

fr\_FR.png [...version française](#)

## Initialization of a Computree development environment

### Initialization of a Computree development environment for Windows (64 bits, 7, 8 ou 10)

#### Installing the development environment

1. Install "Microsoft Visual studio 2015 Express": <https://www.visualstudio.com/fr/post-download-vs/?sku=xdesk>
2. Install "Windows SDK": <https://developer.microsoft.com/fr-fr/windows/downloads/windows-10-sdk> (Optional step, allowing to compile in debug mode)
3. Install "Qt": <https://www.qt.io/download-open-source/> (check at least Qt 5.9.1 msvc 2015 64 bits)
4. Install a SVN client, as for example "TortoiseSVN": <https://tortoisevn.net> (During the installation, think about activating the installation of *command line client tools*)

#### Obtaining the source code of Computree and its plugins

1. Create a *Computree* root directory where you want
2. Download [Computree development kit \(Windows\)](#), which contains:
  - A *all.pro* file with all projects distributed as standard (core + open-source plugins)
  - A *recuperer\_depots.bat* file, containing svn statements, to retrieve the source code
  - A file *LISEZ-MOI.txt*, containing these instructions
3. Unzip *kit\_dev\_windows\_fr.zip* at the root of the *Computree* directory
4. Run the *recuperer\_depots.bat* script

#### Installing Dependencies

Download and unzip [Computree v5 dependencies folder](#), and paste at the root of the computree directory.

If you want to install dependencies in different locations (for example in c:/program files), you can, for each *LIBNAME\_default\_path.pri* file in computreev5 directory, duplicate it and rename it *LIBNAME\_user\_path.pri*. After that you just have to modify this second file to use your local paths.

#### Computree Compilation

1. Launch Qt Creator
2. Open the project *all.pro*, select the compiler MSVC 2015 64bits, with version release and/or debug

If you have not installed PCL, delete/comment the following line in *all.pro*:

```
Computreev5/library/ctlibpcl/ctlibpcl.pro \
```

3. On the Project tab, disable shadow builds (check box), for release and/or debug

#### 4. Run qmake on all.pro, then compile the project

After updating the source code, if the core of Computree has been modified significantly, it may be necessary to run qmake on each subproject and then to do Recompile on all.pro.

## Execution of Computree

Once compiled, **to be run, Computree needs all the dependency dlls**, accessible from the location of the generated *CompuTreeGui.exe* file.

For that copy dlls to ComputreeInstallRelease folder (for release version) and / or ComputreeInstallDebug folder (for debug version).

Dll are available for download here: [Computree v5 DLL](#)

Then you can run from Qt-Creator (green arrow or run on all.pro).

## Configure your plugin if you want to use PCL in your code

If you want to use PCL for your development some preparation steps are required:

You must configure the .pro file of your plugin (.pro) as follows (beginning of the file):

```
CT_PREFIX = ../../computreev5  
  
Include ($$ {CT_PREFIX} /shared.pri)  
Include ($$ {PLUGIN_SHARED_DIR} /include.pri)  
  
COMPUTREE += ctlibpcl  
  
Include ($$ {CT_PREFIX} /include_ct_library.pri)
```

**Do not forget to compile the libpcl project into the computreev5/library/ctlibpcl folder (open the ctlibpcl.pro file and compile it with QtCreator)**

Just make a *qmake* on the project of your plugin (right click → qmake) and compile it.

## Initialization of a Computree Development Environment on Ubuntu 16.04 LTS

### Installing the development environment

#### 1. Installing subversion

*In a terminal (CTRL + ALT + T):*

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

#### 2. Installing Qt (5.9.1)

- Download last Qt installer : <https://www.qt.io/download-open-source/>
  - You will have to create a Qt user account
  - If you are behind a proxy, you need to go in settings section to parameter it
- Install Qt

