

# Sistemas de paquetes software

Departamento de Sistemas Telemáticos y Computación (GSyC)

gsvc-profes (arroba) gsvc.es

Octubre de 2010



©2010 GSyC  
Algunos derechos reservados.  
Este trabajo se distribuye bajo la licencia  
Creative Commons Attribution Share-Alike 3.0

# Empaquetado de ficheros

Almacenar varios ficheros en uno solo, no necesariamente con compresión

Utilidad:

- Más cómodo de manejar (copiar, enviar por correo, etc)
- Conservar metainformación (permisos) o incluso mayúsculas/minúsculas, tildes, etc si los ficheros van a pasar por un sistema de ficheros diferente
  - ISO9660 (cdrom)
  - vfat (Windows, discos externos, pendrives)
  - ntfs (Windows)

Comprime o descomprime 1 fichero

Extensión: `fichero.z`    `fichero.gz`

- Comprimir y descomprimir (borrando el original):

```
gzip fichero
```

```
gunzip fichero.gz
```

- Comprimir y descomprimir (manteniendo el original):

```
gzip -c fichero > fichero.gz
```

```
zcat fichero.gz > fichero
```

```
zcat fichero.gz | less
```

## tar + gzip

Comprime o descomprime varios ficheros, directorios

Extensión: `fichero.tar.gz`      `fichero.tgz`

- Empaquetar y comprimir:

```
tar -cvzf fichero.tgz fichero1 fichero2
```

- Descomprimir y desempaquetar:

```
tar -xvzf fichero.tgz
```

- Mostrar contenido:

```
tar -tzf fichero.tgz
```

Para empaquetar sin comprimir, basta omitir la opción z

```
tar -cvf fichero.tar *.jpg
```

```
tar -xvf fichero.tar
```

# WinZip

- A causa de unas licencias restrictivas, originalmente no había compresores para Linux. (Pero las aplicaciones Windows saben descomprimir .tgz)
- Descomprimir: `unzip fichero.zip`

Formato que ofrece compresión más alta que .gz, (empleando más CPU y memoria)

- Comprimir y descomprimir 1 fichero, borrando el original

```
bzip2 fichero
```

```
bunzip2 fichero.bz2
```

- Comprimir y descomprimir 1 fichero, manteniendo el original

```
bzip2 -c fichero > fichero.bz2
```

```
bunzip2 -c fichero.bz2 > fichero
```

- Comprimir y descomprimir varios ficheros, manteniendo el original

```
tar -cvjf fichero.bz2 fichero1 fichero2
```

```
tar -xvjf fichero.bz2
```

# Instalación de paquetes

- Método clásico para instalar programas:

Formato `.tgz`

Descomprimir y seguir las instrucciones del fichero `README`

Suele ser del estilo de

```
./configure
```

```
make compile
```

```
make install
```

Pero dentro de un `.tgz` puede haber cualquier cosa, no está normalizado y no puede automatizarse

- Sistema de gestión de paquetes

Colección de herramientas que automatizan la instalación, actualización y eliminación de programas.

- Gestión de paquetes, Debian y derivados  
Paquetes en formato `.deb`  
Se pueden manejar directamente con `dpkg`, o con `apt-get`, `aptitude`, `dselect`, o `synaptic`
- Gestión de paquetes, RedHat y derivados  
Paquetes en formato `.rpm`  
Se pueden manejar directamente con `rpm`, o con `up2date` o `yum`

# El sistema de paquetes de Debian

Los paquetes mantienen *dependencias* entre sí, de forma que la instalación de un paquete puede:

- *depender* de que se instale también otro
- *recomendar* que se instale también otro
- *sugerir* que se instale también otro
- *entrar en conflicto* con otro actualmente instalado

# dpkg

- Es la herramienta básica de gestión de paquetes, que es usada por las otras (dselect, apt-get, aptitude, synaptic).
- Usos principales:
  - `dpkg -i paquete_VVV-RRR.deb`  
Instala un paquete
  - `dpkg -r paquete`  
Desinstala (*remove*) un paquete, elimina todo excepto los ficheros de configuración
  - `dpkg -P paquete`  
Purga un paquete, eliminando incluso los ficheros de configuración
- Tiene muchas opciones. Puede esquivarse el esquema de dependencias (peligroso) con las opciones que empiezan por `--force-...`

