

Mise en place de l'environnement de développement sous Ubuntu 18.04 LTS

Installation de l'environnement de développement

1. Installer subversion

Dans un terminal (CTRL + ALT + T) :

```
sudo apt-get update
sudo apt-get install subversion
```

2. Installer Qt (5.9.x)

Dans un terminal (CTRL + ALT + T) :

```
sudo apt-get install build-essential qtcreator qt5-default qt5-doc qt5-doc-html qtbase5-doc-html qtbase5-examples
```

Récupération du code source de Computree et de ses plugins

1. Créer un répertoire racine *Computree* où vous le souhaitez

2. Télécharger [kit_dev_linux.zip](#), qui contient :

- Un fichier *all.pro* avec tous les projets diffusés en standard (cœur + plugins open-sources)
- Un fichier *recuperer_depots.sh*, contenant des instructions svn, pour récupérer le code source
- Un fichier *LISEZ-MOI.txt*, reprenant ces instructions

3. Décompresser *kit_dev_linux.tar.gz* à la racine du répertoire *Computree*

4. Dans un terminal (CTRL + ALT + T), lancer le script *_recuperer_depots.sh*

Installation des dépendances

1. OpenCV 3.2

- Dans un terminal (CTRL + ALT + T) :

```
sudo apt-get install libopencv-dev
```

2. PCL 1.8 (optionnel, permet d'utiliser les plugins nécessitant PCL)

- Dans un terminal (CTRL + ALT + T) :

```
sudo apt-get install libpcl-dev
```

3. **GDAL 2.2.3** (optionnel, donne accès aux formats SIG vecteur et raster de GDAL/OGR)

- Dans un terminal (CTRL + ALT + T) :

```
sudo apt-get install libgdal-dev
```

4. **GSL 2.4** (optionnel, donne accès à une librairie de calcul numérique utilisée dans certains plugins)

- Dans un terminal (CTRL + ALT + T) :

```
sudo apt-get install libgsl-dev
```

Si vous souhaitez installer des dépendances dans des emplacements différents, vous pouvez, pour chaque fichier *LIBNAME_default_path.pri*, le dupliquer et le renommer *LIBNAME_user_path.pri*. Après cela, vous n'avez qu'à modifier ce deuxième fichier pour utiliser votre chemin local.

Compilation de Computree

1. Lancer Qt Creator

2. Ouvrir le projet *all.pro*

Si vous n'avez pas installé PCL, supprimez / commentez la ligne suivante dans *all.pro* :

```
computreev5/library/ctlibpcl/ctlibpcl.pro \
```

3. Dans l'onglet projet désactiver les shadow builds (case à cocher), pour la release et / ou la debug
shadow_build.png

4. Dans l'onglet projet/run, dans la section "Environnement d'exécution", rajouter " ;. " à la fin de la variables **LD_LIBRARY_PATH**

5. Exécuter qmake sur *all.pro*, puis compiler le projet

Après une mise à jour du code source, si le cœur de Computree a été modifié significativement, il peut être nécessaire d'exécuter qmake sur chaque sous-projet, puis de faire Recompile sur *all.pro*.

6. Pour obtenir toutes les traductions, utilisez le script suivant dans un terminal : *computreev5/scripts/unix_updateLanguage.sh*

Exécution de Computree

Une fois compilé, vous pouvez lancer l'exécution depuis Qt-Creator (flèche verte ou run/exécuter sur *all.pro*).

Configurer votre plugin si vous souhaitez utiliser PCL dans votre code

Si vous souhaitez utiliser PCL pour vos développements quelques étapes de préparation sont nécessaires :

Vous devez configurer le fichier *.pro* de votre plugin (*.pro*) comme suit (début du fichier) :

```
CT_PREFIX = ../../computreev5
```

```
include($${CT_PREFIX}/shared.pri)
include($${PLUGIN_SHARED_DIR}/include.pri)

COMPUTREE += ctlibpcl

include($${CT_PREFIX}/include_ct_library.pri)
```

N'oubliez pas de compiler le projet libpcl dans le dossier computreev5/library/ctlibpcl (ouvrez le fichier ctlibpcl.pro et le compiler avec QtCreator)

Il suffit maintenant de faire un *qmake* sur le projet de votre plugin (clic droit → qmake) et de le compiler.

Files

kit_dev_linux.zip	2.44 KB	07/31/2018	Piboule Alexandre
-------------------	---------	------------	-------------------