

# Rcpp

## SEAMLESS R AND C++ INTEGRATION

---

Matt P. Dziubinski

CppCon 2015

`matt@math.aau.dk // @matt_dz`

Department of Mathematical Sciences, Aalborg University  
CREATES (Center for Research in Econometric Analysis of Time Series)

# OUTLINE

R

Rcpp: R & C++

Setup

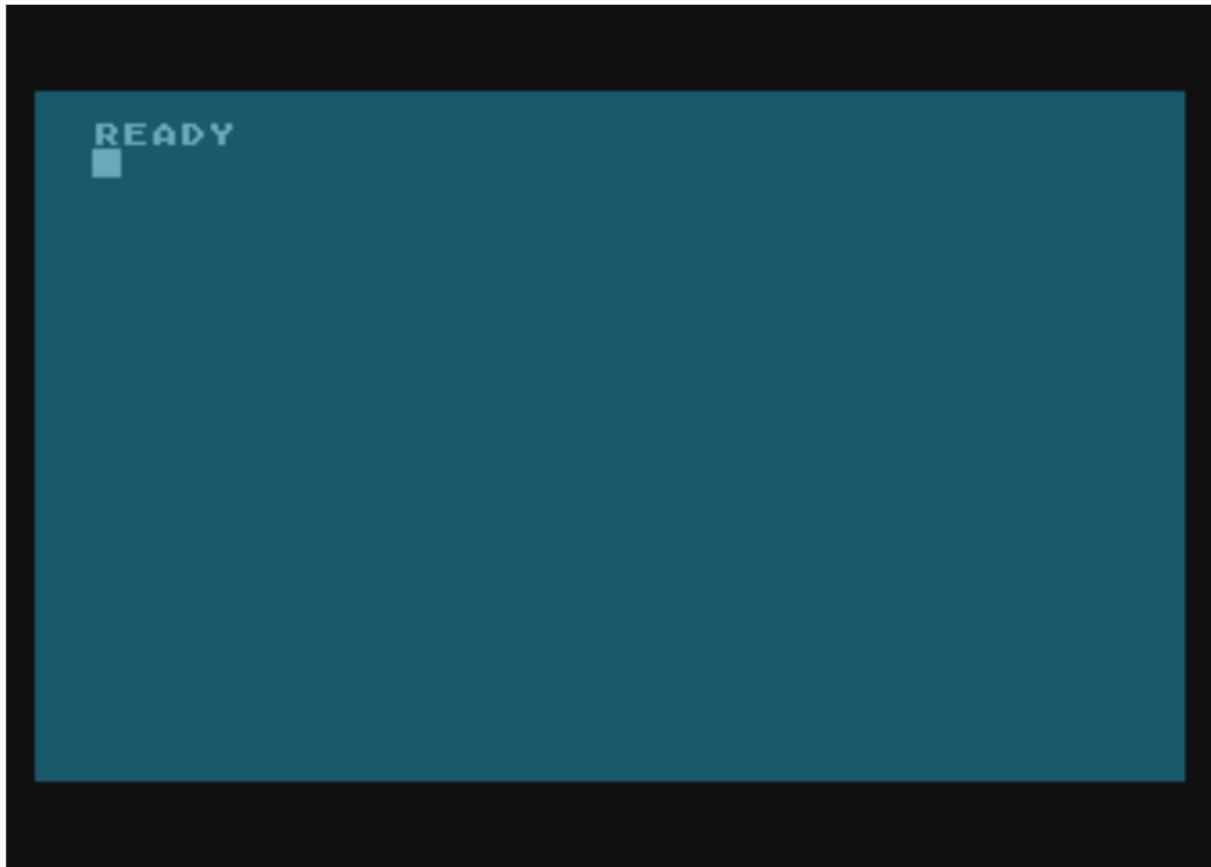
Data Structures

Resources

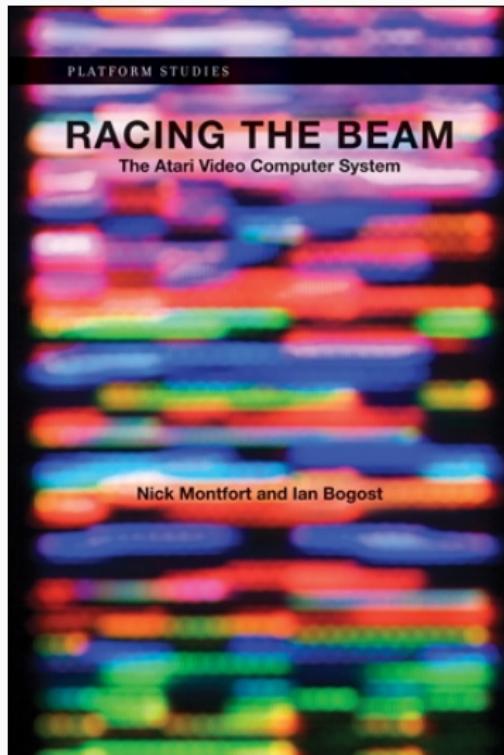
# INTRO



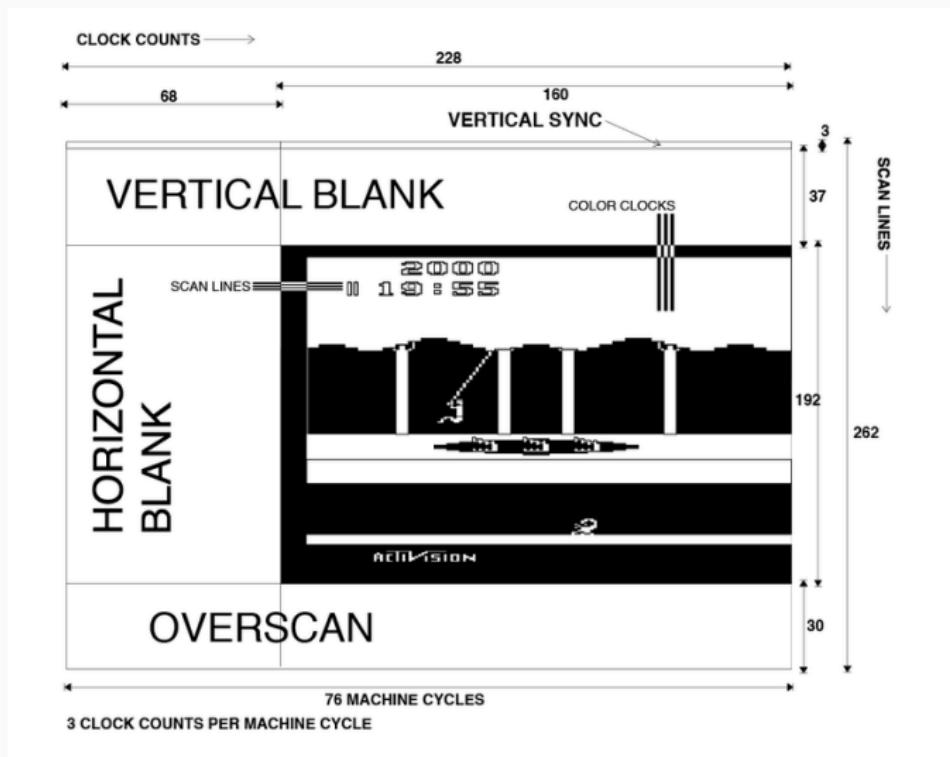
# INTRO



# INTRO



# INTRO



## INTRO

```
READY  
? PEEK(710)  
148
```

```
READY  
POKE 712, 148■
```

## INTRO

```
READY  
? PEEK(710)  
148
```

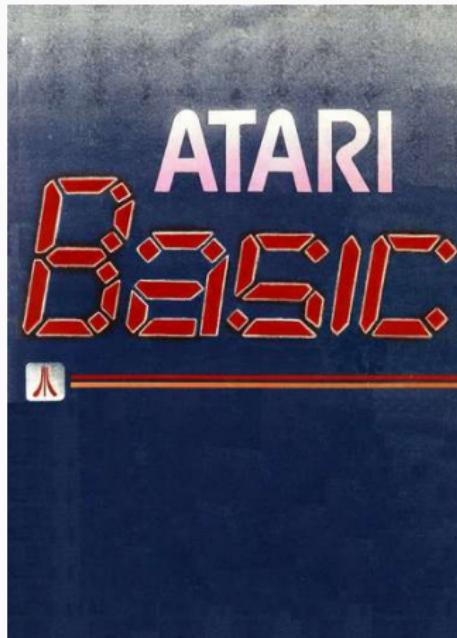
```
READY  
POKE 712, 148
```

```
READY  
■
```

## INTRO

```
READY  
POKE 712, PEEK(710)
```

```
READY  
■
```



## INTRO

```
5 FOR I = 1536 TO 1543
15 READ X
25 POKE I,X
35 NEXT I
45 DATA 104,173,198,2,141,200,2,96
```

## INTRO

```
LDA $710  
STA $712
```



## INTRO

```
PLA  
LDA $710  
STA $712  
RTS
```



## INTRO

```
PLA  
LDA $710  
STA $712  
RTS
```

```
104  
173 198 2  
141 200 2  
96
```



## INTRO

```
READY
5 FOR I = 1536 TO 1543
15 READ X
25 POKE I, X
35 NEXT I
45 DATA 104, 173, 198, 2, 141, 200, 2,
96
■
```

## INTRO

```
READY
5 FOR I = 1536 TO 1543
15 READ X
25 POKE I, X
35 NEXT I
45 DATA 104, 173, 198, 2, 141, 200, 2,
96
V = USR(1536)
READY
■
```

## INTRO: SPEEDUP

```
READY
POKE 540, 255:FOR I = 1 TO 10:POKE 712
, PEEK(710):NEXT I:TIME = PEEK(540):?
TIME:? 255 - TIME:? (255 - TIME)/60
250
```

5

0.0833333333

```
READY
```

```
POKE 540, 255:FOR I = 1 TO 10:U = USR(
1536):NEXT I:TIME = PEEK(540):? TIME:?
255 - TIME:? (255 - TIME)/60
```

252

3

0.05

```
READY
```

? 5 / 3

1.66666666

```
READY
```



## INTRO: TAKEAWAYS

- <http://c2.com/cgi/wiki?AlternateHardAndSoftLayers>
- <http://c2.com/cgi/wiki?ForeignFunctionInterface>

# INTRO: THE TIMES THEY ARE A-CHANGIN'



# INTRO: THE TIMES THEY ARE A-CHANGIN'?



R

---

<https://www.r-project.org/>



# WHAT IS R?

<https://www.r-project.org/about.html>



[Home]

**Download**

CRAN

**R Project**

About R

Contributors

What's New?

Mailing Lists

Bug Tracking

Conferences

Search

**R Foundation**

Foundation

Board

Members

Donors

## What is R?

### Introduction to R

R is a language and environment for statistical computing and graphics. It is a GNU project which is similar to the S language and environment which was developed at Bell Laboratories (formerly AT&T, now Lucent Technologies) by John Chambers and colleagues. R can be considered as a different implementation of S. There are some important differences, but much code written for S runs unaltered under R.

R provides a wide variety of statistical (linear and nonlinear modelling, classical statistical tests, time-series analysis, classification, clustering, ...) and graphical techniques, and is highly extensible. The S language is often the vehicle of choice for research in statistical methodology, and R provides an Open Source route to participation in that activity.

One of R's strengths is the ease with which well-designed publication-quality plots can be produced, including mathematical symbols and formulae where needed. Great care has been taken over the defaults for the minor design choices in graphics, but the user retains full control.

R is available as Free Software under the terms of the Free Software Foundation's GNU General Public License in source code form. It compiles and runs on a wide variety of UNIX platforms and similar systems (including FreeBSD and Linux), Windows and MacOS.

