# How to build PyQt5 for Autodesk Maya 2018 64bit

Written by Cyrille Fauvel – Autodesk Developer Network (April 2013)

Updated by Chengxi Li - Autodesk Developer Network (June 2018)

Building SIP, and PyQt for Maya 2018 PyQt [http://www.riverbankcomputing.co.uk] is a python binding to the Qt library. Because Maya uses Qt internally, you can use the PyQt modules in Maya python scripts to create custom UI. PyQt does not have the same licensing as Maya, Qt, or Python. Please consult the PyQt website for information about licensing for PyQt.

Download PyQt: <http://www.riverbankcomputing.com/software/pyqt/download>

Download SIP: <http://www.riverbankcomputing.com/software/sip/download>

The following are instructions for building a copy of the PyQt modules that have been known to work with Maya.

Maya 2018 uses Qt5.6.1 which is binary compatible with the latest version of PyQt – 5.9 / SIP - 4.19.2

Note that it’s important to use the Maya modified version of the Qt source code. A copy of the customized Qt 5.6.1 source is available from Autodesk's Open Source web-site (http://www.autodesk.com/lgplsource) and includes text files describing how to configure, build and install Qt for each platform supported by Maya.

However, be aware that with Maya 2018, there is no more need to build PySide as it is coming by default in Maya, nor you have to rebuild Qt itself as the main Qt tools to build PyQt are now included in the Maya distributions (I.e. qmake, moc, …)

libxml, openSSL, OpenAL, python2.7, qt-5.6.1, and tbb are also coming by default in the Maya include and lib folders, so unless you have a very specific need, you would not need to rebuild any of those libraries like before. Note as well that there is a 'MAYA\_LOCATION/support/opensource' folder now which contains some of the community source.

Important: Maya 2018 now ships without the devkit, include and mkspecs folders. You can get the Maya 2018 devkit from the Autodesk App Store [here](https://apps.autodesk.com/MAYA/en/Detail/Index?id=8656206734503135164&appLang=en&os=Win64) for Windows, OSX, and Linux. Download the devkit and unzip the files into your Maya root folder. Make sure to read the instructions to install the devkit, include and mkspecs folders properly on your system.

The scripts used in this document are now also posted on [Github](https://github.com/cyrillef/Maya-PyQt-Scripts).

Download SIP and PyQt source from '<http://www.riverbankcomputing.co.uk>' - here I downloaded 'sip- 4.19.3' and 'PyQt5\_gpl-5.9'. Unzip them in one folder.

## Mac

/Users/cyrille/Documents/\_Maya2018Scripts/sip-4.19.3

/Users/cyrille/Documents/\_Maya2018Scripts/PyQt5\_gpl-5.9 '

/Users/cyrille/Documents/\_Maya2018Scripts' being my local folder.

Here are the instructions and scripts for building SIP and PyQt.

Follow the instructions from the API docs to setup your environment (Developer Resources > API Guide > Setting up your build environment > Mac OS X environment, in the Maya Documentation)

If you would like to use Xcode 7.3.1 to compile it and you are having multiple installation of Xcode. Please backup /Applications/Xcode.app and use Xcode 7.3.1 to replace it.

Use xcode-select to change active xcode like below:

sudo xcode-select -switch /Applications/Xcode.app/Contents/Developer

The qt.conf file uses **MAYA\_LOCATION** and **DEVKIT\_LOCATION** to locate the expected header/library files. Therefore, users must set both environment variables before building the PyQt5.

DEVKIT\_LOCATION should point to the directory where the devkit include, mkspecs, cmake directories are located.

Modify devkit/bin/qt.conf as below:

[Paths]  
Prefix=  
Libraries=$(MAYA\_LOCATION)/MacOS  
Binaries=$(DEVKIT\_LOCATION)/devkit/bin  
Headers=$(DEVKIT\_LOCATION)/include/Qt  
ArchData=$(DEVKIT\_LOCATION)  
Data=$(DEVKIT\_LOCATION)  
HostData=$(DEVKIT\_LOCATION)  
HostBinaries=$(DEVKIT\_LOCATION)/devkit/bin  
HostLibraries=$(MAYA\_LOCATION)/MacOS

Untar the include/qt-5.6.1-include.tar.gz into /include/Qt

Untar the qt-5.6.1-mkspecs.tar.gz into /Applications/Autodesk/maya2017/mkspecs. Make sure the qconfig.pri looks like this:

