

# Package ‘bindrcpp’

December 12, 2023

**Title** An 'Rcpp' Interface to Active Bindings

**Version** 0.2.3

**Date** 2023-12-11

**Description** Provides an easy way to fill an environment with active bindings that call a C++ function.

**License** MIT + file LICENSE

**URL** <https://github.com/krlmlr/bindrcpp>,  
<https://krlmlr.github.io/bindrcpp/>

**BugReports** <https://github.com/krlmlr/bindrcpp/issues>

**Imports** bindr (>= 0.1.1), Rcpp (>= 0.12.16)

**Suggests** testthat

**LinkingTo** plogr, Rcpp

**Config/Needs/check** RcppCore/Rcpp

**Encoding** UTF-8

**RoxygenNote** 7.2.3

**NeedsCompilation** yes

**Author** Kirill Müller [aut, cre] (<<https://orcid.org/0000-0002-1416-3412>>),  
RStudio [cph]

**Maintainer** Kirill Müller <[kirill@cynkra.com](mailto:kirill@cynkra.com)>

**Repository** CRAN

**Date/Publication** 2023-12-12 04:20:02 UTC

## R topics documented:

bindrcpp-package . . . . .	2
<b>Index</b>	<b>3</b>

**Description**

Provides an easy way to fill an environment with active bindings that call a C++ function.

**Details**

Use `LinkingTo: bindrcpp` in `DESCRIPTION` and `#include <bindrcpp.h>` in your C++ headers and/or modules to access the C++ functions provided by this package:

- `create_env_string()` creates an environment with active bindings, with names given as a character vector. Access of these bindings triggers a call to a C++ function with a fixed signature (`GETTER_FUNC_STRING`); this call contains the name of the binding (as character) and an arbitrary payload (`PAYLOAD`, essentially a wrapped `void*`).
- `create_env_symbol()` is similar, the callback function accepts the name of the binding as symbol instead of character (`GETTER_FUNC_SYMBOL`).
- `populate_env_string()` and `populate_env_symbol()` populate an existing environment instead of creating a new one.

**Author(s)**

**Maintainer:** Kirill Müller <krlmlr+r@mailbox.org> ([ORCID](#))

Other contributors:

- RStudio [copyright holder]

**See Also**

Useful links:

- <https://github.com/krlmlr/bindrcpp>
- <https://krlmlr.github.io/bindrcpp/>
- Report bugs at <https://github.com/krlmlr/bindrcpp/issues>

# Index

[bindrcpp \(bindrcpp-package\), 2](#)  
[bindrcpp-package, 2](#)