo-e porque se describe así mismo como 'lightweight' (ligero). Así que lo instalas con `urpmi sylpheed`.

Advertencia: haciendo un `urpmi nombrepaquete` usualmente pero no necesariamente instala la última versión del paquete que está siendo instalado. Urpmi comprobará el número de versión del paquete instalado con el mismo paquete en la fuente urpmi. Los excepciones son cuando una mejora importante de algunos paquetes está disponible bajo un nombre levemente diferente de modo que la vieja y nueva versión sigan estando disponible. Es a veces necesario hacerlo así para propósitos de compatibilidad hacia atrás.

Aquí hay un ejemplo notable de `urpmi autoconf` no dará como resultado la última versión que está siendo instalada, para la versión más alta bajo el nombre del paquete 2.13. Puedes hacer `urpmi autoconf2.5` en su lugar, que te dará la última 2.5\*. Nota que en tu sistema Mandriva, ambas versiones se instalaran: Mandriva elegirá automáticamente la más apropiada. Mira `/usr/share/doc/autoconf-2.13/IMPORTANT.README`.

*MDK* para más información de este ejemplo particular.

Mira `urpmq --fuzzy` antes para aprender tales problemas de versiones.

Un ejemplo típico del uso de `urpmi` es actualizar tu sistema con las últimas actualizaciones de seguridad y arreglos de errores.

```
urpmi.update updates && urpmi --update --auto-select
```

Este ejemplo asume que tienes una fuente `urpmi` llamada 'updates' que fue declarada para ser 'update source' cuando fue añadida. Porque el contenido de update sources cambia frecuentemente es necesario realizar un **urpmi.update** para comprobar nuevos paquetes. El carácter '&&' significa que si el primer comando tiene éxito, entonces ejecuta el segundo comando. En la segunda parte del comando el '**--update**' significa solo una búsqueda de repositorios declarados como fuentes de actualización, y el '**--auto-select**' significa 'buscar las últimas versiones de los paquetes ya instalados'. `Urpmi` mostrará una lista de paquetes actualizables y te preguntará si estás de acuerdo.

Si añadiste '**--auto**' el proceso será automático y todos los paquetes actualizables se actualizarán sin pedir confirmación. Si prefieres usar una GUI prueba la GUI de Actualización de Mandriva en el Centro de Control de Mandriva.

## urpme

Este es el comando que desinstala el software de tu sistema. Como `urpmi`, comprueba las dependencias y te dice si algún software que todavía esté instalado depende de lo que deseas desinstalar, dándote la ocasión de abortar el procedimiento de desinstalación del paquete y los que dependan de él. El uso es:

```
urpme nombrepaquete
```

Por ejemplo, Sylpheed es tan bueno que quieres librarte de Evolution, tu cliente de correo-e anterior. Puedes liberar espacio en tu disco con `urpme evolution`.

## urpmq

el `urpmq` permite que preguntes a la base de datos del `urpmi`, así puedes descubrir información sobre los paquetes que has instalado o que puedes instalar así como otras cosas en la base de datos, como qué fuentes de los medios has configurado, como esto:

```
[root@isis root]# urpmq --list-media  
contrib  
MandrivaClub  
club.comm_i586_9.0  
cooker  
plf
```

Advertencia: al usar el `urpmq` para buscar paquetes, asegúrate de entender la diferencia entre usarla con o sin la opción `--fuzzy`. Puede ser que falte algunos paquetes que buscabas si no tienes cuidado. Si el `urpmq` encuentra un paquete con el nombre exacto que preguntaste, volverá este paquete solamente y por defecto,

no verás otros paquetes que incluyan el mismo nombre.

