

# sprof internal

July 16, 2013

**Encoding** UTF-8

**Type** Package

**Title** Profiling, timing and optimization utilitites

**Version** 0.0-5

**Date** 2013-07-09

**Author** Günther Sawitzki

**Maintainer** Günther Sawitzki <gsawitzki@r-forge.r-project.org>

**Description** Support utilities for profiling and dynamic code analysis.

**License** GPL-2 | GPL-3

**Suggests** wordcloud, timeit

**URL** <http://http://sintro.r-forge.r-project.org>

**ByteCompile** FALSE

**KeepSource** TRUE

**BuildVignettes** FALSE

## R topics documented:

sprof-package . . . . .	2
adjacency . . . . .	2
list.as.matrix . . . . .	3
plot.sprof . . . . .	4
plot_nodes . . . . .	5
plot_profiles . . . . .	7
plot_stacks . . . . .	8
print.sprof . . . . .	9
print_profiles . . . . .	10
profiles_matrix . . . . .	11
readRprof . . . . .	11
rrle . . . . .	13
rrleb . . . . .	14
sampleRprof . . . . .	15

sprof01lm . . . . .	16
stackstoadj . . . . .	16
str_prof . . . . .	17
summary.sprof . . . . .	17
summary_terminals . . . . .	18
writeRprof . . . . .	19

<b>Index</b>	<b>21</b>
--------------	-----------

---

sprof-package	<i>sprof: Analysis of R profiles</i>
---------------	--------------------------------------

---

**Description**

Profiling, timing and optimization utilitites

**Details**

Package: sprof  
Type: Package  
Version: 0.0-5  
Date: 2013-07-09  
License: GPL-2 | GPL-3

readProf reads a profile file from [Rprof](#) or other profilers and returns a composite structure of class sprof.

To create a profile on the fly, use [sampleRprof](#).

To import profile information written by [Rprof](#) or other profilers, use [readRprof](#).

For sprof, the usual access functions are supported.

**Author(s)**

Günther Sawitzki <[gsawitzki@r-forge.r-project.org](mailto:gsawitzki@r-forge.r-project.org)>

**Examples**

```
## Not run:  
data(sprof01lm)  
str(sprof01lm)  
plot(sprof01lm)  
  
## End(Not run)
```

---

adjacency	<i>sprof to adjacency matix</i>
-----------	---------------------------------

---

**Description**

convert node information from a sprof structure to adjacency matrix.

**Usage**

```
adjacency(sprof, keep.names = TRUE, rmzero=TRUE)
```

**Arguments**

sprof	a sprof structure.
keep.names	boolean. Copy node names as row- and column names.
rmzero	boolean. Remove nodes with no edges.

**Value**

a correspondence matrix

**Author(s)**

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

**References**

See the vignette of package sprof.

**Examples**

```
data(sprof01lm)
adjacency(sprof01lm)
```

---

list.as.matrix

*Convert list to matrix*

---

**Description**

Convert list to matrix. List entries go to matrix columns, filled for equal length.

**Usage**

```
list.as.matrix(x, filler = NA)
```

**Arguments**

x	a list of numeric vectors.
filler	a value to be used as a filler

**Value**

A matrix with the values from x, filled to matrix shape.

**Author(s)**

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

**Examples**

```
x <- list(x1=c(1,2,3),x2=3, x3=4:8)
list.as.matrix(x)
list.as.matrix(x,filler=0)
```

---

plot.sprof

*plot for profiles*

---

**Description**

plot a plot for the output of class scode.

**Usage**

```
## S3 method for class 'sprof'
plot(x, ...)
```

**Arguments**

x	A data structure as returned by <a href="#">readRprof</a> .
...	further arguments passed to or from other methods.

**Value**

subject to change

**Note**

See the vignette for in-context explanations.  
Displays of the graph structure are given in the vignette.

**References**

<http://sintro.r-forge.r-project.org/>

**See Also**

[summaryRprof](#)  
[plot\\_profiles](#)  
[plot\\_nodes](#) [plot\\_stacks](#)

**Examples**

```
data(sprof01lm)

oldpar <- par(mfrow=c(2,2))
plot_nodes(sprof01lm)
par(oldpar)

oldpar <- par(mfrow=c(2,2))
plot_stacks(sprof01lm)
par(oldpar)

oldpar <- par(mfrow=c(2,2))
plot_profiles(sprof01lm)
par(oldpar)
```

---

plot_nodes	<i>Plot profiling information on node level.</i>
------------	--

---

**Description**

Various plots of a profile.

