Package 'rdocdump'

June 18, 2025

Title Dump 'R' Package Source, Documentation, and Vignettes into One
File

Version 0.1.0

Description Dump source code, documentation and vignettes of an 'R'
package into a single file. Supports installed packages, tar.gz
archives, and package source directories. If the package is not
installed, only its source is automatically downloaded from CRAN for
processing. The output is a single plain text file or a character
vector, which is useful to ingest complete package documentation and
source into a large language model (LLM) or pass it further to other
tools, such as 'ragnar' <https://github.com/tidyverse/ragnar> to
create a Retrieval-Augmented Generation (RAG) workflow.

License MIT + file LICENSE

URL <https://github.com/e-kotov/rdocdump>,
<https://www.ekotov.pro/rdocdump/>

BugReports <https://github.com/e-kotov/rdocdump/issues>

Suggests curl, quarto, testthat (>= 3.0.0), withr

VignetteBuilder quarto

Config/testthat/edition 3

Encoding UTF-8

Language en

RoxygenNote 7.3.2

NeedsCompilation no

Author Egor Kotov [aut, cre, cph] (ORCID:
<https://orcid.org/0000-0001-6690-5345>)

Maintainer Egor Kotov <kotov.egor@gmail.com>

Repository CRAN

Date/Publication 2025-06-18 11:50:06 UTC

Contents

---

rdd_extract_code   *Extract R Source Code from a Package*

Description

This function extracts the R source code from a package. For installed packages, it retrieves the package namespace and deparses all functions found in the package. For package source directories or archives (non-installed packages), it reads all `.R` files from the R directory and, optionally, from the `tests` directory. Optionally, it can include roxygen2 documentation from these files.

Usage

```
rdd_extract_code(
  pkg,
  file = NULL,
  include_tests = FALSE,
  include_roxygen = FALSE,
  force_fetch = FALSE,
  cache_path = getOption("rdocdump.cache_path"),
  keep_files = "none",
  repos = getOption("rdocdump.repos", getOption("repos"))
)
```

Arguments

pkg   A `character` string specifying the package. This can be:

- an installed package name,
- a full path to a package source directory,
- a full path to a package archive file (tar.gz), or
- a package name not installed (which will then be downloaded from CRAN).

file   Optional. Save path for the output text file. If set, the function will return the path to the file instead of the combined text. Defaults to `NULL`.

include_tests   `logical`. If `TRUE`, for non-installed packages, the function will also include R source code from the `tests` directory. Defaults to `FALSE`.

include_roxygen

                `logical`. If `TRUE`, roxygen2 documentation lines (lines starting with "#'") from R files will be included in the output. Defaults to `FALSE`.

force_fetch	logical. If TRUE, the package source will be fetched from CRAN even if the package is installed locally. Default is FALSE.
cache_path	A character string specifying the directory to use as a cache. Defaults to the value of getOption("rdocdump.cache_path").
keep_files	A character value controlling whether temporary files should be kept. Possible values are: • "none": Delete both the tar.gz archive and the extracted files (default). • "tgz": Keep only the tar.gz archive. • "extracted": Keep only the extracted files. • "both": Keep both the tar.gz archive and the extracted files.
repos	A character vector of repository URLs. By default, it uses the value of getOption("rdocdump.repos") which sets the repository URLs to the default R repositories and is itself set to c("CRAN" = "https://cloud.r-project.org") on package load to prevent accidental downloads of pre-built packages from Posit Package Manager and R Universe.

Value

A single string containing the combined R source code (and, optionally, roxygen2 documentation) from the package.

Examples

```
# Extract only R source code (excluding roxygen2 documentation) from an installed package.
code <- rdd_extract_code("splines")
cat(substr(code, 1, 1000))


# Extract R source code including roxygen2 documentation from a package source directory.

# set cache directory for `rdocdump`
rdd_set_cache_path(paste0(tempdir(), "/rdocdump_cache"))

local({
 code_with_roxygen <- rdd_extract_code(
  "ini",
  include_roxygen = TRUE,
  force_fetch = TRUE,
  repos = c("CRAN" = "https://cran.r-project.org")
)
 cat(substr(code_with_roxygen, 1, 1000))
})

# Extract R source code from a package source directory,
# including test files but excluding roxygen2 docs.
local({
 code_with_tests <- rdd_extract_code(
  "ini",
  include_roxygen = TRUE,
  include_tests = TRUE,
  force_fetch = TRUE,
```

```
  repos = c("CRAN" = "https://cran.r-project.org")
)
 cat(substr(code_with_tests, 1, 1000))
})
# clean cache directory
unlink(getOption("rdocdump.cache_path"), recursive = TRUE, force = TRUE)
```

---

rdd_get_repos   *Get Current* rdocdump *Repository Options*

Description

This function returns the current repository URLs used by rdocdump. The default is set to the
CRAN repository at "https://cloud.r-project.org". This does not affect the repositories used by
install.packages() in your current R session and/or project. To set repository options, use
[rdd_set_repos](#).

