

Package ‘releaser’

November 6, 2025

Type Package

Title Help with Preparing a New Version of an R Package

Version 1.0.0

Description Helps to prepare a release. Before releasing an R package it is important to update the DESCRIPTION file and the changelog. This package prepares these files and also updates the versions according to the branches. It relies heavily on the ‘desc’ packages.

License MIT + file LICENSE

URL <https://github.com/TanguyBarthelemy/releaser>,
<https://tanguybarthelemy.github.io/releaser/>

BugReports <https://github.com/TanguyBarthelemy/releaser/issues>

Encoding UTF-8

RoxygenNote 7.3.3

Depends R (>= 4.1)

Imports gh, desc, base64enc

Suggests rlang, spelling, testthat (>= 3.0.0)

Config/testthat/edition 3

Language en-GB

NeedsCompilation no

Author Tanguy Barthelemy [aut, cre, art]

Maintainer Tanguy Barthelemy <tanguy.barthelemy@insee.fr>

Repository CRAN

Date/Publication 2025-11-06 10:10:02 UTC

Contents

| | |
|--|---|
| change_remotes_field | 2 |
| get_changes | 3 |
| get_different_future_version | 3 |
| get_github_branches | 4 |

| | |
|-----------------------------------|---|
| get_latest_version | 5 |
| set_latest_deps_version | 5 |
| update_news_md | 6 |

change_remotes_field *Change the Remotes field in DESCRIPTION*

Description

Update the Remotes field of a package DESCRIPTION file so that dependencies point to specific development targets (develop, snapshot, or main).

Usage

```
change_remotes_field(
  path,
  target = c("develop", "snapshot", "main"),
  verbose = TRUE
)
```

Arguments

| | | |
|---------|-------------|---|
| path | [character] | Path to the package root directory. |
| target | [character] | Target branch or type of remote: must be one of "develop", "snapshot", or "main". |
| verbose | [logical] | Whether to print current and new remote fields (default: TRUE). |

Value

Invisibly returns the new vector of remote specifications (character).

Examples

```
path_rjd3workspace <- file.path(tempdir(), "rjd3workspace")
file.copy(
  from = system.file("rjd3workspace", package = "releaser"),
  to = dirname(path_rjd3workspace),
  recursive = TRUE
)
change_remotes_field(path = path_rjd3workspace, target = "develop")
```

| | |
|-------------|--|
| get_changes | <i>Extract changelog entries for a given version</i> |
|-------------|--|

Description

Extracts the section of NEWS.md corresponding to a given version.

Usage

```
get_changes(path, version_number, verbose = TRUE)
```

Arguments

| | | |
|----------------|-------------|---|
| path | [character] | Path to the package root directory. |
| version_number | [character] | Current version number string (e.g. "1.2.3"). |
| verbose | [logical] | Whether to print current and new remote fields (default: TRUE). |

Value

A character string containing the formatted changelog for the given version.

Examples

```
path_rjd3workspace <- system.file("rjd3workspace", package = "releaser")  
  
get_changes(path = path_rjd3workspace, version_number = "Unreleased")  
get_changes(path = path_rjd3workspace, version_number = "3.2.4")  
get_changes(path = path_rjd3workspace, version_number = "3.5.1")
```

| | |
|------------------------------|--|
| get_different_future_version | <i>Compute possible future semantic versions</i> |
|------------------------------|--|

Description

Given a current package version, compute the potential next patch, minor, and major versions following semantic versioning rules.

Usage

```
get_different_future_version(version_number, verbose = TRUE)
```

Arguments

| | | |
|----------------|-------------|---|
| version_number | [character] | Current version number string (e.g. "1.2.3"). |
| verbose | [logical] | Whether to print current and new remote fields (default: TRUE). |

Value

A named character vector with:

- `current_version` – the input version,
- `future_patch_version` – next patch version,
- `future_minor_version` – next minor version,
- `future_major_version` – next major version.

Examples

```
get_different_future_version("1.2.3")
```

`get_github_branches` *List GitHub repository branches*

Description

Retrieve all branch names from a GitHub repository.

Usage

```
get_github_branches(  
  gh_repo = file.path("rjdverse", "rjd3toolkit"),  
  verbose = TRUE  
)
```

Arguments

| | |
|----------------------|--|
| <code>gh_repo</code> | <code>[character]</code> GitHub repository in the format "owner/repo". |
| <code>verbose</code> | <code>[logical]</code> Whether to print current and new remote fields (default: TRUE). |

Value

A character vector with branch names.

Examples

```
get_github_branches("r-lib/usesthis")
```

```
get_latest_version      Get latest GitHub release version
```

Description

Retrieve the version number of the latest GitHub release for a repository and optionally print versions found across all branches.

Usage

```
get_latest_version(  
  gh_repo = file.path("rjdverse", "rjd3toolkit"),  
  verbose = TRUE  
)
```

Arguments

| | | |
|---------|-------------|---|
| gh_repo | [character] | GitHub repository in the format "owner/repo". |
| verbose | [logical] | Whether to print current and new remote fields (default: TRUE). |

Value

A character string with the version of the latest release.

Examples

```
get_latest_version("r-lib/usethis")
```

```
set_latest_deps_version  
      Set latest versions for rjd3* dependencies
```

Description

Update the DESCRIPTION file of a package so that all dependencies beginning with "rjd3" require the latest released version from GitHub.

Usage

```
set_latest_deps_version(path, verbose = TRUE)
```

Arguments

| | | |
|---------|-------------|---|
| path | [character] | Path to the package root directory. |
| verbose | [logical] | Whether to print current and new remote fields (default: TRUE). |

Value

Invisibly updates the DESCRIPTION file in place.

Examples

```
path_rjd3workspace <- file.path(tempdir(), "rjd3workspace")
file.copy(
  from = system.file("rjd3workspace", package = "releaser"),
  to = dirname(path_rjd3workspace),
  recursive = TRUE
)
set_latest_deps_version(path = path_rjd3workspace)
```

`update_news_md`

Update NEWS.md for a new release

Description

Modify the NEWS.md file of a package to replace the "Unreleased" section with a new version heading and update GitHub comparison links.

Usage

```
update_news_md(path, version_number, verbose = TRUE)
```

Arguments

| | |
|-----------------------------|--|
| <code>path</code> | <code>[character]</code> Path to the package root directory. |
| <code>version_number</code> | <code>[character]</code> Current version number string (e.g. "1.2.3"). |
| <code>verbose</code> | <code>[logical]</code> Whether to print current and new remote fields (default: TRUE). |

Details

The argument `version_number` is the new version number to update the changelog.

Value

Invisibly returns TRUE if the file was successfully updated.

Examples

```
path_rjd3workspace <- file.path(tempdir(), "rjd3workspace")
file.copy(
  from = system.file("rjd3workspace", package = "releaser"),
  to = dirname(path_rjd3workspace),
  recursive = TRUE
)
update_news_md(path = path_rjd3workspace, version_number = "1.2.3")
```

Index

change_remotes_field, 2
character, 2–6

get_changes, 3
get_different_future_version, 3
get_github_branches, 4
get_latest_version, 5

logical, 2–6

set_latest_deps_version, 5

update_news_md, 6