# Recovering the source code of Computree and its plugins

1. Create a *Computree* root directory where you want
2. Download [Computree development kit \(Linux\)](#), which contains:
  - A *all.pro* file with all projects distributed as standard (core + open-source plugins)
  - A *recuperer\_depots.sh* file containing svn statements to retrieve the source code
  - A file *README.txt*, containing these instructions
3. Unzip *kit\_dev\_linux.tar.gz* at the root of the *Computree* directory
4. In a terminal (CTRL + ALT + T), run the *\_recuperer\_depots.sh* script

## Installing Dependencies

1. **OpenCV 3.3.0** (optional but highly recommended, allows to use images / rasters in Computree)
  - Follow instruction given in official OpenCV site: [http://docs.opencv.org/3.3.0/d7/d9f/tutorial\\_linux\\_install.html](http://docs.opencv.org/3.3.0/d7/d9f/tutorial_linux_install.html)
2. **PCL 1.8.0** (optional, allows to use plugins requiring PCL)
  - In a terminal (CTRL + ALT + T) :

```
sudo apt-get install git build-essential linux-libc-dev cmake cmake-gui libusb-1.0-0-dev libusb-1.0-0-dev libudev-dev openmpi-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
```

- Download PCL 1.8.0 source code here: <https://github.com/PointCloudLibrary/pcl/archive/pcl-1.8.0.tar.gz>
- Unzip the file *pcl-1.8.0.tar.gz* (in an explorer: right click, extract here)
- In a 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 2.2.1** (optional, gives access to GDAL/OGR vector and raster formats)

- Download version 2.2.1 of GDAL, here: <http://download.osgeo.org/gdal/2.2.1/gdal-2.2.1.tar.gz>
- Unzip the file *gdal-2.2.1.tar.gz* (in an explorer: right click, extract here)
- Open a terminal in the *gdal-2.2.1* folder (in an explorer: select the folder, right click, open in a terminal)
- Launch the following commands:

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

4. **GSL** (optional, gives access to a numerical calculation library used in some plugins)

- In a terminal (CTRL + ALT + T):

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

If you want to install dependencies in different locations (for example in c:/program files), you can, for each *LIBNAME\_default\_path.pri* file, duplicate it and rename it *LIBNAME\_user\_path.pri*. After that you just have to modify this second file to use your local path.

