NAME

apt-cache − query the APT cache

SYNOPSIS

apt−cache [**−agipns**] [**−o=***config_string*] [**−c=***config_file*] {gencaches | showpkg *pkg*... | showsrc *pkg*... |
stats | dump | dumpavail | unmet | search *regex*... |
show *pkg* [{=*pkg_version_number* | /*target_release*}]... |
depends *pkg* [{=*pkg_version_number* | /*target_release*}]... |
rdepends *pkg* [{=*pkg_version_number* | /*target_release*}]... | pkgnames [*prefix*] |
dotty *pkg* [{=*pkg_version_number* | /*target_release*}]... |
xvcg *pkg* [{=*pkg_version_number* | /*target_release*}]... | policy [*pkg*...] | madison*pkg*... |
{−v | −−version} | {−h | −−help}}

DESCRIPTION

apt−cache performs a variety of operations on APT's package cache. **apt−cache** does not manipulate the state of the system but does provide operations to search and generate interesting output from the package metadata. The metadata is acquired and updated via the 'update' command of e.g. **apt−get**, so that it can be outdated if the last update is too long ago, but in exchange **apt−cache** works independently of the availability of the configured sources (e.g. offline).

Unless the **−h**, or **−−help** option is given, one of the commands below must be present.

gencaches

gencaches creates APT's package cache. This is done implicitly by all commands needing this cache if it is missing or outdated.

showpkg *pkg***...**

showpkg displays information about the packages listed on the command line. Remaining arguments are package names. The available versions and reverse dependencies of each package listed are listed, as well as forward dependencies for each version. Forward (normal) dependencies are those packages upon which the package in question depends; reverse dependencies are those packages that depend upon the package in question. Thus, forward dependencies must be satisfied for a package, but reverse dependencies need not be. For instance, **apt−cache showpkg libreadline2** would produce output similar to the following:

Package: libreadline2
Versions: 2.1−12(/var/state/apt/lists/foo_Packages),
Reverse Depends:
libreadlineg2,libreadline2
libreadline2−altdev,libreadline2
Dependencies:
2.1−12 − libc5 (2 5.4.0−0) ncurses3.0 (0 (null))
Provides:
2.1−12 −
Reverse Provides:

Thus it may be seen that libreadline2, version 2.1−12, depends on libc5 and ncurses3.0 which must be installed for libreadline2 to work. In turn, libreadlineg2 and libreadline2−altdev depend on libreadline2. If libreadline2 is installed, libc5 and ncurses3.0 (and ldso) must also be installed; libreadlineg2 and libreadline2−altdev do not have to be installed. For the specific meaning of the remainder of the output it is best to consult the apt source code.

stats

stats displays some statistics about the cache. No further arguments are expected. Statistics reported are:

- Total package names is the number of package names found in the cache.
- Normal packages is the number of regular, ordinary package names; these are packages that bear a one−to−one correspondence between their names and the names used by other packages

for them in dependencies. The majority of packages fall into this category.

- Pure virtual packages is the number of packages that exist only as a virtual package name; that is, packages only "provide" the virtual package name, and no package actually uses the name. For instance, "mail−transport−agent" in the Debian system is a pure virtual package; several packages provide "mail−transport−agent", but there is no package named "mail−transport−agent".
- Single virtual packages is the number of packages with only one package providing a particular virtual package. For example, in the Debian system, "X11−text−viewer" is a virtual package, but only one package, xless, provides "X11−text−viewer".
- Mixed virtual packages is the number of packages that either provide a particular virtual package or have the virtual package name as the package name. For instance, in the Debian system, "debconf" is both an actual package, and provided by the debconf−tiny package.
- Missing is the number of package names that were referenced in a dependency but were not provided by any package. Missing packages may be an evidence if a full distribution is not accessed, or if a package (real or virtual) has been dropped from the distribution. Usually they are referenced from Conflicts or Breaks statements.
- Total distinct versions is the number of package versions found in the cache. If more than one distribution is being accessed (for instance, "stable" and "unstable"), this value can be considerably larger than the number of total package names.
- Total dependencies is the number of dependency relationships claimed by all of the packages in the cache.