**qconfig.pri**

#configuration  
 CONFIG += release def\_files\_disabled exceptions no\_mocdepend stl x86\_64 qt #qt\_framework   
QT\_ARCH = macosx   
QT\_EDITION = OpenSource   
QT\_CONFIG += minimal-config small-config medium-config large-config full-config no-pkg-config dwarf2 phonon phonon-backend accessibility opengl reduce\_exports ipv6 getaddrinfo ipv6ifname getifaddrs png no-freetype system-zlib nis cups iconv openssl corewlan concurrent xmlpatterns multimedia audio-backend svg script scripttools declarative release x86\_64 qt #qt\_framework  
#versioning   
QT\_VERSION = 5.6.1   
QT\_MAJOR\_VERSION = 5   
QT\_MINOR\_VERSION = 6   
QT\_PATCH\_VERSION = 1  
  
#namespaces   
QT\_LIBINFIX =  
QT\_NAMESPACE =   
QT\_NAMESPACE\_MAC\_CRC =

### Build & Install SIP

Please use the script below, you can also find it in GitHub.

#!/usr/bin/env bash  
   
MAYAQTBUILD="`dirname \"$0\"`" # Relative  
export MAYAQTBUILD="`( cd \"$MAYAQTBUILD\" && pwd )`" # Absolutized and normalized  
pushd $MAYAQTBUILD  
   
export SIPDIR=$MAYAQTBUILD/sip-4.19.3  
export MAYA\_LOCATION=/Applications/Autodesk/maya2018  
   
pushd $SIPDIR  
$MAYA\_LOCATION/Maya.app/Contents/bin/mayapy ./configure.py --arch=x86\_64  
make  
sudo make install  
popd  
  
popd

### Build & Install PyQt

Please use the script below, you can also find it in GitHub.

#!/usr/bin/env bash  
   
MAYAQTBUILD="`dirname \"$0\"`" # Relative  
export MAYAQTBUILD="`( cd \"$MAYAQTBUILD\" && pwd )`" # Absolutized and normalized  
pushd $MAYAQTBUILD  
   
export MAYA\_LOCATION=/Applications/Autodesk/maya2018/Maya.app/Contents  
export DEVKIT\_LOCATION=/Applications/Autodesk/maya2018  
export QTDIR=$DEVKIT\_LOCATION/devkit  
export QMAKESPEC=$DEVKIT\_LOCATION/mkspecs/macx-clang  
export INCDIR\_QT=$DEVKIT\_LOCATION/include/Qt  
export LIBDIR\_QT=$MAYA\_LOCATION/MacOS  
  
error=0  
if [ ! -f $QMAKESPEC/qmake.conf ];  
then  
  echo "You need to install qt-5.6.1-mkspecs.tar.gz in $QTDIR/mkspecs !"  
  error=1  
fi  
if [ ! -f $INCDIR\_QT/QtCore/qdir.h ];  
then  
  echo "You need to uncompress $MAYA\_LOCATION/devkit/include/qt-5.6.1-include.tar.gz in $INCDIR\_QT !"  
  error=1  
fi  
# qt.conf - /Applications/Autodesk/maya2018/Maya.app/Contents/Resources  
if [ ! -f $QTDIR/bin/qt.conf ];  
then  
  echo "You need to copy $QTDIR/Resources/qt.conf in $QTDIR/bin !"  
  error=1  
fi  
   
test=`grep 'Data=$(DEVKIT\_LOCATION)' $QTDIR/bin/qt.conf`  
if [ -z "$test" ];  
then  
  echo "You need to edit $QTDIR/bin/qt.conf to use 'Data=\$(DEVKIT\_LOCATION)'"  
  error=1  
fi  
test=`grep 'Headers=$(DEVKIT\_LOCATION)/include/Qt' $QTDIR/bin/qt.conf`  
if [ -z "$test" ];  
then  
  echo "You need to edit $QTDIR/bin/qt.conf to use 'Headers=\$(DEVKIT\_LOCATION)/include/Qt'"  
  error=1  
fi  
test=`grep 'Libraries=$(MAYA\_LOCATION)/MacOS' $QTDIR/bin/qt.conf`  
if [ -z "$test" ];  
then  
  echo "You need to edit $QTDIR/bin/qt.conf to use 'Libraries=\$(MAYA\_LOCATION)/MacOS'"  
  error=1  
fi  
  
if [ $error -eq 1 ];  
then  
    exit  
fi  
   
export DYLD\_LIBRARY\_PATH=$MAYA\_LOCATION/MacOS  
export DYLD\_FRAMEWORK\_PATH=$MAYA\_LOCATION/Frameworks  
   
