

en_US.png [...english version](#)

Mettre en place son environnement de développement Computree

Mise en place de l'environnement de développement sous Windows (64 bits, 7, 8 ou 10)

Installation de l'environnement de développement

1. Installer "Microsoft Visual studio 2015 Express": <https://www.visualstudio.com/fr/post-download-vs/?sku=xdesk>
2. Installer "Windows SDK": <https://developer.microsoft.com/fr-fr/windows/downloads/windows-10-sdk> (étape optionnelle, permettant de pouvoir compiler en mode debug)
3. Installer "Qt": <https://www.qt.io/download-open-source/> (check at least Qt 5.9.1 msvc 2015 64 bits)
4. Installer un client SVN, comme par exemple "TortoiseSVN": <https://tortoisesvn.net> (pendant l'installation, penser à activer l'installation à activer *command line client tools*)

Récupération du code source de Computree et de ses plugins

1. Créer un répertoire racine *Computree* où vous le souhaitez
2. Télécharger le [Computree development kit \(Windows\)](#), qui contient :
 - Un fichier *all.pro* avec tous les projets diffusés en standard (cœur + plugins open-sources)
 - Un fichier *recuperer_depots.bat*, contenant des instructions svn, pour récupérer le code source
 - Un fichier *LISEZ-MOI.txt*, reprenant ces instructions
3. Décompresser *kit_dev_windows_fr.zip* à la racine du répertoire *Computree*
4. Lancer le script *recuperer_depots.bat*

Installation des dépendances

Télécharger et décompresser [Computree v5 dependencies folder](#), et le déplacer à la racine du répertoire *computree*

Si vous souhaitez installer des dépendances dans des emplacements différents (par exemple dans les *c:/program files*), vous pouvez, pour chaque fichier *LIBNAME_default_path.pro* du répertoire *computreev5*, le dupliquer et le renommer *LIBNAME_user_path.pro*. 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*, sélectionner le compilateur MSVC 2015 64bits, avec les version release et / ou debug

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

4. 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 Recompiler sur all.pro.

Exécution de Computree

Une fois compilé, **pour pouvoir être exécuté, Computree a besoin de toutes les dll des dépendances**, accessibles depuis l'emplacement du fichier *CompuTreeGui.exe* généré.

Pour cela, copier les dlls dans le répertoire ComputreeInstallRelease (pour la version release) et / ou le répertoire ComputreeInstallDebug (pour la version debug).

Les DLL sont disponibles au téléchargement ici : [Computree v5 DLL](#)

Ensuite 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.

Mise en place de l'environnement de développement sous Ubuntu 16.04 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 (5.9.1)

- Télécharger la dernière version de l'installateur Qt installer : <https://www.qt.io/download-open-source/>
 - Il faudra créer un compte utilisateur Qt

- Si vous êtes derrière un proxy, il faudra le paramétrer dans la section paramètres
- Installer Qt

Récupération du code source de Computree et de ses plugins

1. Créer un répertoire racine *Computree* où vous le souhaitez
2. Télécharger le [Computree development kit \(Linux\)](#), 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 3.3.0** (optionnel mais fortement conseillé, permet d'utiliser des image / rasters dans Computree)
 - Suivre les instructions données sur le site officiel d'OpenCV : http://docs.opencv.org/3.3.0/d7/d9f/tutorial_linux_install.html
2. **PCL 1.8.0** (optionnel, permet d'utiliser les plugins nécessitant PCL)
 - Dans un terminal (CTRL + ALT + T) :

```
sudo apt-get install git build-essential linux-libc-dev cmake cmake-gui libusb-1.0-0-dev libusb-dev libudev-dev mpi-default-dev openmpi-bin openmpi-common libflann1.8 libflann-dev libeigen3-dev libboost-all-dev libvtk5.10-qt4 libvtk5.10 libvtk5-dev libqhull* libgtest-dev freeglut3-dev pkg-config libxmu-dev libxi-dev mono-complete qt-sdk openjdk-8-jdk openjdk-8-jre libproj-dev
```

- Télécharger le code source PCL 1.8.0 ici <https://github.com/PointCloudLibrary/pcl/archive/pcl-1.8.0.tar.gz>
- Décompresser le fichier *pcl-1.8.0.tar.gz* (dans un explorateur: clic droit, extraire ici)
- Dans un terminal (CTRL + ALT + T) :

```
cd pcl-pcl-1.8.0
mkdir build
cd build
cmake -DCMAKE_BUILD_TYPE=Release -DCMAKE_INSTALL_PREFIX=/usr \ -DCMAKE_INSTALL_PREFIX=/usr ..
make -j7
sudo make install
```