Por ejemplo:

```
// Imagina que tu mejor amigo te habló sobre vegastrike, un estupendo juego 3D.
```

```
[root@localhost augustin]# urpmq vegastrike
```

```
vegastrike
```

```
// ¡Estupendo! ¡Hay un paquete de Mandriva con este nombre exacto!
```

```
// ¡Pero no ves otros paquetes que incluyan este nombre!
```

```
// Preguntamos otra vez: nota la 'e' que falta en el final del nombre del paquete.
```

```
[root@localhost augustin]# urpmq vegastrik
```

The following packages contain vegastrik:

```
vegastrike
```

```
vegastrike-data
```

```
vegastrike-data-music
```

```
// Esta vez el resultado de la consulta es muy diferente:
```

```
// Vemos no uno, sino tres paquetes.
```

```
// Compara la siguiente consulta con las dos anteriores.
```

```
[root@localhost augustin]# urpmq --fuzzy vegastrike
```

The following packages contain vegastrike:

```
vegastrike
```

```
vegastrike-data
```

```
vegastrike-data-music
```

```
// Ve la diferencia que la opción --fuzzy hace.
```

**urpmq --fuzzy** es también estupendo ver lo que puede ser la otra versión disponible allí de un paquete: una nueva versión importante no proporciona a veces por completo compatibilidad posterior. Por esta razón la más nueva versión será distribuida bajo nombre levemente diferente del paquete. Ahora, estás seguro de que no falta nada.

Por ejemplo:

```
[root@localhost augustin]# urpmi mplayer
```

```
// Esto NO instalará la última versión de mplayer.
```

```
[root@localhost augustin]# urpmq --fuzzy mplayer
```

The following packages contain mplayer:

kmplayer

mplayer

mplayer-fonts

mplayer-gui

mplayer-skins

mplayer1.0

mplayer1.0-gui

mplayerplugin

transcode

xmms-mplayer

```
// Ves que: la nueva edición 1.0 está en un paquete separado.
```

```
// ¡¡Has sido advertido!!
```

## Instalando un nuevo kernel

Los núcleos nunca se actualizan, ellos están instalados siempre. Nunca verás tu kernel actualizado cuando ejecutas la actualización de Mandriva. Para instalar un nuevo kernel desde la línea de comando:-

```
# urpmi.update updates /obtienes la lista de actualizaciones disponibles desde tu fuente de actualización/
```

```
# urpmq --fuzzy kernel /lista los kernels disponibles /
```

```
# urpmi -y kernel /método alternativo listar los kernels disponibles/
```

The following packages contain kernel:

kernel-2.4.22.10mdk



kernel-2.4.22.26mdk

kernel-2.6-source

kernel-2.6.0-0.test5.1mdk

kernel-BOOT-2.4.22.10mdk

kernel-enterprise-2.4.22.10mdk

kernel-enterprise-2.4.22.26mdk

kernel-i686-up-4GB-2.4.22.10mdk

kernel-i686-up-4GB-2.4.22.26mdk

kernel-multimedia-2.4.22.21mm.1mdk

kernel-multimedia-2.4.22.26mm.1mdk

kernel-multimedia-smp-2.4.22.21mm.1mdk

kernel-multimedia-smp-2.4.22.26mm.1mdk

kernel-multimedia-source

kernel-secure-2.4.22.10mdk

kernel-secure-2.4.22.26mdk

kernel-smp-2.4.22.10mdk

kernel-smp-2.4.22.26mdk

kernel-smp-2.6.0-0.test5.1mdk

kernel-source

#urpmi kernel-2.4.22.26mdk /toma el kernel que quieres instalar/

El nuevo kernel se instalará en /boot al lado de tu viejos kernel. La configuración del Lilo se ajustará automáticamente en nueva línea para permitirte arrancar el viejo kernel. La línea por defecto de 'linux' arrancará el nuevo kernel. Si lo deseas puedes utilizar el software de instalación Mandriva con el GUI para instalar un nuevo kernel, pero debes ejecutar el GUI de actualización de Mandriva primero para forzar una actualización de la base de datos del urpmi.

Notarás que uno de los paquetes disponibles es 'kernel-source' que éste es el código de fuente del núcleo más reciente disponible en la fuente de la actualización (en este ejemplo 2.4.22.26mdk) necesitas solamente instalar el paquete de la fuente del kernel si necesitas compilar cualquier cosa como un controlador de

dispositivo. Asegúrate de que has instalado la misma versión del kernel-source que el kernel que tienes funcionando, o bien el controlador que compilas no funcionará.