**Usage**

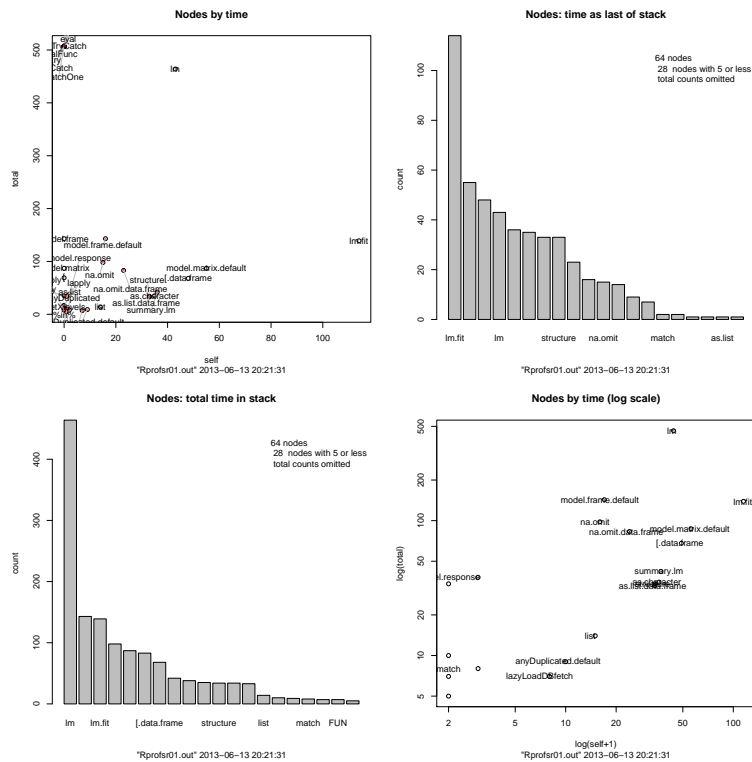
```
plot_nodes(x, which = c(1L, 2L, 3L, 4L), col = NULL,
ask = prod(par("mfcoll")) < length(which) && dev.interactive(),
src = NULL, mincount = 5, ...)
```

**Arguments**

x	preferably a sprof object. Other data structures may be extended
which	Selector of plots to show.
col	Colour table
ask	boolean. Ask for a new page?
src	String to be used as source identifier.
mincount	minimum total frequency count for node to be shown in barcharts.
...	passed.

**Details**

Plots are from this collection:



## Value

To come.

## Note

See the vignette of package sprof.

## Author(s)

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

## References

See the vignette of package sprof.

## See Also

See Also as [plot.sprof](#), ~~~

## Examples

```
data(sprof01lm)
oldpar <- par(mfrow=c(2,2))
plot_nodes(sprof01lm)
par(oldpar)
```

plot\_profiles

*Plot profiling information on profile level.*

## Description

Various plots of a profile.

