

# INTÉGRER VTK DANS UN WIDGET QT5

par

Jérémi Panneton

en date du

1<sup>er</sup> octobre 2015

# Prérequis

- **Visual Studio 2013 (ou 2015 avec compilateur v120) :**  
<https://www.visualstudio.com/fr-ca/downloads/download-visual-studio-vs.aspx>
- **Qt (installateur VS2013) :** <http://www.qt.io/download-open-source/>
- **CMake :** <https://cmake.org/download/>
- **VTK :** <http://www.vtk.org/download/>

Versions utilisées (testées et fonctionnelles) :

- Visual Studio 2013 Ultimate
- Qt 5.5.0
- PCL 1.7.2
- CMake 3.3.2
- VTK 6.2.0

Note : PCL 1.7.2 utilise VTK 6.2.0 à l'interne. Il est donc recommandable d'obtenir la même version de VTK que PCL utilise à l'interne.

Le but de ce pas-à-pas est d'intégrer une version de PCL compilée avec VS2015 qui ne possède pas les modules Qt nécessaires (QVTKWidget ne vient pas par défaut). Il faut donc recompiler VTK, qui est le module dans PCL nécessitant les fichiers manquants, de zéro de sorte à remplacer celui par défaut de PCL. VTK ne pouvant être compilé avec VS2015 (problème de version de Qt), il faut donc le compiler avec le compilateur de VS2013 (et ça fonctionne quand même!). Il faut également utiliser une version de Qt installée avec l'installateur et non à partir des sources! Voici les étapes à suivre pour construire et intégrer VTK.

# Étapes à suivre

1. Installer Qt en 32/64 bits et CMake.
2. Extraire VTK (fichiers sources) dans le répertoire désiré.
3. Ouvrir CMake.

Where is the source code : *C:/chemin/vers/VTK-6.2.0*

Where to build the binaries : *C:/chemin/vers/VTK-6.2.0/build*

4. Sélectionner Visual Studio 12 2013 (32 bits) ou Visual Studio 12 2013 Win64 (64 bits) comme générateur (important) et configurer CMake :

Nom	Valeur
<i>BUILD_SHARED_LIBS</i>	Coché
<i>CMAKE_CONFIGURATION_TYPES</i>	Debug;Release
<i>CMAKE_INSTALL_PREFIX</i>	C:/Program Files (x86)/VTK
<i>VTK_Group_Imaging</i>	Coché
<i>VTK_Group_Qt</i>	Coché
<i>VTK_Group_Rendering</i>	Coché
<i>VTK_Group_StandAlone</i>	Coché
<i>VTK_Group_Views</i>	Coché
<i>Options avancés</i>	
<i>VTK_QT_VERSION</i>	5
<i>Module_vtkGUISupportQt</i>	Coché
<i>Module_vtkGUISupportQtOpenGL</i>	Coché
<i>Module_vtkGUISupportQtSQL</i>	Coché
<i>Module_vtkGUISupportQtWebkit</i>	Coché
<i>Module_vtkRenderingQt</i>	Coché
<i>Module_vtkViewsQt</i>	Coché

5. Créer deux nouvelles entrées (Add Entry) :

Nom	Type	Nom
<i>CMAKE_PREFIX_PATH</i>	PATH	C:/Program Files (x86)/Windows Kits/8.1/Lib/winv6.3/um/x86  <b>ou (64 bits):</b>  C:/Program Files (x86)/Windows Kits/8.1/Lib/winv6.3/um/x64
<i>CMAKE_DEBUG_POSTFIX</i>	STRING	-gd

Note : CMAKE\_DEBUG\_POSTFIX permet de suffixer les librairies avec -gd lorsqu'elles sont compilées en debug (pour les distinguer de celles en release).

6. Cliquer sur Configure.

7. Reconfigurer CMake :

Nom	Nom
<i>QT_QMAKE_EXECUTABLE</i>	C:/Qt/Qt5.5.0/5.5/msvc2013/bin/qmake.exe

8. Cliquer sur Configure.

9. Reconfigurer CMake :

Name	Value (example)
<i>Qt5Core_DIR</i>	C:/Qt/Qt5.5.0/5.5/msvc2013/lib/cmake/Qt5Core
<i>Qt5Designer_DIR</i>	C:/Qt/Qt5.5.0/5.5/msvc2013/lib/cmake/Qt5Designer
<i>Qt5Gui_DIR</i>	C:/Qt/Qt5.5.0/5.5/msvc2013/lib/cmake/Qt5Gui
<i>Qt5Network_DIR</i>	C:/Qt/Qt5.5.0/5.5/msvc2013/lib/cmake/Qt5Network
<i>Qt5OpenGL_DIR</i>	C:/Qt/Qt5.5.0/5.5/msvc2013/lib/cmake/Qt5OpenGL
<i>Qt5Sql_DIR</i>	C:/Qt/Qt5.5.0/5.5/msvc2013/lib/cmake/Qt5Sql
<i>Qt5UiPlugin_DIR</i>	C:/Qt/Qt5.5.0/5.5/msvc2013/lib/cmake/Qt5UiPlugin
<i>Qt5WebKit_DIR</i>	C:/Qt/Qt5.5.0/5.5/msvc2013/lib/cmake/Qt5WebKit
<i>Qt5WebKitWidgets_DIR</i>	C:/Qt/Qt5.5.0/5.5/msvc2013/lib/cmake/Qt5WebKitWid gets
<i>Qt5Widgets_DIR</i>	C:/Qt/Qt5.5.0/5.5/msvc2013/lib/cmake/Qt5Widgets
<i>Qt5Xml_DIR</i>	C:/Qt/Qt5.5.0/5.5/msvc2013/lib/cmake/Qt5Xml

10. Cliquer sur Generate (si tout a bien fonctionné).

11. Aller dans le dossier */build* du répertoire de VTK.

12. Ouvrir la solution *VTK.sln* avec Visual Studio 2013 (si possible).

13. Définir le projet ALL\_BUILD comme projet de démarrage, sélectionner le mode Debug et *Générer -> Générer la solution*.

