dpkg Debian package tool

CONCEPTS

dpkg works on single .deb files and the local database. It does not fetch packages or resolve dependencies: use apt for that.

.deb file ar archive
↓ `--unpack`
Unpacked files placed
↓ `--configure`
Installed maintainer scripts run
↓ `-r`
Removed config kept (rc)
↓ `-P`
Purged config gone

The two-letter state in dpkg -l: ii is installed, rc is removed with config left, hi is installed and held.

The package database lives in /var/lib/dpkg; the action log is /var/log/dpkg.log.

INSTALL AND REMOVE

`sudo dpkg -i <file.deb>`
Install or upgrade from a file

`sudo dpkg -i <dir>/*.deb`
Install several files at once

`sudo dpkg -R <dir>`
Install every .deb in a directory

`sudo dpkg -r <pkg>`
Remove, keep config files

`sudo dpkg -P <pkg>`
Purge: remove with config files

`sudo dpkg --configure -a`
Finish interrupted configuration

`sudo dpkg-reconfigure <pkg>`
Rerun a package's configuration questions

`sudo dpkg --configure <pkg>`
Configure one unpacked package

Prefer sudo apt install ./<file.deb>: it installs a local file and pulls in dependencies.

QUERY INSTALLED

`dpkg -l`
List all packages

`dpkg -l '<pattern>'`
List packages matching a glob

QUERY INSTALLED (cont.)

`dpkg -s <pkg>`
Show status, version, depends

`dpkg -L <pkg>`
List files owned by a package

`dpkg -S <path>`
Find the package owning a file

`dpkg -l | grep ^ii`
Only installed packages

`dpkg -l | grep ^rc`
Removed packages with config left

`dpkg -C`
Audit for half-installed packages

`dpkg -V [<pkg>]`
Verify installed files against the database

`dpkg --print-architecture`
Native architecture

`dpkg --add-architecture <arch>`
Enable a foreign architecture

`dpkg --compare-versions <a> lt <b>`
Test version order (exit code)

DPKG-QUERY

`dpkg-query -W <pkg>`
Package and version

`dpkg-query -W -f='${Package} ${Version}\n'`
Custom output for all packages

`dpkg-query -W -f='${Installed-Size}\t${Package}\n' | sort -n`
Largest packages last

HOLDS AND SELECTIONS

`echo '<pkg> hold' | sudo dpkg --set-selections`
Hold a package at its version

`echo '<pkg> install' | sudo dpkg --set-selections`
Release a hold

`dpkg --get-selections`
List selections

`dpkg --get-selections | grep hold`
Show held packages

HOLDS AND SELECTIONS (cont.)

`dpkg --get-selections > <file>`
Save the package list

`sudo dpkg --set-selections < <file>`
Restore a saved list

`apt-mark hold <pkg>`
apt's equivalent of a hold

INSPECT .DEB FILES

`dpkg-deb -I <file.deb>`
Show control info

`dpkg-deb -c <file.deb>`
List contained files

`dpkg-deb -W <file.deb>`
Name and version only

`dpkg-deb -f <file.deb> <field>`
Print one control field

`dpkg-deb -x <file.deb> <dir>`
Extract files without installing

`dpkg-deb -e <file.deb> <dir>`
Extract control files and scripts

`dpkg-deb -R <file.deb> <dir>`
Extract files and control together

`dpkg-deb -b <dir> <file.deb>`
Build a .deb from a tree

DIVERT AND OVERRIDE

`sudo dpkg-divert --add --rename <path>`
Move a file aside for upgrades

`sudo dpkg-divert --remove --rename <path>`
Undo a diversion

RECIPES

Which package owns this command

```
# resolve symlinks first
dpkg -S "$(readlink -f "$(command -v <cmd>)")"
```

Look inside a download before installing

```
dpkg-deb -I <file.deb>
dpkg-deb -c <file.deb>
dpkg-deb -x <file.deb> /tmp/peek
```

Recover from a broken install

```
sudo dpkg --configure -a
sudo apt -f install
dpkg -C
```