

# Package ‘RcppExamples’

March 17, 2025

**Title** Examples using 'Rcpp' to Interface R and C++

**Version** 0.1.10

**Date** 2025-03-17

**Description** Examples for Seamless R and C++ integration

The 'Rcpp' package contains a C++ library that facilitates the integration of R and C++ in various ways. This package provides some usage examples. Note that the documentation in this package currently does not cover all the features in the package. The site <<https://gallery.rcpp.org>> regroups a large number of examples for 'Rcpp'.

**Imports** Rcpp

**LinkingTo** Rcpp

**URL** <https://github.com/eddelbuettel/rcppexamples>,  
<https://dirk.eddelbuettel.com/code/rcpp.examples.html>

**BugReports** <https://github.com/eddelbuettel/rcppexamples/issues>

**License** GPL (>= 2)

**RoxygenNote** 6.0.1

**NeedsCompilation** yes

**Author** Dirk Eddelbuettel [aut, cre] (<<https://orcid.org/0000-0001-6419-907X>>),  
Romain Francois [aut] (<<https://orcid.org/0000-0002-2444-4226>>)

**Maintainer** Dirk Eddelbuettel <edd@debian.org>

**Repository** CRAN

**Date/Publication** 2025-03-17 12:40:08 UTC

## Contents

RcppExamples-package . . . . .	2
factor2char . . . . .	2
RcppDataFrame . . . . .	3
RcppDateExample . . . . .	4
RcppListExample . . . . .	5

RcppMatrixExample . . . . .	6
RcppNumericVectorExample . . . . .	7
RcppRNGsExample . . . . .	8
RcppStringVectorExample . . . . .	9

<b>Index</b>	<b>10</b>
--------------	-----------

---

RcppExamples-package    *Examples for the Rcpp R/C++ Interface library*

---

## Description

This package shows some simple examples for the use of **Rcpp**.

It can also serve as a working template to create packages that use **Rcpp** to interface C++ code or libraries.

## Details

The **Rcpp** package provides a number of C++ classes that ease access to C++ from R. This comprises both passing parameters to functions, as well as returning results back from C++ to R.

The **RcppExamples** package provides some simple examples for use of **Rcpp**. At this point the documentation is not complete in the sense of not covering all accessible classes. However, several basic use cases are illustrated,

## Author(s)

Dominick Samperi wrote the initial versions of Rcpp (and RcppTemplate) during 2005 and 2006. Dirk Eddelbuettel made some additions, and became maintainer in 2008. Dirk Eddelbuettel and Romain Francois have been extending Rcpp since 2009.

## See Also

The <https://gallery.rcpp.org> site regroups a number of examples.

---

factor2char                    *Convert Index and String Vector into CharacterVector, and vice versa*

---

## Description

These two functions are an illustration of how `as.character` and `as.factor` may be reimplemented at the C++ level.

## Usage

```
factor2char(iv)
```

```
char2factor(sv)
```

**Arguments**

iv	A Integer Vector corresponding to numeric representation of the factor This vector is also expected to have an attribute 'levels' with the factor levels
sv	A String Vector

**Value**

A Character Vector which at each position contains the level value of the corresponding index, or a Factor, depending on the function

**Examples**

```
f <- as.factor(c("red", "green", "blue", "red"))
factor2char(f);
f <- as.factor(c("red", "green", "blue", "red"))
v <- factor2char(f);
char2factor(v)
```

---

RcppDataFrame

*Rcpp::DataFrame example for Rcpp*


---

**Description**

A DataFrame can be passed C++ and can be instantiated as a corresponding C++ object using the Rcpp API.

This example shows (in the corresponding C++ code) how to access, modify and create a data frame.

**Details**

Usage of Rcpp::DataFrame is fully defined in the respective header file.

