

Vim IDE python

- Objet : du tuto 
- Niveau requis : 
[débutant, avisé](#)
- Commentaires : *Contexte d'utilisation du sujet du tuto.* 
- Débutant, à savoir : [Utiliser GNU/Linux en ligne de commande, tout commence là !](#) 😊
- Références :
<https://github.com/Valloric/YouCompleteMe/wiki/Building-Vim-from-source>
https://github.com/thesheff17/youtube/blob/master/vim/vim_notes.txt

Installer VIM IDE pour python 3

Introduction

- Si vim est déjà installé il n'est peut être pas configuré avec un interpréteur pour python ou python 3 :

```
vim --version | grep python
```

+cryptv	+linebreak	-python	+viminfo
+cscope	+lispindent	-python3	+vreplace

⇒ -python -python3 😞

- Si l'on veut repartir à neuf :

```
dpkg --status vim
```

```
Package: vim
Status: install ok installed
Priority: optional
Section: editors
Installed-Size: 2233
Maintainer: Debian Vim Maintainers <pkg-vim-maintainers@lists.alioth.debian.org>
Architecture: amd64
Version: 2:7.4.488-7
Provides: editor
Depends: vim-common (= 2:7.4.488-7), vim-runtime (= 2:7.4.488-7), libacl1 (>= 2.2.51-8), libc6 (>= 2.15), libgpm2 (>= 1.20.4), libselinux1 (>= 1.32), libtinfo5
Suggests: ctags, vim-doc, vim-scripts
Description: Vi IMproved - enhanced vi editor
 Vim is an almost compatible version of the UNIX editor Vi.
<...>
```

On supprime la version installée et ses dépendances

```
apt-get remove vim vim-common vim-runtime
```

- Sur un système Jessie tout neuf la commande : `apt-get install vim`

installera les paquets `vim-doc` et `vim-scripts`. Il est suggéré d'installer aussi `vim-common` et `vim-runtime`...

Allons-y :

```
apt-get install vim
```

Reconfiguration de vim

Pour utiliser vim avec python3 :

- on installe pour ce qui concerne python 3 :

```
apt-get install libncurses5-dev libgnome2-dev libgnomeui-dev \
  libgtk2.0-dev libatk1.0-dev libbonoboui2-dev \
  libcairo2-dev libx11-dev libxpm-dev libxt-dev python-dev \
  ruby-dev mercurial
```

- Pour ce concerne la configuration de vim :

```
apt-get install devscripts
```

- Changer les options de vim

On crée un répertoire où travailler :

```
mkdir temp && cd temp
```

On récupère les sources :

```
apt-get source vim
```

Ce qui donne :

```
ls
```

```
vim-7.4.488  vim_7.4.488-7.debian.tar.xz  vim_7.4.488-7.dsc
vim_7.4.488.orig.tar.gz
```

On se déplace dans le dossier `vim-7.4.488` :

```
cd vim-7.4.488
```

- Pour reconstruire vim

D'abord on installe toutes les dépendances nécessaires à la reconstruction du paquet :

```
apt-get build-dep vim
```

Dans ce répertoire vim-7.4.488 il y a le répertoire debian :

```
ls
```

```
configure      Filelist      README_amibin.txt.info  README_dos.txt
README_srcdos.txt  README_w32s.txt  vimdir.info
Contents        libs          README_amisrc.txt      README_extra.txt
README_src.txt    runtime       Vim.info
Contents.info    Makefile      README_amisrc.txt.info  README_mac.txt
README.txt        runtime.info  vimtutor.bat
csdpmi4b.zip     nsis          README_ami.txt         README_ole.txt
README.txt.info   src           vimtutor.com
debian           pixmaps       README_ami.txt.info     README_os2.txt
README_unix.txt   src.info      Xxd.info
farsi            README_amibin.txt  README_bindos.txt      README_os390.txt
README_vms.txt    uninstal.txt
```

Et dans ce répertoire debian il y a un fichier rules :

```
ls -la debian/rules
```

```
-rwxr-xr-x 1 hypathie hypathie 18825 mars  21  2015 rules
```

Puis on édite rules pour obtenir la ligne suivante :

```
vim rules
```

```
ALLINTERPFLAGS+=--enable-python3interp --with-python3-config-dir=$(shell
python3-config --configdir)
```

On choisit son éditeur par défaut et on lui ajoute sa touche personnelle :

```
dch -i
```

Par exemple :

```
vim (2:7.4.488-7.1) UNRELEASED; urgency=medium
* Non-maintainer upload.
* vim_python3
```

On reconstruit le paquet :

```
debuild binary
```

Il fait ses tests 😊 ...