## Usando interface gráfica

### Los Drakes

Mandriva ofrece un número de agradables herramientas gráficas para realizar las mismas tareas. Mira las [Drakes tools](#).

### urpmi.setup

urpmi.setup es un GUI para urpmi.addmedia y urpmi.remove, y está basado en la lista de espejos del Penguin Liberation Front.

## Una cuestión de seguridad

### Cuales son los elementos de seguridad

Urpmi debe ser ejecutado como root para instalar paquetes. Una inteligente pero maliciosa persona tiene la posibilidad de engañar a un no tan inteligente usuario de Mandriva (sé que todos los usuarios de Mandriva son listos, pero déjame asumir que hay un usuario incauto de Mandriva en alguna parte, solo para el motivo de la discusión) para instalar un paquete que parece un paquete regular usado por muchos pero que contiene un Caballo de Troya, un cierto código malvado. Una vez instalada, la aplicación puede dejar algún virus, gusano o incluso algún spyware... Hasta ahora, no hay noticia de que este exploit haya ocurrido, pero en unos pocos años, cuando GNU/Linux sea más ampliamente usado como SO, es posible que algunas personas podrían intentar realizar tal hazaña. Afortunadamente, fallarán, porque una preocupación sana por seguridad ha sido parte siempre de la comunidad de GNU/Linux. La mayoría de los paquetes son firmados por los embaladores para probar la autenticidad. Mira por favor:

Al lado de una persona malévola que intenta trampearle en la instalación de su malware, lo cuál es más probable que suceda es que usted intenta instalar algún paquete que obtuviste corrupto, durante descarga de ftp, o debido a un rasguño en tu Cd de instalación... Antes de instalar, el urpmi comprobará la integridad del paquete usando el código hash MD5. Mira la sección relevante abajo para los detalles.

### **rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format**

A veces, al usar urpmi, la gente acaba teniendo filas de:

```
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
```

```
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
```

```
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
```

```
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
```

```
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
```

```
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
```

```
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
```

```
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
```

Es confuso conocer cual es el origen del problema: el archivo se obtuvo quizá corrupto durante una actualización o cuando un proceso del urpmi fue matado.

La solución es quitar el archivo rpmdb: /var/lib/rpm/Pubkeys e importar las claves públicas otra vez.

P: Me gustaría ver más información aquí sobre importar claves públicas para urpmi y arreglar problemas con firmas no válidas. No conozco bastante para añadir esta información... --SimonOosthoek

R: Mira en <http://mandrake.vmlinuz.ca/bin/view/Main/GnuPG>, párrafo titulado "rpm package validation with [GnuPG](#) keys.

## MD5 Hash

El MD5 hash es un código dentro del RPM que comprueba si el RPM esté corrupto o no. Si por cualquier razón se corrompe el archivo, el RPM (y el urpmi) rechazarán instalarlo.

Aquí hay un ejemplo que me ocurrió recientemente:

```
[root@localhost augustin]# urpmi kdesdk
```

```
rpmdb: /var/lib/rpm/Pubkeys: unexpected file type or format
```

```
error: cannot open Pubkeys index using db3 - Invalid
```

```
argument (22)
```

The following packages have bad signatures:

```
/var/cache/urpmi/rpms/kdesdk-3.1.3-9mdk.i586.rpm: Invalid
```

```
signature ((SHA1) DSA sha1 MD5 GPG GPG#70771ff3 NOT OK)
```

Do you want to continue installation ? (y/N) y

```
installing /var/cache/urpmi/rpms/kdesdk-3.1.3-9mdk.i586.rpm
```

```
error: /var/cache/urpmi/rpms/kdesdk-3.1.3-9mdk.i586.rpm: MD5
```

```
digest: BAD Expected(97f2ba5a91888cd3af40f89be6b65868) !=
```

```
(393221db35071aa90eaa73816a9a5ba8)
```

unable to install package

```
/var/cache/urpmi/rpms/kdesdk-3.1.3-9mdk.i586.rpm
```

Fijate en las dos líneas Invalid signature ((SHA1) DSA sha1 MD5 GPG GPG#70771ff3 NOT OK) y MD5 digest: BAD Expected(97f2ba5a91888cd3af40f89be6b65868) ! (393221db35071aa90eaa73816a9a5ba8)= . The file kdesdk-3.1.3-9mdk.i586.rpm is corrupted and won't be installed.