*"Currently, the CRAN package repository features 7176 available packages."*

- <https://cran.r-project.org/web/packages/>
- <https://cran.r-project.org/web/views/>

# CRAN Task Views

## CRAN Task Views

<a href="#">Bayesian</a>	Bayesian Inference
<a href="#">ChemPhys</a>	Chemometrics and Computational Physics
<a href="#">ClinicalTrials</a>	Clinical Trial Design, Monitoring, and Analysis
<a href="#">Cluster</a>	Cluster Analysis & Finite Mixture Models
<a href="#">DifferentialEquations</a>	Differential Equations
<a href="#">Distributions</a>	Probability Distributions
<a href="#">Econometrics</a>	Econometrics
<a href="#">Envirometrics</a>	Analysis of Ecological and Environmental Data
<a href="#">ExperimentalDesign</a>	Design of Experiments (DoE) & Analysis of Experimental Data
<a href="#">Finance</a>	Empirical Finance
<a href="#">Genetics</a>	Statistical Genetics
<a href="#">Graphics</a>	Graphic Displays & Dynamic Graphics & Graphic Devices & Visualization
<a href="#">HighPerformanceComputing</a>	High-Performance and Parallel Computing with R
<a href="#">MachineLearning</a>	Machine Learning & Statistical Learning
<a href="#">MedicalImaging</a>	Medical Image Analysis
<a href="#">MetaAnalysis</a>	Meta-Analysis
<a href="#">Multivariate</a>	Multivariate Statistics
<a href="#">NaturalLanguageProcessing</a>	Natural Language Processing
<a href="#">NumericalMathematics</a>	Numerical Mathematics
<a href="#">OfficialStatistics</a>	Official Statistics & Survey Methodology
<a href="#">Optimization</a>	Optimization and Mathematical Programming
<a href="#">Pharmacokinetics</a>	Analysis of Pharmacokinetic Data
<a href="#">Phylogenetics</a>	Phylogenetics, Especially Comparative Methods
<a href="#">Psychometrics</a>	Psychometric Models and Methods
<a href="#">ReproducibleResearch</a>	Reproducible Research
<a href="#">Robust</a>	Robust Statistical Methods
<a href="#">SocialSciences</a>	Statistics for the Social Sciences
<a href="#">Spatial</a>	Analysis of Spatial Data
<a href="#">SpatioTemporal</a>	Handling and Analyzing Spatio-Temporal Data
<a href="#">Survival</a>	Survival Analysis
<a href="#">TimeSeries</a>	Time Series Analysis
<a href="#">WebTechnologies</a>	Web Technologies and Services
<a href="#">gR</a>	gRaphical Models in R

To automatically install these views, the `ctv` package needs to be installed, e.g., via

```
install.packages("ctv")
library("ctv")
and then the views can be installed via install.views or update.views (which first assesses which of the packages are already installed and up-to-date), e.g.,
install.views("Econometrics")
or
update.views("Econometrics")
```

- <https://cran.r-project.org/web/views/MachineLearning.html>
  - <https://cran.r-project.org/web/packages/caret/>
    - <https://topepo.github.io/caret/modelList.html>
    - <https://topepo.github.io/caret/bytag.html>
    - <https://topepo.github.io/caret/training.html>
  - <https://cran.r-project.org/web/packages/elasticnet/>
  - <https://cran.r-project.org/web/packages/glmnet/>
    - Authors: Jerome Friedman, Trevor Hastie, Noah Simon, Rob Tibshirani
    - [https://cran.r-project.org/web/packages/glmnet/vignettes/glmnet\\_beta.html](https://cran.r-project.org/web/packages/glmnet/vignettes/glmnet_beta.html)

# INTRODUCTION TO STATISTICAL LEARNING (ISL)

<http://www.statlearning.com/>

## An Introduction to Statistical Learning with Applications in R

Gareth James, Daniela Witten, Trevor Hastie and Robert Tibshirani

[Home](#)

[About this Book](#)

[R Code for Labs](#)

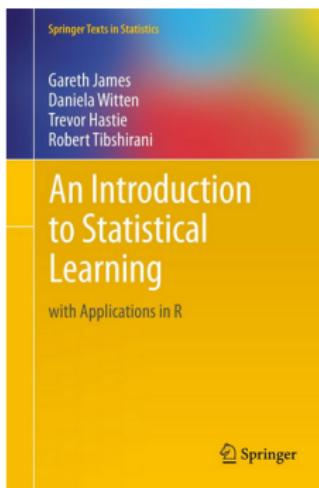
[Data Sets and Figures](#)

[ISLR Package](#)

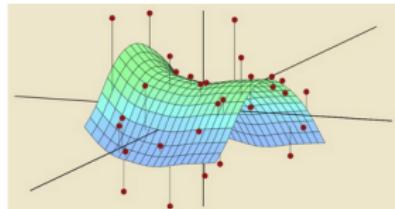
[Get the Book](#)

[Author Bios](#)

[Errata](#)



[Download the book PDF](#)  
(corrected 4th printing)



*Statistical Learning MOOC covering the entire ISL book offered by Trevor Hastie and Rob Tibshirani.*

# ELEMENTS OF STATISTICAL LEARNING (ESL)

<http://www-stat.stanford.edu/ElemStatLearn>

## THE ELEMENTS OF STATISTICAL LEARNING

Trevor Hastie, Robert Tibshirani, and Jerome Friedman

About this book

How to order

Table of contents

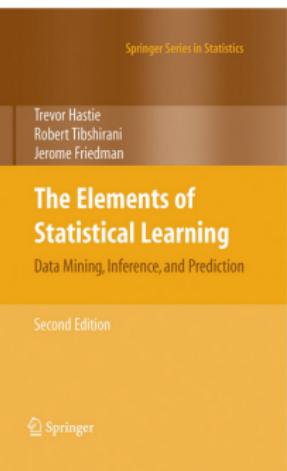
Data

Errata

R Functions

Complements

Short course:  
Statistical Learning  
and Data Mining

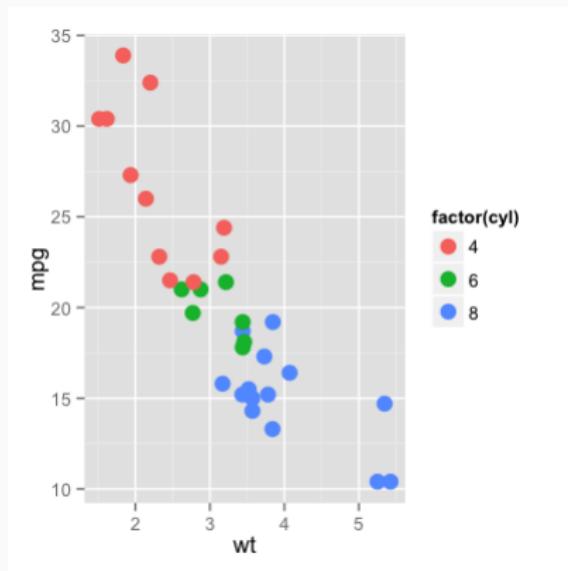


The Elements of Statistical Learning  
Data Mining, Inference, and Prediction  
Second Edition  
Springer

***The Elements of Statistical Learning:***  
**Data Mining, Inference, and Prediction.**  
**Second Edition**  
**February 2009**

[Trevor Hastie](#)  
[Robert Tibshirani](#)  
[Jerome Friedman](#)

<http://docs.ggplot2.org/current/>



[http://docs.ggplot2.org/current/aes\\_group\\_order.html](http://docs.ggplot2.org/current/aes_group_order.html)  
<https://stat.ethz.ch/R-manual/R-devel/library/datasets/html/mtcars.html>

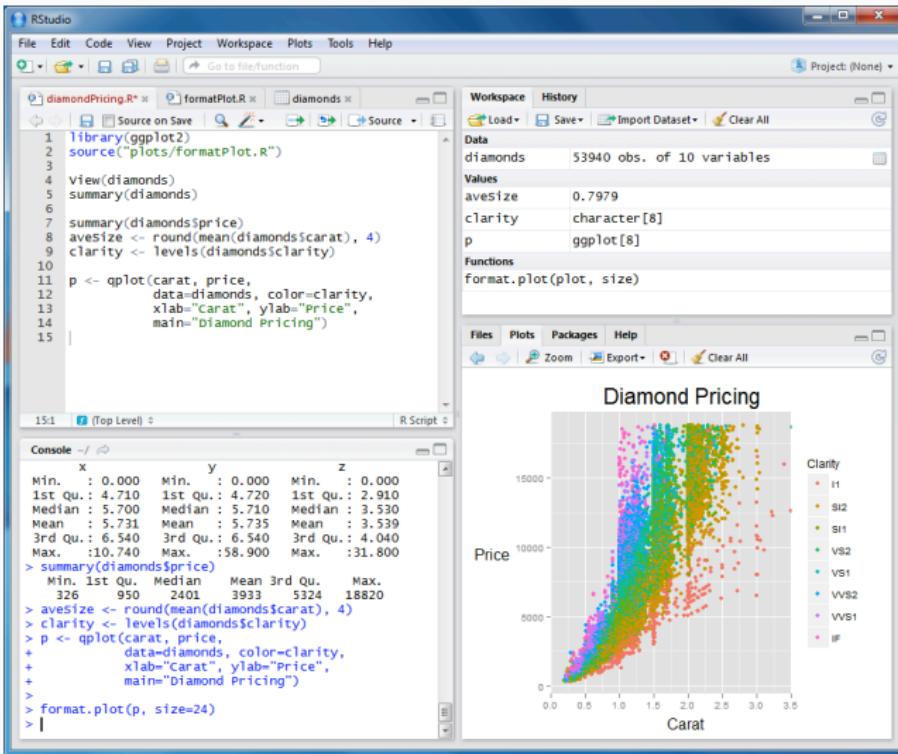
<https://www.rstudio.com/>

<https://github.com/rstudio/rstudio/>



# RStudio IDE

<https://www.rstudio.com/products/RStudio/>



```
install.packages("ggplot2")

library("ggplot2")

ggplot(diamonds, aes(x = carat, y = price, col = clarity)) + geom_point()
```

[https://ateucher.github.io/rcourse\\_site/03-plotting.html](https://ateucher.github.io/rcourse_site/03-plotting.html)

<http://www.ats.ucla.edu/stat/r/faq/packages.htm>

# R & RSTUDIO IDE: DEMO - PACKAGE NOT INSTALLED

```
Console c:/bin/prog/R/RStudio/bin/ ↵

R version 3.2.2 Patched (2015-09-13 r69384) -- "Fire Safety"
Copyright (C) 2015 The R Foundation for statistical computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> ggplot(diamonds, aes(x = carat, y = price, col = clarity)) + geom_point()
Error: could not find function "ggplot"
```

# R & RSTUDIO IDE: DEMO - INSTALLING PACKAGE

```
> install.packages("ggplot2")    --  
also installing the dependencies 'stringi', 'magrittr', 'colorspace', 'Rcpp', 'stringr', 'RColorBrewer', 'dichromat', 'munsell',  
'labeling', 'plyr', 'digest', 'gttable', 'reshape2', 'scales', 'proto'  
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.2/ggplot2_1.0.1.zip'  
Content type 'application/zip' length 2676922 bytes (2.6 MB)  
downloaded 2.6 MB  
  
package 'stringi' successfully unpacked and MD5 sums checked  
package 'magrittr' successfully unpacked and MD5 sums checked  
package 'colorspace' successfully unpacked and MD5 sums checked  
package 'Rcpp' successfully unpacked and MD5 sums checked  
package 'stringr' successfully unpacked and MD5 sums checked  
package 'RColorBrewer' successfully unpacked and MD5 sums checked  
package 'dichromat' successfully unpacked and MD5 sums checked  
package 'munsell' successfully unpacked and MD5 sums checked  
package 'labeling' successfully unpacked and MD5 sums checked  
package 'plyr' successfully unpacked and MD5 sums checked  
package 'digest' successfully unpacked and MD5 sums checked  
package 'gttable' successfully unpacked and MD5 sums checked  
package 'reshape2' successfully unpacked and MD5 sums checked  
package 'scales' successfully unpacked and MD5 sums checked  
package 'proto' successfully unpacked and MD5 sums checked  
package 'ggplot2' successfully unpacked and MD5 sums checked
```

# R & RStudio IDE: DEMO - LOADING/ATTACHING & USING PACKAGE

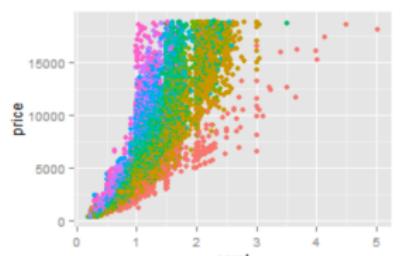
Screenshot of the RStudio IDE interface showing the loading and use of the ggplot2 package.

**Console:** Displays the R code used to create a scatter plot.

```
> library("ggplot2")
> ggplot(diamonds, aes(x = carat, y = price, col = clarity)) +
  geom_point()
>
>
```

**Environment:** Shows the Global Environment pane, which is currently empty.

**Plots:** Displays a scatter plot of price versus carat, colored by clarity. The plot includes a legend for clarity levels: I1, SI2, SI1, VS2, VS1, VVS2, VVS1, and IF.



The scatter plot shows a positive correlation between carat weight and price. The points are color-coded according to their clarity grade. The legend indicates the following mapping:

- I1 (Red)
- SI2 (Yellow)
- SI1 (Green)
- VS2 (Cyan)
- VS1 (Blue)
- VVS2 (Light Blue)
- VVS1 (Purple)
- IF (Pink)

http://shiny.rstudio.com/

The screenshot shows the RStudio interface with a ggvis application running. The top navigation bar includes tabs for Environment, History, Files, Plots, Packages, Help, and Viewer. The main workspace displays an R script named 'server.R' containing code to create a density plot of 'wt' from the 'mtcars' dataset. The plot is centered around 3.5 with a bandwidth adjustment of 1. The console output shows the application was run locally on port 4470.

```
library(ggvis)
function(input, output, session) {
  iadjust <- reactive(input$adjust)
  ikernel <- reactive(input$kernel)

  mtcars %>%
    ggvis(x = ~wt) %>%
    layer_densities(
      adjust = iadjust,
      kernel = ikernel) %>%
    bind_shiny("ggvis", "ggvis_ui")
}

<function>(input, output, session) <--> server.R

shiny::runApp(~Desktop')

Listening on http://127.0.0.1:4470
```

# R MARKDOWN

<http://rmarkdown.rstudio.com/>

Screenshot of RStudio showing an R Markdown document and its generated output.