## Usage

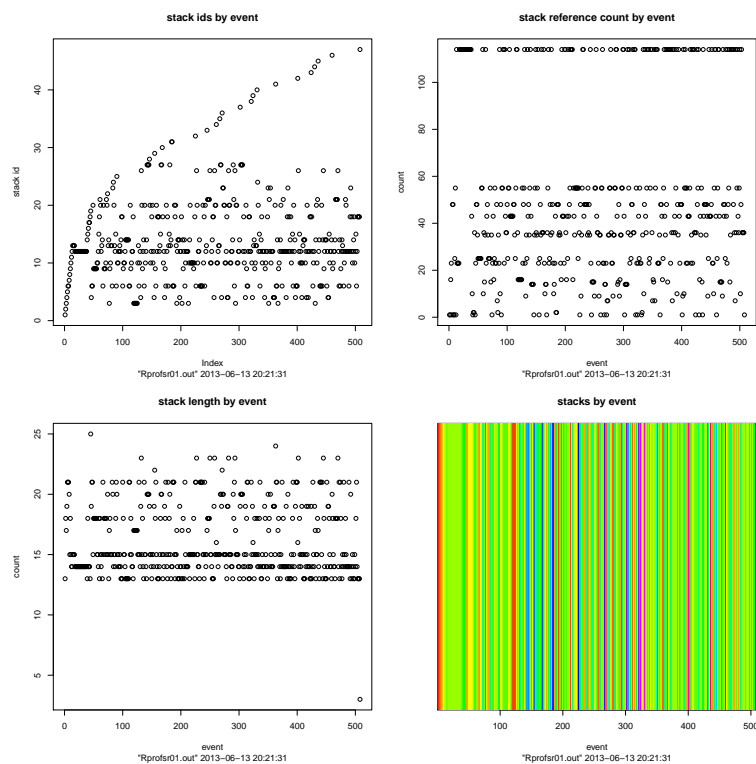
```
plot_profiles(x, which = c(1L, 2L, 3L, 4L), col,
ask = prod(par("mfcol")) < length(which) && dev.interactive(),
src = NULL, ...)
```

## Arguments

x	preferably a sprof object. Other data structures may be extended
which	Selector of plots to show.
col	Colour table
ask	boolean. Ask for a new page?
src	String to be used as source identifier.
...	passed.

## Details

Plots are from this collection:



**Note**

See the vignette of package sprof.

**Author(s)**

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

**References**

See the vignette of package sprof.

**See Also**

See Also as [plot.sprof](#), ~~~

**Examples**

```
data(sprof01lm)
oldpar <- par(mfrow=c(2,2))
plot_profiles(sprof01lm)
par(oldpar)
```

---

plot\_stacks

---

*Plot profiling information on stack level.*


---

**Description**

Various plots of a profile.

**Usage**

```
plot_stacks(x, which = c(1L, 2L), ask = prod(par("mfcol")) < length(which) && dev.interactive(), src = NULL)
```

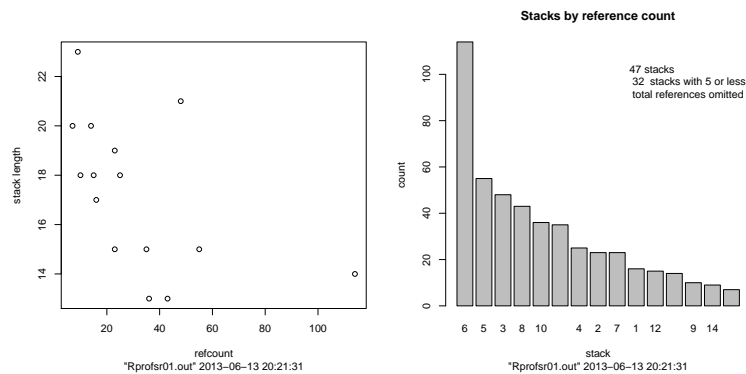
**Arguments**

x	preferably a sprof object. Other data structures may be extended
which	Selector of plots to show.
ask	boolean. Ask for a new page?
src	String to be used as source identifier.
mincount	minimum total frequency count for stack to be shown in barcharts.
...	passed.



## Details

Plots are from this collection:



## Value

To come.

## Note

See the vignette of package sprof.

## Author(s)

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

## References

See the vignette of package sprof.

## See Also

See Also as [plot.sprof](#), ~~~

## Examples

```
data(sprof01lm)
oldpar <- par(mfrow=c(2,2))
plot_stacks(sprof01lm)
par(oldpar)
```

---

print.sprof

*print for profiles*

---

## Description

Print a print for the output of class scode.

**Usage**

```
## S3 method for class 'sprof'
print(x, ...)
```

**Arguments**

**x** A data structure as returned by [readRprof](#).  
**...** further arguments passed to or from other methods.

**Value**

None.

**References**

<http://sintro.r-forge.r-project.org/>

**See Also**

[summaryRprof](#) [plot.sprof](#)

**Examples**

```
data(sprof01lm)
print(sprof01lm)
```

---

print_profiles	<i>Print profile information</i>
----------------	----------------------------------

---

**Description**

Print profile information.