export SIPDIR=$MAYAQTBUILD/sip-4.19.3  
export PYQTDIR=$MAYAQTBUILD/PyQt5\_gpl-5.9  
   
export SIP\_EXE=$MAYA\_LOCATION/Frameworks/Python.framework/Versions/2.7/bin/sip  
export SIP\_INCLUDE=$MAYA\_LOCATION/Frameworks/Python.framework/Versions/2.7/include/python2.7  
  
pushd $PYQTDIR  
export PATH=$QTDIR/bin:$PATH  
  
echo   
echo Environment  
echo -----------  
set  
echo -----------  
echo QT Settings  
echo -----------  
qmake -query  
echo -----------  
echo  
$MAYA\_LOCATION/bin/mayapy ./configure.py QMAKE\_MAC\_SDK=macosx10.11 QMAKE\_RPATHDIR+=$LIBDIR\_QT --sip=$SIP\_EXE --sip-incdir=$SIP\_INCLUDE -w --no-designer-plugin   
make -j 8  
sudo make install  
popd  
  
popd

Note that I am compiling against Mac OS X SDK 10.11 which is same as the developer environment. If you want to compile against other versions, please modify the script(macosx10.11).

You're done! Please check the testing paragraph at the end of the article.

## Linux

/home/cyrille/Documents/\_Maya2018Scripts/sip-4.19.3

/home/cyrille/Documents/\_Maya2018Scripts/PyQt5\_gpl-5.9

/home/cyrille/Documents/\_Maya2018Scripts' being my local folder.

Here are the instructions and scripts for building SIP and PyQt.

Follow the instructions from the API docs to setup your environment (Developer Resources > API Guide > Setting up your build environment > Linux environments (64 bit), in the Maya Documentation).

The qt.conf file uses **MAYA\_LOCATION** and **DEVKIT\_LOCATION** to locate the expected header/library files. Therefore, users must set both environment variables before building the PyQt5.

DEVKIT\_LOCATION should point to the directory where the devkit include, mkspecs, cmake directories are located.

Please backup your qt.conf first, you'll need to restore it after building PyQt5. Replace …/bin/qt.conf with below:

[Paths]   
Prefix=   
Libraries=$(MAYA\_LOCATION)/lib   
Binaries=$(DEVKIT\_LOCATION)/bin  
Headers=$(DEVKIT\_LOCATION)/include/Qt   
ArchData=$(DEVKIT\_LOCATION)   
Data=$(DEVKIT\_LOCATION)   
HostData=$(DEVKIT\_LOCATION)   
HostBinaries=$(DEVKIT\_LOCATION)/bin

Untar the /include/qt-5.6.1-include.tar.gz into /include/Qt

Untar the /mkspecs/qt-5.6.1-mkspecs.tar.gz into /mkspecs

Make qmake, moc executables from the Maya bin directory

sudo chmod aog+x /usr/autodesk/maya2018/bin/moc   
sudo chmod aog+x /usr/autodesk/maya2018/bin/qmake

### Build & Install SIP

Please use the script below, you can also find it in GitHub.

#!/usr/bin/env bash  
   
MAYAQTBUILD="`dirname \"$0\"`" # Relative  
export MAYAQTBUILD="`( cd \"$MAYAQTBUILD\" && pwd )`" # Absolutized and normalized  
pushd $MAYAQTBUILD  
   
export SIPDIR=$MAYAQTBUILD/sip-4.19.3  
export MAYA\_LOCATION=/usr/autodesk/maya2018  
   
pushd $SIPDIR  
$MAYA\_LOCATION/bin/mayapy ./configure.py  
make  
sudo make install  
popd  
  
popd

### Build & Install PyQt

Please use the script below, you can also find it in GitHub.