The left pane shows the R Markdown file (`welcome.Rmd`):

```
1: ---  
2: title: "Welcome to R Markdown"  
3: author: "RStudio"  
4: date: "December 18, 2014"  
5: output: html_document  
6: ---  
7:  
8: This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.  
9:  
10: When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document.  
11:  
12: `{{r}}`  
13: coef(lm(dist ~ speed, data = cars))  
14:  
15:  
16: You can also embed plots, for example:  
1: (Top Level) R Markdown
```

The right pane shows the generated HTML output:

## Welcome to R Markdown

RStudio

December 18, 2014

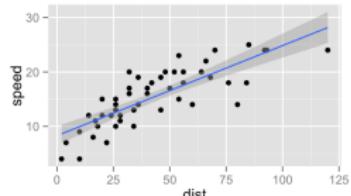
This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the Knit button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document.

```
coef(lm(dist ~ speed, data = cars))
```

```
## (Intercept)      speed  
## -17.579095   3.932409
```

You can also embed plots, for example:



37

# R MARKDOWN - PDF

<http://rmarkdown.rstudio.com/>

The screenshot shows the RStudio interface with two panes. The left pane displays the R Markdown source code for a document named 'welcome.Rmd'. The right pane shows the generated PDF output.

**R Markdown Source (welcome.Rmd):**

```
1: ---
2: title: "Welcome to R Markdown"
3: author: "RStudio"
4: date: "December 18, 2014"
5: output: pdf_document
6: ---
7
8 This is an R Markdown document.
9 simple formatting syntax for aut
10 PDF, and MS Word documents. For
11 on using R Markdown see <http://rstudio.com>.
12
13 When you click the **Knit** butt
14 will be generated that includes
15 as well as the output of any emb
16 chunks within the document.
17
18 ````{r}
19 coef(lm(dist ~ speed, data = car
20 )
21
22 You can also embed plots... for ex
23 (Top Level) :
```

**PDF Output:**

Welcome to R Markdown  
RStudio  
December 18, 2014

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the Knit button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document.

```
## (Intercept)    speed
## -37.579445   3.932469
```

You can also embed plots, for example:

A scatter plot showing the relationship between 'dist' (distance) and 'speed'. The x-axis ranges from 0 to 125, and the y-axis ranges from 0 to 30. A blue regression line is drawn through the data points, which show a positive correlation.

Note that the echo = FALSE parameter was added to the code chunk to prevent printing of the R code that generated the plot.

The bottom right corner of the PDF shows a small thumbnail of the same scatter plot.

# R MARKDOWN - SLIDES

<http://rmarkdown.rstudio.com/>

The screenshot shows the RStudio interface with two main panes. The left pane displays the R Markdown source code for a presentation titled "Welcome to R Markdown". The right pane shows the generated PDF output, which includes a title slide and a page with a scatter plot.

**R Markdown Source:**

```
1 ---  
2 title: "Welcome to R Markdown"  
3 author: "RStudio"  
4 date: "December 18, 2014"  
5 output: beamer_presentation  
6 ---  
7  
8 This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.  
9  
10 When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document.  
11  
12 ***  
13  
14 `{{r}}`  
15 coef(lm(dist ~ speed, data = cars))  
16 `{{r}}`  
5:9 (Top Level) ▾
```

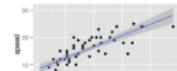
**PDF Output:**

- Title Slide:** Welcome to R Markdown, RStudio, December 18, 2014.
- Content Page:** This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see <http://rmarkdown.rstudio.com>.

When you click the Knit button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document.

```
resFile<-lm(dist ~ speed, data = cars)  
resFile  
## Call:  
## lm(formula = dist ~ speed, data = cars)  
## Coefficients:  
## (Intercept) speed  
## -17.579050 3.932409
```

You can also embed plots, for example:



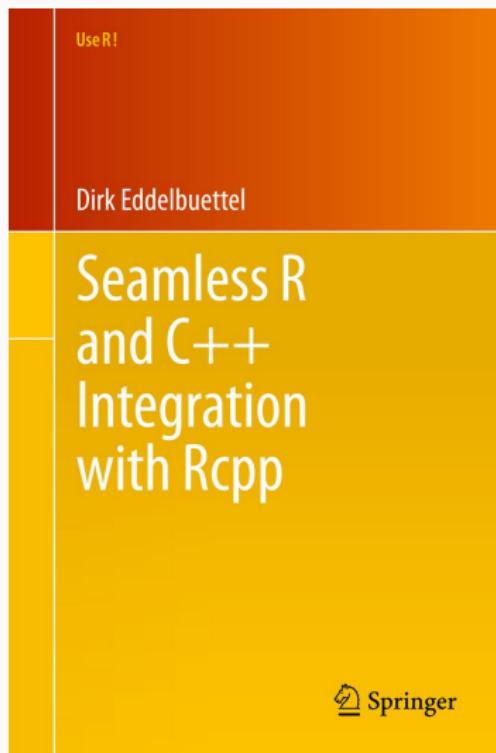
<http://pandoc.org/>

# R - Books

<https://www.r-project.org/doc/bib/R-books.html>

 Data Mining with R: The Art of Modeling Data for Knowledge Discovery (2nd Ed.) [R] Aug 4, 2011 <b>\$39.09 - \$62.34</b> ✓Phone Pearson, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 Statistical Methods for Data Science with R [R] Nov 10, 2010 <b>\$29.44 - \$55.57</b> ✓Phone Pearson, Kindle Edition Only 1 left in stock - order soon. ★★★★★ 1 □ + 1	 Introductory Time Series with R [R] Jun 1, 2009 <b>\$43.18</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 A Beginner's Guide to R (2nd Ed.) [R] Jul 1, 2009 <b>\$24.36 - \$25.71</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 An Introduction to Applied Multivariate Analysis with R [R] Mar 22, 2011 <b>\$40.01 - \$44.31</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Biostatistics with R: An Introduction to Biostatistical Methods Through Biological Data (2nd Ed.) [R] Oct 11, 2010 <b>\$36.51 - \$44.30</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Applied Spatial Data Analysis with R [R] Jan 31, 2010 <b>\$26.41 - \$57.56</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 R for Marketing Research and Analytics [R] Mar 15, 2010 <b>\$11.74</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Data Manipulation with R [R] Mar 15, 2008 <b>\$9.99 - \$99.94</b> ✓Phone Pearson, Kindle Edition - New Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 Introducing Monte Carlo Methods with R [R] Dec 15, 2008 <b>\$16.58 - \$41.43</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1
 ggplot2: Elegant Graphics for Data Analysis (2nd Ed.) [R] Feb 22, 2010 <b>\$27.54 - \$62.34</b> ✓Phone Hadley Wickham ★★★★★ 1 □ + 1	 Empirical Ecology with R [R] Jun 1, 2009 <b>\$42.49 - \$76.79</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Modern Optimization with R [R] Mar 21, 2011 <b>\$40.99</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Analysis of Phylogenetics and Evolution with R [R] Mar 21, 2011 <b>\$39.32 - \$75.02</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Business Analytics for Managers [R] Aug 22, 2010 <b>\$25.10 - \$43.23</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Bayesian Networks in R: With Applications in Systems Biology [R] Apr 27, 2010 <b>\$37.43 - \$49.39</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Generalized Linear Models with R [R] Sep 1, 2010 <b>\$42.10 - \$89.00</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Solving Differential Equations in R [R] Sep 2, 2010 <b>\$24.36 - \$41.37</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 R for Big Data [R] Nov 22, 2011 <b>\$25.33 - \$46.00</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 Six Sigma with R [R] Oct 1, 2010 <b>\$44.95 - \$69.95</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1
 Multistate Analysis of Life Histories with R [R] Oct 14, 2014 <b>\$43.95</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Chemometrics with R: Multivariate Data Analysis in the Natural Sciences and Life Sciences [R] May 19, 2011 <b>\$52.14</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Nonlinear Regression with R [R] Nov 21, 2008 <b>\$69.82</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Competing Risks and Multistate Models with R [R] Mar 13, 2010 <b>\$29.14 - \$61.49</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Statistical Analysis of Network Data with R [R] Mar 22, 2014 <b>\$28.10 - \$66.29</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Analyzing Compositional Data with R [R] Jan 30, 2010 <b>\$28.10 - \$56.99</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 A Primer of Ecology with R [R] Oct 6, 2010 <b>\$39.45 - \$41.45</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Applied Econometrics with R [R] Aug 28, 2008 <b>\$15.19 - \$41.43</b> ✓Phone Springer, Kindle Edition ★★★★★ 1 □ + 1	 Bayesian Methods for Data Analysis (3rd Ed.) [R] Aug 28, 2010 <b>\$69.09 - \$92.34</b> ✓Phone Braza, Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 Generalized Linear Models with R [R] Oct 1, 2010 <b>\$40.95 - \$69.95</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1
 Data Mining with R and R: The Art of Modeling Data for Knowledge Discovery (2nd Ed.) [R] Aug 4, 2011 <b>\$39.09 - \$62.34</b> ✓Phone Pearson, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 Wavelet Methods in Statistics with Applications to Finance [R] Aug 18, 2008 <b>\$109.44 - \$142.22</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 Bayesian Computation with R [R] Jun 1, 2009 <b>\$109.44 - \$142.22</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 Primer to Analysis of Economic Data Using R (2nd Ed.) [R] Mar 22, 2010 <b>\$40.66</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 Statistical R and C++ Integration with Rcpp [R] Oct 1, 2010 <b>\$26.10 - \$59.25</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 Lattice: Multivariate Data Visualization with R [R] Aug 28, 2009 <b>\$43.95 - \$66.52</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 Applied Statistical Genetics with R: For Population-based Association Studies (2nd Ed.) [R] Apr 17, 2008 <b>\$109.44 - \$142.22</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 Functional and Phylogenetic Ecology in R [R] Aug 28, 2008 <b>\$109.44 - \$142.22</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 R Through Excel: A Spreadsheet Interface for Statistics, Data Analysis, and Graphics using R [R] Aug 28, 2008 <b>\$109.44 - \$142.22</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1	 Chemotax [R] Oct 1, 2010 <b>\$40.95 - \$69.95</b> ✓Phone Springer, Kindle Edition Only 12 left in stock - order soon. ★★★★★ 1 □ + 1

<http://rcpp.org/book/>



## R - LOOPS VS. VECTORIZED CODE

```
1 sum_loop = function(numbers)
2 {
3     result = 0
4     for (number in numbers)
5         result = result + number
6     result
7 }
8
9 some_numbers = as.numeric(1:1e7)
10 str(some_numbers)
11
12 system.time({result = sum_loop(some_numbers)}); show(result)
13
14 system.time({result = sum(some_numbers)}); show(result)
15
```

15:1 (Top Level) ▾

Console G:/docs/Projects/AD/ADmetrics-docs/R/ ↵

```
> sum_loop = function(numbers)
+ {
+     result = 0
+     for (number in numbers)
+         result = result + number
+     result
+ }
>
> some_numbers = as.numeric(1:1e7)
> str(some_numbers)
num [1:10000000] 1 2 3 4 5 6 7 8 9 10 ...
>
> system.time({result = sum_loop(some_numbers)}); show(result)
  user  system elapsed
  3.93    0.00   3.94
[1] 5e+13
> system.time({result = sum(some_numbers)}); show(result)
  user  system elapsed
  0.02    0.00   0.01
[1] 5e+13
```

# R - LOOPS & BYTE CODE COMPILER VS. VECTORIZED CODE

```
16 sum_loop_bc = compiler::cmpfun(sum_loop)
17 system.time({result = sum_loop_bc(some_numbers)}); show(result)
18
19 compiler::disassemble(sum_loop_bc)
20
21
22
23
24
25
26
27
28
29
30
31:1 (Top Level) ⇡
```

Console G:/docs/Projects/AD/ADmetrics-docs/R/ ↵

```
> system.time({result = sum_loop(some_numbers)}); show(result)
  user  system elapsed
 3.93    0.00   3.94
[1] 5e+13
> system.time({result = sum(some_numbers)}); show(result)
  user  system elapsed
 0.02    0.00   0.01
[1] 5e+13
> sum_loop_bc = compiler::cmpfun(sum_loop)
> system.time({result = sum_loop_bc(some_numbers)}); show(result)
  user  system elapsed
 0.43    0.00   0.44
[1] 5e+13
> compiler::disassemble(sum_loop_bc)
list(.Code, list(8L, LDCONST.OP, 1L, SETVAR.OP, 2L, POP.OP, GETVAR.OP,
  3L, STARTFOR.OP, 5L, 4L, 21L, GETVAR.OP, 2L, GETVAR.OP, 4L,
  ADD.OP, 6L, SETVAR.OP, 2L, POP.OP, STEPFOR.OP, 12L, ENDFOR.OP,
  POP.OP, GETVAR.OP, 2L, RETURN.OP), list({
    result = 0
    for (number in numbers) result = result + number
    result
  }, 0, result, numbers, number, for (number in numbers) result = result +
  number, result + number))
```