La solución es eliminar el archivo de /var/cache/urpmi/rpms/ y descargarlo otra vez. Intenta instalarlo otra vez con urpmi.

Si todavía no funciona, significa que su espejo tiene un archivo corrupto. Intenta descargarlo manualmente a partir de dos o tres otros espejos y guárdalo en la carpeta /var/cache/urpmi/rpms/. Quita siempre manualmente el archivo previamente descargado antes de procurar utilizar otro espejo. Entonces instálalo otra vez con urpmi.

Sucede que el mismo archivo corrupto está reflejado en servidores de cada ftp. Si has intentado varios servidores y te ocurre siempre el mismo problema, ve a un foro de usuario de Mandriva o una lista de correo si otros usuarios tienen el mismo problema con el mismo archivo. Si éste es el caso, un archivo nuevo será lanzado probablemente pronto.

## Problemas

### RPM database locked

Como superusuario, escribe:

```
killall urpmi urpmi.update urpme rpm urpmi.addmedia
```

```
rm -f /var/lib/urpmi/.LOCK /var/lib/rpm/RPMLOCK
```

Si eso no desbloquea la base de datos, reiniciar es siempre la solución.

### Reconstruir la base de datos

Si la base de datos tiene inconsistencias (una inconsistencia puede conducir a el RPM para decirle que un "paquete está instalado ya" cuando no lo está realmente, y viceversa).

```
rm -rf /var/lib/rpm/___db*
```

```
rpm --rebuilddb
```

### El error "el medio contrib usa una lista no válida"

```
rm /var/lib/urpmi/list.contrib
```

Esto parará el error del suceso sin interferir con la instalación del paquete.

# Actualizar a la última versión de Mandriva usando urpmi

Urpmi también puede ser usado para [actualizar](#) tu sistema a una nueva versión de Mandriva Linux.

## Trucos usando urpmi

### Instalar desde url o archivo local

urpmi puede usarse para instalar archivos locales rpm, y entonces resolver dependencias. Ejemplo, has recompilado o creado foo-1.0-1bar.rpm. Puedes fácilmente usar urpmi ./foo-1.0-1bar.rpm. Incluso trabaja con http o ftp url ( e incluso con ssh, dado que el rsync está instalado en ambos lados).

### Obteniendo la lista de dependencias antes de instalar

Usando el **urpmq**, puedes encontrar porciones de información con respecto al RPM. Si utilizas el **urpmq -d**, consigues una lista de todos los RPM que sean necesarios para instalar un software. Prueba **urpmq -d bash**, para ver cuales son los rpm requeridos. Añadiendo el **-m**, verás solamente lo no instalado en el sistema en el momento. Y con **--sources**, obtendrás las fuentes.

Así pues, con el **urpmq -d -m --sources**, puedes fácilmente obtener una lista del URL que corresponden a la RPM que necesitas conseguir, absolutamente práctica si careces una buena conexión al Internet.

### Obteniendo información del rpm.

**urpmq -i** puede darte información sobre el rpm que vas a instalar. Prueba **urpmq -i bash**. Solo funciona con hdlists completas, no con synthesis.

## Instalando otro software

En lo posible, debes utilizar urpmi para instalar software, usando los paquetes de una fuente correcta del urpmi para tu versión actual de Mandriva. Casi todo que podrías desear está disponible, si tienes fuentes definidas para el contrib y el plf: ¡esto es una manera de evitar mucho dolor! Sin embargo, puedes desear a veces instalar un paquete diferente, o más reciente que el disponible de las fuentes de Mandriva.