**Usage**

```
print_profiles(x)
```

**Arguments**

**x** a sprof data structure.

**Value**

none

**Author(s)**

Günther Sawitzki <[gsawitzki@r-forge.r-project.org](mailto:gsawitzki@r-forge.r-project.org)>

**Examples**

```
data(sprof01lm)
print_profiles(sprof01lm)
```

---

profiles_matrix	<i>Extract a node incidence matrix from profile information.</i>
-----------------	--

---

**Description**

Extract a node incidence matrix from profile information.

**Usage**

```
profiles_matrix(x)
```

**Arguments**

x                      an sprof data structure.

**Value**

an incidence matrix, NA filled.

**Author(s)**

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

**Examples**

```
data(sprof01lm)
smat <- profiles_matrix(sprof01lm)
image(smat)
```

---

readRprof	<i>Read <a href="#">Rprof</a> Output files and Stack Logs</i>
-----------	---

---

**Description**

Read a log of stack entries, such as the output of the [Rprof](#) function, and generate a more accessible representation.

**Usage**

```
readRprof(filename = "Rprof.out", chunksize = 5000,
interval = 0.02,
head = c("auto", "none", "Rprofmem"),
id = NULL)
```

**Arguments**

filename	Name of a file produced by Rprof().
chunksize	Number of lines to read at a time.
interval	Real: time interval between samples, in s.
head	c("auto", "none", "Rprofmem") to interpret control information as provided by Rprof or Rprofmem. See details.
id	An optional identification string. Defaults to filename and date.

**Details**

This function reads a log file of stacks, one stack snapshot per line, stack entries separated by space.

As profiling output file could be very large, it is read in blocks of chunksize lines. Increasing chunksize will make the function run faster if sufficient memory is available.

The input format is controlled by the head argument. Format "auto" tries to detect control lines as interspersed by Rprof. These lines are not included in the output.

"none" ignores all control information and includes these lines as strange stacks.

"Rprofmem" isolates headers as provided by Rprofmem. new page entries are encoded as malloc requests with length 0.

**Value**

This data structure is subject to change.

Temporarily: A list with components

firstline	A verbatim copy of the first line of the input file. Typically this contains timing or formatting information.
nodes	A vector of node names. This may include stray entries from interspersed lines.
stacks	A vector of unique stacks found in input, stored as verbatim copies.
stacksrenc	A list of unique stacks in top down order (top first), encoded as vectors or references to stacks.
data	A vector encoding the data file as references to stacks.
timesRLE	Vector of sampling intervals, in milliseconds. Run-length encoded.
freq	A frequency table summarizing data.

**Author(s)**

Günther Sawitzki <gsawitzki@r-forge.r-project.org>, based on the code of [summaryRprof](#)

**References**

<http://sintro.r-forge.r-project.org/>

**See Also**

[summaryRprof](#)

[summaryRprof](#)

[flatProfile](#) in `library(proftools)`.

`{parse_rprof}` in `library(profr)`.

The chapter on “Tidying and profiling R code” in “Writing R Extensions” (see the ‘doc/manual’ subdirectory of the R source tree).

[Rprof](#) is a sampling profiler.

[tracemem](#) traces copying of an object via the C function `duplicate`.

[Rprofmem](#) is a non-sampling memory-use profiler.

<http://developer.r-project.org/memory-profiling.html>

**Examples**

```
## Not run:
## Rprof() is not available on all platforms
profintrval <- 0.001
simruns <- 100

n <- 10000
x <- runif(n)
y0 <- 2 + 3 * x

sinknull <- textConnection(NULL, "w"); sink(sinknull)
Rprof(tmp <- tempfile(), interval = profintrval)
for (i in 1:simruns) {y <- y0 + rnorm(n); xxx<- summary(lm(y~x))}
Rprof()

Rprof_out <- readRprof(tmp)

unlink(tmp)
sink(); close(sinknull)

str(Rprof_out)

## End(Not run)
```

---

rrle

---

*Recursive run length encoding.*


---

**Description**

Encode a matrix as run-length, top down. Encoding respects previous runs, e.g line 2 encodes runs in each run of line 1.

**Usage**

```
rrle(x)
```

**Arguments**

x                      a matrix.

