Package: patRoonInst (via r-universe)

August 27, 2024

Type Package

Title Manages 'patRoon' Installations

Version 1.0.0

Author Rick Helmus

Maintainer Rick Helmus <r.helmus@uva.nl>

Description Installs and updates patRoon and its dependencies.

License GPL-3

Encoding UTF-8

LazyData true

URL https://github.com/rickhelmus/patRoonInst,
https://rickhelmus.github.io/patRoonInst/

BugReports https://github.com/rickhelmus/patRoon/issues

Depends R (>= 4.0)

Roxygen list(markdown = TRUE)

RoxygenNote 7.2.3

Imports methods, checkmate, jsonlite, remotes, BiocManager, withr

Collate 'direct_deps.R' 'install-main.R' 'install-repos.R' 'install-PD.R' 'install-RU.R' 'install.R' 'legacy.R' 'purge.R' 'utils.R' 'zzz.R'

Suggests testthat (>= 3.0.0)

Config/testthat/edition 3

Repository https://rickhelmus.r-universe.dev

RemoteUrl https://github.com/rickhelmus/patRoonInst

RemoteRef HEAD

RemoteSha 060d125a20dd46a51567defe9e2f17c1bd24a8ac

Contents

`patRoonInst-package` *Manage patRoon installations*

Description

Installs and updates patRoon and its dependencies.

Package options

The following package options (see `options()`) can be set:

- `patRoonInst.repos.patRoonDeps` The URL to the `patRoonDeps` R package repository.
- `patRoonInst.repos.runiverse` The URL to the r-universe R package repository.
- `patRoonInst.path.legacy` The directory where legacy files were installed.

See Also

Useful links:

- `https://github.com/rickhelmus/patRoonInst`
- `https://rickhelmus.github.io/patRoonInst/`
- Report bugs at `https://github.com/rickhelmus/patRoon/issues`

`getDirectDeps` *Installation data for* **patRoon** *and its direct dependencies*

Description

Returns information for **patRoon** and its direct R package dependencies.

Usage

```
getDirectDeps()
```

Details

The information returned by this function may change over time, and is downloaded from the internet.

Value

A `list` with miscellaneous package information.

`installing` *Managing* **patRoon** *installations*

Description

Functions to install, update and synchronize **patRoon** and its dependencies from various repositories.

Usage

```
install(
  pkgs = NULL,
  ignorePkgs = NULL,
  origin = NULL,
  lib.loc = NULL,
  ask = TRUE,
  force = FALSE,
  quiet = TRUE
)

update(
  pkgs = NULL,
  ignorePkgs = NULL,
  origin = NULL,
  lib.loc = NULL,
  ask = TRUE,
  quiet = TRUE
)

sync(
  pkgs = NULL,
  ignorePkgs = NULL,
  origin = NULL,
  lib.loc = NULL,
  allDeps = FALSE,
  ask = TRUE,
  quiet = TRUE
)
```

Arguments

`pkgs`, `ignorePkgs`

A `character` vector with packages to consider/ignore. Execute `names(getDirectDeps())` to obtain valid package options. If `NULL` then all packages are considered/none are ignored. The `ignorePkgs` argument can also include `"big"` to exclude large packages (*e.g.* **patRoonDeps**), and will override `pkgs` in case of conflicts.

`origin` Where **patRoon** and its R dependencies are installed from. Valid values are: `"patRoonDeps"`, `"runiverse"`, `"regular"`. If `NULL` then the default on Windows is `"patRoonDeps"` and `"runiverse"` otherwise. See below for more details.

`lib.loc` The path to the R library where packages will be installed. Set to `NULL` for the default R library.

`ask` Set to `TRUE` to ask before proceeding package installations. No effect on non-interactive R sessions.

`force` If `TRUE` then packages will *always* be installed, even if already present and with the correct version.

`quiet` If `TRUE` the installations are performed more quietly (sets the `quiet` option to `install.packages()`.

