

# Package ‘spam64’

January 5, 2022

**Type** Package

**Title** 64-Bit Extension of the SPARse Matrix R Package 'spam'

**Version** 2.8-0

**Date** 2022-01-05

**Description** Provides the Fortran code of the R package 'spam' with 64-bit integers. Loading this package together with the R package spam enables the sparse matrix class spam to handle huge sparse matrices with more than  $2^{31}-1$  non-zero elements. Documentation is provided in Gerber, Moesinger and Furrer (2017) <[doi:10.1016/j.cageo.2016.11.015](https://doi.org/10.1016/j.cageo.2016.11.015)>.

**Suggests** spam (== 2.8-0)

**License** LGPL-2 | BSD\_3\_clause + file LICENSE

**URL** <https://git.math.uzh.ch/reinhard.furrer/spam>

**NeedsCompilation** yes

**Author** Reinhard Furrer [aut, cre] (<<https://orcid.org/0000-0002-6319-2332>>), Florian Gerber [aut] (<<https://orcid.org/0000-0001-8545-5263>>), Roman Flury [aut] (<<https://orcid.org/0000-0002-0349-8698>>), Daniel Gerber [ctb], Kaspar Moesinger [ctb], Youcef Saad [ctb] (SPARSEKIT <http://www-users.cs.umn.edu/~saad/software/SPARSKIT/>), Esmond G. Ng [ctb] (Fortran Cholesky routines), Barry W. Peyton [ctb] (Fortran Cholesky routines), Joseph W.H. Liu [ctb] (Fortran Cholesky routines), Alan D. George [ctb] (Fortran Cholesky routines), Lehoucq B. Rich [ctb] (ARPACK), Maschhoff Kristi [ctb] (ARPACK), Sorensen C. Danny [ctb] (ARPACK), Yang Chao [ctb] (ARPACK)

**Maintainer** Reinhard Furrer <[reinhard.furrer@math.uzh.ch](mailto:reinhard.furrer@math.uzh.ch)>

**Repository** CRAN

**Date/Publication** 2022-01-05 20:50:02 UTC

## R topics documented:

spam64-package . . . . . 2

**Index** 4

spam64-package      *64-bit extension for the SPArse Matrix Package spam*

### Description

Provides the Fortran code of the R package **spam** with 64-bit integers. Loading this package together with the R package **spam** enables the sparse matrix class `spam` to handle huge sparse matrices with more than  $2^{31}-1$  non-zero elements.

### Note

It is intended to use **spam64** together with **spam**. To avoid issues on 32-bit platforms we did not link the packages **spam** and **spam64** using dependencies.

Conversion between the structures happens when calling low-level functions and for some other selected operations.

Some **spam64** functions have been successfully tested with 64-bit matrices. However, we expect that some functions of **spam** do not work with 64-bit matrices (yet). Please do not hesitate to contact us via email or <https://git.math.uzh.ch/reinhard.furrer/spam> in case you would like to use a `spam` function with 64-bit matrices that is not working properly in the current version.

### Author(s)

Reinhard Furrer [aut, cre], Florian Gerber [aut], Roman Flury [aut] and many contributors.

### References

F. Gerber, K. Moesinger, R. Furrer (2017), Extending R packages to support 64-bit compiled code: An illustration with `spam64` and GIMMS NDVI3g data, *Computer & Geoscience* 104, 109-119, <https://doi.org/10.1016/j.cageo.2016.11.015>.

`spam64` uses the R package `dotCall64` to call compiled code: F. Gerber, K. Moesinger, R. Furrer (2018), `dotCall64`: An R package providing an efficient interface to compiled C, C++, and Fortran code supporting long vectors. *SoftwareX*, 7, 217-221, <https://doi.org/10.1016/j.softx.2018.06.002>.

### Examples

```
library("spam")
library("spam64")

tiny <- spam(1)
pad(tiny) <- c(3,2^32)
tiny
```

```
str(tiny)      # tiny matrix big time

print(A <- spam_random(3))
options(spam.force64 = TRUE)  # forcing 64-bit structure
print( B <- spam_random(3))
A+B

options(spam.force64 = FALSE)
B      # No operations, structure is preserved
A+B    # Lowlevel operation, structure is adapted
```

# Index

- \* **documentation**

- spam64-package, [2](#)

- \* **package**

- spam64-package, [2](#)

SPAM64 (spam64-package), [2](#)

Spam64 (spam64-package), [2](#)

spam64 (spam64-package), [2](#)

spam64-package, [2](#)