NAME

org.debian.apt – the main interface of aptdaemon

SYNOPSIS

The daemon is accessed through the D-Bus object at */org/debian/apt*. Which provides the following interface.

DESCRIPTION

METHODS

AddRepository

AddRepository (in 's' *type*,
in 's' *uri*,
in 's' *dist*,
in 'as' *comps*,
in 's' *comment*,
in 's' *sourcesfile***)**

Add a new repository to the sources list file. The repository has to define the type (e.g. deb or deb-src), uri (e.g. http://archive.debian.org/debian), the distribution (e.g. stable) and a list of components (e.g. main).

Optionally a comment and an alternative sources.list file can be specified.

EnableDistroComponent

EnableDistroComponent (in 's' *comp***)**

Enable a component for all distro repositories, e.g. main or universe.

GetTrustedVendorKeys

GetTrustedVendorKeys (out 'as' *keys***)**

Return the GnuPG keys which are used to authenticate packages.

GetAtiveTransactions

GetActiveTransactions (out 'as' *tids***)**

Return an array of the currently queued transactions.

Quit

Quit ()

Request the shutdown of the daemon. The daemon will finish a currently running transaction before.

TRANSACTION BASED METHODS

The following methods are handled by transactions. By calling the methods a new transaction will be created for the task. The return value of the method is the D-Bus object path of the corresponding transaction. The transaction can be modified, to use an http proxy or to run in a specific language by setting the corresponding properties on the **org.debian.apt.transaction**(7) interface. Afterwards the transaction has to be queued for processing by calling its Run method on the **org.debian.apt.transaction**(7) interface.

UpdateCache

UpdateCache (out 's' *tid***)**

Return the id of a newly created transaction which will fetch the latest meta data from the repositories and rebuild the cache of available and installed packages.

UpdateCachePartially

UpdateCachePartially (in 's' *sources_list*,
out 's' *tid***)**

Return the id of a newly created transaction which will fetch the latest meta data from the repositories specified in the given sources.list snippet only and rebuild the cache of available and installed packages.

InstallPackages

InstallPackages (in 'as' *package_names*,
out 's' *tid*)

Return the id of a newly created transaction which will install the packages of the given names.

Optionally the to be installed version can be specified by appending a "=" and the version to the package name, e.g. "xterm=256-1".

InstallFile

InstallFile (in 's' *path*,
out 's' *tid*)

Return the id of a newly create transaction which will install a local *.deb package file at the given *path*.

UpgradePackages

UpgradePackages (in 'as' *package_names*,
out 's' *tid*)

Return the id of a newly created transaction which will upgrade the packages of the given names to their latest version.

Optionally the to be installed version can be specified by appending a "=" and the version to the package name, e.g. "xterm=256-1".

RemovePackages

RemovePackages (in 'as' *package_names*,
out 's' *tid*)

Return the id of a newly created transaction which will remove the packages of the given names.

Optionally the version of the to be removed packages can be specified by appending a "=" and the version to the package name, e.g. "xterm=256-1".

UpgradeSystem

UpgradeSystem (in 'b' *safe_mode*,
out 's' *tid*)

Return the id of a newly created transaction which will upgrade the whole system.

If in safe mode only already installed packages will be updated. Updates which require to remove installed packages or to install additional packages will be skipped.

CommitPackages

CommitPackages (in 'as' *install*,
in 'as' *reinstall,*
in 'as' *remove,*
in 'as' *purge,*
in 'as' *upgrade*,
in 'as' *downgrade*,
out 's' *tid*)

Return the id of a newly created transaction which will perform a complex install/removal task at once. *Install*, *reinstall*, *remove*, *purge* and *upgrade* are lists of package names.

Optionally the version of the to be removed packages or the version of the package which should be installed can be specified by appending a "=" and the version to the package name, e.g. "xterm=256-1".

AddVendorKeyFromFile

AddVendorKeyFromFile (in 's' *path*,
out 's' *tid*)

Return the id of a newly created transaction which will install the public GnuPG key at the given *path* to the list of trusted software vendors. The list will be used to authenticate packages.

AddVendorKeyFromKeyserver

AddVendorKeyFromKeyserver (in 's' *keyid*,
in 's' *keyserver,*
out 's' *tid***)**

Return the id of a newly created transaction which will download and install the public GnuPG key of the *keyid* from the given *keyserver* to the list of trusted software vendors. The list will be used to authenticate packages.

RemoveVendorKey

RemoveVendorKey (in 's' *fingerprint*,
out 's' *tid***)**

Return the id of a newly created transaction which will remove the public GnuPG key with the given *fingerprint* from the list of trusted software vendors. The list will be used to authenticate packages.

FixBrokenDepends

FixBrokenDepends (out 's' *tid***)**

Return the id of a newly created transaction which will try to resolve unsatisfied dependencies by installing required packages or removing conflicting ones.

FixIncompleteInstall

FixIncompleteInstall (out 's' *tid***)**

Return the id of a newly created transaction which will try to complete previously failed installations by calling "dpkg --configure -a".

SIGNALS

ActiveTransactionsChanged

ActiveTransactionsChanged ('s' *active*,
'as' *queued***)**

The signal is used to report changes of the currently running or queued transactions. If there's any active transaction active will be an empty string.

HOMEPAGE

https://launchpad.net/aptdaemon

BUGS

You can report bugs at the Launchpad site of aptdaemon: https://bugs.launchpad.net/aptdaemon/+filebug

AUTHOR

Sebastian Heinlein <devel at glatzor dot de>

SEE ALSO

org.debian.apt.transaction(7), **aptd**(2), **aptdcon**(2)