

# Package ‘rworkflows’

January 5, 2023

**Type** Package

**Title** Test, Document, Containerise, and Deploy R Packages

**Version** 0.99.5

**Description** Continuous integration for R packages.  
Automates testing, documentation website building,  
and containerised deployment.

**URL** <https://github.com/neurogenomics/rworkflows>, <https://cran.r-project.org/web/packages/rworkflows/index.html>

**BugReports** <https://github.com/neurogenomics/rworkflows/issues>

**Encoding** UTF-8

**Depends** R (>= 4.1)

**biocViews** Genetics, FunctionalGenomics, SystemsBiology

**Imports** stats,  
here,  
yaml,  
utils,  
desc,  
badger

**Suggests** markdown,  
rmarkdown,  
magick,  
remotes,  
knitr,  
BiocStyle,  
covr,  
testthat (>= 3.0.0),  
hexSticker,  
htmltools,  
data.table

**RoxygenNote** 7.2.3

**VignetteBuilder** knitr

**License** GPL-3

**Config/testthat/edition** 3

## R topics documented:

bioc_r_versions . . . . .	2
construct_runners . . . . .	3
use_badges . . . . .	4
use_dockerfile . . . . .	5
use_issue_template . . . . .	6
use_readme . . . . .	7
use_vignette_docker . . . . .	8
use_vignette_getstarted . . . . .	9
use_workflow . . . . .	10
<b>Index</b>	<b>12</b>

---

bioc_r_versions	<i>Bioconductor / R versions</i>
-----------------	----------------------------------

---

### Description

Get the respective version of R for a given version of **Bioconductor**.

### Usage

```
bioc_r_versions(bioc_version = NULL)
```

### Arguments

**bioc\_version** Version of Bioc to return info for. Can be:

- "devel" Get the current development version of Bioc.
- "release" Get the current release version of Bioc.
- <numeric> A specific Bioc version number (e.g. 3.16).
- NULL Return info for all Bioc versions.

### Value

Named list of Bioc/R versions

### Examples

```
ver <- bioc_r_versions(bioc_version="devel")
```

---

construct_runners	<i>Construct runners</i>
-------------------	--------------------------

---

## Description

Construct runner configurations across multiple Operating Systems (OS) for GitHub Actions workflow.

## Usage

```
construct_runners(
  os = c("ubuntu-latest", "macOS-latest", "windows-latest"),
  bioc = stats::setNames(list("devel", "release", "release"), os),
  r = stats::setNames(list("auto", "auto", "auto"), os),
  cont = stats::setNames(list(paste0("bioconductor/bioconductor_docker:", bioc[[1]]),
    NULL, NULL), os),
  rspm = stats::setNames(list(paste0("https://packagemanager.rstudio.com/",
    "cran/__linux__/focal/release"), NULL, NULL), os),
  versions_explicit = FALSE
)
```

## Arguments

os	Which OS to launch GitHub Actions on.
bioc	Which Bioconductor version to use on each OS. See <a href="#">bioc_r_versions</a> documentation for all options.
r	Which R version to use on each OS.
cont	Which Docker container to use on each OS (NULL means no container will be used for that OS). See <a href="#">here</a> for a list of all official Bioconductor Docker container versions.
rspm	Which R repository manager to use on each OS (NULL means the default will be used for that OS).
versions_explicit	Specify R/Bioc versions explicitly (e.g. r: 4.2.0, bioc: 3.16) as opposed to flexibly (e.g. r: "latest", bioc: "release").

## Value

Named list of configurations for each runner OS.

## Examples

```
runners <- construct_runners()
```

use\_badges

*Use badges***Description**

Create one or more badges showing the status of your R package. Uses the package **badger**.