Puis on lance :

```
dpkg -i {vim-gtk,vim-common,vim-gui-common,vim-runtime}*.deb
```

Cette dernière commande se lance en étant dans le répertoire racine, où il y a les fichiers qui ont été créés par la commande `debuild binary` :

```
cd temp/ && ls
```



```
vim-7.4.488                                vim-doc_7.4.488-7.1_all.deb
vim_7.4.488-7.1_amd64.deb                  vim-
gnome_7.4.488-7.1_amd64.deb
vim_7.4.488-7.debian.tar.xz               vim-gtk_7.4.488-7.1_amd64.deb
vim_7.4.488-7.dsc                          vim-gui-
common_7.4.488-7.1_all.deb
vim_7.4.488.orig.tar.gz                   vim-
lesstif_7.4.488-7.1_all.deb
vim-athena_7.4.488-7.1_amd64.deb          vim-nox_7.4.488-7.1_amd64.deb
vim-common_7.4.488-7.1_amd64.deb          vim-
runtime_7.4.488-7.1_all.deb
vim-dbg_7.4.488-7.1_amd64.deb             vim-tiny_7.4.488-7.1_amd64.deb
```

Et voilà !

```
vim --version | grep python*
```

+cryptv	+linebreak	-python	+viminfo
+cscope	+lispindent	+python3	+vreplace

vim IDE pour python 3

- On installe git

```
apt-get install git
```

- On télécharge ce script

[configVim.sh](#)

```
#!/bin/sh

var1=~/.vim/autoload/"
var2=~/.vim/bundle/"
var3=~/.vim/colors/"
var4=~/.vim/ftplugin/"

if [ -f "$var1" ]
```

```
then
    echo "le fichier existe déjà !"
else
    mkdir -p ~/vim/autoload/
fi
if [ -f "$var2" ]
then
    echo "le fichier existe !"
else
    mkdir -p ~/.vim/bundle/
if [ -f "$var3" ]
then
    echo "le fichier existe !"
else
    mkdir -p ~/.vim/colors/
fi
if [ -f "$var4" ]
then
    echo "le fichier existe !"
else
    mkdir -p ~/.vim/ftplugin/
fi

ls -la ~/.vim/
cd ~/.vim/autoload
wget
https://raw.githubusercontent.com/tpope/vim-pathogen/master/autoload/pathogen.vim
cd ~/.vim/bundle
git clone git clone https://github.com/scrooloose/nerdtree && git clone
https://github.com/jistr/vim-nerdtree-tabs.git && git clone
https://github.com/tpope/vim-sensible.git && git clone
https://github.com/kien/ctrlp.vim.git && git clone
https://github.com/klen/python-mode.git && git clone
https://github.com/Lokaltog/vim-powerline.git && git clone
https://github.com/othree/html5.vim.git && git clone
https://github.com/feix760/css3complete.vim.git
cd ~/.vim/colors
wget
https://raw.githubusercontent.com/thesheff17/youtube/master/vim/wombat2
56mod.vim
cd ~/.vim/ftplugin
wget
https://raw.githubusercontent.com/thesheff17/youtube/master/vim/python_
editing.vim
cd ~
wget
https://raw.githubusercontent.com/thesheff17/youtube/master/vim/vimrc
ls -la ~/vimrc
printf "Le fichier ~/vimrc a bien été reçu.\n"
mv vimrc ~/.vimrc
```

```
printf "Le fichier ~/vimrc est dans ~/.vimrc\n"  
printf "Et voilà !\n"
```

- On installe cette configuration

```
mv ~/Téléchargements/configVim.sh ~/
```

```
chmod +x configVim.sh
```

```
./configVim.sh
```



Pour la supprimer, c'est simple :

```
rm -rf ~/.vim/ && rm -r ~/.vimrc
```

- Pour avoir la même configuration quand on lance vim en root

```
cd ~
```

```
wget https://raw.githubusercontent.com/thesheff17/youtube/master/vim/vim.py
```

```
chmod +x ./vim.py
```

```
./vim.py -u root
```



Et pour supprimer cette configuration de vim utiliser en tant que root :

```
cd ~ && rm -rf ~/.vim/ && rm -r ~/.vimrc
```

Configuration

Pour la gestion des téléchargements des plugins :

```
apt-get install git-core && apt-get install wget
```

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

Permanent link:
<http://debian-facile.org/utilisateurs:hyathie:tutos:vim-ide-python>

Last update: **29/03/2016 05:48**



