RcppClassic: Deprecated Rcpp API

by Dirk Eddelbuettel and Romain François

December 2, 2010

This document presents the `RcppClassic` package. This package has been factored out of `Rcpp` and only contains code that is considered deprecated.

This package is released for the sole purpose of allowing package authors that are still using the classic `Rcpp` API to keep their package buildable. This document explains the changes needed in a package to use both the current and classic Rcpp APIs.

1 Using RcppClassic

A few changes are needed in packages that want to use the classic Rcpp API that is contained in RcppClassic.

1.1 The DESCRIPTION file

The client package must declare that it depends on both Rcpp and RcppClassic, in the Depends field and the LinkingTo field, so it must contain this:

```
Depends: RcppClassic, Rcpp
LinkingTo: RcppClassic, Rcpp
```

1.2 Makevars

The Makevars file must be updated so that user libraries for both Rcpp and RcppClassic are used:

```
PKG_LIBS = `$(R_HOME)/bin/Rscript -e "Rcpp:::LdFlags()"` \
           `$(R_HOME)/bin/Rscript -e "RcppClassic:::LdFlags()"`
```

1.3 Makevars.win

The Makevars.win must also be updated for the same reason:

```
PKG_LIBS = $(shell "${R_HOME}/bin${R_ARCH_BIN}/Rscript.exe" -e "Rcpp:::LdFlags()")
PKG_LIBS += $(shell "${R_HOME}/bin${R_ARCH_BIN}/Rscript.exe" -e "RcppClassic:::LdFlags()")
```

1.4 include RcppClassic.h instead of Rcpp.h

Finally, all instances of this line :

```
#include <Rcpp.h>
```

need to be replaced by:

```
#include <RcppClassic.h>
```

2   Not Using RcppClassic