Using R for HPC: [http://www.nimbios.org/tutorials/TT\\_RforHPC](http://www.nimbios.org/tutorials/TT_RforHPC)

## C++ - LOOPS, RCPP, RESULT

```
25 sum_cpp_txt = '
26 double sum_cpp(double from, double to)
27 {
28     asm("#begin sum_cpp");
29     double result = 0.;
30     for (double number = from; number < to; ++number)
31         result += number;
32     asm("#end sum_cpp");
33     return result;
34 }'
35
36 sum_cpp = Rcpp::cppFunction(sum_cpp_txt)
37 system.time({result = sum_cpp(1., 1.e7)}); show(result)
38
```

37:56 | (Top Level) ▾

```
Console g:/ ↵
> sum_cpp_txt = '
+ double sum_cpp(double from, double to)
+ {
+     asm("#begin sum_cpp");
+     double result = 0.;
+     for (double number = from; number < to; ++number)
+         result += number;
+     asm("#end sum_cpp");
+     return result;
+ }'
>
> sum_cpp = Rcpp::cppFunction(sum_cpp_txt)
> system.time({result = sum_cpp(1., 1.e7)}); show(result)
  user  system elapsed
 0.04    0.00    0.04
```

# C++ - LOOPS, X86

<https://gcc.godbolt.org/> <https://goo.gl/DAKTUA>  
<https://github.com/mattgodbolt/gcc-explorer>

Interactive compiler - C++   Tip 2 tips   Flatt 4 Share About

Source: Examples Name: max array

Load Save Save as... Permalink

Compiler: x86 gcc 5.2.0

Compiler options: -Ofast -march=native -mtune=native -mfpmath=sse

Filter: Unused labels Directives Comment-only lines Intel syntax Colourise

Fork me on GitHub

Code editor

```
1 double sum_cpp(double from, double to)
2 {
3     asm("#begin sum_cpp");
4     double result = 0;
5     for (double number = from; number < to; ++number)
6         result += number;
7     asm("#end sum_cpp");
8     return result;
9 }
```

Assembly output

```
1 sum_cpp(double, double):
2     vmovapd xmm2, xmm0
3     #APP
4     # 3 "/tmp/gcc-explorer-compiler115817-68-lu1xvvw/example.cpp" 1
5     #begin sum_cpp
6     # 0 """
7     #NO APP
8     vcomisd xmm2, xmm1
9     vxorpd xmm0, xmm0, xmm0
10    jnb .L2
11    vmovsd xmm3, QWORD PTR .LC1[rip]
12    .L3:
13    vaddsd xmm0, xmm0, xmm2
14    vaddsd xmm2, xmm2, xmm3
15    vcomisd xmm2, xmm1
16    jb .L3
17    .L2:
18    #APP
19    # 7 "/tmp/gcc-explorer-compiler115817-68-lu1xvvw/example.cpp" 1
20    #end sum_cpp
21    # 0 """
22    #NO APP
23    ret
```

Compiler output — x86 gcc 5.2.0 (g++ (GCC-explorer-build) 5.2.0)

Compiled ok

# C++ - LOOPS, RCPP, X86

```
3709 # 9 "file1294d1f4299.cpp" 1
3710 →#begin sum_cpp
3711 # 0 "" 2
3712 .LVL362:
3713 /NO_APP
3714 .LBB1565:
3715 →.loc 13 11 0
3716 →vcomisd→xmm1, xmm2
3717 →vxorpd→xmm0, xmm0, xmm0
3718 .LVL363:
3719 →jbe→.L344
3720 →vmovsd→xmm3, QWORD PTR .LC17[rip]
3721 .LVL364:
3722 →.p2align 4,,10
3723 .L346:
3724 →.loc 13 12 0 discriminator 3
3725 →vaddsd→xmm0, xmm0, xmm2
3726 .LVL365:
3727 →.loc 13 11 0 discriminator 3
3728 →vaddsd→xmm2, xmm2, xmm3
3729 .LVL366:
3730 →vcomisd→xmm1, xmm2
3731 →ja→.L346
3732 .LVL367:
3733 .L344:
3734 .LBE1565:
3735 →.loc 13 13 0
3736 /APP
3737 # 13 "file1294d1f4299.cpp" 1
3738 →#end sum_cpp
```

# C++ - LOOPS, RCPP, RSTUDIO

```
26 sum_cpp_txt = '  
27 double sum_cpp(double from, double to)  
28 {  
29     asm("#begin sum_cpp");  
30     double result = 0.;  
31     for (double number = from; number < to; ++number)  
32         result += number;  
33     asm("#end sum_cpp");  
34     return result;  
35 }'  
36  
37 sum_cpp = Rcpp::cppFunction(sum_cpp_txt, verbose = TRUE, rebuild = TRUE)  
38
```

35:3 (Top Level)

R Script

Console G:/docs/Projects/AD/ADmetrics-docs/R/

Building shared library

DIR: C:/Users/MATT~1.R2D/AppData/Local/Temp/1/RtmpINKEYm/sourcecpp\_1294563c3d4b

```
c:/bin/sci/R/R-DEVE~2/bin/x64/R CMD SHLIB -o "sourceCpp_3136.dll" --preclean "" "file1294d1f4299.cpp"  
g++ -m64 -I"c:/bin/sci/R/R-DEVE~2/include" -DNDEBUG -isystem"c:/CPP/Boost/boost_1_58_0" -isystem"c:/C  
PP/POCO/poco-current/include" -I"c:/bin/sci/R/R-devel-current/library/Rcpp/include" -I"C:/Users/matt.  
R2D2/AppData/Local/Temp/1/RtmpINKEYm" -I"d:/RCompile/r-compiling/local/local320/include" -O2 -Wall  
-mtune=native -fopenmp -D_GLIBCXX_USE_INT128 -fasm=intel -c
```

# R - LOOPS, RANDOM NORMAL NUMBERS

```
69 set.seed(42)
70 some_numbers = rnorm(1e7)
71 str(some_numbers)
72
73 system.time({result = sum_loop(some_numbers)}); show(result)
74
75 sum_loop_bc = compiler::cmpfun(sum_loop)
76 system.time({result = sum_loop_bc(some_numbers)}); show(result)
77
78 system.time({result = sum(some_numbers)}); show(result)
79
80
81
```

69:1 (Top Level)

Console

```
> set.seed(42)
> some_numbers = rnorm(1e7)
> str(some_numbers)
num [1:10000000] 1.371 -0.565 0.363 0.633 0.404 ...
>
> system.time({result = sum_loop(some_numbers)}); show(result)
  user  system elapsed
  3.85    0.02   3.88
[1] 4752.874
>
> sum_loop_bc = compiler::cmpfun(sum_loop)
> system.time({result = sum_loop_bc(some_numbers)}); show(result)
  user  system elapsed
  0.39    0.00   0.39
[1] 4752.874
>
> system.time({result = sum(some_numbers)}); show(result)
  user  system elapsed
  0.02    0.00   0.02
[1] 4752.874
```

# R - LOOPS, RANDOM NORMAL NUMBERS

```
86 sum_Rcpp_txt = '
87 double sum_Rcpp(Rcpp::NumericVector numbers)
88 {
89     asm("#begin sum_Rcpp");
90     double result = 0.;
91     auto size = numbers.size();
92     for (decltype(size) i = 0; i < size; ++i)
93         result += numbers[i];
94     asm("#end sum_Rcpp");
95     return result;
96 }'
97
98 sum_Rcpp = Rcpp::cppFunction(sum_Rcpp_txt, plugins = "cpp11")
99
100 str(some_numbers)
101
102 system.time({result = sum_Rcpp(some_numbers)}); show(result)
103
104:1 (Top Level) #
```

```
Console g:/ ↵
> sum_Rcpp_txt = '
+ double sum_Rcpp(Rcpp::NumericVector numbers)
+ {
+     asm("#begin sum_Rcpp");
+     double result = 0.;
+     auto size = numbers.size();
+     for (decltype(size) i = 0; i < size; ++i)
+         result += numbers[i];
+     asm("#end sum_Rcpp");
+     return result;
+ }'
>
> sum_Rcpp = Rcpp::cppFunction(sum_Rcpp_txt, plugins = "cpp11")
>
> str(some_numbers)
num [1:10000000] 1.371 -0.565 0.363 0.633 0.404 ...
>
> system.time({result = sum_Rcpp(some_numbers)}); show(result)
  user  system elapsed
  0.01    0.00    0.01
[1] 4752.874
```

# C++ - LOOPS, RCPP, OPENMP - RESULT

```
117 set.seed(42)
118
119 some_numbers = rnorm(1e8)
120
121 str(some_numbers)
122 system.time({result = sum(some_numbers)}); show(result)
123 system.time({result = sum_Rcpp(some_numbers)}); show(result)
124 system.time({result = sum_Rcpp_omp(some_numbers)}); show(result)
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
449
450
```

Console Source Cpp ×

g/ ↻

```
> set.seed(42)
>
> some_numbers = rnorm(1e8)
>
> str(some_numbers)
num [1:100000000] 1.371 -0.565 0.363 0.633 0.404 ...
> system.time({result = sum(some_numbers)}); show(result)
  user  system elapsed
  0.15    0.00   0.16
[1] 4143.983
> system.time({result = sum_Rcpp(some_numbers)}); show(result)
  user  system elapsed
  0.06    0.00   0.07
[1] 4143.983
> system.time({result = sum_Rcpp_omp(some_numbers)}); show(result)
  user  system elapsed
  0.38    0.00   0.03
[1] 4143.983
```

# C++ - LOOPS, RCPP, OPENMP - CODE

```
1 // [[Rcpp::plugins(cpp11)]]
2 #include <cstddef>
3 #include <Rcpp.h>
4
5 // [[Rcpp::export]]
6 double sum_Rcpp(Rcpp::NumericVector numbers)
7 {
8     double result = 0.;
9     auto size = numbers.size();
10    for (decltype(size) i = 0; i < size; ++i)
11        result += numbers[i];
12    return result;
13 }
14
15 // [[Rcpp::export]]
16 double sum_Rcpp_omp(Rcpp::NumericVector numbers)
17 {
18     double result = 0.;
19     auto size = numbers.size();
20     #pragma omp parallel for reduction(+: result)
21     for (decltype(size) i = 0; i < size; ++i)
22         result += numbers[i];
23     return result;
24 }
```

25:1 (Top Level) ↴

Console Source Cpp ✎

```
g/ ↤
> Rcpp::sourceCpp('summation.cpp')
> str(sum_Rcpp)
function (numbers)
> str(sum_Rcpp_omp)
function (numbers)
>
> sum_Rcpp
function (numbers)
.Primitive(".Call")(<pointer: 0x000000006b182e00>, numbers)
> sum_Rcpp_omp
function (numbers)
.Primitive(".Call")(<pointer: 0x000000006b1826c0>, numbers)
```

Ulrich Drepper

# Utilizing the other 80% of your system's performance: Starting with Vectorization

02-26-2015 @ 3:45 - 4:45

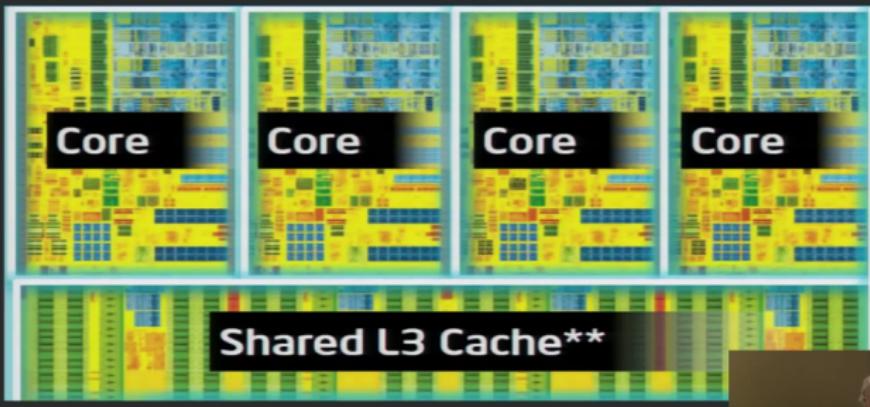
**Ulrich Drepper** — Ulrich worked on free software for more than 20 years and contributed to many different projects, mostly in the system infrastructure area. He currently works at Goldman Sachs in the area of data science.

Vectorization, as opposed to parallelization, is less utilized as a means of exploiting the full capabilities of a processor. This is a problem since even today this means only  $\frac{1}{4}$  to  $\frac{1}{2}$  of the performance of the CPU is used. This is only getting worse in future, especially as accelerators are becoming more prevalent.

After an introduction to the basics and history of vectorization the talk will introduce various techniques available for vectorization of compiled code. This talk will focus on gcc and for some details on Linux but the knowledge should be transferable if the features are fully implemented elsewhere.

