

Gérer un meta-paquet sans supprimer ses dépendances

- Objet : Éviter de tout bousiller votre Debian en supprimant un simple logiciel
- Niveau requis : Intermédiaire
[intermédiaire](#), [avisé](#)
- Commentaires : *Suite à quelques mauvaises expériences et poussé par un désir d'alléger mon installation afin d'accélérer une vieille machine, j'ai compris qu'il fallait retirer des paquets imbriqués..*
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande, tout commence là !](#) 😊
- Suivi :
[en-chantier](#), [à-tester](#), [à-placer](#)
- 🛠️ **Fix Me!** Pour choisir, voir les autres Tags possibles dans [l'Atelier](#).
 - Création par [Patriboom](#) 13/01/2016
 - Testé par <...> le <...> 🛠️ **Fix Me!**
- Commentaires sur le forum : [Lien vers le forum concernant ce tuto](#) ¹⁾

Nota :

Contributeurs, les 🛠️ **Fix Me!** sont là pour vous aider, supprimez-les une fois le problème corrigé ou le champ rempli !

Introduction

Nous verrons ici comment supprimer un paquet sans tout supprimer de nos logiciels préférés qui lui sont associés par un meta-paquet. Pour y arriver, nous verrons d'abord qu'est-ce qu'un meta-paquet, puis nous verrons ensemble les étapes de suppression de notre paquet.

Qu'est-ce qu'un meta-paquet ?

Un meta-paquet est une série de commandes informatiques qui associe plusieurs paquets pour former un ensemble cohérent et complet. Dans l'exemple ci-bas, nous parlerons du meta-paquet « gnome » tandis que nous verrons comment supprimer le paquet « tracker » et ses dépendances.

Un meta-paquet peut contenir des paquets, des meta-paquets ou les deux. Le meta-paquet, c'est un peu comme un emballage cadeau qui enveloppe un produit. L'emballage cadeau peut contenir plusieurs boîtes, lesquelles peuvent contenir plusieurs appareils. Prenons l'exemple d'une cargaison de crayons. Le camion (meta-paquet) transporte plusieurs boîtes (meta-paquets) de crayons (produits). Le camionneur, lui, a un crayon (produit) libre dans son camion (meta-paquet) sans que son crayon soit dans une boîte.

Nous avons donc

La route (système) -meta-paquet : camion

1. qui contient : des boîtes (meta-paquets)
 - qui contiennent des crayons (produits)
 - qui contiennent des cartouches de remplissage
 - bla bla bla
2. qui contient des crayons non-emballés (produits)

-meta-paquet : auto

1. qui contient des passagers

De la même façon, nos paquets Linux peuvent être dans des meta-paquets à plusieurs niveaux. Dans le cas qui nous intéresse, gnome, nous avons cette structure

Linux (système) -meta-paquet : gnome (meta-paquet)

1. gnome-about (meta-paquet)
2. gnome-applets (meta-paquet)
3. gnome-control-center (meta-paquet)
4. gnome-disk-utility (meta-paquet)
5. gnome-icon-theme (meta-paquet)
6. gnome-keyring (meta-paquet)
7. gnome-menus (meta-paquet)
8. gnome-panel (meta-paquet)
9. gnome-power-manager (meta-paquet)
10. gnome-screensaver (meta-paquet)
11. bla bla bla

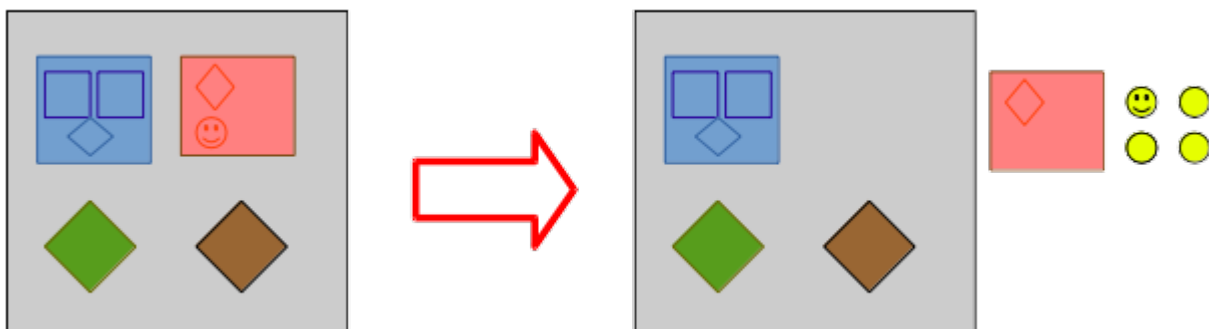
Pour en connaître la liste complète :

```
apt show gnome-core
```