The C++ source file corresponding to the this function does the following:

```
// we receive a 'DF' data.frame object
// and access each column by name
Rcpp::IntegerVector a = DF["a"];
Rcpp::CharacterVector b = DF["b"];
Rcpp::DateVector c = DF["c"];

// do something
a[2] = 42;
b[1] = "foo";
c[0] = c[0] + 7;                               // move up a week

// create a new data frame
Rcpp::DataFrame NDF =
```

```

Rcpp::DataFrame::create(Rcpp::Named("a")=a,
Rcpp::Named("b")=b,
Rcpp::Named("c")=c);

    // and return old and new in list
    return(Rcpp::List::create(Rcpp::Named("origDataFrame")=DF,
Rcpp::Named("newDataFrame")=NDF));

```

### Author(s)

Dirk Eddelbuettel and Romain Francois

### Examples

```

## Not run:
RcppDataFrame()

## End(Not run)

```

---

RcppDateExample	<i>C++ classes for interfacing date and datetime R objects</i>
-----------------	--

---

### Description

Rcpp has the classes Rcpp::Date, Rcpp::Datetime, Rcpp::DateVector and Rcpp::DatetimeVector.

### Details

In the C++ code for the RcppDateExample.cpp file:

```

// [[Rcpp::export]]
List DateExample(DateVector & dv, DatetimeVector & dtv) {
    Function formatDate("format.Date");
    Function formatDatetime("format.POSIXct");

    Rprintf("\nIn C++, seeing the following date value\n");
    for (int i=0; i<dv.size(); i++) {
        Rcout << as<std::string>(formatDate(wrap(dv[i]))) << std::endl;
        dv[i] = dv[i] + 7; // shift a week
    }
    Rprintf("\nIn C++, seeing the following datetime value\n");
    for (int i=0; i<dtv.size(); i++) {
        Rcout << as<std::string>(formatDatetime(wrap(dtv[i]))) << std::endl;
        dtv[i] = dtv[i] + 0.250; // shift 250 millisec
    }

    // Build result set to be returned as a list to R.

```

```

        return List::create(Named("date", dv),
                           Named("datetime", dtv));
    }

```

### Author(s)

Dominick Samperi wrote the initial versions of Rcpp (and RcppTemplate) during 2005 and 2006. Dirk Eddelbuettel made some additions, and became maintainer in 2008. Dirk Eddelbuettel and Romain Francois have been extending Rcpp since 2009.

### References

*Writing R Extensions*, available at <https://www.r-project.org>.

### Examples

```

# set up date and datetime vectors
dvec <- Sys.Date() + -2:2
dtvec <- Sys.time() + (-2:2)*0.5

# call the underlying C++ function
result <- RcppDateExample(dvec, dtvec)

# inspect returned object
result

```

---

RcppListExample      *Examples of uses of List*

---

### Description

List is an Rcpp class that can be used to manipulate R lists.

### Arguments

params            A heterogeneous list specifying method (string), tolerance (double), maxIter (int) and startDate (Date in R, RcppDate in C++).

### Value

RcppListExample returns a list containing:

method	string input paramter
tolerance	double input paramter
maxIter	int input parameter
startDate	Date type with starting date
params	input parameter list (this is redundant because we returned the input parameters above)

**Author(s)**

Dominick Samperi wrote the initial versions of Rcpp (and RcppTemplate) during 2005 and 2006. Dirk Eddelbuettel made some additions, and became maintainer in 2008. Dirk Eddelbuettel and Romain Francois have been extending Rcpp since 2009.

**References**

*Writing R Extensions*, available at <https://www.r-project.org>.

**Examples**

```
# set up some value
params <- list(method='BFGS',
              tolerance=1.0e-5,
              maxIter=100,
              startDate=as.Date('2006-7-15'))

# call the underlying C++ function
result <- RcppListExample(params)

# inspect returned object
result
```

---

RcppMatrixExample      *Example of using Rcpp NumericMatrix*

---

**Description**

The NumericMatrix class represents numeric matrices

**Details**

The C++ code presented in the MatrixExample.cpp file:

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

// suncc needs help to disambiguate between sqrt( float ) and sqrt(double)
inline static double sqrt_double(double x) { return ::sqrt(x); }

using namespace Rcpp;

// [[Rcpp::export]]
List MatrixExample(const NumericMatrix & orig) {
    NumericMatrix mat(orig.nrow(), orig.ncol());

    // we could query size via
```

```

    // int n = mat.nrow(), k=mat.ncol();
    // and loop over the elements, but using the STL is so much nicer
    // so we use a STL transform() algorithm on each element
    std::transform(orig.begin(), orig.end(), mat.begin(), sqrt_double );

    return List::create(Named("result") = mat,
                       Named("original") = orig);
}

```

### Author(s)

Dominick Samperi wrote the initial versions of Rcpp (and RcppTemplate) during 2005 and 2006. Dirk Eddelbuettel made some additions, and became maintainer in 2008. Dirk Eddelbuettel and Romain Francois have been extending Rcpp since 2009.

### References

*Writing R Extensions*, available at <https://www.r-project.org>.

### Examples