**Value**

list of run length encoded lines

**Author(s)**

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

**See Also**

[rrleb](#), [~~~](#)

**Examples**

```
x <- matrix(c(
  1,1,1,2,2,
  3,3,4,4,4,
  5,5,6,6,7,
  8,9,9,0,0
),nrow=4, ncol =5, byrow=TRUE)
xrrle <- rrle(x)
xrrle

t(sapply(xrrle, inverse.rle))
```

---

rrleb

---

*Recursive run length encoding bottom up.*


---

**Description**

Encode a matrix as run-length, bottom up. Encoding respects previous runs, e.g line n-1 encodes runs in each run of line n.

May be removed.

**Usage**

```
rrleb(x)
```

**Arguments**

x                      a matrix.

**Value**

list of run length encoded lines

**Author(s)**

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

**See Also**

[rle](#), [~~~](#)

**Examples**

```
x <- matrix(c(
  3,3,4,4,4,
  5,5,6,6,7,
  8,9,9,0,0,
  1,1,1,2,2

),nrow=4, ncol =5, byrow=TRUE)
xrrleb <- rrleb(x)
xrrleb

t(sapply(xrrleb, inverse.rle))
```

---

sampleRprof

*Get a sample profile*


---

**Description**

Get a sample profile and return it as a sprof data structure.

**Usage**

```
sampleRprof(expr, runs = NULL, gcFirst = TRUE, interval = 0.001, ...)
```

**Arguments**

expr	an expression to be profiled.
runs	nr of runs to profile.
gcFirst	boolean. Profile GC.
interval	Real: time interval between samples, in s.
...	additional parameters, passed to Rprof

**Value**

A list of type sprof

**Author(s)**

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

**References**

<http://sintro.r-forge.r-project.org/>

**Examples**

```
res_lm <- sampleRprof(for (i in 1:1000) yy<- lm(runif(1000)~rnorm(1000)), runs=100)
```

---

sprof01lm	<i>sprof sample data</i>
-----------	--------------------------

---

**Description**

An example data set for the functions in package sprof.

**Usage**

```
data(sprof01lm)
```

**Format**

The format is: A List of 4 \$ info : 'data.frame': 1 obs. of 8 variables: \$ nodes : 'data.frame': 62 obs. of 5 variables: \$ stacks : 'data.frame': 50 obs. of 7 variables: \$ profiles: List of 4

**References**

See the vignette of package sprof.

**Examples**

```
data(sprof01lm)
str(sprof01lm)
plot(sprof01lm)
```

---

stackstoadj	<i>Stacks to adjacency matrix</i>
-------------	-----------------------------------

---

**Description**

convert stack information to adjacency matrix

**Usage**

```
stackstoadj(xstacks, xfreq, maxnode)
```

**Arguments**

xstacks	list of stack ids
xfreq	vector of frequencies or weights
maxnode	maximum of nodes (maybe higher then in stacks)

**Value**

the adjacency matrix



**Author(s)**

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

**Examples**

```
#
```

---

str_prof	<i>str for sprof objects</i>
----------	------------------------------

---

**Description**

str for sprof objects

**Usage**

```
str_prof(x)
```

**Arguments**

x                      an sprof object

**Author(s)**

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

**Examples**

```
data(sprof01lm)
str_prof(sprof01lm)
```

---

summary.sprof	<i>Summary for profiles</i>
---------------	-----------------------------

---

**Description**

Print a summary for the output of class scode.

**Usage**

```
## S3 method for class 'sprof'
summary(object, ...)
```

**Arguments**

object                  A data structure as returned by [readRprof](#).  
 ...                    further arguments passed to or from other methods.

**Value**

None.

**References**

<http://sintro.r-forge.r-project.org/>

**See Also**

[summaryRprof](#)

**Examples**

```
## Not run:
## Rprof() is not available on all platforms
profininterval <- 0.001
simruns <- 100

n <- 10000
x <- runif(n)
y0 <- 2 + 3 * x

sinknull <- textConnection(NULL, "w"); sink(sinknull)
Rprof(tmp <- tempfile(), interval = profinterval)
for (i in 1:simruns) {y <- y0 + rnorm(n); xxx<- summary(lm(y~x))}
Rprof()

Rprof_out <- readProf(tmp)

unlink(tmp)
sink(); close(sinknull)

summary(Rprof_out)

## End(Not run)
```

---

summary_terminals	<i>Tabulate leaf nodes</i>
-------------------	----------------------------

---

**Description**

Tabulate leaf nodes

**Usage**

```
summary_terminals(x)
```

**Arguments**

x                      an sprof data structure.