#!/usr/bin/env bash  
   
MAYAQTBUILD="`dirname \"$0\"`" # Relative  
export MAYAQTBUILD="`( cd \"$MAYAQTBUILD\" && pwd )`" # Absolutized and normalized  
pushd $MAYAQTBUILD  
   
export MAYA\_LOCATION=/usr/autodesk/maya2017  
export QTDIR=$MAYA\_LOCATION  
export DEVKIT\_LOCATION=$MAYA\_LOCATION  
export QMAKESPEC=$QTDIR/mkspecs/linux-g++-64  
export INCDIR\_QT=$MAYA\_LOCATION/include/Qt  
export LIBDIR\_QT=$QTDIR/lib  
  
error=0  
if [ ! -f $QMAKESPEC/qmake.conf ];  
then  
  echo "You need to install qt-5.6.1-mkspecs.tar.gz in $QTDIR/mkspecs !"  
  error=1  
fi  
if [ ! -f $INCDIR\_QT/QtCore/qdir.h ];  
then  
  echo "You need to uncompress $MAYA\_LOCATION/include/qt-5.6.1-include.tar.gz in $INCDIR\_QT !"  
  error=1  
fi  
# qt.conf - $QTDIR/bin/qt.conf  
if [ ! -f $QTDIR/bin/qt.conf ];  
then  
  echo "You need to copy $QTDIR/Resources/qt.conf in $QTDIR/bin !"  
  error=1  
fi  
  
# The grep string should be in single quote('), if it is in double quote (""),   
# shell will expand the variable, hence the intension of the below grep will fail   
test=`grep 'Headers=$(DEVKIT\_LOCATION)/include/Qt' $QTDIR/bin/qt.conf`  
if [ -z "$test" ];  
then  
  echo "You need to edit $QTDIR/bin/qt.conf to use 'Headers=$(DEVKIT\_LOCATION)/include/Qt'"  
  error=1  
fi  
  
if [ $error -eq 1 ];  
then  
    exit  
fi  
   
export SIPDIR=$MAYAQTBUILD/sip-4.19.3  
export PYQTDIR=$MAYAQTBUILD/PyQt5\_gpl-5.9  
   
pushd $PYQTDIR  
export PATH=$QTDIR/bin:$PATH  
$QTDIR/bin/mayapy ./configure.py LIBDIR\_QT=$LIBDIR\_QT INCDIR\_QT=$INCDIR\_QT MOC=$QTDIR/bin/moc -w --no-designer-plugin   
make -j 8  
sudo make install  
popd  
  
popd

You're done! Please check the testing paragraph at the end of the article.

# Windows

D:\\_\_sdkext\\_Maya2018 Scripts\sip-4.19.3

D:\\_\_sdkext\\_Maya2018 Scripts\PyQt5\_gpl-5.9

D:\\_\_sdkext\\_Maya2018 Scripts being my local folder.

Here are the instructions and scripts for building SIP and PyQt.

Follow the instructions from the API docs to setup your environment (Developer Resources > API Guide > Setting up your build environment > Windows environment (64-bit), in the Maya Documentation)

Please backup your qt.conf first, you'll need to restore it after building PyQt5. Replace …/bin/qt.conf with below:

[Paths]  
Prefix=$(MAYA\_LOCATION)  
Libraries=lib   
Binaries=bin   
Headers=include/Qt  
Data=.  
Plugins=qt-plugins   
Translations=qt-translations   
Qml2Imports=qml

Unzip the /include/qt-5.6.1-include.tar.gz into /include/Qt

Unzip the /mkspecs/qt-5.6.1-mkspecs.tar.gz into /mkspecs

Please run following build scripts with VS2012 x64 Native Tools Command Prompt. If your Maya installed in folders required administrator privilege(e.g. Program files), please run the command prompt as Administrator.

### Environment Setup

@echo off  
  
set MAYAVERSION=2018  
set ADSKQTVERSION=5.6.1  
set SIPVERSION=4.19.3  
set PYQTVERSION=5.9  
set MAYADRIVE=m:  
set BUILDDRIVE=v:  
  
if exist %MAYADRIVE%\nul subst %MAYADRIVE% /d  
subst %MAYADRIVE% "C:\Program Files\Autodesk\Maya%MAYAVERSION%"  
set MAYA\_LOCATION=%MAYADRIVE%  
  
set MAYAPYQTBUILD=%~dp0  
rem Removing trailing \  
set MAYAPYQTBUILD=%MAYAPYQTBUILD:~0,-1%  
  
if exist %BUILDDRIVE%\nul subst %BUILDDRIVE% /d  
subst %BUILDDRIVE% "%MAYAPYQTBUILD%"  
  
set SIPDIR=%BUILDDRIVE%\sip-%SIPVERSION%  
set PYQTDIR=%BUILDDRIVE%\PyQt5\_gpl-%PYQTVERSION%  
rem set ADSKQTDIR=%BUILDDRIVE%\qt-%ADSKQTVERSION%  
set QTDIR=%MAYA\_LOCATION%  
  
set PATH=%QTDIR%\bin;%PATH%  
set MSVC\_VERSION=2015  
set MSVC\_DIR=C:\Program Files (x86)\Microsoft Visual Studio 14.0  
set QMAKESPEC=%QTDIR%\mkspecs\win32-msvc%MSVC\_VERSION%  
set \_QMAKESPEC\_=win32-msvc%MSVC\_VERSION%  
  
if ["%LIBPATH%"]==[""] call "%MSVC\_DIR%\VC\vcvarsall" amd64  
  
set INCLUDE=%INCLUDE%;%MAYA\_LOCATION%\include\python2.7  
set LIB=%LIB%;%MAYA\_LOCATION%\lib