`allDeps` Consider *all* dependencies when synchronizing, including recursive dependencies. This is currently only supported for `origin="patRoonDeps"`. Note that handling of recursive dependencies currently are not influenced by the `pkgs` and `ignorePkgs` arguments.

Details

The installation of patRoon involves installing several dependencies, including R packages not available from common repositories (CRAN, BioConductor) and software outside the R environment. The following three functions automate this process and are used to install and maintain **patRoon** and its dependencies:

- `install()` Installs **patRoon** and any missing dependencies.
- `update()` Like `install()` and updates outdated packages.
- `sync()` Like `update()` but may also downgrade packages to fully synchronize package versions with the repository.

The R packages are currently sourced from the following repositories (set by the `origin` argument):

- `"patRoonDeps"`: a specialized **miniCRAN** repository contains binary R packages with versions that are automatically tested with **patRoon**. Currently this is *only* supported for Windows.
- `"runiverse"`: an r-universe repository containing the latest versions of **patRoon** and its dependencies.
- "regular": the last versions are sourced from regular CRAN/BioConductor repositories and GitHub. The latter means that suitable build tools (*e.g.* Rtools on windows).

Note that some large R packages, currently `patRoonDeps`, `patRoonExt` and `MetaCleanData`, are always sourced directly from GitHub.

Synchronization with `sync()` was mainly designed to be used in combination with the `patRoonDeps` repository, since it contains package versions already tested with `patRoon`. Doing a synchronization is most commonly used with **patRoon** bundle installations or when **patRoon** is installed in a separate R library (*i.e.* using the `lib.loc` argument). Furthermore, the `allDeps` argument should be set to `TRUE` to ensure *all* dependencies are synchronized.

Packages that originate from GitHub (even when obtained via `patRoonDeps/runiverse`) will *also* be synchronized with `update()`, since these packages typically involve 'snapshots' with unreliable version information.

Value

All functions return `NULL` invisibly.

`legacy` *Manage legacy* **patRoon** *installations*

Description

Toggles or removes installations of patRoon that were installed by the legacy installation script (`install_patRoon.R`).

Usage

```
toggleLegacy(enable = NULL, ask = TRUE)

removeLegacy(restoreRProfile = FALSE, ask = TRUE)
```

Arguments

`enable`	Enables (`enable=TRUE`), disables (`enable=FALSE`) or toggles (`enable=NULL`) a legacy **patRoon** installation.
`ask`	Set to `TRUE` to ask before proceeding. No effect on non-interactive R sessions.
`restoreRProfile`	If `TRUE` then the modifications to the user's `.Rprofile` file will be removed that were automatically performed by the legacy installation script.

Details

Previous versions of **patRoon** (`<2.3`) could be installed *via* a (now legacy) installation R script (`install_patRoon.R`). The functions documented here can (temporarily) undo the changes by this script, so that the installation functions in this package can be used to manage **patRoon** installations.

The `toggleLegacy()` function is used to enable or disable legacy installations temporarily.

The `removeLegacy()` function is used to undo legacy installations.

Value

All functions return `NULL` invisibly.

Background

This section is purely informative, but may be of use when you want to manually manage legacy installations.

The legacy installation script typically results in the following changes:

1. `~/patRoon-install`: a directory with external dependencies (`MetFrag`, `SIRIUS`, ...) and may contain R packages from the `patRoonDeps` repository depending on user input during the installation.
2. `~/.Rprofile-patRoon.R`: An R script that should be loaded during R startup, and initializes the configuration needed to use the files (1).
3. Code in `~/.Rprofile` to actually load (2) on R startup (only if it exists)

When `toggleLegacy()` is used to disable legacy installations the function simply renames the initialization script (2) so it won't be loaded on R startup. The `removeLegacy()` function removes the files from (1-2) and optionally the changes from (3).

Note

The R session should be restarted to make the changes effective.

Index