**Value**

A table of frequencies, bystack.

**Author(s)**

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

**Examples**

```
data(sprof01lm)
summary_terminals(sprof01lm)
```

---

writeRprof	<i>Write profile data</i>
------------	---------------------------

---

**Description**

Write a profile data file from a sprof data structure.

**Usage**

```
writeRprof(sprof, filename = "Rprof.Out")
```

**Arguments**

sprof	a data structure from package sprof
filename	The file to be used for exporting the profiling results.

**Details**

writeRprof only writes the stack entries for the profile.

This can be used to export information after preprocessing with sprof to some package designed for Rprof output.

**Value**

An invisible list with the profile entries, headers removed.

**Note**

See the vignette for in-context explanations.

Displays of the graph structure are given in the vignette.

**Author(s)**

Günther Sawitzki <gsawitzki@r-forge.r-project.org>

**References**

<http://sintro.r-forge.r-project.org/>

**Examples**

```
## Not run:  
data(sprof01lm)  
writeRprof(sprof01lm)  
  
## End(Not run)
```

# Index

- \*Topic **datasets**
    - sprof01lm, [16](#)
  - \*Topic **hplot**
    - plot\_nodes, [5](#)
    - plot\_profiles, [7](#)
    - plot\_stacks, [8](#)
  - \*Topic **list**
    - list.as.matrix, [3](#)
  - \*Topic **manip**
    - rrle, [13](#)
    - rrleb, [14](#)
  - \*Topic **matrix**
    - list.as.matrix, [3](#)
    - rrle, [13](#)
    - rrleb, [14](#)
  - \*Topic **methods**
    - plot.sprof, [4](#)
    - print.sprof, [9](#)
    - summary.sprof, [17](#)
  - \*Topic **utilities**
    - list.as.matrix, [3](#)
    - plot.sprof, [4](#)
    - print.sprof, [9](#)
    - readRprof, [11](#)
    - sampleRprof, [15](#)
    - sprof-package, [2](#)
    - summary.sprof, [17](#)
    - writeRprof, [19](#)
  - \*Topic **util**
    - adjacency, [2](#)
    - print\_profiles, [10](#)
    - profiles\_matrix, [11](#)
    - rrle, [13](#)
    - rrleb, [14](#)
    - stackstoadj, [16](#)
    - str\_prof, [17](#)
    - summary\_terminals, [18](#)
- adjacency, [2](#)
- flatProfile, [13](#)
- list.as.matrix, [3](#)
- plot.sprof, [4](#), [6](#), [8–10](#)
- plot\_nodes, [4](#), [5](#)
- plot\_profiles, [4](#), [7](#)
- plot\_stacks, [4](#), [8](#)
- print.sprof, [9](#)
- print\_nodes (print.sprof), [9](#)
- print\_profiles, [10](#)
- print\_stacks (print.sprof), [9](#)
- profiles\_matrix, [11](#)
- readRprof, [2](#), [4](#), [10](#), [11](#), [17](#)
- rle, [15](#)
- Rprof, [2](#), [11](#), [13](#)
- Rprofmem, [13](#)
- rrle, [13](#)
- rrleb, [14](#), [14](#)
- sampleRprof, [2](#), [15](#)
- sprof (sprof-package), [2](#)
- sprof-package, [2](#)
- sprof01lm, [16](#)
- stackstoadj, [16](#)
- str\_prof, [17](#)
- summary.sprof, [17](#)
- summary\_nodes (summary.sprof), [17](#)
- summary\_profiles (summary.sprof), [17](#)
- summary\_stacks (summary.sprof), [17](#)
- summary\_terminals, [18](#)
- summaryRprof, [4](#), [10](#), [12](#), [13](#), [18](#)
- tracemem, [13](#)
- writeRprof, [19](#)