## Computree Compilation

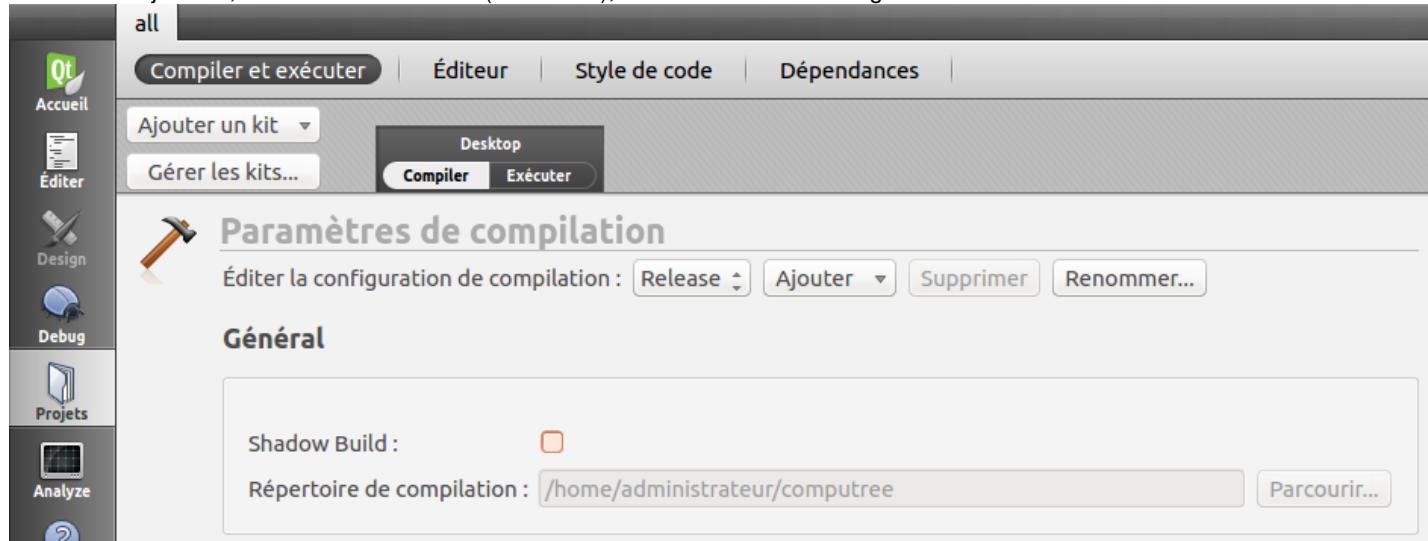
1. Launch Qt Creator

2. Open the project *all.pro*

If you have not installed PCL, delete/comment the following line in *all.pro*:

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

3. On the Project tab, disable shadow builds (check box), for release and/or debug



4. Run qmake on *all.pro*, then compile the project

After updating the source code, if the core of Computree has been modified significantly, it may be necessary to run qmake on each subproject and then to do Recompile on *all.pro*.

## Execution of Computree

Once compiled, you can run from Qt-Creator (green arrow or run on *all.pro*).

## Configure your plugin if you want to use PCL in your code

If you want to use PCL for your development some preparation steps are required:

You must configure the .pro file of your plugin (.pro) as follows (beginning of the file):

```
CT_PREFIX = ../../computreev5  
  
include($$CT_PREFIX)/shared.pri  
include($$PLUGIN_SHARED_DIR)/include.pri
```

```

COMPUTREE += ctlibpcl

include(${CT_PREFIX}/include_ct_library.pri)

```

**Do not forget to compile the libpcl project into the computreev5/library/ctlibpcl folder (open the ctlibpcl.pro file and compile it with QtCreator)**

Just make a *qmake* on the project of your plugin (right click → qmake) and compile it.

## List of svn repositories

If you want to add repositories, or do a manual installation without the scripts, you will find in the table below a list of repositories for all Computree plugins.

To access a repository, of course, you must have adequate rights for the project.

In general, the name of a repository is <http://rdinnovation.onf.fr/svn/nom-du-projet>. The name of the project being the name that appears in the address bar of the browser.

Plugin	Plugin code	Project	Svn Repository
<b>Computree (base)</b>	CT	computree	<a href="http://rdinnovation.onf.fr/svn/computree">http://rdinnovation.onf.fr/svn/computree</a>
<b>ComputreeDevTools</b>	-	computreedevtools	<a href="http://rdinnovation.onf.fr/svn/computreedevtools">http://rdinnovation.onf.fr/svn/computreedevtools</a>
<b>Plugin Onf</b>	ONF	plugin-onf	<a href="http://rdinnovation.onf.fr/svn/plugin-onf">http://rdinnovation.onf.fr/svn/plugin-onf</a>
<b>Plugin Arts Free</b>	ARFR	plugin-arts-free	<a href="http://rdinnovation.onf.fr/svn/plugin-arts-free">http://rdinnovation.onf.fr/svn/plugin-arts-free</a>
<b>Plugin Onf Lsis</b>	OL	plugin-onf-lsis	<a href="http://rdinnovation.onf.fr/svn/plugin-onf-lsis">http://rdinnovation.onf.fr/svn/plugin-onf-lsis</a>
<b>Plugin Generate</b>	GEN	plugin-generate	<a href="http://rdinnovation.onf.fr/svn/plugin-generate">http://rdinnovation.onf.fr/svn/plugin-generate</a>
<b>Plugin ToolKit</b>	TK	plugin-toolkit	<a href="http://rdinnovation.onf.fr/svn/plugin-toolkit">http://rdinnovation.onf.fr/svn/plugin-toolkit</a>
<b>Plugin LVox</b>	LVOX	plugin-lvox	<a href="http://rdinnovation.onf.fr/svn/plugin-lvox">http://rdinnovation.onf.fr/svn/plugin-lvox</a>

[Back to home](#)

### Files

shadow_build.png	62.5 KB	12/22/2016	Piboule Alexandre
kit_qt_551.png	158 KB	12/22/2016	Piboule Alexandre