### Build & Install SIP

Please use the script below, you can also find it in GitHub.

@echo off  
set XXX=%~dp0  
if ["%MAYAPYQTBUILD%"]==[""] call "%XXX%setup.bat"  
  
pushd %SIPDIR%  
rem "%MAYA\_LOCATION%\bin\mayapy" configure-ng.py --spec %\_QMAKESPEC\_%  
"%MAYA\_LOCATION%\bin\mayapy" configure.py  
nmake  
nmake install  
popd

### Build & Install PyQt

Please use the script below, you can also find it in GitHub.

@echo off  
set XXX=%~dp0  
if ["%MAYAPYQTBUILD%"]==[""] call "%XXX%setup.bat"  
  
set QMAKESPEC=%QTDIR%\mkspecs\%\_QMAKESPEC\_%  
if not exist "%QMAKESPEC%\qmake.conf" (  
    echo "You need to uncompress %MAYA\_LOCATION%\mkspecs\qt-5.6.1-mkspecs.tar.gz !"  
    goto end  
)  
if not exist "%MAYA\_LOCATION%\include\Qt\QtCore\qdir.h" (  
    echo "You need to uncompress %MAYA\_LOCATION%\include\qt-5.6.1-include.tar.gz in %MAYA\_LOCATION%\include\Qt !"  
    goto end  
)  
findstr /L /C:"Headers=include/Qt" "%MAYA\_LOCATION%\bin\qt.conf" >nul 2>&1  
if ERRORLEVEL 1 (  
    echo "You need to edit %MAYA\_LOCATION%\bin\qt.conf to use 'Headers=include/Qt'"  
    goto end  
)  
findstr /L /C:"-lqtmain -lshell32" "%QTDIR%\mkspecs\common\msvc-desktop.conf" >nul 2>&1  
if ERRORLEVEL 1 (  
    echo "You need to edit %QTDIR%\mkspecs\common\msvc-desktop.conf to use 'QMAKE\_LIBS\_QT\_ENTRY     = -lqtmain -lshell32'"  
    goto end  
)  
if not exist "%MAYA\_LOCATION%\include\Qt\qtnfc.disabled" (  
    echo "You need to rename %MAYA\_LOCATION%\include\Qt\qtnfc to %MAYA\_LOCATION\include\Qt\qtnfc.disabled"  
    goto end  
)  
      
pushd %PYQTDIR%  
  
"%MAYA\_LOCATION%\bin\mayapy" configure.py --spec %QMAKESPEC% LIBDIR\_QT="%QTDIR%\lib" INCDIR\_QT="%QTDIR%\include\Qt" MOC="%QTDIR%\bin\moc.exe" --sip="%QTDIR%\Python\sip.exe" --sip-incdir="%QTDIR%\Python\include" -w --no-designer-plugin  
nmake  
nmake install  
popd  
  
:end

You're done! Please check the testing paragraph at the end of the article.

# Testing

Copy and paste this example in the Maya Script Editor (in a Python tab), and execute the code:

import sys   
from PyQt5.QtWidgets import (QWidget, QToolTip, QPushButton)   
from PyQt5.QtGui import QFont       
   
class Example(QWidget):  
    def \_\_init\_\_(self):  
        super(Example,self).\_\_init\_\_()  
        self.initUI()  
  
    def initUI(self):  
        QToolTip.setFont(QFont('SansSerif', 10))  
        self.setToolTip('This is a <b>QWidget</b> widget')  
        btn = QPushButton('Button', self)  
        btn.setToolTip('This is a <b>QPushButton</b> widget')  
        btn.resize(btn.sizeHint())  
        btn.move(50, 50)   
        self.setGeometry(300, 300, 300, 200)  
        self.setWindowTitle('Tooltips')  
        self.show()  
          
ex = Example()

 If you see the dialog showing, you are all set.