Étapes de suppression de notre paquet

Étapes de suppression de notre paquet

Selon une logique simple, il suffirait d'isoler notre paquet pour le supprimer (sortir la boîte du camion, sortir le crayon de la boîte et s'en servir).



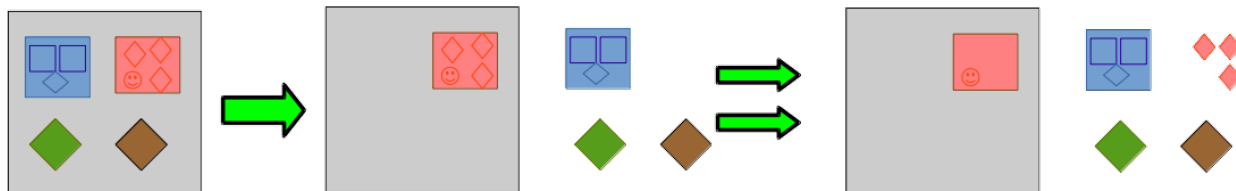


Les carrés représentent des meta-paquets Les losanges représentent des paquets complémentaires ou nécessaires à l'application. Le bonhomme sourire ici représente l'application que nous voulons supprimer.

Pourtant, penser ainsi c'est oublier la sécurité et la cohérence de l'ensemble. En effet, tous les programmes, paquets et meta-paquets qui sont installés par un meta-paquet sont aussi munis d'une attache qui les garde liés à leur meta-paquet installateur (en alimentation, on appelle ça la « traçabilité » des produits).

Dans notre analogie du transport d'un crayon par camion, il faut se rappeler que nous ne voulons pas utiliser le crayon, mais bien le retourner à l'expéditeur (supprimer un paquet). Ce que nous devons donc faire, ce sera de vider la cargaison au complet – sauf le crayon – et retourner le camion à l'expéditeur. Nous devons donc sortir toutes les boîtes du camion, sortir tous les crayons des boîtes et retourner le camion contenant le seul crayon que nous ne voulons pas.

Nous ferons donc ainsi :



Dans un premier temps vider notre meta-paquet de tout sauf des éléments que nous voulons supprimer et même vider le meta-paquet qui contient notre logiciel de ses autres composantes.

Comme avec l'analogie du camionnage, on ne détruit rien. On ne fait que casser les liens, ouvrir les cartons.

Détruire les liens, dans notre cas, ça s'appelle « ôter la marque *Installer automatiquement* »

Première étape: étudier le cas

Il est important de connaître la chaîne des dépendances à laquelle est fixé notre logiciel.

Il faut repérer le bon emballage, la bonne boîte, le bon camion auxquels est attaché notre crayon.

Liste des liens ascendants

```
apt-cache rdepends -i --no-pre-depends nom_du_logiciel
```

Notre exemple (tracker) nous donnera ceci:

```
apt-cache rdepends -i --no-pre-depends tracker
```

Extrait du retour de la commande « `apt-cache rdepends -i --no-pre-depends tracker` »

```
tracker
```

```
Reverse Depends:
  rygel-tracker
  tracker-utils
  tracker-miner-fs
  tracker-gui
  tracker-extract
  tracker-dbg
  gnome-photos  *
  gnome-music  *
  gnome-documents  *
  gnome-boxes  *
  bijiben  *
```



Si vous aviez fait **apt-cache rdepends -i -no-pre-depends nom_du_logiciel > NomDeFichier** le résultat aurait été conservé dans un fichier du nom de *NomDeFichier*. Ceci pourrait être utile avec la prochaine commande qui générera une plus longue liste.