```

M <- matrix((1:16)^2, 4)
RcppMatrixExample(M)

```

---

RcppNumericVectorExample  
*Rcpp NumericVector example*

---

### Description

Example on how to use a NumericVector and manipulate it with the STL.

### Details

```

NumericVector orig ; // from R
NumericVector vec(orig.size()); // create a target vector of the same size

// we could query size via
// int n = vec.size();
// and loop over the vector, but using the STL is so much nicer
// so we use a STL transform() algorithm on each element
std::transform(orig.begin(), orig.end(), vec.begin(), sqrt_double );

return List::create(Named("result") = vec,
                   Named("original") = orig);

```

As shown in the example section, provided the seed is reset, the exact same draws can be obtained in R itself – which is important for reproducibility.

### Author(s)

Dirk Eddelbuettel and Romain Francois

### Examples

```
RcppNumericVectorExample(seq(1,9)^2)
```

---

RcppRNGsExample	<i>Rcpp RNGs example</i>
-----------------	--------------------------

---

### Description

Rcpp sugar provides numerous *p/q/d/r* functions for numerous distributions.

This example shows (in the corresponding C++ code) how to draw from three different distributions and returns a data frame.

### Details

The various header file, and the Rcpp sugar vignette, provide full documentation for Rcpp sugar.

The C++ source file corresponding to the this function does the following:

```
int n; // length passed in from R

NumericVector rn = rnorm(n);
NumericVector rt = rt(n, 1.0);
NumericVector rp = rpois(n, 1.0);

// create a new data frame to return draws
return DataFrame::create(Named("rnorm") = rn,
                        Named("rt") = rt,
                        Named("rpois") = rp);
```

As shown in the example section, provided the seed is reset, the exact same draws can be obtained in R itself – which is important for reproducibility.

### Author(s)

Dirk Eddelbuettel and Romain Francois



**Examples**

```
set.seed(42)
X <- RcppRNGsExample(10L)
set.seed(42)
Y <- data.frame(rnorm=rnorm(10),rt=rt(10,1),rpois=rpois(10,1))
all.equal(X,Y)
```

---

RcppStringVectorExample

*Example of using Rcpp StringVector (aka CharacterVector)*

---

**Description**

The StringVector (aka CharacterVector) class represents character vectors.

**Details**

The C++ code presented in the StringVectorExample.cpp file:

```
#include <Rcpp.h>
using namespace Rcpp ;

// [[Rcpp::export]]
List StringVectorExample(const StringVector & orig) {
  StringVector vec(orig.size());

  std::transform(orig.begin(), orig.end(), vec.begin(),
    make_string_transformer(tolower));

  return List::create(Named("result") = vec,
    Named("original") = orig);
}
```

**Author(s)**

Dominick Samperi wrote the initial versions of Rcpp (and RcppTemplate) during 2005 and 2006. Dirk Eddelbuettel made some additions, and became maintainer in 2008. Dirk Eddelbuettel and Romain Francois have been extending Rcpp since 2009.

**References**

*Writing R Extensions*, available at <https://www.r-project.org>.

**Examples**

```
RcppStringVectorExample(c("Tick", "Tack", "Tock"))
```

# Index

## \* **interface**

- RcppDataFrame, 3
- RcppDateExample, 4
- RcppListExample, 5
- RcppMatrixExample, 6
- RcppNumericVectorExample, 7
- RcppRNGsExample, 8
- RcppStringVectorExample, 9

## \* **package**

- RcppExamples-package, 2

## \* **programming**

- RcppDataFrame, 3
- RcppDateExample, 4
- RcppListExample, 5
- RcppMatrixExample, 6
- RcppNumericVectorExample, 7
- RcppRNGsExample, 8
- RcppStringVectorExample, 9

char2factor (factor2char), 2

factor2char, 2

RcppDataFrame, 3  
RcppDateExample, 4  
RcppExamples (RcppExamples-package), 2  
RcppExamples-package, 2  
RcppListExample, 5  
RcppMatrixExample, 6  
RcppNumericVectorExample, 7  
RcppRNGsExample, 8  
RcppStringVectorExample, 9