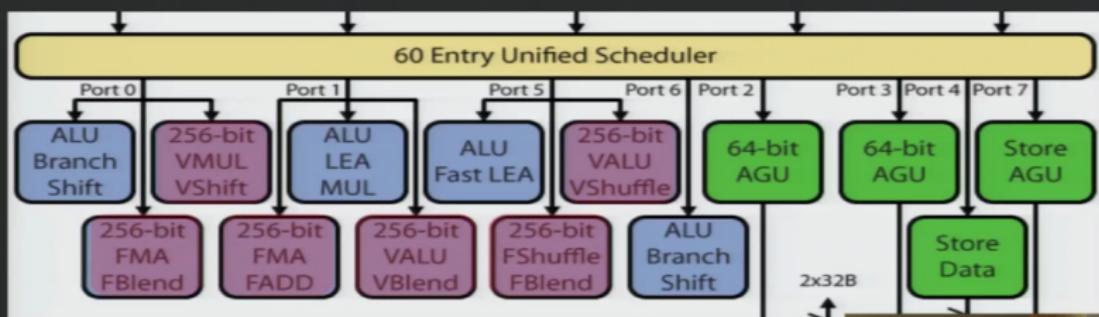
<http://applicative.acm.org/speaker-UlrichDrepper.html>

## Parallelism: Sure, it's covered!



# UTILIZING THE OTHER 80%... - VECTORIZATION

## EX in details



Similarly

## Vector Register Use



Integer: 8-, 16-, 32-, and 64-bit: **97%, 94%, 88%, and 75% unused**

FP: 16-, 32-, and 64-bit: **94%, 88%, and 75% unus**



## Rcpp: R & C++

---

<http://dirk.eddelbuettel.com/code/rcpp.html>

Dirk Eddelbuettel

Blog

Code

Publications

Talks

Other ▾

## History

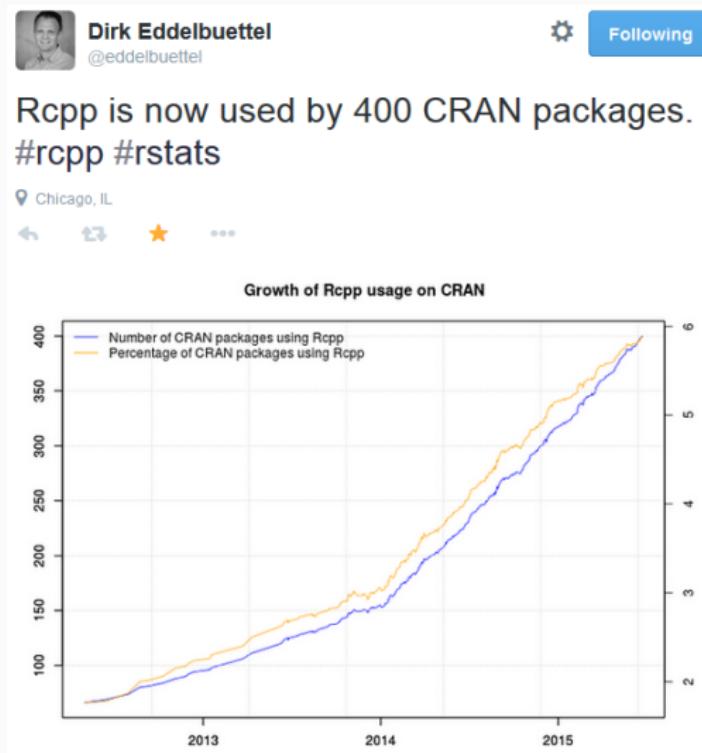
Rcpp was initially written by Dominick Samperi to ease contributions to the [RQuantLib](#) project, and then released as a project in its own right. During 2006, Dominick made several releases under the RCpp name (versions 1.0 to 1.4) before he changed the name to RCppTemplate and made more releases (1.5 to 5.2). His project saw no public releases for the thirty-five months period from November 2006 to November 2009.

As a user of Rcpp, I (Dirk) chose to adopt Rcpp during 2008, made a first release 0.6.0 in November 2008 and have made a number of new releases since -- see the [ChangeLog](#) for details. Rcpp is open for contributions and patches some of which have already been integrated.

[Romain Francois](#) joined the effort just before the 0.7.0 release and brought along a lot of energy and new ideas. We now have a [mailing list](#) for discussions around Rcpp. If you have ideas or suggested changes, send an email there.

# Rcpp TIMELINE

<https://twitter.com/eddelbuettel/status/613235012939464704>



[http://dirk.eddelbuettel.com/blog/2015/09/10/  
#rcpp\\_0.12.1](http://dirk.eddelbuettel.com/blog/2015/09/10/#rcpp_0.12.1)

Dirk Eddelbuettel      Blog    Code    Publications    Talks      Other ▾

Thu, 10 Sep 2015

## Rcpp 0.12.1: First boat load of fixes!

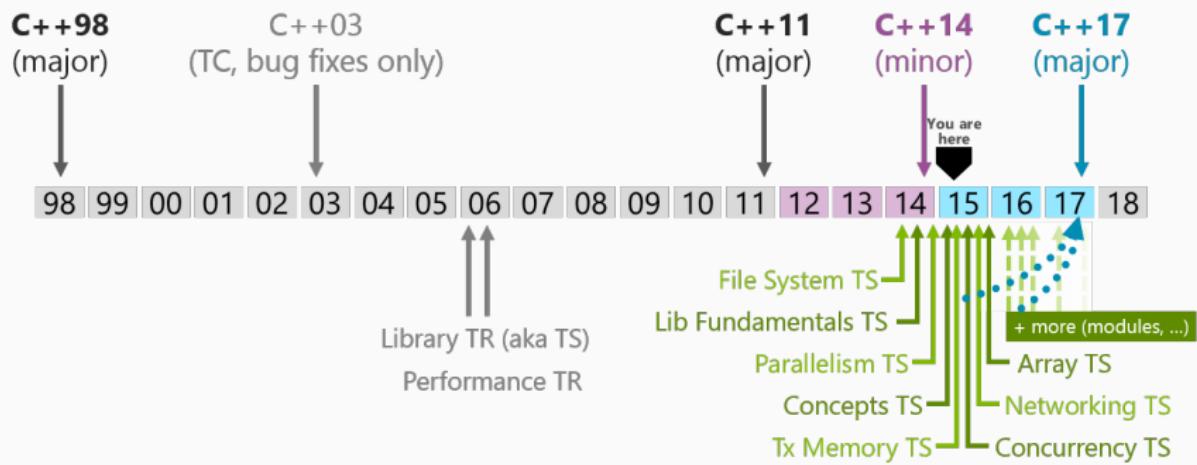
The first update in the 0.12.\* series of [Rcpp](#) is now on the [CRAN](#) network for [GNU R](#) this morning, and I will push a [Debian](#) package. This follows the [0.12.0 release](#) from late July which started to some serious new features.

[Rcpp](#) has become *the* most popular way of enhancing [GNU R](#) with C++ code. [As of today, 461 packages on CRAN depend on Rcpp for making analyses go faster and further.](#)

This release once again features contributions from a number of new contributors, with [Florian Plaza Oñate](#) and [Dan Dillon](#) joining the fray with several pull requests each. [Kurt Hornik](#) also helped with a fix for a bad interaction between very recent R (3.2.2) changes and Rcpp; another related one was also addressed. A big *Thanks!* to everybody helping with code, bug reports or documentation, See below for a detailed list of changes extracted from the [NEWS](#) file.

# C++ TIMELINE

<https://isocpp.org/std/status>



*"C++11 feels like a new language: The pieces just fit together better than they used to and I find a higher-level style of programming more natural than before and as efficient as ever."* — Bjarne Stroustrup.

## BEFORE C++11

```
#include <iostream>
#include <vector>

int main()
{
    std::vector<int> v(5);

    int element = 0;
    for (std::vector<int>::size_type i = 0; i < v.size(); ++i)
        v[i] = element++;

    int sum = 0;
    for (std::vector<int>::size_type i = 0; i < v.size(); ++i)
        sum += v[i];
    std::cout << "sum = " << sum;
}
```

- Q.: Is it **immediately clear** what this code does?

```
#include <iostream>
#include <vector>

int main()
{
    const std::vector<int> v {0, 1, 2, 3, 4};
    auto sum = 0;
    for (auto element : v) sum += element;
    std::cout << "sum = " << sum;
}
```

- How about now?
- (*Not Your Father's*) C++ — Herb Sutter
  - <https://channel9.msdn.com/Events/Lang-NEXT/Lang-NEXT-2012/-Not-Your-Father-s-C>

## BEFORE RCPP

```
#include <R.h>
#include <Rinternals.h>

// not quite right
int fibonacci_c_impl(int n)
{
    if (n < 2) return n;
    return fibonacci_c_impl(n - 1) + fibonacci_c_impl(n - 2);
}

SEXP fibonacci_c(SEXP n)
{
    SEXP result = PROTECT(allocaVector(INTSXP, 1));
    INTEGER(result)[0] = fibonacci_c_impl(asInteger(n));
    UNPROTECT(1);
    return result;
}

fibonacci = function(n) .Call("fibonacci_c", n)
```

```
// still not quite right
// [[Rcpp::export]]
int fibonacci(int n)
{
    if (n < 2) return n;
    return fibonacci(n - 1) + fibonacci(n - 2);
}
```

- Function **fibonacci** available in R automatically.
- 400 CRAN packages *may be onto something ;-)*

# SETUP

---

## SIMPLE EXAMPLE #0

The screenshot shows the RStudio interface. The top panel is a code editor for a file named `fibonacci_example.cpp`. It contains the following C++ code:

```
// [[Rcpp::export]]
int fibonacci(int n)
{
    if (n < 2) return n;
    return fibonacci(n - 1) + fibonacci(n - 2);
}
```

The bottom panel is a terminal window titled "Console" showing the output of running R. The output includes the R version information, the standard copyright notice, and the standard R welcome message.

```
R version 3.2.2 Patched (2015-09-13 r69384) -- "Fire Safety"
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> |
```

## SIMPLE EXAMPLE #0

The screenshot shows the RStudio interface. The top panel is a code editor with the file 'fibonacci\_example.cpp' open. The code defines a function 'fibonacci' that uses recursion to calculate the nth Fibonacci number. The bottom panel is a terminal window (Console) running on a Windows 64-bit system. It displays the standard R startup message, including the warranty notice, contributor information, and help instructions. At the bottom of the terminal, there is an error message indicating that the 'Rcpp' package is not found.

```
fibonacci_example.cpp *
Source on Save | ? | Source
1 // [[Rcpp::export]]
2 int fibonacci(int n)
3 {
4     if (n < 2) return n;
5     return fibonacci(n - 1) + fibonacci(n - 2);
6 }
7

4:23 fibonacc(int n) ▾ C/C++ ▾

Console c:/bin/prog/R/RStudio/bin/
Platform: x86_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> Rcpp::sourceCpp('G:/notes/confs/cppCon/2015/cppCon2015slides/Rcpp/src/fibonacci.cpp')
Error in loadNamespace(name) : there is no package called 'Rcpp'
>
```

## SIMPLE EXAMPLE #0

The screenshot shows the RStudio interface. The top panel displays a C++ code editor with the file 'fibonacci\_example.cpp'. The code defines a function 'fibonacci' that uses the Rcpp::export attribute and implements the recursive formula for the Fibonacci sequence. The bottom panel shows an R console window with the following output:

```
fibonacci_example.cpp * | Source on Save | ? | Source | C/C++ | 4:23 | fibonacci(int n) | C/C++ |
```

```
Console c:/bin/prog/R/RStudio/bin/ ↵
> install.packages("Rcpp")
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.2/Rcpp_0.12.1.zip'
Content type 'application/zip' length 3189676 bytes (3.0 MB)
downloaded 3.0 MB

package 'Rcpp' successfully unpacked and MD5 sums checked

The downloaded binary packages are in
      c:\Users\Matt\AppData\Local\Temp\1\Rtmpcz13gk\downloaded_packages
>
>
>
>
>
> |
```

# SIMPLE EXAMPLE #1

The screenshot shows the RStudio interface. In the top-left corner, there's a code editor window titled "fibonacci\_example.cpp". The code contains a simple recursive function for calculating Fibonacci numbers:

```
// [[Rcpp::export]]
int fibonacci(int n)
{
    if (n < 2) return n;
    return fibonacci(n - 1) + fibonacci(n - 2);
}
```

Below the code editor is a terminal console window titled "Console c:/bin/prog/R/RStudio". It displays the following R session:

```
Type 'q()' to quit R.

> Rcpp::sourceCpp('G:/fibonacci_example.cpp')
Warning message:
running command 'make -f "c:/bin/prog/R/R-CURR~1/etc/x64/Makeconf" -f "c:/bin/prog/R/R-CURR~1/share/make/winshlib.mk" SHLIB_LDFLAGS='$(SHLIB_CXXLD)' SHLIB_LD='$(SHLIB_CXXLD)' SHLIB="sourceCpp_1.dll" WIN=64 TCLBIN=64 OBJECTS="fibonacci_example.o"' had status 127
Error in Rcpp::sourceCpp("G:/notes/confs/cppCon/2015/cppCon2015slides/Rcpp/src/fibonacci_example.cpp") :
  Error 1 occurred building shared library.
WARNING: Rtools is required to build R packages but is not currently installed. Please download and install the appropriate version of Rtools before proceeding:
```

At the bottom of the terminal, a red URL is displayed: <https://cran.rstudio.com/bin/windows/Rtools/>.