showsrc *pkg***...**
showsrc displays all the source package records that match the given package names. All versions are shown, as well as all records that declare the name to be a binary package. Use **−−only−source** to display only source package names.

dump
dump shows a short listing of every package in the cache. It is primarily for debugging.

dumpavail
dumpavail prints out an available list to stdout. This is suitable for use with **dpkg**(1) and is used by the **dselect**(1) method.

unmet
unmet displays a summary of all unmet dependencies in the package cache.

show *pkg***...**
show performs a function similar to **dpkg −−print−avail**; it displays the package records for the named packages.

search *regex***...**
search performs a full text search on all available package lists for the POSIX regex pattern given, see **regex**(7). It searches the package names and the descriptions for an occurrence of the regular expression and prints out the package name and the short description, including virtual package names. If **−−full** is given then output identical to show is produced for each matched package, and if **−−names−only** is given then the long description is not searched, only the package name and provided packages are.

Separate arguments can be used to specify multiple search patterns that are and'ed together.

depends *pkg***...**
depends shows a listing of each dependency a package has and all the possible other packages that can fulfill that dependency.

rdepends *pkg***...**

rdepends shows a listing of each reverse dependency a package has.

pkgnames [*prefix*]
This command prints the name of each package APT knows. The optional argument is a prefix match to filter the name list. The output is suitable for use in a shell tab complete function and the output is generated extremely quickly. This command is best used with the **−−generate** option.

Note that a package which APT knows of is not necessarily available to download, installable or installed, e.g. virtual packages are also listed in the generated list.

dotty *pkg***...**
dotty takes a list of packages on the command line and generates output suitable for use by dotty from the **GraphViz**[1] package. The result will be a set of nodes and edges representing the relationships between the packages. By default the given packages will trace out all dependent packages; this can produce a very large graph. To limit the output to only the packages listed on the command line, set the APT::Cache::GivenOnly option.

The resulting nodes will have several shapes; normal packages are boxes, pure virtual packages are triangles, mixed virtual packages are diamonds, missing packages are hexagons. Orange boxes mean recursion was stopped (leaf packages), blue lines are pre−depends, green lines are conflicts.

Caution, dotty cannot graph larger sets of packages.

xvcg *pkg***...**
The same as dotty, only for xvcg from the **VCG tool**[2].

policy [*pkg*...]
policy is meant to help debug issues relating to the preferences file. With no arguments it will print out the priorities of each source. Otherwise it prints out detailed information about the priority selection of the named package.

madison *pkg***...**
apt−cache's madison command attempts to mimic the output format and a subset of the functionality of the Debian archive management tool, madison. It displays available versions of a package in a tabular format. Unlike the original madison, it can only display information for the architecture for which APT has retrieved package lists (APT::Architecture).

OPTIONS

All command line options may be set using the configuration file, the descriptions indicate the configuration option to set. For boolean options you can override the config file by using something like **−f−**,**−−no−f**, **−f=no** or several other variations.

−p, **−−pkg−cache**
Select the file to store the package cache. The package cache is the primary cache used by all operations. Configuration Item: Dir::Cache::pkgcache.

−s, **−−src−cache**
Select the file to store the source cache. The source is used only by gencaches and it stores a parsed version of the package information from remote sources. When building the package cache the source cache is used to avoid reparsing all of the package files. Configuration Item: Dir::Cache::srcpkgcache.

−q, **−−quiet**
Quiet; produces output suitable for logging, omitting progress indicators. More q's will produce more quietness up to a maximum of 2. You can also use **−q=#** to set the quietness level, overriding the configuration file. Configuration Item: quiet.

−i, **−−important**
Print only important dependencies; for use with unmet and depends. Causes only Depends and Pre−Depends relations to be printed. Configuration Item: APT::Cache::Important.

