

# Package: pactaverse (via r-universe)

October 30, 2024

**Title** Easily Install and Load the 'PACTA-verse'

**Version** 0.0.0.9000

**Description** The 'pactaverse' is a set of packages that work to help R users implement various functionality related to the PACTA open source project.

**License** MIT + file LICENSE

**URL** <https://rmi-pacta.github.io/pactaverse/>

**BugReports** <https://github.com/RMI-PACTA/pactaverse/issues>

**Imports** cli, dplyr, magrittr, pacta.data.scraping (>= 0.0.0.9000), pacta.data.validation (>= 0.0.0.9000), pacta.executive.summary (>= 0.0.1.9000), pacta.interactive.plot (>= 0.0.0.9000), pacta.portfolio.allocate (>= 0.0.0.9000), pacta.portfolio.audit (>= 0.0.0.9000), pacta.portfolio.import (>= 0.0.0.9000), pacta.portfolio.report (>= 0.0.1.9000), pacta.portfolio.utils (>= 0.0.1.9000), purrr, rlang, rstudioapi, tibble

**Remotes** RMI-PACTA/pacta.data.scraping,  
RMI-PACTA/pacta.data.validation,  
RMI-PACTA/pacta.executive.summary,  
RMI-PACTA/pacta.interactive.plot,  
RMI-PACTA/pacta.portfolio.allocate,  
RMI-PACTA/pacta.portfolio.audit,  
RMI-PACTA/pacta.portfolio.import,  
RMI-PACTA/pacta.portfolio.report,  
RMI-PACTA/pacta.portfolio.utils

**Suggests** covr, mockr, testthat (>= 3.0.0)

**Config/testthat/edition** 3

**Config/Needs/website** rmi-pacta/pacta.pkgdown.rmitemplate

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.2.3

**Repository** <https://rmi-pacta.r-universe.dev>

**RemoteUrl** <https://github.com/rmi-pacta/pactaverse>

**RemoteRef** HEAD

**RemoteSha** fb0e0a57a5ca1192da2de24555ec22176989d416

## Contents

pactaverse_conflicts . . . . .	2
pactaverse_deps . . . . .	3
pactaverse_logo . . . . .	3
pactaverse_packages . . . . .	4
pactaverse_sitrep . . . . .	4
pactaverse_update . . . . .	5
<b>Index</b>	<b>6</b>

---

pactaverse\_conflicts *Conflicts between the pactaverse and other packages*

---

## Description

This function lists all the conflicts between packages in the pactaverse and other packages that you have loaded.

## Usage

```
pactaverse_conflicts(only = NULL)
```

## Arguments

**only** Set this to a character vector to restrict to conflicts only with these packages.

## Details

There are four conflicts that are deliberately ignored: `intersect`, `union`, `setequal`, and `setdiff` from `dplyr`. These functions make the base equivalents generic, so shouldn't negatively affect any existing code.

## Examples

```
pactaverse_conflicts()
```

---

pactaverse_deps	<i>List all pactaverse dependencies</i>
-----------------	---

---

**Description**

List all pactaverse dependencies

**Usage**

```
pactaverse_deps(recursive = FALSE, repos = getOption("repos"))
```

**Arguments**

recursive	If TRUE, will also list all dependencies of pactaverse packages.
repos	The repositories to use to check for updates. Defaults to <code>getOption("repos")</code> .

---

pactaverse_logo	<i>The pactaverse logo, using ASCII or Unicode characters</i>
-----------------	---

---

**Description**

Use `cli::ansi_strip()` to get rid of the colors.

**Usage**

```
pactaverse_logo(unicode = cli::is_utf8_output())
```

**Arguments**

unicode	Whether to use Unicode symbols. Default is TRUE on UTF-8 platforms.
---------	---

**Examples**

```
pactaverse_logo()
```

pactaverse\_packages    *List all packages in the pactaverse*

---

**Description**

List all packages in the pactaverse

**Usage**

```
pactaverse_packages(include_self = TRUE)
```

**Arguments**

include\_self    Include pactaverse in the list?

**Examples**

```
pactaverse_packages()
```

---

pactaverse\_sitrep    *Get a situation report on the pactaverse*

---

**Description**

This function gives a quick overview of the versions of R and RStudio as well as all pactaverse packages. It's primarily designed to help you get a quick idea of what's going on when you're helping someone else debug a problem.

**Usage**

```
pactaverse_sitrep()
```

---

pactaverse\_update      *Update pactaverse packages*

---

**Description**

This will check to see if all pactaverse packages (and optionally, their dependencies) are up-to-date, and will install after an interactive confirmation.

**Usage**

```
pactaverse_update(recursive = FALSE, repos = getOption("repos"))
```

**Arguments**

`recursive`      If TRUE, will also list all dependencies of pactaverse packages.  
`repos`            The repositories to use to check for updates. Defaults to `getOption("repos")`.

**Examples**

```
## Not run:  
pactaverse_update()  
  
## End(Not run)
```

# Index

`cli::ansi_strip()`, 3

`pactaverse_conflicts`, 2

`pactaverse_deps`, 3

`pactaverse_logo`, 3

`pactaverse_packages`, 4

`pactaverse_sitrep`, 4

`pactaverse_update`, 5