

boost_1_60_0 .zip installation in windows

I have performed below steps for windows 7

Boost link http://www.boost.org/doc/Libs/1_49_0/Libs/python/doc/

1. **Downloaded** boost_1_60_0 .zip
2. `cd \boost_1_60_0\tools\build`
3. **Ran** below command **in** command prompt

```
bootstrap.bat gcc
bjam --toolset=gcc "--prefix=C:\DestinationFolder" install
```

Now i have b2.exe & bjam.exe **in** C:\DestinationFolder\bin

Can anyone please help me in what to do next as I am unable to process further

c++ boost

asked Feb 5 '16 at 6:20



[user3812837](#)

74 ● 1 ● 10

Which compiler do you have? Are you really sure that it is gcc? – [Sergei Nikulov](#) Feb 5 '16 at 6:53

I have MinGW installed – [user3812837](#) Feb 5 '16 at 7:19

2 And i took reference from stackoverflow.com/questions/7947542/building-boost-on-windows – [user3812837](#) Feb 5 '16 at 7:19

1 I would rather use the [prebuilt Windows binaries](#). – [RHertel](#) Feb 5 '16 at 7:42

1 I have done this a while ago, so I don't remember every step; but I think it was fairly easy: Download the latest version, `boost_1_60_0-msvc-14.0-64.exe` or `boost_1_60_0-msvc-14.0-32.exe` (64 or 32 bit, depending on your system), and run the exe file (e.g., open in a folder and double-click). This should start an interactive installer which, as I recall, of the usual type. Sorry that I can't be more specific. The only part I found somewhat more difficult was to ensure that the compiler finds the libraries. I think I needed to add the installation directory to the `PATH` environment variable. – [RHertel](#) Feb 5 '16 at 7:59

1 Answer

Since yours is now the third "How do I build boost on Windows?" question that I've seen since `1.60.0` was released here are my own personal Windows boost build notes:

Windows does not directly support boost, so you can download it and put it wherever you want. The boost user guide recommends creating a `BOOST_ROOT` environment variable with the location of boost.

Note: in the following examples `2>&1 | tee ??_build.txt` is optional, but it's useful to keep a build log...

Building for Visual Studio 2015

In a Visual Studio tools Command Prompt:

```
cd boost_1_xx_0
call bootstrap.bat
```

For static libraries:

```
b2 -j8 toolset=msvc-14.0 address-model=64 architecture=x86 link=static threading=multi
runtime-link=shared --build-type=complete stage
2>&1 | tee msvc_static_build.txt
```

Note: thread must be built with dynamic linking see: <https://studiodreya.com/2015/05/20/the-simplest-way-of-building-boost-1-58-for-32-bit-and-64-bit-architectures-with-visual-studio/>

For dynamic thread library:

```
b2 -j8 toolset=msvc-14.0 address-model=64 architecture=x86 link=shared threading=multi
runtime-link=shared --with-thread --build-type=minimal stage
2>&1 | tee msvc_thread_build.txt
```

For all as dynamic libraries:

```
b2 -j8 toolset=msvc-14.0 address-model=64 architecture=x86 link=shared threading=multi
runtime-link=shared --build-type=complete stage
2>&1 | tee msvc_dynamic_build.txt
```

Building for MinGw

Ensure that gcc/mingw is in the path, e.g.: C:\Qt\Tools\mingw491_32\bin

```
cd boost_1_xx_0
bootstrap.bat mingw

b2 toolset=gcc link=shared threading=multi --build-type=complete stage
2>&1 | tee mingw_build.txt
```

Note: since boost 1.61.0 you may need to change: bootstrap.bat mingw to bootstrap.bat gcc

edited May 21 '16 at 15:00

answered Feb 5 '16 at 11:38



kenba

1,217 ● 1 ● 6 ● 14