**Usage**

```
use_badges(
  add_hex = TRUE,
  add_actions = "rworkflows",
  add_doi = NULL,
  add_github_version = TRUE,
  add_commit = TRUE,
  add_code_size = TRUE,
  add_codecov = TRUE,
  add_license = TRUE,
  add_authors = TRUE,
  add_bioc_release = FALSE,
  add_bioc_download_month = FALSE,
  add_bioc_download_total = FALSE,
  add_bioc_download_rank = FALSE,
  add_cran_release = FALSE,
  add_cran_checks = FALSE,
  add_cran_download_month = FALSE,
  add_cran_download_total = FALSE,
  branch = "master",
  as_list = FALSE,
  sep = "\n",
  hex_height = 600,
  colors = list(github = "black", bioc = "green", cran = "blue", default = "blue"),
  verbose = TRUE
)
```

**Arguments**

add_hex	Add a hex sticker. If add_hex=TRUE, will assume the sticker is located at the following relative path: "inst/hex/hex.png". If add_hex is a character string, this will instead be used as the relative hex path (e.g. "/images/mysticker.png").
add_actions	The name of one or more GitHub Actions to show the status for with <a href="#">badge_github_actions</a> (e.g. c("rworkflows", "rworkflows_static")).
add_doi	Add the DOI of a given package or publication associated with the package. Must be provided as a character string.
add_github_version	Add package version with <a href="#">badge_github_version</a> .
add_commit	Add the last GitHub repo commit date with <a href="#">badge_last_commit</a> .
add_code_size	Add code size with <a href="#">badge_code_size</a> .
add_codecov	Add CodeCov status with <a href="#">badge_codecov</a> .

add_license	Add license info with <a href="#">badge_license</a> .
add_authors	Add author names inferred from the DESCRIPTION file.
add_bioc_release	Add Bioc release version with <a href="#">badge_bioc_release</a> .
add_bioc_download_month	Add the number of Bioc downloads last month <a href="#">badge_bioc_download</a> .
add_bioc_download_total	Add the number of Bioc downloads total <a href="#">badge_bioc_download</a> .
add_bioc_download_rank	Add the download rank of the package on Bioc <a href="#">badge_bioc_download_rank</a> .
add_cran_release	Add Bioc release version with <a href="#">badge_cran_release</a> .
add_cran_checks	Add whether package is passing all checks on CRAN with <a href="#">badge_cran_checks</a> .
add_cran_download_month	Add the number of CRAN downloads last month <a href="#">badge_cran_download</a> .
add_cran_download_total	Add the number of CRAN downloads total <a href="#">badge_cran_download</a> .
branch	Name of the GitHub repository branch to use.
as_list	Return the header as a named list (TRUE), or a collapsed text string (default: FALSE).
sep	Character to separate each item in the list with using <a href="#">paste</a> .
hex_height	Height of the hex sticker in pixels (when add_hex=TRUE).
colors	Colors to assign to each group of badges (when possible).
verbose	Print messages.

### Value

A named list of selected badges in markdown format.

### Examples

```
## Causes issues bc examples can't find the the DESCRIPTION file.
## Not run:
rworkflows::use_badges()

## End(Not run)
```

---

use\_dockerfile

*Use Dockerfile*

---

### Description

Creates a Docker file to be used with the GitHub Actions (GHA) workflows distributed by **rworkflows**.

**Usage**

```
use_dockerfile(
  save_dir = here::here(),
  path = file.path(save_dir, "Dockerfile"),
  force_new = FALSE,
  show = FALSE,
  verbose = TRUE
)
```

**Arguments**

save_dir	Directory to save the Docker file to.
path	Path to the Docker file.
force_new	If a Docker file already exists, overwrite it (default: FALSE).
show	Print the contents of the Docker file in the R console.
verbose	Print messages.

**Value**

Path to Docker file.

**Examples**

```
path <- use_dockerfile(save_dir=tempdir())
```

---

use_issue_template	<i>Use Issue Template</i>
--------------------	---------------------------

---

**Description**

Creates one or more Issue Templates to be used in a GitHub repository.

**Usage**

```
use_issue_template(
  templates = c("bug_report.md", "feature_request.md"),
  save_dir = here::here(".github", "ISSUE_TEMPLATE"),
  path = file.path(save_dir, templates),
  force_new = FALSE,
  show = FALSE,
  verbose = TRUE
)
```

**Arguments**

templates	The names of templates to be used.
save_dir	Directory to save the Docker file to.
path	Path to the Docker file.
force_new	If a Docker file already exists, overwrite it (default: FALSE).
show	Print the contents of the Docker file in the R console.
verbose	Print messages.