Usage

```
rdd_get_repos()
```

Value

A character vector of repository URLs.

Examples

```
# Get current rdocdump repository options
rdd_get_repos()
```

---

rdd_set_cache_path   *Set* rdocdump *Cache Path in the Current R Session*

Description

This function sets the cache path used by rdocdump to store temporary files (downloaded tar.gz
archives and/or extracted directories) for the current R session. The cache path is stored in the option
"rdocdump.cache_path", which can be checked with getOption("rdocdump.cache_path"). The
path is created if it does not exist.

Usage

```
rdd_set_cache_path(path)
```

Arguments

path   A `character` string specifying the directory to be used as the cache path.

Value

Invisibly returns the new cache path.

Examples

```
# set cache directory for `rdocdump`
rdd_set_cache_path(paste0(tempdir(), "/rdocdump_cache"))
# default cache directory
unlink(getOption("rdocdump.cache_path"), recursive = TRUE)
```

---

rdd_set_repos *Set* rdocdump *Repository Options*

Description

This function sets the package repository URLs used by rdocdump when fetching package sources. May be useful for setting custom repositories or mirrors. This does not affect the repositories used by install.packages() in your current R session and/or project.

Usage

```
rdd_set_repos(repos)
```

Arguments

repos   A character vector of repository URLs.

Value

Invisibly returns the new repository URLs.

Examples

```
# Set rdocdump repository options
rdd_set_repos(c("CRAN" = "https://cloud.r-project.org"))
```

---

rdd_to_txt *Dump Package Source, Documentationm and Vignettes into Plain Text*

Description

This function produces a single text output for an R package by processing its documentation (Rd files from the package source or the documentation from already installed packages), vignettes, and/or R source code.

Usage

```
rdd_to_txt(
  pkg,
  file = NULL,
  content = "all",
  force_fetch = FALSE,
  keep_files = "none",
  cache_path = getOption("rdocdump.cache_path"),
  repos = getOption("rdocdump.repos", getOption("repos"))
)
```

Arguments

pkg   A `character` string specifying the package. This can be:

- an installed package name,
- a full path to a package source directory,
- a full path to a package archive file (tar.gz), or
- a package name not installed (which will then be downloaded from CRAN).

file   Optional. Save path for the output text file. If set, the function will return the path to the file instead of the combined text. Defaults to `NULL`.

content   A character vector specifying which components to include in the output. Possible values are:

- `"all"`: Include Rd documentation, vignettes, and R source code (default).
- `"docs"`: Include only the Rd documentation.
- `"vignettes"`: Include only the vignettes.
- `"code"`: Include only the R source code. When extracting code for non-installed packages, the function will not include roxygen2 documentation, as the documentation can be imported from the Rd files. If you want to extract the R source code with the roxygen2 documentation, use [rdd_extract_code](#) and set include_roxygen to TRUE.

You can specify multiple options (e.g., `c("docs", "code")` to include both documentation and source code).

force_fetch   `logical`. If `TRUE`, the package source will be fetched from CRAN as a tar.gz archive even if the package is already installed locally. Default is `FALSE`.

keep_files   A character value controlling whether temporary files should be kept. Possible
                    values are:

- "none": Delete both the tar.gz archive and the extracted files (default).
- "tgz": Keep only the tar.gz archive.
- "extracted": Keep only the extracted files.
- "both": Keep both the tar.gz archive and the extracted files.

cache_path   A character string specifying the directory where kept temporary files will be
                    stored. By default, it uses the value of getOption("rdocdump.cache_path")
                    which sets the cache directory to the temporary directory of the current R ses-
                    sion.

repos   A character vector of repository URLs. By default, it uses the value of getOption("rdocdump.repos")
                    which sets the repository URLs to the default R repositories and is itself set
                    to c("CRAN" = "https://cloud.r-project.org") on package load to prevent
                    accidental downloads of pre-built packages from Posit Package Manager and R
                    Universe.

Value

A single string containing the combined package documentation, vignettes, and/or code as specified
by the content argument. If the file argument is set, returns the path to the file.

Examples

```
# Extract documentation for built-in `stats` package (both docs and vignettes).
docs <- rdd_to_txt("splines")
cat(substr(docs, 1, 500))


# set cache directory for `rdocdump`
rdd_set_cache_path(paste0(tempdir(), "/rdocdump_cache"))

# Extract only documentation for rJavaEnv by downloading its source from CRAN
docs <- rdd_to_txt(
  "rJavaEnv",
  force_fetch = TRUE,
  content = "docs",
  repos = c("CRAN" = "https://cran.r-project.org")
)
lines <- unlist(strsplit(docs, "\n"))
# Print the first 3 lines
cat(head(lines, 3), sep = "\n")
# Print the last 3 lines
cat(tail(lines, 3), sep = "\n")

# clean cache directory
unlink(getOption("rdocdump.cache_path"), recursive = TRUE, force = TRUE)
```

Index