## Versiones de Ubuntu:

nombre año.mes

Warty Warthog 4.10

Hoary Hedgehog 5.04

Breezy Badger 5.10

Dapper Drake 6.06 LTS

Edgy Eft 6.10

Feisty Fawn 7.04

Gutsy Gibbon 7.10

Hardy Heron 8.04 LTS

Intrepid Ibex 8.10

Jaunty Jackalope 9.04

Karmic Koala 9.10

Lucid Lynx 10.04 LTS

Maverick Meerkat 10.10

Natty Narwhal 11.04

Versión estándar: soportada 18 meses

LTS: Long Term Support: 3 años en escritorio y 5 en servidor

Ubuntu Desktop / Ubuntu Server Edition / Ubuntu Server Edition  
JeOS

Variantes de Ubuntu: Kubuntu, Xubuntu, Edubuntu, Ubuntu  
Netbook Edition, Ubuntu MID Edition, Ubuntu Studio

# apt

- La herramienta más sencilla de usar y más potente.
- Usa *repositorios*: sitios centralizados donde se almacenan paquetes
- Las direcciones de los repositorios se indican en el fichero `/etc/apt/sources.list`
- Los repositorios de ubuntu se dividen en 4 componentes
  - 1 *Main*. Soportado oficialmente por ubuntu. Libre
  - 2 *Restricted*. Soportado oficialmente. No libre
  - 3 *Universe*. No soportado oficialmente. Libre
  - 4 *Multiverse*. No soportado oficialmente. No libre

Además, se pueden añadir componentes de terceros

```
# deb cdrom:[Ubuntu 6.06 _Dapper Drake_ - Release i386 (20060531)]/ dapper main
deb http://archive.ubuntu.com/ubuntu edgy main restricted
deb http://security.ubuntu.com/ubuntu edgy-security main restricted
deb http://archive.ubuntu.com/ubuntu edgy-updates main restricted

## All community supported packages, including security- and other updates
deb http://archive.ubuntu.com/ubuntu edgy universe multiverse
deb http://security.ubuntu.com/ubuntu edgy-security universe multiverse
deb http://archive.ubuntu.com/ubuntu edgy-updates universe multiverse

# Google Picasa for Linux repository
deb http://dl.google.com/linux/deb/ stable non-free
```

# Uso básico de apt

El primer *front-end* fue *dselect*, muy potente y con un pésimo interfaz de usuario

Desde línea de mandatos se puede usar `apt-get`

A partir de 2005, Debian recomienda usar *aptitude*, que tiene la misma sintaxis que `apt-get`

- `aptitude update`  $\equiv$  `apt-get update`  
Consulta el repositorio para saber qué paquetes hay disponibles (no instala nada)
- `aptitude safe-upgrade`  $\equiv$  `apt-get upgrade`  
Actualiza todos los paquetes instalados a la última versión disponible (sin cambiar de distribución)
- `aptitude install paquete`  $\equiv$  `apt-get install paquete`  
Instala un paquete (resolviendo conflictos)

**ERROR FRECUENTE:** Hacer solo *aptitude update* no actualiza los paquetes. Normalmente lo que queremos es  
`sudo aptitude update; sudo aptitude upgrade`

Aunque indiquemos a nuestro sistema de paquetería que instale la última versión de un paquete, tal vez no sea posible. Se dice que el paquete está *retenido* (*hold*)

- El paquete depende de otro no incluido en la distribución actual
- El administrador lo ha retenido *a mano* (no le gusta, da problemas...)

Un paquete retenido para `apt-get` puede no estar retenido para `aptitude`. Y viceversa.

- `aptitude`:  
`sudo aptitude hold nombre_del paquete`  
`sudo aptitude unhold nombre_del paquete`
- `apt-get`:  
`sudo install feta`  
`sudo feta hold nombre_del paquete`  
`sudo feta unhold nombre_del paquete`

- `aptitude remove paquete`  
Desinstalar un paquete (resolviendo conflictos)
- `aptitude --purge remove paquete`  
Purgar un paquete (resolviendo conflictos)
- `aptitude dist-upgrade`  
Actualiza *agresivamente* todos los paquetes instalados, lo que puede incluir el paso a la versión más reciente de la distribución
- `aptitude clean`  
Borrar las copias descargadas de los `.deb`

# Otros mandatos interesantes

En los repositorios hay muchos paquetes ¿Cómo saber cuál necesito?

- `aptitude search cadena`  
Buscar una cadena en el nombre o descripción de un paquete. Indica el estado del paquete (instalado, no instalado, borrado...)
- `aptitude show paquete`  
Muestra descripción del paquete
- `dpkg-reconfigure paquete`  
Reconfigurar un paquete

# El sistema de paquetes de OpenWRT

Basado en opkg (fork de ipkg).  
Similar al de Debian, más sencillo

- `/etc/opkg.conf`  
URL del repositorio
- `opkg update`
- `opkg upgrade <paquete/s>`
- `opkg install <paquete/s>`
- `opkg reconfigure <paquete/s>`
- `opkg remove <paquete/s>`