Nous avons ici les paquets sur lesquels compte tracker pour fonctionner (dont tracker dépend). Voyons maintenant quelles sont les dépendances des paquet « mère » de ceux-ci (quels autres produits sont dans la boîte de notre crayon). Sont ci-haut marqués d'un astérisque (*) les paquets qui ne contiennent pas le mot *tracker*, les paquets - donc - qui sont susceptibles d'être bons et qui devraient par conséquent rester installés.

Notons au passage que trois de ces paquets sont de la forme : « *gnome-quelque chose* ». Afin de simplifier les démarches, nous remonterons de ce pas à la source (la paquet-mère) : « *gnome* ».

Liste des liens descendants

Faisons donc:

```
apt-cache depends -i gnome
apt-cache dpends -i bijiben
```

Nous obtiendrons de belles longues listes:

Résultat de «[apt-cache depends -i gnome](#)»

```
Dépend: gnome-core
Dépend: desktop-base
Dépend: network-manager-gnome
Dépend: bijiben
Dépend: brasero
Dépend: cheese
```

```
Dépend: evolution
Dépend: evolution-plugins
Dépend: file-roller
Dépend: gedit
Dépend: gnome-clocks
Dépend: gnome-color-manager
Dépend: gnome-documents
Dépend: gnome-games
Dépend: gnome-getting-started-docs
Dépend: gnome-logs
Dépend: gnome-maps
Dépend: gnome-music
Dépend: gnome-nettool
Dépend: gnome-photos
Dépend: gnome-sound-recorder
Dépend: gnome-tweak-tool
Dépend: nautilus-sendto
Dépend: gnome-orca
Dépend: polari
Dépend: rygel-playbin
Dépend: rygel-tracker
Dépend: seahorse
Dépend: vinagre
Dépend: alacarte
Dépend: avahi-daemon
Dépend: gimp
Dépend: hamster-applet
Dépend: inkscape
Dépend: libreoffice-evolution
Dépend: libreoffice-gnome
Dépend: libreoffice-writer
Dépend: libreoffice-calc
Dépend: libreoffice-impress
Dépend: rhythmbox
Dépend: simple-scan
|Dépend: goobox
Dépend: sound-juicer
Dépend: transmission-gtk
Dépend: xdg-user-dirs-gtk
Dépend: cups-pk-helper
Dépend: gedit-plugins
Dépend: gnome-shell-extension-weather
Dépend: gstreamer1.0-libav
Dépend: gstreamer1.0-plugins-ugly
Dépend: rhythmbox-plugins
Dépend: rhythmbox-plugin-cdrecorder
Dépend: telepathy-gabble
Dépend: telepathy-rakia
Dépend: telepathy-salut
Dépend: totem-plugins
```

Dépend: libgtk2-perl

Résultat de la commande « apt-cache depends -i bijiben »

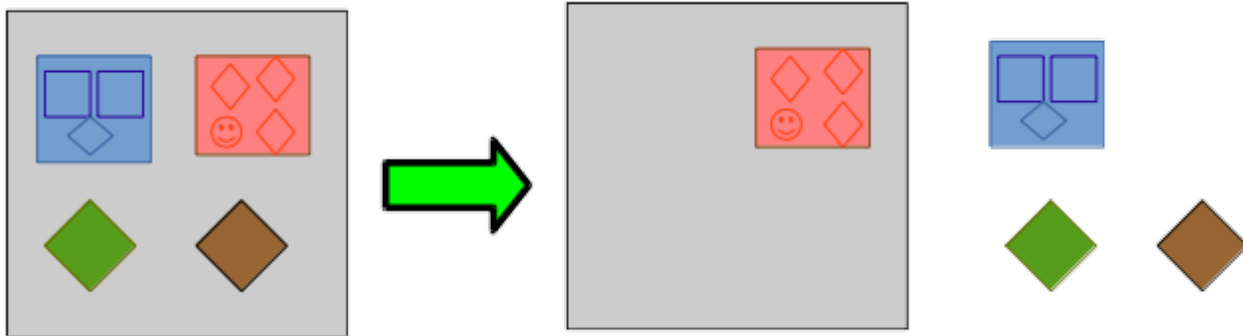
```
|Dépend: dconf-gsettings-backend
Dépend: <gsettings-backend>
  gconf-gsettings-backend
  dconf-gsettings-backend
Dépend: libc6
Dépend: libcairo-gobject2
Dépend: libcairo2
Dépend: libecal-1.2-16
Dépend: libedataserver-1.2-18
Dépend: libgdk-pixbuf2.0-0
Dépend: libglib2.0-0
Dépend: libgoa-1.0-0b
Dépend: libgtk-3-0
Dépend: libical1
Dépend: libpango-1.0-0
Dépend: libpangocairo-1.0-0
Dépend: libtracker-sparql-1.0-0
Dépend: libuuid1
Dépend: libwebkitgtk-3.0-0
Dépend: libxml2
Dépend: libzeitgeist-2.0-0
```