−−no−pre−depends, **−−no−depends**, **−−no−recommends**, **−−no−suggests**, **−−no−conflicts**,

−−no−breaks, **−−no−replaces**, **−−no−enhances**
Per default the **depends** and **rdepends** print all dependencies. This can be tweaked with these flags which will omit the specified dependency type. Configuration Item: APT::Cache::Show*DependencyType* e.g. APT::Cache::ShowRecommends.

−−implicit
Per default **depends** and **rdepends** print only dependencies explicitly expressed in the metadata. With this flag it will also show dependencies implicitly added based on the encountered data. A Conflicts: foo e.g. expresses implicitly that this package also conflicts with the package foo from any other architecture. Configuration Item: APT::Cache::ShowImplicit.

−f, **−−full**
Print full package records when searching. Configuration Item: APT::Cache::ShowFull.

−a, **−−all−versions**
Print full records for all available versions. This is the default; to turn it off, use **−−no−all−versions**. If **−−no−all−versions** is specified, only the candidate version will be displayed (the one which would be selected for installation). This option is only applicable to the show command. Configuration Item: APT::Cache::AllVersions.

−g, **−−generate**
Perform automatic package cache regeneration, rather than use the cache as it is. This is the default; to turn it off, use **−−no−generate**. Configuration Item: APT::Cache::Generate.

−−names−only, **−n**
Only search on the package and provided package names, not the long descriptions. Configuration Item: APT::Cache::NamesOnly.

−−all−names
Make pkgnames print all names, including virtual packages and missing dependencies. Configuration Item: APT::Cache::AllNames.

−−recurse
Make depends and rdepends recursive so that all packages mentioned are printed once. Configuration Item: APT::Cache::RecurseDepends.

−−installed
Limit the output of depends and rdepends to packages which are currently installed. Configuration Item: APT::Cache::Installed.

−−with−source *filename*
Adds the given file as a source for metadata. Can be repeated to add multiple files. Supported are currently *.deb, *.dsc, *.changes, Sources and Packages files as well as source package directories. Files are matched based on their name only, not their content!

Sources and Packages can be compressed in any format apt supports as long as they have the correct extension. If you need to store multiple of these files in one directory you can prefix a name of your choice with the last character being an underscore ("_"). Example: my.example_Packages.xz

Note that these sources are treated as trusted (see **apt-secure**(8)). Configuration Item: APT::Sources::With.

−h, **−−help**
Show a short usage summary.

−v, **−−version**
Show the program version.

−c, **−−config−file**
Configuration File; Specify a configuration file to use. The program will read the default configuration file and then this configuration file. If configuration settings need to be set before the default configuration files are parsed specify a file with the **APT_CONFIG** environment variable. See

apt.conf(5) for syntax information.

−o, **−−option**
Set a Configuration Option; This will set an arbitrary configuration option. The syntax is **−o Foo::Bar=bar**. **−o** and **−−option** can be used multiple times to set different options.

FILES

/etc/apt/sources.list
Locations to fetch packages from. Configuration Item: Dir::Etc::SourceList.

/etc/apt/sources.list.d/
File fragments for locations to fetch packages from. Configuration Item: Dir::Etc::SourceParts.

/var/lib/apt/lists/
Storage area for state information for each package resource specified in **sources.list**(5) Configuration Item: Dir::State::Lists.

/var/lib/apt/lists/partial/
Storage area for state information in transit. Configuration Item: Dir::State::Lists (partial will be implicitly appended)

SEE ALSO

apt.conf(5), **sources.list**(5), **apt-get**(8)

DIAGNOSTICS

apt−cache returns zero on normal operation, decimal 100 on error.

BUGS

APT bug page[3]. If you wish to report a bug in APT, please see /usr/share/doc/debian/bug−reporting.txt or the **reportbug**(1) command.

AUTHORS

Jason Gunthorpe

APT team

NOTES

1. GraphViz
   http://www.research.att.com/sw/tools/graphviz/
2. VCG tool
   http://rw4.cs.uni-sb.de/users/sander/html/gsvcg1.html
3. APT bug page
   http://bugs.debian.org/src:apt