

Package: universe (via r-universe)

September 20, 2026

Type Package

Title Tools for Working with R-universe <<https://r-universe.dev>>

Version 1.0.0.9000

Description Utilities to interact with the R-universe platform.

Includes functions to manage local package repositories, as well as API wrappers for retrieving data and metadata about packages in r-universe.

License Apache License (>= 2)

URL <https://docs.ropensci.org/universe/>,

<https://github.com/ropensci/universe>

BugReports <https://github.com/ropensci/universe/issues>

Depends R (>= 4.1.0)

Imports cli, curl, fs, httr2, jsonlite, rlang, withr

Suggests httptest2, knitr, rmarkdown, testthat (>= 3.0.0), usethis

VignetteBuilder knitr

Config/Needs/build moodymudskipper/devtag

Config/testthat/edition 3

Encoding UTF-8

Roxygen list(markdown = TRUE, roclets = c("`collate", "`rd",
 "`namespace", "`devtag::dev_roclet"))

RoxygenNote 7.3.3.9000

Config/pak/sysreqs cmake make libuv1-dev libssl-dev

Repository <https://ropensci.r-universe.dev>

Date/Publication 2026-02-02 14:54:53 UTC

RemoteUrl <https://github.com/ropensci/universe>

RemoteRef main

RemoteSha 5174f6f780ebb36c7bb1f1786079170d3db178e6

Contents

add	2
everyone	3
global_search	3
repo_snapshot	4
universe_all_packages	5
universe_ls	5
universe_one_package	6
universe_search	7
use_custom_manifest	7
use_universe_action	8
Index	9

add	<i>Enable a package repository from r-universe</i>
-----	--

Description

Adds r-universe package repositories to your options("repos") such that they are used by default in install.packages(). If the universe was already enabled, it will not be added again, hence it is harmless to call this function multiple times.

Usage

```
add(universe = "ropensci")

remove(universe)
```

Arguments

universe vector with name(s) of the universe(s), i.e. the subdomain part of https://ropensci.r-universe.dev.

Details

Note that changes to your options are not permanent. To automatically enable a repository for every R session, you can call this function in your `~/.Rprofile` script.

Value

the updated list of repositories

Examples

```
add("ropensci")
```

everyone	<i>All universes</i>
----------	----------------------

Description

All universes

Usage

```
everyone(type = c("all", "universes", "maintainers"))
```

Arguments

type Type of universe to query: "all", only organizations ("universe"), personal accounts ("maintainer").

Value

A character vector of all universes.

Examples

```
head(everyone())
```

global_search	<i>Search among all universes</i>
---------------	-----------------------------------

Description

Search among all universes

Usage

```
global_search(query, limit = 100L)
```

Arguments

query Query string. See [R-universe docs](#).
limit Number of results to return (integer of length 1)

Value

A list with query results. The total field indicates the total number of results and can be used as limit value in a second call.

Examples

```
global_search(query = '"weather data"', limit = 1)
global_search(query = 'needs:httr2', limit = 1)
```

repo_snapshot	<i>Download snapshot</i>
---------------	--------------------------

Description

Downloads a full copy (snapshot) of an R package repository.

Usage

```
repo_snapshot(
  repo,
  destdir = NULL,
  types = c("src", "win", "mac", "linux", "wasm", "docs"),
  bin_versions = r_version(),
  verbose = interactive()
)

r_version()
```

Arguments

repo	url of the cran-like repository to snapshot
destdir	path to directory where to save the snapshots
types	which files to include. Must be subset of "src", "win", "mac", "linux", "wasm", "docs".
bin_versions	vector with versions of R to download binary packages. Set to NULL to download all binaries available.
verbose	print some progress

Value

None, used for its side-effects.

Examples

```
repo_snapshot("https://jeroen.r-universe.dev")
```

universe_all_packages *Info on all packages in an universe*

Description

Info on all packages in an universe

Usage

```
universe_all_packages(universe, limit = 100L)
```

Arguments

universe	Name of the universe (character of length 1)
limit	Number of results to return (integer of length 1)

Value

A list with information on all packages in the universe.

See Also

Other universe: [universe_ls\(\)](#), [universe_one_package\(\)](#), [universe_search\(\)](#)

Examples

```
universe_all_packages("jeroen")
universe_all_packages("ropensci")
```

universe_ls *All packages in an universe*

Description

All packages in an universe

Usage

```
universe_ls(universe)
```

Arguments

universe	Name of the universe (character of length 1)
----------	--

Value

A character vector of all packages in the universe.

See Also

Other universe: [universe_all_packages\(\)](#), [universe_one_package\(\)](#), [universe_search\(\)](#)

Examples

```
universe_ls("jeroen")
universe_ls("ropensci")
```

universe_one_package	<i>Info on a single packages in an universe</i>
----------------------	---

Description

Info on a single packages in an universe

Usage

```
universe_one_package(universe, package)
```

Arguments

universe	Name of the universe (character of length 1)
package	Name of the package (character of length 1)

Value

A list with information on the package.

See Also

Other universe: [universe_all_packages\(\)](#), [universe_ls\(\)](#), [universe_search\(\)](#)

Examples

```
universe_one_package("jeroen", package = "curl")
```

universe_search	<i>Search within a single universe</i>
-----------------	--

Description

Search within a single universe

Usage

```
universe_search(universe, query, limit = 100L)
```

Arguments

universe	Name of the universe (character of length 1)
query	Query string. See R-universe docs .
limit	Number of results to return (integer of length 1)

Value

A list with query results. The `total` field indicates the total number of results and can be used as `limit` value in a second call.

See Also

Other universe: [universe_all_packages\(\)](#), [universe_ls\(\)](#), [universe_one_package\(\)](#)

Examples

```
universe_search("ropensci", query = '"weather data"')
universe_search("ropensci", query = 'needs:httr2')
```

use_custom_manifest	<i>Create packages.json for existing universe</i>
---------------------	---

Description

Create `packages.json` for existing universe

Usage

```
use_custom_manifest(universe, path = "packages.json")
```

Arguments

universe	Name of the universe, e.g. "jeroen"
path	Absolute path to which the JSON file could be saved.

Value

The path to the JSON file it created.

Examples

```
json_file <- withr::local_tempfile()
use_custom_manifest("jeroen", json_file)
file.edit(json_file)
```

use_universe_action	<i>Use R-universe GitHub Actions workflow</i>
---------------------	---

Description

Test the R-universe build workflow from your own GitHub repository

Usage

```
use_universe_action(universe = NULL, path = ".")
```

Arguments

universe	Universe to set the context to, by default the owner of the repository. This affects where R package dependencies are downloaded from (besides the default repositories).
path	Path to the local repository for which you want to use the workflow.

Value

The path to the workflow

Examples

```
temp_dir <- withr::local_tempdir()
workflow_path <- use_universe_action("maelle", temp_dir)
file.edit(workflow_path)
```

Index

* **universe**

- universe_all_packages, [5](#)
- universe_ls, [5](#)
- universe_one_package, [6](#)
- universe_search, [7](#)
- ~/.Rprofile, [2](#)

add, [2](#)

everyone, [3](#)

global_search, [3](#)

r_version (repo_snapshot), [4](#)

remove (add), [2](#)

repo_snapshot, [4](#)

universe_all_packages, [5](#), [6](#), [7](#)

universe_ls, [5](#), [5–7](#)

universe_one_package, [5](#), [6](#), [6](#), [7](#)

universe_search, [5](#), [6](#), [7](#)

use_custom_manifest, [7](#)

use_universe_action, [8](#)