Deuxième étape: casser les liens (les bons)

La liste ci-haut nous sera très utile. C'est la liste des paquets que nous devrons délier du meta-paquet *gnome*. Nous le ferons en deux étapes: 1) vider le camion de toutes les boîtes à conserver; 2) vider la boîte contenant notre crayon de tout ce qui n'est pas notre crayon.

Cassage des liens descendants

De notre camion, nous sortons toutes les boîtes qui ne contiennent pas les crayons similaires à celui que nous cherchons.



```
apt-mark manual nom_du_paquet
```

pour CHACUN des paquets listés ci-haut et que nous souhaitons conserver.

Notre exemple de désinstallation du paquet *Tracker* donne donc ceci:

Les lignes à inscrire pour isoler les paquets identifiés par «*apt-cache depends -i gnome* »

```
apt-mark manual gnome-core
apt-mark manual desktop-base
apt-mark manual network-manager-gnome
apt-mark manual bijiben
apt-mark manual brasero
apt-mark manual cheese
apt-mark manual evolution
apt-mark manual evolution-plugins
apt-mark manual file-roller
apt-mark manual gedit
apt-mark manual gnome-clocks
apt-mark manual gnome-color-manager
apt-mark manual gnome-documents
apt-mark manual gnome-games
apt-mark manual gnome-getting-started-docs
apt-mark manual gnome-logs
apt-mark manual gnome-maps
apt-mark manual gnome-music
apt-mark manual gnome-nettool
apt-mark manual gnome-photos
apt-mark manual gnome-sound-recorder
apt-mark manual gnome-tweak-tool
apt-mark manual nautilus-sendto
apt-mark manual gnome-orca
apt-mark manual polari
apt-mark manual rygel-playbin
apt-mark manual rygel-tracker
apt-mark manual seahorse
apt-mark manual vinagre
apt-mark manual alacarte
apt-mark manual avahi-daemon
apt-mark manual gimp
```

```
apt-mark manual hamster-applet
apt-mark manual inkscape
apt-mark manual libreoffice-evolution
apt-mark manual libreoffice-gnome
apt-mark manual libreoffice-writer
apt-mark manual libreoffice-calc
apt-mark manual libreoffice-impress
apt-mark manual rhythmbox
apt-mark manual simple-scan
apt-mark manual goobox
apt-mark manual sound-juicer
apt-mark manual transmission-gtk
apt-mark manual xdg-user-dirs-gtk
apt-mark manual cups-pk-helper
apt-mark manual gedit-plugins
apt-mark manual gnome-shell-extension-weather
apt-mark manual gstreamer1.0-libav
apt-mark manual gstreamer1.0-plugins-ugly
apt-mark manual rhythmbox-plugins
apt-mark manual rhythmbox-plugin-cdrecorder
apt-mark manual telepathy-gabble
apt-mark manual telepathy-rakia
apt-mark manual telepathy-salut
apt-mark manual totem-plugins
apt-mark manual libgtk2-perl
apt-mark manual gnome
```



Si vous travaillez comme moi, vous copiez la liste obtenue par la recherche inverse des dépendances et vous copiez cette liste dans un traitement de texte (gnome, mousepad).