3. **GDAL 2.1.2** (optionnel, donne accès aux formats SIG vecteur et raster de GDAL/OGR)
 - Télécharger la version 2.1.2 de GDAL, ici : <http://download.osgeo.org/gdal/2.1.2/gdal-2.1.2.tar.gz>
 - Décompresser le fichier *gdal-2.1.2.tar.gz* (dans un explorateur : clic droit, extraire ici)
 - Ouvrir un terminal dans le dossier *gdal-2.1.2* (dans un explorateur : sélectionner le dossier, clic droit, ouvrir dans un terminal)
 - Lancer les commandes suivantes :

```
./configure
make
sudo make install
sudo ldconfig
```

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

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

```
sudo apt-get install -y gsl-bin libgsl0-dev
```

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

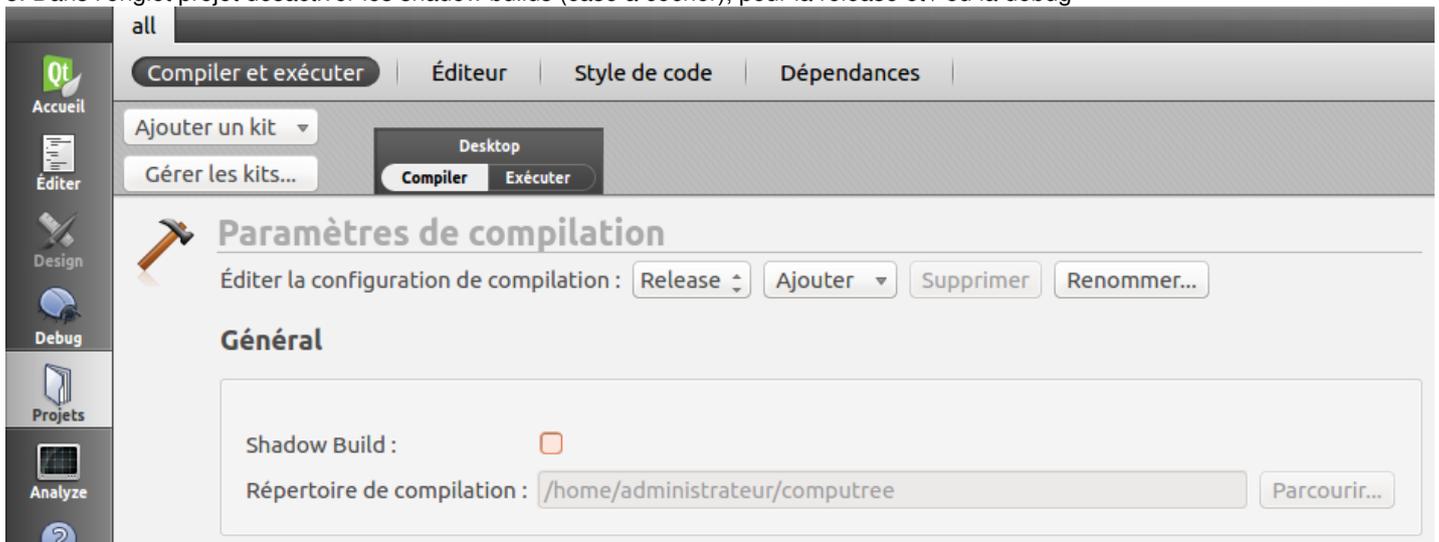
Compilation de Computree

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



4. 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*.

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.

Liste des dépôts svn

Si vous souhaitez ajouter des dépôts, ou faire une installation manuelle sans les scripts, vous trouverez dans le tableau ci-dessous la liste des dépôts pour tous les plugins Computree.

Pour accéder à un dépôt il faut bien sûr avoir les droits adéquats pour le projet considéré.

D'une façon générale le nom d'un dépôt est <http://rdinnovation.onf.fr/svn/nom-du-projet>. Le nom du projet étant celui qui s'affiche dans la barre d'adresse du navigateur.

Plugin	Code plugin	Projet	Dépôt svn
Computree (base)	CT	computree	http://rdinnovation.onf.fr/svn/computree
ComputreeDevTools	-	computredevtools	http://rdinnovation.onf.fr/svn/computredevtools
Plugin Onf	ONF	plugin-onf	http://rdinnovation.onf.fr/svn/plugin-onf
Plugin Arts Free	ARFR	plugin-arts-free	http://rdinnovation.onf.fr/svn/plugin-arts-free
Plugin Onf Lsis	OL	plugin-onf-lsis	http://rdinnovation.onf.fr/svn/plugin-onf-lsis
Plugin Generate	GEN	plugin-generate	http://rdinnovation.onf.fr/svn/plugin-generate
Plugin ToolKit	TK	plugin-toolkit	http://rdinnovation.onf.fr/svn/plugin-toolkit
Plugin LVox	LVOX	plugin-lvox	http://rdinnovation.onf.fr/svn/plugin-lvox

[Retour à l'accueil](#)

Files

kit_qt_551.png	158 KB	12/20/2016	Piboule Alexandre
shadow_build.png	62.5 KB	12/20/2016	Piboule Alexandre