**Value**

Path to Issue Templates.

**Examples**

```
path <- use_issue_template(save_dir=tempdir())
```

---

use_readme	<i>Use README</i>
------------	-------------------

---

**Description**

Creates an rmarkdown README file that autofills using metadata from the R package *DESCRIPTION* file.

**Usage**

```
use_readme(  
  save_dir = here::here(),  
  path = file.path(save_dir, "README.Rmd"),  
  force_new = FALSE,  
  show = FALSE,  
  verbose = TRUE  
)
```

**Arguments**

save_dir	Directory to save the vignette file to.
path	Path to the vignette file.
force_new	If the file already exists, overwrite it (default: FALSE).
show	Print the contents of the vignette file in the R console.
verbose	Print messages.

**Value**

Path to README file.

**Examples**

```
## use default save_dir in practice  
path <- use_readme(save_dir = tempdir())
```

---

use\_vignette\_docker      *Use vignette: Docker*

---

## Description

Creates a vignette rmarkdown file demonstrates how to create a Docker/Singularity image from a container stored in **Dockerhub**.

## Usage

```
use_vignette_docker(  
  docker_org,  
  title = "Docker/Singularity Containers",  
  vignette_index_entry = "docker",  
  save_dir = here::here(),  
  path = file.path(save_dir, "vignettes", "docker.Rmd"),  
  force_new = FALSE,  
  show = FALSE,  
  verbose = TRUE  
)
```

## Arguments

docker_org	DockerHub organization name. Can simply be your Dockerhub username instead.
title	Title of vignette.
vignette_index_entry	Index entry of the vignette, which is used when creating the navigation bar in the <b>pkgdown</b> site.
save_dir	Directory to save the vignette file to.
path	Path to the vignette file.
force_new	If the file already exists, overwrite it (default: FALSE).
show	Print the contents of the vignette file in the R console.
verbose	Print messages.

## Value

Path to vignette file.

## Examples

```
path <- use_vignette_docker(docker_org = "neurogenomicslab",  
                           ## use default save_dir in practice  
                           save_dir = tempdir())
```



---

`use_vignette_getstarted`*Use vignette: Get started*

---

## Description

Creates a "Get started" rmarkdown vignette file.

## Usage

```
use_vignette_getstarted(  
  package,  
  title = "Get started",  
  vignette_index_entry = package,  
  save_dir = here::here(),  
  path = file.path(save_dir, "vignettes", paste0(package, ".Rmd")),  
  force_new = FALSE,  
  show = FALSE,  
  verbose = TRUE  
)
```

## Arguments

<code>package</code>	R package name.
<code>title</code>	Title of vignette.
<code>vignette_index_entry</code>	Index entry of the vignette, which is used when creating the navigation bar in the <b>pkgdown</b> site.
<code>save_dir</code>	Directory to save the vignette file to.
<code>path</code>	Path to the vignette file.
<code>force_new</code>	If the file already exists, overwrite it (default: FALSE).
<code>show</code>	Print the contents of the vignette file in the R console.
<code>verbose</code>	Print messages.

## Value

Path to vignette file.

## Examples

```
path <- use_vignette_getstarted(package = "mypackage",  
                                ## use default save_dir in practice  
                                save_dir = tempdir())
```

---

use_workflow	<i>Use GitHub Actions workflow</i>
--------------	------------------------------------

---

## Description

Create workflow that calls an **rworkflows** **GitHub Actions (GHA)**

## Usage

```
use_workflow(
  name = "rworkflows",
  tag = "@master",
  on = c("push", "pull_request"),
  branches = c("master", "main", "RELEASE_*"),
  runners = construct_runners(),
  run_bioccheck = FALSE,
  run_rcmdcheck = TRUE,
  as_cran = TRUE,
  run_vignettes = TRUE,
  has_testthat = TRUE,
  run_covr = TRUE,
  run_pkgdown = TRUE,
  has_runit = FALSE,
  has_latex = FALSE,
  run_docker = FALSE,
  github_token = "${ secrets.PAT_GITHUB }",
  docker_user = NULL,
  docker_org = docker_user,
  docker_token = "${ secrets.DOCKER_TOKEN }",
  cache_version = "cache-v1",
  enable_act = FALSE,
  save_dir = here::here(".github", "workflows"),
  return_path = TRUE,
  force_new = FALSE,
  preview = FALSE,
  verbose = TRUE
)
```