A modal dialog box titled "Install Build Tools" is centered over the terminal. It contains the message: "Compiling C/C++ code for R requires installation of additional build tools." Below the message is a question: "Do you want to install the additional tools now?" with "Yes" and "No" buttons.

## SIMPLE EXAMPLE #2

The screenshot shows the RStudio interface. The top panel is a code editor with the file "fibonacci\_example.cpp" open. The code defines a recursive function for calculating Fibonacci numbers:

```
// [[Rcpp::export]]
int fibonacci(int n)
{
    if (n < 2) return n;
    return fibonacci(n - 1) + fibonacci(n - 2);
}
```

The bottom panel is a terminal window titled "Console" showing the build process and an error message:

```
c:/bin/prog/R/RStudio/ ↵
_CXXLDFLAGS)' SHLIB_LD='${SHLIB_CXXLD}' SHLIB="sourceCpp_1.dll" WIN=64 ^
TCLBIN=64 OBJECTS="fibonacci_example.o"' had status 127
Error in Rcpp::sourceCpp("G:/notes/confs/CppCon/2015/CppCon2015slides/
Rcpp/src/fibonacci_example.cpp") :
  Error 1 occurred building shared library.
WARNING: Rtools is required to build R packages but is not currently i
nstalled. Please download and install the appropriate version of Rtool
s before proceeding:
https://cran.rstudio.com/bin/windows/Rtools/trying URL 'https://cran.r
studio.com/bin/windows/Rtools/Rtools33.exe'
Content type 'application/x-msdos-program' length 32481209 bytes (31.0
 MB)
downloaded 31.0 MB

>
> |
```

## SIMPLE EXAMPLE #3

The screenshot shows the RStudio interface. The top panel displays a C++ code editor with the file 'fibonacci\_example.cpp'. The code defines a function 'fibonacci' that uses the Rcpp::export attribute and implements the recursive formula for Fibonacci numbers. The bottom panel is a 'Console' window showing the R environment. It starts with the standard R welcome message, followed by information about R being a collaborative project, and then a command-line session where the user runs Rcpp code to calculate the 10th Fibonacci number.

```
// [[Rcpp::export]]
int fibonacci(int n)
{
    if (n < 2) return n;
    return fibonacci(n - 1) + fibonacci(n - 2);
}

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> Rcpp::sourcecpp('G:/notes/confs/CppCon/2015/cppcon2015slides/Rcpp/src/fibonacci_example.cpp')
> fibonacci(10)
[1] 55
> |
```

## SIMPLE EXAMPLE #4

The screenshot shows the RStudio interface with two panes. The top pane is the 'Source' editor containing C++ code for a Fibonacci function. The bottom pane is the 'Console' window showing R's startup message and a command-line session.

```
fibonacci_example.cpp *
Source on Save | ? | Source
1 // [[Rcpp::export]]
2 int fibonacci(int n)
3 {
4     if (n < 2) return n;
5     return fibonacci(n - 1) + fibonacci(n - 2);
6 }
7
8 /*** R
9 fibonacci(10)
10 */

8:7 (R Code Chunk) C/C++ ▾
```

Console ~ /

You are welcome to use R under certain conditions.  
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.

```
> Rcpp::sourceCpp('G:/notes/confs/CppCon/2015/CppCon2015slides/Rcpp/src/fibonacci.cpp')

> fibonacci(10)
[1] 55
> |
```

# LINUX SETUP EXAMPLE #0

The screenshot shows the RStudio interface on a Linux system. The left pane displays a C++ code editor with a file named `hello.cpp`. The code contains a simple Rcpp script that prints "Hello world". The right pane shows the R environment, which is currently empty. Below the environment is a file browser window titled "Home" showing the directory structure. The bottom pane is an R console window displaying the standard R startup message and license information.

```
1 #include <Rcpp.h>
2
3 // [[Rcpp::export]]
4 void Rcpp_hello()
5 {
6     Rcpp::Rcout << "Hello world" << '\n';
7 }
8
9 /**
10 Rcpp_hello()
11 */
```

R version 3.2.1 (2015-06-18) -- "World-Famous Astronaut"  
Copyright (C) 2015 The R Foundation for Statistical Computing  
Platform: x86\_64-unknown-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.

> |

# LINUX SETUP EXAMPLE #1

The screenshot shows the RStudio interface on a Linux system. The left pane displays the code for 'hello.cpp':

```
1 #include <Rcpp.h>
2
3 // [[Rcpp::export]]
4 void Rcpp_hello()
5 {
6     Rcpp::Rcout << "Hello world" << '\n';
7 }
8
9 /*** R
10 Rcpp_hello()
11 */
```

The right pane shows the R console output:

```
Environment is empty
```

The bottom-left pane shows the R startup message:

```
Copyright (C) 2015 The R Foundation for Statistical Computing
Platform: x86_64-unknown-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.
You are welcome to redistribute it under certain conditions.
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.
Type 'contributors()' for more information and
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or
'help.start()' for an HTML browser interface to help.
Type 'q()' to quit R.

> Rcpp::sourceCpp('r/hello.cpp')
Error in loadNamespace(name) : there is no package called 'Rcpp'
> |
```

# LINUX SETUP EXAMPLE #2

The screenshot shows the RStudio interface on a Linux system. The left pane displays a C++ code editor with the file `hello.cpp`. The code includes an Rcpp header and a function `Rcpp_hello()` that prints "Hello world". The right pane shows the R environment, which is currently empty. Below the environment is a file browser window titled "Files" showing the directory structure under "Home". The bottom pane is a terminal window displaying the R startup message and the command `> install.packages("Rcpp")`.

```
1 #include <Rcpp.h>
2
3 // [[Rcpp::export]]
4 void Rcpp_hello()
5 {
6     Rcpp::Rcout << "Hello world" << '\n';
7 }
8
9 /*** R
10 Rcpp_hello()
11 */
```

Environment is empty

Files Plots Packages Help Viewer

Name	Size	Modified
.History	49 B	Sep 20, 2015, 11:51 AM
builds		
cpp		
Desktop		
Downloads		
r		

Console -> Copyright (C) 2015 The R Foundation for Statistical Computing  
Platform: x86\_64-unknown-linux-gnu (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.

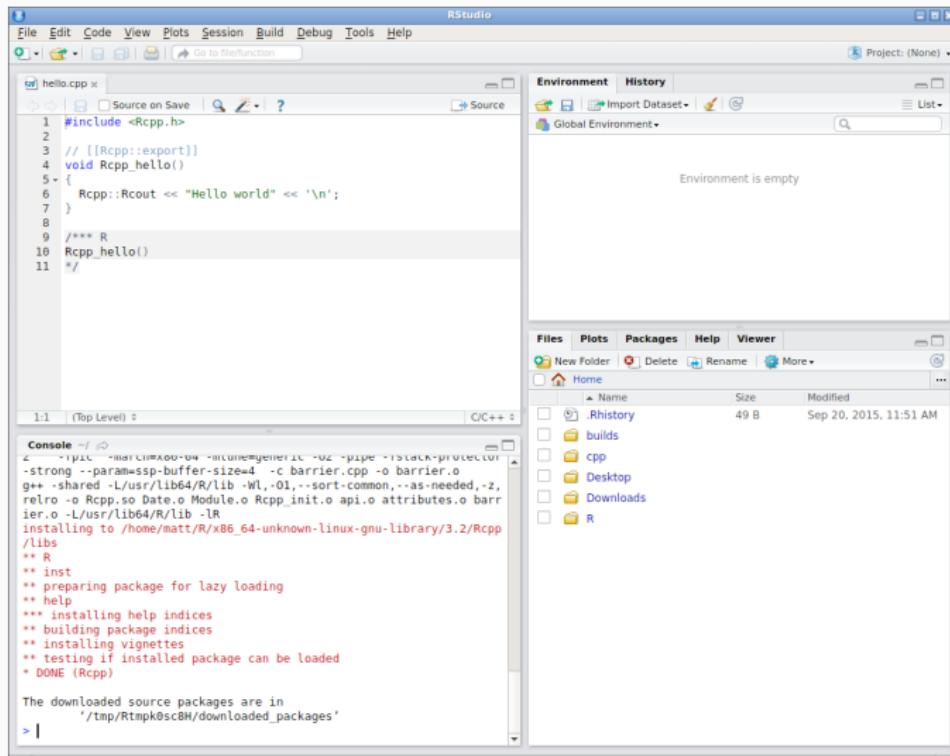
Natural language support but running in an English locale

R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.

```
> Rcpp::sourceCpp('r/hello.cpp')
Error in loadNamespace(name) : there is no package called 'Rcpp'
> install.packages("Rcpp")
```

## LINUX SETUP EXAMPLE #3



# LINUX SETUP EXAMPLE #4

The screenshot shows the RStudio interface with the following components:

- File Editor:** Displays the code for `hello.cpp`. The code includes an Rcpp header, a function definition, and an R interface block.
- Environment View:** Shows the global environment with a single entry: `Rcpp_hello` (function).
- File Browser:** Shows the local directory structure under "Home".
- Console:** Displays the output of the R command `Rcpp::sourceCpp('r/hello.cpp')`, which installs the package and shows the results of the `Rcpp_hello()` function.

```
1 #include <Rcpp.h>
2
3 // [[Rcpp::export]]
4 void Rcpp_hello()
5 {
6   Rcpp::Rcout << "Hello world" << '\n';
7 }
8
9 /*** R
10 Rcpp_hello()
11 **/
```

```
11:3 (Top Level) # C/C++ #
```

```
Console -> R
R version 3.2.0 -- "Lemonade" (2015-04-16)
Copyright (C) 2015 The R Foundation for Statistical Computing
ISBN 978-3-900051-07-0
Platform: x86_64-pc-linux-gnu (64-bit)

installing to /home/matt/R/x86_64-unknown-linux-gnu-library/3.2/Rcpp
/lib
** R
** inst
** preparing package for lazy loading
** help
*** installing help indices
** building package indices
** installing vignettes
** testing if installed package can be loaded
* DONE (Rcpp)

The downloaded source packages are in
  '/tmp/Rtmpk0scBH/downloaded_packages'
> Rcpp::sourceCpp('r/hello.cpp')

> Rcpp_hello()
Hello world
> |
```

- R language — C API
  - Writing R Extensions:  
<https://cran.r-project.org/doc/manuals/r-release/R-exts.html>
- Rcpp — C++ API — ABI implications
  - <https://isocpp.org/wiki/faq/compiler-dependencies#binary-compat>
- Most platforms: GNU Compiler Collection
  - Windows: Rtools, <https://cran.r-project.org/bin/windows/Rtools/>
    - R-SIG-windows, <https://stat.ethz.ch/mailman/listinfo/r-sig-windows>
  - Frequently Asked Questions about Rcpp - What compiler can I use?  
<http://dirk.eddelbuettel.com/code/rcpp/Rcpp-FAQ.pdf>
  - <https://cran.r-project.org/doc/manuals/R-admin.html#Platform-notes>

- "It is necessary to know something about how R objects are handled in C code.
- All the R objects you will deal with will be handled with the type SEXP, which is a pointer to a structure with `typedef SEXPREC`.
- SEXP is an acronym for Simple EXPression, common in LISP-like language syntaxes.
- Think of this structure as a variant type that can handle all the usual types of R objects, that is vectors of various modes, functions, environments, language objects and so on."