La regla de oro es, **nunca desvíes la base de datos del paquete RPM**, si puedes ayudarle posiblemente, y nunca instales un paquete forzándolo. Si sigues esta regla, aunque el problema puede aparecer, tienes la garantía de nunca tener un sistema en estado inconsistente ¡Si rpm/urpmi no te deja instalar algo, hay generalmente una buena razón!

### Compilando de las fuentes

En vez de la familiar trilogía de:

```
./configure
```

```
make
```

make install

usa checkinstall en su lugar:

```
./configure
```

```
make
```

```
checkinstall
```

Entonces conseguirás un RPM que se pueda instalar usando urpmi, y puedes también desinstalarlo cuando lo desees.

## Instalando binarios como acrobat reader, o java de sun.

Estos programas deben estar en el directorio `/usr/local/`. Este directorio se piensa para los paquetes oficiosos. Sin embargo, puedes desear agregar algunos symlinks si no están en su `$PATH`. Ejemplo: `/usr/bin/java - /usr/java/j2re1.4.2_04/bin/java`

## Scripts

Sugiero que guardes los scripts que se escribes en `$HOME/bin`. (si esto no está en tu `$PATH`, agréguelo a su `bashrc`).

## El controlador de Nvidia

El controlador binario de nvidia, si lo usas, tiene su propio instalador horrible, que rompe todas las reglas antedichas. ¡El resultado funciona muy bien, pero el proceso no es agradable!

## Reconstruyendo RPMS

Si tienes una RPM construido para otro sistema con excepción tuyo propio, es posible reconstruirlo.

## Verificación de paquetes RPM

Una herramienta realmente útil para comprobar un sistema roto debes verificar los paquetes contra la base de datos de la RPM.

```
rpm -Va
```

Esto te dirá que qué paquetes han cambiado desde los instalaste (es decir enumera todas las maneras de las cuales tu sistema es inconsistente con la base de datos del RPM) por ejemplo, si tienes un apagón durante el arranque, algunos archivos pueden llegar a estar corruptos. (o si has hecho algo tonto). Sabiendo esto, puedes reparar tu sistema para estar tan bien como nuevo reinstalando los paquetes referidos. (esto se puede incluso hacer en un sistema no arrancable, si utilizas un CD de rescate para arrancar y repararlo)

Para comprobar un solo paquete usa `rpm -V nombrepaquete` (el nombre es suficiente, números de versión etc. se pueden omitir).

Para verificar en uno o más paquetes la firma apropiada de md5sum, el hash y el gpg: rpm -K foo.123.rpm, o rpm -K foo\*

## Madrugador lento

Si el msec está verificando tu sistema utiliza la base de datos de la RPM, pero eso se hace en 19, y generalmente muy temprano por la mañana. En un PC lento esto puede tomar un tiempo largo. Si ps -e muestra el msec, rpmv, grep y sort, puedes acelerar el rpmdrake/urpmi reiniciando el rpmw, grep y sort a 0. (si deseas tener un pequeño script para iniciar el rpmdrake, pero primero comprueba si el msec está haciendo esto, Correo-e [DickGevers](#)).

*Nota: El nombre del autor original se ha perdido, y se ha restaurado por ASAP*

-- [AnneWilson](#) - 18 Nov 2004

-- [DickGevers](#) - 26 Dic 2004 (early bird)

-- [DickGevers](#) - 13 Mayo de 2005 (Mandriva nombre cambiado; URLs< algunas correcciones)

-- [EricHuff](#) - 16 Jul 2005 Copiando rpms (incluyendo instalador de CDs)... (con ayuda de Dick G y James S)

-- [DickGevers](#) - 16 Jul 2005 (Nota <sup>(1)</sup>); idea de respaldo para base de datos RPM; ruta en .bash\_profile preferido . bashrc; algunos tipos ; rpmbuild desde tarball en 'compilación de fuente' ; consejo para hombre navegador.)



Copyright © 2003-2005 by the contributing authors. All material on this collaboration platform is the property of the contributing authors and is licensed under a [Creative Commons License](#) unless otherwise noted.

Ideas, requests, problems regarding TWiki? [Send feedback](#)