## Arguments

name	Workflow name. <ul style="list-style-type: none"> <li>"rworkflows" A short workflow script that calls the GitHub action from the GitHub Marketplace. The action is continually updated so users do not need to worry about maintaining it.</li> <li>"rworkflows_static" A longer workflow scripts that explicitly copies all steps from the <b>rworkflows</b> action into a static file. Users may need to update this file themselves over time, though this does allow for a fully customisable workflow.</li> </ul>
tag	Which version of the rworkflows action to use. Can be a branch name on the <b>GitHub repository</b> (e.g. "@master"), or a <b>Release Tag</b> (e.g. "@v1").

on	GitHub trigger conditions.
branches	GitHub trigger branches.
runners	Runner configurations for multiple Operating Systems (OS), including R versions, Bioc versions, and container sources. Can use the <a href="#">construct_runners</a> functions to assist in constructing customized runners configurations.
run_bioccheck	Run Bioconductor checks using <code>BiocCheck::BiocCheck()</code> . Must pass in order to continue workflow.
run_rcmdcheck	Run R CMD checks using <code>rcmdcheck::rcmdcheck()</code> . Must pass in order to continue workflow.
as_cran	When running R CMD checks, use the '-as-cran' flag to apply CRAN standards
run_vignettes	Build and check R package vignettes.
has_testthat	Run unit tests and report results.
run_covr	Run code coverage tests and publish results to codecov.
run_pkgdown	Knit the <i>README.Rmd</i> (if available), build documentation website, and deploy to <i>gh-pages</i> branch.
has_runit	Run R Unit tests.
has_latex	Install a suite of LaTeX dependencies used for rendering Sweave (.rnw) and other documentation files.
run_docker	Whether to build and push a Docker container to DockerHub.
github_token	Token for the repo. Can be passed in using secrets.PAT_GITHUB .
docker_user	DockerHub username.
docker_org	DockerHub organization name. Is the same as docker_user by default.
docker_token	DockerHub token.
cache_version	Name of the cache subdirectory to be used when reinstalling software in GHA.
enable_act	Whether to add extra lines to the yaml to enable local workflow checking with <code>act</code> .
save_dir	Directory to save workflow to.
return_path	Return the path to the saved <i>yaml</i> workflow file (default: TRUE), or return the <i>yaml</i> object directly.
force_new	If the GHA workflow yaml already exists, overwrite with new one (default: FALSE).
preview	Print the yaml file to the R console.
verbose	Print messages.

**Value**

Path or yaml object.

**Source**

Issue reading in "on:"/"y","n" elements.

Issue writing "on:" as "'as':"

**Examples**

```
path <- use_workflow(save_dir = file.path(tempdir(), ".github", "workflows"))
```

# Index

badge\_bioc\_download, [5](#)  
badge\_bioc\_download\_rank, [5](#)  
badge\_bioc\_release, [5](#)  
badge\_code\_size, [4](#)  
badge\_codecov, [4](#)  
badge\_cran\_checks, [5](#)  
badge\_cran\_download, [5](#)  
badge\_cran\_release, [5](#)  
badge\_github\_actions, [4](#)  
badge\_github\_version, [4](#)  
badge\_last\_commit, [4](#)  
badge\_license, [5](#)  
bioc\_r\_versions, [2](#), [3](#)  
  
construct\_runners, [3](#), [11](#)  
  
paste, [5](#)  
  
use\_badges, [4](#)  
use\_dockerfile, [5](#)  
use\_issue\_template, [6](#)  
use\_readme, [7](#)  
use\_vignette\_docker, [8](#)  
use\_vignette\_getstarted, [9](#)  
use\_workflow, [10](#)