wxWidgets for Qt installation

-----------------------------

IMPORTANT NOTE:

If you experience problems installing, please re-read these

instructions and other related files (todo.txt, bugs.txt and

osname.txt for your platform if it exists) carefully before

mailing wxwin-users or the author. Preferably, try to fix the

problem first and then send a patch to the author.

When sending bug reports tell us what version of wxWidgets you are

using (including the beta) and what compiler on what system. One

example: wxQt 3.1.0, GCC 4.8.1, Ubuntu 14.04

\* The simplest case

-------------------

If you compile wxWidgets on Linux for the first time and don't like to read

install instructions just do (in the base dir):

> mkdir buildqt

> cd buildqt

> ../configure --with-qt

> make

> su <type root password>

> make install

> ldconfig

[if you get "ldconfig: command not found", try using "/sbin/ldconfig"]

If you don't do the 'make install' part, you can still use the libraries from

the buildgtk directory, but they may not be available to other users.

If you want to remove wxWidgets on Unix you can do this:

> su <type root password>

> make uninstall

> ldconfig

\* The simplest errors

---------------------

For any configure errors: please look at config.log file which was generated

during configure run, it usually contains some useful information.

configure reports, that you don't have Qt installed although you are very

sure you have. Well, you have installed it, but you also have another

version of the Qt installed, which you may need to remove. Or maybe you

installed it in a non-default location and configure can't find it there,

so please check that your PATH variable includes the path to the correct

qtconfig/pkg-config. Also check that your LD\_LIBRARY\_PATH or equivalent

variable contains the path to Qt libraries if they were installed in a

non-default location.

\* The simplest program

----------------------

Now create your super-application myfoo.cpp and compile anywhere with

g++ myfoo.cpp `wx-config --libs --cxxflags` -o myfoo

\* GUI libraries

---------------

wxWidgets/Qt requires the Qt library to be installed on your system. It has

to be a stable version, preferably Qt 5.2.1 or later.

\* Building wxQT on Ubuntu

-------------------------

Install latest Qt5 packages (qt5-default). To build unit tests, libcppunit-dev

is required. You will need to install other dependencies to compile wxWidgets

depending on the features you'll want to use (build-essential libjpeg-dev

libtiff5-dev ubuntu-restricted-extras freeglut3 freeglut3-dev libsdl1.2-dev

libgstreamer-plugins-base0.10-dev)

Then create a build directory, configure and compile:

mkdir bldqt5

cd bldqt5

../configure --with-qt --enable-debug

make

make samples

If everything is ok, you can do the make install as specified before.

Optionally, you can build and run Unit Tests:

cd tests

make

./test\_gui

\* Building wxGT on Android

--------------------------

Download Android Native Development Kit (NDK), tandalone Android Software

Development Kit (SDK), install them and perform the following instructions to

prepare the cross-compilation tool-chain to (change NDK and other paths):

NDK=~/src/android-ndk-r9d

SDK=~/src/android-sdk-linux

export ANDROID\_NDK\_ROOT=$NDK

$NDK/build/tools/make-standalone-toolchain.sh \

--toolchain=arm-linux-androideabi-4.8 --platform=android-9 \

--install-dir=/tmp/ndk

export PATH=/tmp/ndk/bin:$PATH

export CC=arm-linux-androideabi-gcc

export CXX=arm-linux-androideabi-g++

Also, you'll need to download the Qt library bundle that matches your operating

system installed package (5.2.1 in this case installed in ~/src/qt, you'll need

the android\_armv5/ android\_armv7/ android\_x86/ pre-compiled folders to

cross-compile for that architectures)

Then, create a build directory (under the wxWidgets folder), configure for

Andrid (disable currently unsupported/uneeded features) and run make:

cd ~/src/wxWidgets

mkdir bldqt5droid

cd bldqt5droid

../configure --with-qt --enable-debug --build=x86\_64-unknown-linux-gnu \

--host=arm-linux-androideabi --disable-compat28 --disable-shared \

--disable-arttango --enable-image --disable-dragimage --disable-sockets \

--with-libtiff=no --without-opengl --disable-baseevtloop --disable-utf8

make

You can now compile and link your app against this build, and finally

package it for Android using standard APK tools.

\* Create your configuration

---------------------------

Usage:

./configure options

If you want to use system's C and C++ compiler,

set environment variables CC and CXX as

% setenv CC cc

% setenv CXX CC

% ./configure [options]

to see all the options please use:

./configure --help

It is recommended to build wxWidgets in another directory (maybe a

subdirectory of your wxWidgets installation) as this allows you to

have multiple configurations (for example, debug and release or GTK

and Motif) simultaneously.

\* Feature Options

-----------------

When producing an executable that is linked statically with wxQt

you'll be surprised at its immense size. This can sometimes be

drastically reduced by removing features from wxWidgets that

are not used in your program.

Please see the output of "./configure --help" for comprehensive list

of all configurable options.

Apart from disabling certain features you can very often "strip"

the program of its debugging information resulting in a significant

reduction in size.

----------------------

In the hope that it will be useful,

The wxWidgets Team