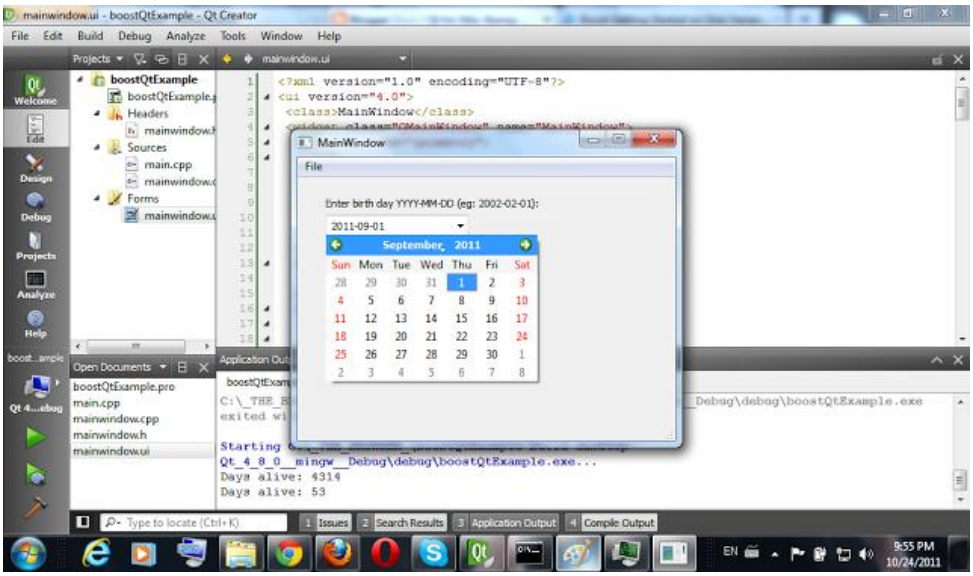
Qt / Boost for Beginners (step by step example)

Boost is must-have set of headers and libraries for any C++ coder and in some moment you'll need to use it with Qt. Most of the boost headers are ... headers only, but sometimes you'll need the real libraries. If you are new to boost, below I'll provide you with step-by-step instructions on how to compile boost libraries and how to use them from Qt Creator. I presume, however, that you are at least basically familiar with Qt: you have Qt SDK installed on your Windows machine and you know how to use Qt Creator. For this example I am using MingW compiler (because this is the default compiler for Qt)

In this tutorial we will:
1) install *boost* platform
2) build *boost* libraries for MingW (windows)
3) build GUI Qt app that uses the *boost* library

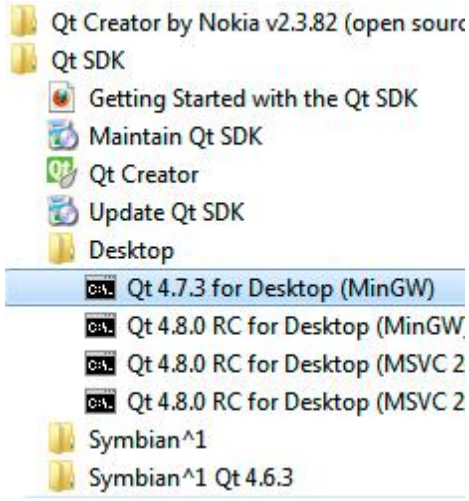
As in any good tutorial, the resul first:



[http://4.bp.blogspot.com/-RDaxXfnC1x4/TqW081ULMqI/AAAAAAAAAHc/y9SQFFfR7HA/s1600/tutor_result.png]

So, let's start:
1. Download and install *Boost* on the same machinw where Qt is installed
1.1. Go to http://www.boost.org/users/download/ [http://www.boost.org/users/download/] and get the latest version of Boost (I recommend downloading the .7z version or, if you prefer zip format, don't use the windows internal unzip utility because it is too slow, really way too slow) At the time of this writing the latest version of Boost was 1.47.0 so down below you will see directories like *boost_1_47_0*
1.2. Decompress the archive on your disk. In my case I decompressed the libraries in C:\boost and got the root path of boost like this: *C:/boost/boost_1_47_0*

2. Now we must build the libraries (remember, although most of the *Boost*platform is just header files, here we want to use the whole thing). In our case we will use MingW/gcc so:
2.1. Open the toolchain command prompt:



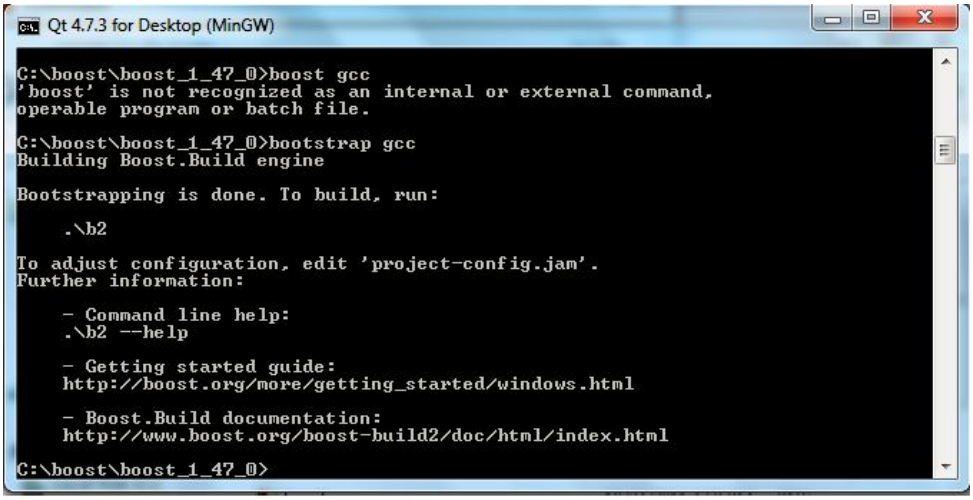
[http://1.bp.blogspot.com/-S8Wi593ybr0/TqWke0zy7II/AAAAAAAAAHM/NCeoHkNm8Xk/s1600/tutor_command_prompt.png]

2.2. In the command prompt, go to *Boost* root folder. In my case:

```
cd /boost/boost_1_47_0
```

2.3. Build the building tool... Yes, the first step is to create the *bjam* tool that will later build the libraries. We build bjam tool with this command:

```
bootstrap gcc
```



[http://3.bp.blogspot.com/-4hTf88OFT30/TqWlVn5E5VI/AAAAAAAAAHU/tgdXW8uQ3Xs/s1600/tutor_bootstrap.png]

2.3. Now we can build the libraries themselves -- type the following command and press enter:

```
bjam --build-dir=c:/boost_lib toolset=gcc stage
```

This will take a long time (depending on your computer). While waiting, you may want to read what the above command does here: http://www.boost.org/doc/libs/1_41_0/more/getting_started/unix-variants.html#easy-build-and-install [http://www.boost.org/doc/libs/1_41_0/more/getting_started/unix-variants.html#easy-build-and-install]

As you will see in the above link, there is easier way but in my case I prefer to

know what I am doing (and which version of gcc I am using, because there are plenty of them on my PC), that's why I am installing the libraries with few more commands (and with much more control)

Now, when the build process finishes, according tho Boost documentation, you should find the Boost library binaries in the `stage/lib/` subdirectory of your build directory. In our case *c:\boost_lib\stage\lib*): but that didn't happen (may be because I didn't created the target directory first?) However, don't panic! The libraries were built inside the boost folder itself, in:

```
C:\boost\boost_1_47_0\bin.v2\libs
```

So I just moved all the libraries from this place to the actual dir I was planning to use for them (*c:\boost_lib*)

Now we are ready to go with the Qt part of the example.

3) Download the Qt example project:

qtBoostExample.zip [http://wonderwebware.com/qtBoostExample.zip]

And uncompress somewhere on your hdd. The example is slightly modified version of the original *Boost* example located here:
http://www.boost.org/doc/libs/1_42_0/doc/html/date_time/examples.html#date_time.examples.days_alive [http://www.boost.org/doc/libs/1_42_0/doc/html/date_time/examples.html#date_time.examples.days_alive]

The important parts are these:
3.1) In the project file you will see the include path to boost:

```
#Dont forget to add the path to your boost install folder
INCLUDEPATH += c:/boost/boost_1_47_0
```

(replace with the actual folder where you decompressed boost)

3.2) And don't forget to link against the static libraries needed (as in our example):

```
LIBS   +=   C:\boost_lib\date_time\build\gcc-mingw-
4.4.0\release\link-static\threading-
multi\libboost_date_time-mgw44-mt-1_47.a
```

(this is, obviously, the location of the static library on my hdd; if you ended with libraries in other place, just point to the exact one)

That's it, Ctrl+R and you should get your first *Boost*-enabled application. This tutorial is not about the actual application (there is nothing special inside) so, for now -- Enjoy!

PS: The whole tutorial was written at once, while experimenting and is here only because I want to keep my experiments online for future reference so -- please --

do not judge me too hard.

PPS: Did you know that Qt works on Atom 1.66GHz CPU with 1GB RAM? The whole tutorial was tested on and written with my Lenovo s10-3t toy. Not the fastest dev.environment, but still impressive :)

Posted 24th October 2011 by Vladislav Hristov

Labels: boost, qt, windows

2 View comments

Alexander Oblovatniy 5 December 2012 at 08:50

thank you. such a quest for windows. for linux so more easier:)

Reply

Brendon Bolin 17 April 2013 at 15:32

Thanks for the write-up. It was quite useful. I would recommend one slight change, you can just link to the directory of the libs then you don't have to worry about which are needed.

Something like,
LIBS += -LC:/boost_lib/boost/bin.v2/libs

Reply

Enter your comment...

Comment as: Select profile...

Publish Preview