[https://cran.r-project.org/doc/manuals/r-release/R-exts.html#Calling-\\_002eCall](https://cran.r-project.org/doc/manuals/r-release/R-exts.html#Calling-_002eCall)

*"The R object types are represented by a C structure defined by a `typedef SEXPREC` in `Rinternals.h`. It contains several things among which are pointers to data blocks and to other SEXPRECs. A SEXP is simply a pointer to a SEXPREC."*

- PROTECT a UNPROTECT macros – R's GC

<https://cran.r-project.org/doc/manuals/r-release/R-exts.html#Garbage-Collection>  
<http://adv-r.had.co.nz/C-interface.html>

## COMPILATION, INLINE — EXAMPLE — RCPP::AS & RCPP::WRAP

```
fibonacci_impl = '
int fibonacci(int n)
{
  if (n < 2) return n;
  return fibonacci(n - 1) + fibonacci(n - 2);
}

fibonacci_body = '
  int n = Rcpp::as<int>(in_n);
  return Rcpp::wrap(fibonacci(n));
'

# install.packages("inline")
fibonacci_function = inline::cxxfunction(signature(in_n = "integer"),
                                             body = fibonacci_body,
                                             inc = fibonacci_impl,
                                             plugin = "Rcpp")

fibonacci_function(10) # returns 55
```

## COMPILATION, INLINE — VERBOSE OUTPUT |

```
fibonacci_function = inline::cxxfunction(..., verbose = TRUE)
```

>> Program source :

```
1 :  
2 : // includes from the plugin  
3 :  
4 : #include <Rcpp.h>  
5 :  
6 :  
7 : #ifndef BEGIN_RCPP  
8 : #define BEGIN_RCPP  
9 : #endif  
10 :  
11 : #ifndef END_RCPP  
12 : #define END_RCPP  
13 : #endif  
14 :
```

## COMPILATION, INLINE — VERBOSE OUTPUT II

```
15 : using namespace Rcpp;  
16 :  
17 :  
18 : // user includes  
19 :  
20 : int fibonacci(int n)  
21 : {  
22 :     if (n < 2) return n;  
23 :     return fibonacci(n - 1) + fibonacci(n - 2);  
24 : }  
25 :  
26 :  
27 : // declarations  
28 : extern "C" {  
29 : SEXP filece83d074c9d( SEXP in_n) ;  
30 : }  
31 :  
32 : // definition  
33 :
```

## COMPILATION, INLINE — VERBOSE OUTPUT III

```
34 : SEXP filece83d074c9d( SEXP in_n ){
35 : BEGIN_RCPP
36 :
37 :     int n = Rcpp::as<int>(in_n);
38 :     return Rcpp::wrap(fibonacci(n));
39 :
40 : END_RCPP
41 : }
42 :
43 :
```

## COMPILATION, RCPP CPPFUNCTION — EXAMPLE

```
fibonacci_source = '
int fibonacci(int n)
{
  if (n < 2) return n;
  return fibonacci(n - 1) + fibonacci(n - 2);
}'

fibonacci_cpp = Rcpp::cppFunction(code = fibonacci_source)

fibonacci_cpp(10) # returns 55
```

## COMPILATION, RCPP CPPFUNCTION — VERBOSE OUTPUT |

```
fibonacci_cpp = Rcpp::cppFunction(code = fibonacci_source, verbose = TRUE)
```

Generated code **for** function definition:

---

```
#include <Rcpp.h>

using namespace Rcpp;

// [[Rcpp::export]]

int fibonacci(int n)
{
  if (n < 2) return n;
  return fibonacci(n - 1) + fibonacci(n - 2);
}
```

Generated **extern "C"** functions

## COMPILATION, RCPP CPPFUNCTION — VERBOSE OUTPUT II

---

```
#include <Rcpp.h>
// fibonacci
int fibonacci(int n);
RcppExport SEXP sourceCpp_2_fibonacci(SEXP nSEXP) {
BEGIN_RCPP
    Rcpp::RObject __result;
    Rcpp::RNGScope __rngScope;
    Rcpp::traits::input_parameter< int >::type n(nSEXP);
    __result = Rcpp::wrap(fibonacci(n));
    return __result;
END_RCPP
}
```

Generated R functions

---

## COMPILATION, RCPP CPPFUNCTION — VERBOSE OUTPUT III

```
`.sourceCpp_2_DLLInfo` <- dyn.load('C:/Users/Matt/AppData/Local/Temp/1/Rtmp  
fibonacci <- Rcpp:::sourceCppFunction(function(n) {}, FALSE, `.sourceCpp_2_  
rm(`.sourceCpp_2_DLLInfo`)  
Building shared library
```

---

DIR: ...

## COMPILATION, RCPP ATTRIBUTES — EXAMPLE

```
// fibonacci_example.cpp

// [[Rcpp::export]]
int fibonacci(int n)
{
    if (n < 2) return n;
    return fibonacci(n - 1) + fibonacci(n - 2);
}

/** R
fibonacci(10)
*/
```

Rcpp::sourceCpp('fibonacci\_example.cpp') # returns 55

## COMPILATION, RCPP CPPFUNCTION — VERBOSE OUTPUT |

```
Rcpp::sourceCpp('fibonacci_example.cpp', verbose = TRUE)
```

```
> Rcpp::sourceCpp('fibonacci_example.cpp', verbose = TRUE)
```

```
Generated extern "C" functions
```

---

```
#include <Rcpp.h>
// fibonacci
int fibonacci(int n);
RcppExport SEXP sourceCpp_4_fibonacci(SEXP nSEXP) {
BEGIN_RCPP
    Rcpp::RObject __result;
    Rcpp::RNGScope __rngScope;
    Rcpp::traits::input_parameter< int >::type n(nSEXP);
    __result = Rcpp::wrap(fibonacci(n));
```

## COMPILATION, RCPP CPPFUNCTION — VERBOSE OUTPUT II

```
    return __result;
END_RCPP
}
```

Generated R functions

---

```
`.sourceCpp_4_DLLInfo` <- dyn.load('.../sourcecpp_ce857c352c1/sourceCpp_7.d  
fibonacci <- Rcpp:::sourceCppFunction(function(n) {}, FALSE, `.sourceCpp_4_  
rm(`.sourceCpp_4_DLLInfo`)
```

Building shared library

---

```
DIR: .../sourcecpp_ce857c352c1
```

```
.../bin/x64/R CMD SHLIB -o "sourceCpp_7.dll" "" "fibonacci_example.cpp"
```

## COMPILATION, RCPP CPPFUNCTION — VERBOSE OUTPUT III

```
g++ ... -c fibonacci_example.cpp -o fibonacci_example.o  
g++ ... -shared -o sourceCpp_7.dll fibonacci_example.o  
  
> fibonacci(10)  
[1] 55
```

## EXCEPTION HANDLING

---

```
::Rf_error
```

```
throw std::range_error
```

<http://gallery.rcpp.org/articles/intro-to-exceptions/>

# DATA STRUCTURES

---

Foundation and core:

- RObject
  - NumericVector
  - IntegerVector
- RAII instead of manual PROTECT / UNPROTECT
  - <https://isocpp.org/wiki/faq/exceptions#finally>
  - "smart SEXP" (resource)

## INTEGERVECTOR

```
#include <algorithm>

#include <Rcpp.h>

// [[Rcpp::export]]
int accumulate(Rcpp::IntegerVector v)
{
    return std::accumulate(v.begin(), v.end(), 0);
}

/** R
accumulate(1:5) # returns 15
*/
```

# INTEGERVECTOR - LIGHTWEIGHT PROXY OBJECT

Not call-by-value

[https://en.wikipedia.org/wiki/Evaluation\\_strategy](https://en.wikipedia.org/wiki/Evaluation_strategy)

```
#include <Rcpp.h>

// [[Rcpp::export]]
void tweak(Rcpp::IntegerVector v)
{
  if (v.size() > 0) v[0] = 42;
}

/** R
v = 1:5 # 1 2 3 4 5
stopifnot(v == 1:5)
tweak(v) # 42 2 3 4 5
stopifnot(v == c(42, 2:5))
*/
```

# NUMERICVECTOR, REFERENCE SEMANTICS

```
1 #include <algorithm>
2 #include <Rcpp.h>
3
4 // [[Rcpp::export]]
5 void print_sorted(Rcpp::NumericVector v)
6 {
7     Rcpp::Rcout << "[C++ Before: " << v << '\n';
8     std::sort(v.begin(), v.end());
9     Rcpp::Rcout << "[C++ After: " << v << '\n';
10 }
11
12 /*** R
13 v = seq(from = 5.5, to = 1.5)
14 cat('[R] Before: ', v, '\n')
15 print_sorted(v)
16 cat('[R] After: ', v, '\n')
17 */
```

8:33 print\_sorted(Rcpp::NumericVector v): void ↴

Console Source Cpp ×

~/

[R] Before: 5.5 4.5 3.5 2.5 1.5

```
> print_sorted(v)
[C++ Before: 5.5 4.5 3.5 2.5 1.5
[C++ After: 1.5 2.5 3.5 4.5 5.5

> cat('[R] After: ', v, '\n')
[R] After: 1.5 2.5 3.5 4.5 5.5
```

# NUMERICVECTOR, DEEP COPY: RCPP::CLONE

```
1 #include <algorithm>
2 #include <Rcpp.h>
3
4 // [[Rcpp::export]]
5 void print_sorted(Rcpp::NumericVector v_in)
6 {
7     Rcpp::NumericVector v(Rcpp::clone(v_in));
8     Rcpp::Rcout << "[C++ Before: " << v << '\n';
9     std::sort(v.begin(), v.end());
10    Rcpp::Rcout << "[C++ After: " << v << '\n';
11 }
12
13 /*** R
14 v = seq(from = 5.5, to = 1.5)
15 cat('[R] Before: ', v, '\n')
16 print_sorted(v)
17 cat('[R] After: ', v, '\n')
7:41 [ f print_sorted(Rcpp::NumericVector v_in): void ⇨
```

Console Source Cpp ×

~/🔗  
[R] Before: 5.5 4.5 3.5 2.5 1.5

> print\_sorted(v)  
[C++ Before: 5.5 4.5 3.5 2.5 1.5  
[C++ After: 1.5 2.5 3.5 4.5 5.5

> cat('[R] After: ', v, '\n')  
[R] After: 5.5 4.5 3.5 2.5 1.5

- Rcpp::NumericMatrix
- Rcpp::LogicalVector
- Rcpp::CharacterVector
- Rcpp::RawVector

## OTHER DATA STRUCTURES

---

- List / GenericVector
  - Dynamically Heterogeneous
- DataFrame
- Function, Environment
- Rcpp::Named

## Rcpp::Named

```
1 #include <Rcpp.h>
2
3 // [[Rcpp::export]]
4 Rcpp::NumericVector named_values()
5 {
6     return Rcpp::NumericVector::create(
7         Rcpp::Named("alpha") = 1.1,
8         Rcpp::Named("beta") = 2.2,
9         Rcpp::Named("gamma") = 3.3
10    );
11 }
12
13 /*** R
14 named_values()
15 */
16
```

10:5 f named\_values(): Rcpp::NumericVector ▾

Console Source Cpp ✎

~/

```
> named_values()
alpha   beta  gamma
1.1    2.2   3.3
```

## Rcpp::List

```
1 // [[Rcpp::plugins(cpp11)]]
2 #include <Rcpp.h>
3
4 // [[Rcpp::export]]
5 void greet(Rcpp::List list)
6 {
7     const auto year = Rcpp::as<int>(list["year"]);
8     const auto con = Rcpp::as<std::string>(list["con"]);
9     Rcpp::Rcout << "Greetings " << con << ' ' << year << "!\n";
10 }
11
12 /**
13  * @R
14  * r_list = List(year = 2015, con = "CppCon")
15  * greet(r_list)
16  */
17
```

17:1 (Top Level) C

Console ~/ ↵

```
> greet(r_list)
Greetings CppCon 2015!

> show(r_list)
$year
[1] 2015

$con
[1] "CppCon"
```

## Rcpp::Function

```
1 #include <Rcpp.h>
2
3 // [[Rcpp::export]]
4 double apply_function(std::string function_name, double arg)
5 {
6     Rcpp::Function f(function_name);
7     double result = Rcpp::as<double>(f(arg));
8     return result;
9 }
10
11 /**
12 square = function(x) x * x
13 apply_function("square", 2.0)
14 */
15
```

14:1 | C (R Code Chunk)

Console

Source Cpp

~/

```
> apply_function("square", 2.0)
[1] 4
```

- Rmath.h
- PRNGs, Statistical Distributions
- <http://gallery.rcpp.org/articles/using-rmath-functions/>
- <http://gallery.rcpp.org/articles/random-number-generation/>
- <http://dirk.eddelbuettel.com/blog/2012/11/14/>

- <https://cran.r-project.org/web/packages/Rcpp/vignettes/Rcpp-package.pdf>
  - Rcpp.package.skeleton
  - Makevars
  - Makevars.win

- `Rcpp::as` - from R to C++
- `Rcpp::wrap` - from C++ to R
- intrusive and nonintrusive extension - conversion vs. specialization
  - nonintrusive: <http://c2.com/cgi/wiki?OpenClosedPrinciple>
- <http://dirk.eddelbuettel.com/code/rcpp/Rcpp-extending.pdf>
- <http://gallery.rcpp.org/articles/custom-as-and-wrap-example/>

## EXTENDING - Rcpp::wrap - FROM C++ TO R

```
// [[Rcpp::plugins(cpp11)]]
#include <RcppCommon.h>

struct point { double x, y; };

namespace Rcpp { template <> SEXP wrap(const point & p); }

// [[Rcpp::export]]
point wrapped(double x, double y)
{
    return point{x, y};
}

#include <Rcpp.h>
```

## EXTENDING - Rcpp::wrap - FROM C++ TO R

```
namespace Rcpp
{
template <> SEXP wrap(const point & p)
{
    return Rcpp::NumericVector::create(
        Rcpp::Named("x") = p.x,
        Rcpp::Named("y") = p.y);
}
}

/** R
wrapped(1., 2.)
*/
```

## EXTENDING - Rcpp::wrap - FROM C++ TO R

```
8 // [[Rcpp::export]]
9 point wrapped(double x, double y)
10 {
11     return point{x, y};
12 }
13
14 #include <Rcpp.h>
15
16 namespace Rcpp
17 {
18 template <> SEXP wrap(const point & p)
19 {
20     return Rcpp::NumericVector::create(
21         Rcpp::Named("x") = p.x,
22         Rcpp::Named("y") = p.y);
23 }
24 }
25
26 /**
27 wrapped(1., 2.)
28 */
29
```