Sans oublier ...

ceux obtenus par « `apt-cache depends -i bijiben` »

```
apt-mark manual dconf-gsettings-backend
apt-mark manual <gsettings-backend>
  gconf-gsettings-backend
  dconf-gsettings-backend
apt-mark manual libc6
apt-mark manual libcairo-gobject2
apt-mark manual libcairo2
apt-mark manual libecal-1.2-16
apt-mark manual libedataserver-1.2-18
apt-mark manual libgdk-pixbuf2.0-0
apt-mark manual libglib2.0-0
apt-mark manual libgoa-1.0-0b
```



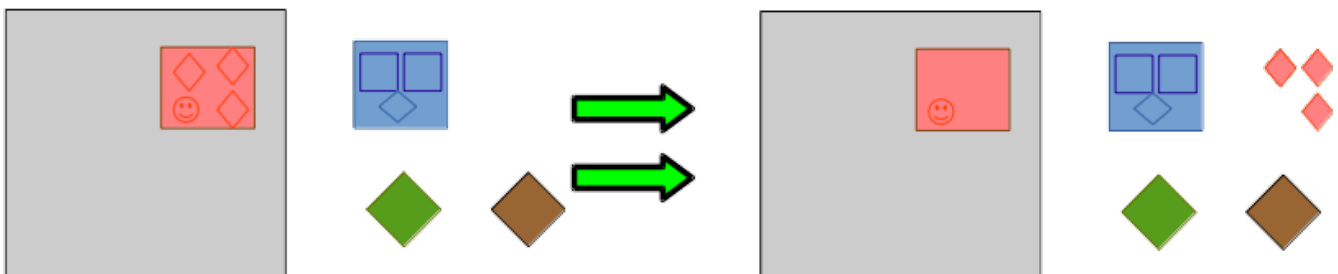
```
apt-mark manual libgtk-3-0
apt-mark manual libical1a
apt-mark manual libpango-1.0-0
apt-mark manual libpangocairo-1.0-0
apt-mark manual libtracker-sparql-1.0-0
apt-mark manual libuuid1
apt-mark manual libwebkitgtk-3.0-0
apt-mark manual libxml2
apt-mark manual libzeitgeist-2.0-0
```



Oui, il faut passer ainsi toutes les listes de dépendance descendante obtenues lors de la deuxième étape. Rappelez-vous: sortir toutes les boîtes du camion, sortir tous les emballages qui ne sont pas ceux du crayon que nous cherchons à retourner à l'expéditeur! Parfois, ce sont de gros camions; parfois ce sont des camions bien chargés; parfois ce sont de gros camions bien chargés. Gnome ainsi ainsi: gros et chargé!

Cassage des liens ascendants

Revenons ici à notre première liste, celle des dépendances inverses, celle qui contenait plusieurs mentions de « tracker ». Vous vous rappelez que nous n'avions conservé de cette liste que les paquets non-tracker, soit gnome et bijiben.



De notre boîte, donc, nous sortons tous les paquets qui ne sont pas liés à notre crayon.

```
apt-mark manual gnome-photos
apt-mark manual gnome-music
apt-mark manual gnome-documents
apt-mark manual gnome-boxes
apt-mark manual bijiben
```



Votre commande, transformée dans le-dit traitement de texte pourra s'écrire en une seule ligne de comme ceci:

apt-mark manual gnome-photos gnome-music gnome-documents gnome-boxes bijiben

Troisième étape: supprimer le paquet

```
apt-get remove tracker
```

Quatrième étape: reconstruire les références du meta-paquet (facultatif)

Voilà le temps que nous remettons les choses dans l'ordre et refassions les liens. C'est beau le libre ... mais quand le camion est parti, c'est encore plus beau un entrepôt bien classé!

```
apt-mark auto gnome-photos  
apt-mark auto gnome-music  
apt-mark auto gnome-documents  
apt-mark auto gnome-boxes  
apt-mark auto bijiben
```

1)

N'hésitez pas à y faire part de vos remarques, succès, améliorations ou échecs !

From:

<http://debian-facile.org/> - **Documentation - Wiki**

Permanent link:

<http://debian-facile.org/atelier:chantier:supprimer-un-paquet-sans-tout-bousiller>



Last update: **03/11/2016 15:48**