

The objective

Install Rcpp on Windows 7 Pro x64 to test out Romain's Rcpp solution to the old r-wiki optimisation challenge

The Problem

I had zero problems installing and running Rcpp code on [Ubuntu Linux](#) 11.10 but just could not get the same to work on Windows 7 Pro x64, often getting the following warning and errors:

```
Error in compileCode(f, code, language = language,
verbose = verbose) :
Compilation ERROR, function(s)/method(s) not created!
```

```
cygwin warning:
MS-DOS style path detected: C:/PROGRA~1
/R/R-214~1.0/etc/x64/Makeconf
Preferred POSIX equivalent is: /cygdrive/c/PROGRA~1
/R/R-214~1.0/etc/x64/Makeconf
CYGWIN environment variable option
“nodosfilewarning” turns off this warning.
Consult the user’s guide for more details about POSIX
paths:
```

<http://cygwin.com/cygwin-ug-net/using.html#using-pathnames>

```
x86_64-w64-mingw32-g++.exe: C:/Program: No such file
or directory
x86_64-w64-mingw32-g++.exe: Files/R/R-2.14.0/library
/Rcpp/lib/x64/libRcpp.a: No such file or directory
```

The Solution

It turns out that there's two issues here:

(1) the cygwin warning which we can ignore as it has nothing to do with Rcpp and is caused by using the 64bit version of MinGW from Rtools when adding it to our PATH variable (easily solved by choosing the 32bit version instead, shown below in step 5 part viii)

(2) the actual Rcpp issue itself which happened because I had originally installed R on a path with a space in it. In fact (and this is something I didn't previously know) it's officially recommended that one does not install R on Windows with a path containing a space (see [rw-FAQ 2.2](#)).

So armed with this knowledge, here is how I got Rcpp working from beginning to end (a lot of the sub-steps below are blindingly obvious but I included them anyway for the sake of completeness in case other people with similar problems come across this post):

1. Install R on a path that does not contain a space.
 - (i) download R from: <http://cran.r-project.org/bin/windows/base/>
 - (ii) double click the downloaded .exe file
 - (iii) press Run when the security warning appears
 - (iv) press Yes when the User Account Control message box appears
 - (v) Choose your language (I choose English)
 - (vi) when prompted for where to install R, choose a location without a space. I choose “C:\R\R-2.14.0”
 - (vii) click Next on all following screens to finish installation (I left the defaults unaltered).
2. Install Rtools for components necessary to build R (this has the tool chain require for C++ code compilation)
 - (i) Go to this webpage: <http://www.murdoch-sutherland.com/Rtools/>
 - (ii) In the downloads section, choose a compatible version for

your version of R. I'm using R-2.14.0 and thus

chose [Rtools214.exe](#)

(iii) press Run when the security warning appears

(iv) press Yes when the User Account Control message box appears

(v) choose your language (I choose English)

(vi) choose where to install (I chose "C:\R\Rtools")

(vii) click Next on all following screens to finish installation (I left the defaults unaltered).

3. Download [batchfiles](#) to always point to the latest version of R on your system when running R from the command line (I have found that this saves a lot of time in the long run)

(i) go to: <http://cran.r-project.org/contrib/extra/batchfiles/>

(ii) choose the latest version (I chose batchfiles_0.6-6.zip)

(iii) when downloaded, right click on file and click Extract All.

(iv) click Browse and choose where you want to put the extracted folder (I chose "C:\R")

4. Download Redmond Path Utility to alter PATH variables in a very user friendly fashion:

(i) go to: http://download.cnet.com/Redmond-Path/3000-2094_4-10811594.html

(ii) click on "Download Now CNET Secure Download"

(iii) double click the downloaded .exe file

(iv) press Run when the security warning appears

(iv) press Yes when the User Account Control message box appears

(v) press Next

(vi) press Decline (unless you want the annoying advertising extra).

(vii) click open

(viii) An explorer window will open with RedmondPath.zip

(ix) click Extract All.

(x) click Browse and choose where you want to put the extracted folder (I chose "C:\R")

5. Edit PATH variable to allow system wide access to the current version of R on the computer and components of Rtools

(i) double click the Redmond Path Utility from step 4 above (mine is in: "C:\R\RedmondPath\Redmond Path.exe")

(ii) click Yes when the User Account Control message box appears

(iii) click the green "+" icon in the top left corner so we can add elements to the PATH variable

(iv) In the window which pops open, navigate to "C:\R\batchfiles_0.6-6" and click OK

(v) click the green "+" icon in the top left corner so we can add elements to the PATH variable

(vi) In the window which pops open, navigate to "C:\R\Rtools\bin" and click OK

(vii) click the green "+" icon in the top left corner so we can add elements to the PATH variable

(viii) In the window which pops open, navigate to "C:\R\Rtools\MinGW\bin" and click OK

6. Restart your computer (this solved an issue where the edits to the PATH variable above had not taken immediate affect)

7. Open R and run the following code

```
1 # install packages
2 install.packages(c("Rcpp", "rbenchmark"),
3
4 # load main two packages
5 library(Rcpp)
6 library(inline)
7
8 # do something with Rcpp to quickly check
9 body <- '
```

```

10 NumericVector xx(x);
11 return wrap( std::accumulate( xx.begin(),
12                               xx.end(), 0,
13                               add ) );
14
15 x <- 1
16 y <- 2
17 res <- add(c(x, y))
18 res
19 #[1] 3

```

And there you have it, Rcpp working on Windows! I was able to run Romain's Rcpp code ([see his blog for the code](#)) without any problems. Awesome stuff.

Useful resources I read in order to work out how to correctly install Rcpp:

[Rcpp-FAQ](#)

<http://lists.r-forge.r-project.org/pipermail/rcpp-devel/>

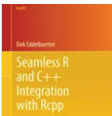


<http://cran.r-project.org/doc/manuals/R-admin.html#The-Windows-toolset>

Comments: 31

13 5 3
 Like Tweet +1

Send
 To leave a comment for the author, please follow the link and comment on his blog:
[Consistently Infrequent » R](#).
 R-bloggers.com offers **daily e-mail updates** about R news and tutorials on topics such as: visualization ([ggplot2](#), [Boxplots](#), [maps](#), [animation](#)), programming ([RStudio](#), [Sweave](#), [LaTeX](#), [SQL](#), [Eclipse](#), [git](#), [hadoop](#), [Web Scraping](#)) statistics ([regression](#), [PCA](#), [time series](#), [trading](#)) and more...

Related aRticles

- 
New book: Seamless R and C++ Integration with Rcpp
 A new book from Dirk Eddelbuettel, co-author of the Rcpp package, is now available. Seamless R and C++ Integration with Rcpp can be ...
- 
Rcpp 0.10.6
 A new maintenance release 0.10.6 of Rcpp is now on the CRAN network for GNU R; binaries for Debian have been uploaded as well. This ...
- 
Slides from Rcpp talk in Chicago
 A couple of days ago, I gave a talk to the Chicago R Users Group which is run ever-so-smoothly by Paul Teetor and Chase ...

If you got this far, why not **subscribe for updates** from the site?
 Choose your flavor: [e-mail](#), [twitter](#), [RSS](#), or [facebook](#)...

Like 13 Send Tweet 5 +1 3

Comments are closed.

Top 3 Posts from the past 2 days

- [The homogenization of scientific](#)