27:16 (R Code Chunk) ▾

Console Source Cpp ×

~/

```
> wrapped(1., 2.)
x y
1 2
```

## EXTENDING - Rcpp::as - FROM R TO C++

```
// [[Rcpp::plugins(cpp11)]]
#include <RcppCommon.h>

struct point { double x, y; };

namespace Rcpp { template <> point as(SEXP coords); }

// [[Rcpp::export]]
double squared_norm(point p)
{
    return p.x * p.x + p.y * p.y;
}

#include <Rcpp.h>
```

## EXTENDING - Rcpp::as - FROM R TO C++

```
namespace Rcpp
{
template <> point as(SEXP coords_in)
{
    Rcpp::NumericVector coords(coords_in);
    auto x = coords[0];
    auto y = coords[1];
    return point{x, y};
}

/** R
squared_norm(c(1., 2.))
*/
```

## EXTENDING - Rcpp::as - FROM R TO C++

```
1 // [[Rcpp::plugins(cpp11)]]
2 #include <RcppCommon.h>
3
4 struct point { double x, y; };
5
6 namespace Rcpp { template <> point as(SEXP coords); }
7
8 // [[Rcpp::export]]
9 double squared_norm(point p)
10 {
11     return p.x * p.x + p.y * p.y;
12 }
```

11:32  squared\_norm(point p): double ↴

Console

Source Cpp 

```
~/
> coordinates = c(1., 2.)
> show(coordinates)
[1] 1 2
> squared_norm(coordinates)
[1] 5
```

# EXPOSING CLASSES, MODULES

---

- Rcpp::Xptr
  - <http://www.r-bloggers.com/external-pointers-with-rcpp/>
  - <http://gallery.rcpp.org/articles/passing-cpp-function-pointers/>
- RCPP\_MODULE
  - inspiration: Boost.Python, <http://boost.org/libs/python>
  - in particular: BOOST\_PYTHON\_MODULE,  
<http://www.boost.org/doc/libs/release/libs/python/doc/tutorial/doc/html>
  - <http://dirk.eddelbuettel.com/code/rcpp/Rcpp-modules.pdf>

```
struct point { double x, y; };
```

```
RCPP_MODULE(point_module) {
  Rcpp::class_<point>("point")
  .field( "x", &point::x )
  .field( "y", &point::y )
  ;
}
```

# Rcpp & Python

<http://gallery.rcpp.org/articles/rcpp-python/>

```
1 #include <Rcpp.h>
2 #include <Python.h>
3
4 // http://gallery.rcpp.org/articles/rcpp-python/
5
6 //[[Rcpp::export]]
7 void initialize_python() {
8     Py_SetProgramName("hello_from_python");
9     Py_Initialize();
10 }
11
12 //[[Rcpp::export]]
13 void finalize_python() {
14     Py_Finalize();
15 }
16
17 //[[Rcpp::export]]
18 void hello_python() {
19     PyRun_SimpleString("import datetime\n"
20                         "now = datetime.datetime.now()\n"
21                         "print('Hello CppCon', now.year, '!')\n");
22 }
23
24 /** R
25
26 initialize_python()
27 hello_python()
28 finalize_python()
29 */
21:12  hello_python(): void 





> initialize_python()  

> hello_python()  

> finalize_python()  

Hello CppCon 2015 !


```

- Syntactic Sugar
- <http://dirk.eddelbuettel.com/code/rcpp/Rcpp-sugar.pdf>

# SUGAR EXAMPLE

```
1 #include <Rcpp.h>
2
3 // [[Rcpp::export]]
4 bool any_less(Rcpp::NumericVector x, Rcpp::NumericVector y)
5 {
6     return is_true(any(x < y));
7 }
8
9 /**
10  * R
11  * x = c(1, 2, 3)
12  * y = c(4, 5, 6)
13  * any_less(x, y)
14  * any_less(y, x)
15 */
```

14:1 (R Code Chunk) ▾

Console Source Cpp ×

~/

```
> x = c(1, 2, 3)
> y = c(4, 5, 6)
> any_less(x, y)
[1] TRUE
> any_less(y, x)
[1] FALSE
```

- Implementation: Expression Templates, CRTP
- <https://cran.r-project.org/web/packages/Rcpp/vignettes/Rcpp-sugar.pdf>
- <http://gallery.rcpp.org/articles/sugar-function-clamp/>
- <http://gallery.rcpp.org/articles/sugar-for-high-level-vector-operations/>

- embedding R in C++ code
- <http://dirk.eddelbuettel.com/code/rinside.html>
- `install.packages("RInside")`

```
[matt@jacobi standard]$ pwd
/home/matt/R/x86_64-unknown-linux-gnu-library/3.2/RInside/examples/standard
[matt@jacobi standard]$ cat rinside_sample0.cpp
// -*- mode: C++; c-indent-level: 4; c-basic-offset: 4; tab-width: 8; -*-
//
// Simple example showing how to do the standard 'hello, world' using embedded R
//
// Copyright (C) 2009 Dirk Eddelbuettel
// Copyright (C) 2010 Dirk Eddelbuettel and Romain Francois
//
// GPL'ed

#include <RInside.h>                                // for the embedded R via RInside

int main(int argc, char *argv[]) {
    RInside R(argc, argv);                          // create an embedded R instance
    R["txt"] = "Hello, world!\n";                   // assign a char* (string) to 'txt'
    R.parseEvalQ("cat(txt)");                        // eval the init string, ignoring any returns
    exit(0);
}
```

```
[matt@jacobi standard]$ make rinside_sample0
g++ -I/usr/include/R/ -I/home/matt/R/x86_64-unknown-linux-gnu-library/3.2/Rcpp/include -I/home/matt/R/x86_64-unknown-linux-gnu-library/3.2/RInside/include -march=h=x86-64 -mtune=generic -O2 -pipe -fstack-protector-strong --param=ssp-buffer-size=4 -Wall -D_FORTIFY_SOURCE=2   rinside_sample0.cpp -Wl,--export-dynamic -fopenmp -L/usr/lib64/R/lib -lR -lpcre -llzma -lbz2 -lz -lrt -ldl -lm -lblas -llapack -L/home/matt/R/x86_64-unknown-linux-gnu-library/3.2/RInside/lib -lRInside -Wl,-rpath,/home/matt/R/x86_64-unknown-linux-gnu-library/3.2/RInside/lib -o rinside_sample0
[matt@jacobi standard]$ ls
GNUmakefile          rinside_sample11.cpp  rinside_sample5.cpp
Makefile.win          rinside_sample12.cpp  rinside_sample6.cpp
cmake                rinside_sample13.cpp  rinside_sample7.cpp
rinside_callbacks0.cpp    rinside_sample14.cpp  rinside_sample8.cpp
rinside_callbacks1.cpp    rinside_sample15.cpp  rinside_sample9.cpp
rinside_module_sample0.cpp    rinside_sample16.cpp  rinside_test0.cpp
rinside_sample0          rinside_sample17.cpp  rinside_test1.cpp
rinside_sample0.cpp      rinside_sample2.cpp   rinside_test2.cpp
rinside_sample1.cpp      rinside_sample3.cpp
rinside_sample10.cpp     rinside_sample4.cpp
[matt@jacobi standard]$ ./rinside_sample0
Hello, world!
```

## BH: Boost C++ Header Files

- <https://cran.r-project.org/web/packages/BH/>
- <http://dirk.eddelbuettel.com/code/bh.html>
- <https://github.com/eddelbuettel/bh>
- <http://gallery.rcpp.org/articles/using-boost-with-bh/>

## RcppArmadillo: Rcpp Integration for the Armadillo Linear Algebra Library

- <http://dirk.eddelbuettel.com/code/rcpp.armadillo.html>
- <https://github.com/RcppCore/RcppArmadillo>

### RcppEigen: Rcpp Integration for the Eigen Linear Algebra Library

- <https://cran.r-project.org/web/packages/RcppEigen/>
- <https://github.com/RcppCore/RcppEigen>

### RcppGSL

- <http://dirk.eddelbuettel.com/code/rcpp.gsl.html>
- <https://github.com/eddelbuettel/rcppgsl>

### RcppParallel

- <https://github.com/RcppCore/RcppParallel>
- <https://cran.r-project.org/web/packages/RcppParallel/>
- <http://rcppcore.github.io/RcppParallel/>
- <http://gallery.rcpp.org/articles/parallel-vector-sum/>

### CRAN Users

- <http://dirk.eddelbuettel.com/code/rcpp.cranusers.html>

- Eigen::Map
  - [http://eigen.tuxfamily.org/dox/group\\_\\_TutorialMapClass.html](http://eigen.tuxfamily.org/dox/group__TutorialMapClass.html)
  - [http://eigen.tuxfamily.org/dox/classEigen\\_1\\_1Map.html](http://eigen.tuxfamily.org/dox/classEigen_1_1Map.html)

## RCPPEIGEN EXAMPLE - NOT AVAILABLE

```
1 // [[Rcpp::depends(RcppEigen)]]
2 #include <RcppEigen.h>
3
4 // [[Rcpp::export]]
5 double elements_sum(Eigen::Map<Eigen::MatrixXd> matrix)
6 {
7     return matrix.sum();
8 }
```

```
8:2 f elements_sum(Eigen::Map<Eigen::MatrixXd> matrix): double
```

C/C++

Console Source Cpp ×

C:/Users/Matt/Downloads/CppCon2015Slides/Rcpp/src/ ↵

> Rcpp::sourceCpp("RcppEigen.cpp")

Error: Package 'RcppEigen' referenced from Rcpp::depends in source file  
RcppEigen.cpp is not available.

## RCPPEIGEN EXAMPLE - SETUP

The screenshot shows the RStudio interface with the following components:

- Code Editor:** Displays the following C++ code:

```
1 // [[Rcpp::depends(RcppEigen)]]
2 #include <RcppEigen.h>
3
4 // [[Rcpp::export]]
5 double elements_sum(Eigen::Map<Eigen::MatrixXd> matrix)
6 {
7     return matrix.sum();
8 }
```

- Console:** Displays the R session history:

```
C:/Users/Matt/Downloads/CppCon2015Slides/Rcpp/src/ ↵
> Rcpp::sourceCpp("RcppEigen.cpp")
Error: Package 'RcppEigen' referenced from Rcpp::depends in source file
RcppEigen.cpp is not available.
> install.packages("RcppEigen")
trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.2/RcppEigen_0
.3.2.5.0.zip'
content type 'application/zip' length 2136315 bytes (2.0 MB)
downloaded 2.0 MB

package 'RcppEigen' successfully unpacked and MD5 sums checked
```

## RCPPEIGEN EXAMPLE - USE

```
1 // [[Rcpp::depends(RcppEigen)]]
2 #include <RcppEigen.h>
3
4 // [[Rcpp::export]]
5 double elements_sum(Eigen::Map<Eigen::MatrixXd> matrix)
6 {
7     return matrix.sum();
8 }
```

```
7:1 f elements_sum(Eigen::Map<Eigen::MatrixXd> matrix): double ↴
```

Console Source Cpp ✖

C:/Users/Matt/Downloads/CppCon2015Slides/Rcpp/src/ ↵

```
> Rcpp::sourceCpp("RcppEigen.cpp")
> matrix = matrix(data = 1.5:3.5, nrow = 3, ncol = 3)
> matrix
 [,1] [,2] [,3]
[1,] 1.5 1.5 1.5
[2,] 2.5 2.5 2.5
[3,] 3.5 3.5 3.5
> elements_sum(matrix)
[1] 22.5
> sum(matrix)
[1] 22.5
```

## RESOURCES

---

## RESOURCES: WHERE TO LEARN MORE

- <https://cran.r-project.org/web/packages/Rcpp/vignettes/>
- <http://dirk.eddelbuettel.com/code/rcpp/Rcpp-quickref.pdf>
- <http://gallery.rcpp.org/>
- <http://www.rcpp.org/book/>
- <http://dirk.eddelbuettel.com/presentations/>
- <http://adv-r.had.co.nz/Rcpp.html>
- <https://cran.r-project.org/doc/manuals/r-release/R-exts.html>

- <https://cran.r-project.org/web/packages/Rcpp/vignettes/>
- <http://news.gmane.org/gmane.comp.lang.r.rcpp>
- <http://stackoverflow.com/tags/rcpp/>

## RESOURCES: LIBRARIES

- <http://dirk.eddelbuettel.com/code/>
- <http://gallery.rcpp.org/articles/using-boost-with-bh/>
- <http://dirk.eddelbuettel.com/code/rquantlib.html>
- <https://rcppcore.github.io/RcppParallel/>
  - <http://bit.ly/1Ltycxk>

Romain François (@romain\_francois) Following

Rcpp/RcppParallel tutorial slides. [bit.ly/1Ltycxk](http://bit.ly/1Ltycxk)

[View translation](#)

...

### News

- <http://www.r-bloggers.com/>
- <http://dirk.eddelbuettel.com/blog/>
- <https://github.com/RcppCore/Rcpp>

### Conferences

- <https://www.r-project.org/conferences.html>
- <http://www.rinfinance.com/>
- <http://www.earl-conference.com/>

## RESOURCES: SLIDES

<https://speakerdeck.com/mattpd>

THANK YOU!  
QUESTIONS?