14. Définir le projet INSTALL comme projet de démarrage et Générer -> Générer  
INSTALL. **Si le projet est en mode Debug, il faut renommer les fichiers**  
***QVTKWidgetPlugin-gd.xxx* du répertoire */build/bin/Debug* en**  
***QVTKWidgetPlugin.xxx*.** Sinon, l'installateur ne trouvera pas les fichiers et  
l'installation échouera.
15. Refaire l'étape 13 et 14 en mode Release en prenant soin avant de **faire un**  
**backup du répertoire** défini plus tôt en tant que *CMAKE\_INSTALL\_PREFIX*  
(dans notre cas *C:/Program Files (x86)/VTK*) pour ne pas l'écraser.
16. Toujours dans le répertoire */build*, copier les fichiers  
*bin/Release/QVTKWidgetPlugin.dll* et *lib/Release/QVTKWidgetPlugin.lib* dans  
le répertoire *plugins/designer* **de la version de Qt utilisée par nos**  
**programmes** (dans notre cas 5.6.0).  
Ex : *C:\Qt\qt5\qt5\qtbase\plugins\designer*
17. Aller dans le répertoire défini plus tôt en tant que *CMAKE\_INSTALL\_PREFIX*  
(dans notre cas *C:/Program Files (x86)/VTK* ou le répertoire du backup).

18. Copier les fichiers suivants dans le dossier VTK existant de PCL :

<b>Dossier</b>	<b>Debug</b>	<b>Release</b>
<i>bin/</i>	vtkGUISupportQt-6.2-gd.dll vtkGUISupportQtOpenGL-6.2-gd.dll vtkGUISupportQtSQL-6.2-gd.dll vtkGUISupportQtWebkit-6.2-gd.dll vtkRenderingQt-6.2-gd.dll vtkViewsQt-6.2-gd.dll	vtkGUISupportQt-6.2.dll vtkGUISupportQtOpenGL-6.2.dll vtkGUISupportQtSQL-6.2.dll vtkGUISupportQtWebkit-6.2.dll vtkRenderingQt-6.2.dll vtkViewsQt-6.2.dll
<i>lib/</i>	vtkGUISupportQt-6.2-gd.lib vtkGUISupportQtOpenGL-6.2-gd.lib vtkGUISupportQtSQL-6.2-gd.lib vtkGUISupportQtWebkit-6.2-gd.lib vtkRenderingQt-6.2-gd.lib vtkViewsQt-6.2-gd.lib	vtkGUISupportQt-6.2.lib vtkGUISupportQtOpenGL-6.2.lib vtkGUISupportQtSQL-6.2.lib vtkGUISupportQtWebkit-6.2.lib vtkRenderingQt-6.2.lib vtkViewsQt-6.2.lib
<i>include/</i>	-	QVTKApplication.h QVTKGraphicsItem.h QVTKInteractor.h QVTKInteractorAdapter.h QVTKInteractorInternal.h QVTKPaintEngine.h QVTKWidget.h QVTKWidget2.h QVTKWin32Header.h vtkGUISupportQtModule.h vtkGUISupportQtOpenGLModule.h vtkGUISupportQtSQLModule.h vtkGUISupportQtWebkitModule.h

	vtkRenderingQtModule.h vtkViewsQtModule.h
--	--

19. Modifier les propriétés du projet C++ dans Visual Studio 2015 afin d'y ajouter les librairies manquantes (sachant que PCL est déjà configuré).

Dépendances supplémentaires	
Debug	Release
vtkGUISupportQt-6.2-gd.lib	vtkGUISupportQt-6.2.lib
vtkGUISupportQtOpenGL-6.2-gd.lib	vtkGUISupportQtOpenGL-6.2.lib
vtkGUISupportQtSQL-6.2-gd.lib	vtkGUISupportQtSQL-6.2.lib
vtkGUISupportQtWebkit-6.2-gd.lib	vtkGUISupportQtWebkit-6.2.lib
vtkRenderingQt-6.2-gd.lib	vtkRenderingQt-6.2.lib
vtkViewsQt-6.2-gd.lib	vtkViewsQt-6.2.lib

20. Voir cette page pour des exemples de code :

[http://pointclouds.org/documentation/tutorials/qt\\_visualizer.php](http://pointclouds.org/documentation/tutorials/qt_visualizer.php)

Note : si la fenêtre VTK affiche une erreur, voir cette page :

[http://www.vtk.org/Wiki/VTK/VTK 6 Migration/Factories now require defines](http://www.vtk.org/Wiki/VTK/VTK_6_Migration/Factories_now_require_defines)



# Références

<https://gist.github.com/UnaNancyOwen/9d16060714ba9b28f90e#file-vtk6-md>

[http://www.vtk.org/Wiki/VTK/Tutorials/QtSetup#Qt\\_Creator](http://www.vtk.org/Wiki/VTK/Tutorials/QtSetup#Qt_Creator)

[http://www.vtk.org/Wiki/VTK/VTK 6 Migration/Factories now require defines](http://www.vtk.org/Wiki/VTK/VTK_6_Migration/Factories_now_require_defines)

[http://pointclouds.org/documentation/tutorials/qt\\_visualizer.php](http://pointclouds.org/documentation/tutorials/qt_visualizer.php)