Package: tsdrtools (via r-universe)

October 18, 2026

Type Package

Title Convenience package for installing R-packages on UiO TSD

Version 0.1.03

Description Some packages are hard to install on the secure closed server of the University of Oslo, TSD This package has some functions to help users more easily install packages to TSD that cannot be installed using the mini-CRAN service within TSD.

License CC BY 4.0

Encoding UTF-8

RoxygenNote 7.2.0

Suggests testthat, covr, knitr, rmarkdown

Imports tools, utils

URL https://github.com/LCBC-UiO/tsdrtools

BugReports https://github.com/LCBC-UiO/tsdrtools/issues

VignetteBuilder knitr

Repository https://capro-uio.r-universe.dev

Date/Publication 2022-06-15 10:18:19 UTC

RemoteUrl https://github.com/capro-uio/tsdrtools

RemoteRef HEAD

RemoteSha c0a5370426079e39dcd2ee8604ab58eec6d93b88

Contents

as_install_error 2
as_tsdrtools_install 2
as_tsdrtools_status 3
core_pkgs 3
error_logs 3
get_dependency_tree 4

install_error 5
is.install_error 5
is.tsdrtools_status 5
is_install_error 6
is_tsdrtools_install 6
is_tsdrtools_status 6
package_workarounds 7
stringi_workaround 7
tsd_package_install 8
tsd_package_prepare 9
tsdrtools_install 10
tsdrtools_status 10

Index **11**

`as_install_error` *Create install_error*

Description

Create install_error

Usage

```
as_install_error(x)
```

Arguments

x	list to make install_error

`as_tsdrtools_install` *Create tsdrtools_install*

Description

Create tsdrtools_install

Usage

```
as_tsdrtools_install(x)
```

Arguments

x	list to make tsdrtools_install

as_tsdrtools_status *Create tsdrtools_status*

Description

Create tsdrtools_status

Usage

```
as_tsdrtools_status(x)
```

Arguments

x	list to make tsdrtools_status

core_pkgs *R core packages*

Description

R core packages

Usage

```
core_pkgs()
```

Value

character vector of R core-packages

error_logs *Extract error logs from tsdrstools_install-object*

Description

Extract error logs from tsdrstools_install-object

Usage

```
error_logs(x, package = NULL)
```

Arguments

x	object of class tsdrstools_install
package	NULL or character vector of package errors to extract. If NULL extracts all.

Value

list of errors

Examples

```
## Not run:
status <- tsd_package_install("devtools.zip")
error_logs(status)

## End(Not run)
```

---

get_dependency_tree *Get package dependency tree*

Description

This function will return all package dependencies for any given package, including backwards dependency for all. The function mainly prints out the dependency tree in the console, but also returns all packages required for build (uniquely).

Usage

```
get_dependency_tree(
  package,
  i = 1,
  type = c("Depends", "Imports"),
  repos = getOption("repos"),
  verbose = TRUE,
  ...
)
```

Arguments

package	name of package. character.
i	integer. Used in recursive call to package dependency retrieval.
type	Character vector, a subset of c("Depends", "Imports", "LinkingTo", "Suggests", "Enhances")
repos	character vector. Addresses of CRAN mirror(s)
verbose	logical. If function should be talkative.
...	additional arguments to functions available.packages

Value

character vector of unique packages

install_error — *Constructor for install_error*

Description

Constructor for install_error

Usage

```
install_error(x)
```

Arguments

x	named list of installation errors

is.install_error — *Validate install_error*

Description

Validate install_error

Usage

```
is.install_error(x)
```

Arguments

x	an object

is.tsdrtools_status — *Validate tsdrtools_status*

Description

Validate tsdrtools_status

Usage

```
is.tsdrtools_status(x)
```

Arguments

x	an object

is_install_error *Validate install_error*

Description

Validate install_error

Usage

```
is_install_error(x)
```

Arguments

x	an object

is_tsdrtools_install *Validate tsdrtools_install*

Description

Validate tsdrtools_install

Usage

```
is_tsdrtools_install(x)
```

Arguments

x	an object

is_tsdrtools_status *Validate tsdrtools_status*

Description

Validate tsdrtools_status

Usage

```
is_tsdrtools_status(x)
```

Arguments

x	an object

package_workarounds *Retrieve package workaround*

Description

Certain R-packages require workarounds to install on offline servers. This function will check if the given package has such a work-around and return said workaround for the type of process asked (prepare or install).

Usage

```
package_workarounds(package, folder, type, verbose = TRUE)
```

Arguments

package	name of package. character.
folder	character. folder to place all necessary files in
type	character vector of either "prepare" or "install". This indicates if the workaround for install-process or prepare-process should be returned.
verbose	logical. If function should be talkative.

stringi_workaround *Function for stringi workaround*

Description

stringi requires download of a bundle to work. If we want to install offline, the bundle needs to be downloaded and ported with the package tar. For `type = "prepare"` the bundle is downloaded into the folder with the pacakge tars. For install, the call to `R CMD INSTALL` is given the path to the bundle during installation, which will stop it from attempting to download the bundle during install.

Usage

```
stringi_workaround(folder, type, verbose = TRUE)
```

Arguments

folder	character. folder to place all necessary files in
type	character vector of either "prepare" or "install". This indicates if the workaround for install-process or prepare-process should be returned.
verbose	logical. If function should be talkative.

tsd_package_install *Install package on TSD from archive*

Description

After downloading and preparing the packages for install outside of TSD using the `tsd_package_prepare` function, install said packages using this function.

Usage

```
tsd_package_install(
  zip_file,
  lib = .libPaths()[1],
  opts = "",
  verbose = TRUE,
  ...
)
```

Arguments

`zip_file`	path to zipped file
`lib`	library folder to install to
`opts`	optional arguments to ‘R CMD INSTALL‘
`verbose`	logical. If function should be talkative.
`...`	additional arguments to functions `install.packages`

Value

vector of successful or failed package installs

Examples

```
## Not run:
# prepare a package first
tsd_package_prepare("devtools")

tsd_package_install("devtools.zip")

## End(Not run)
```

tsd_package_prepare *Prepare R package for TSD install*

Description

This function will download a package with all its dependencies to a zipped archive, to them be installed within TSD using `tsd_package_install`.

Usage

```
tsd_package_prepare(
  package,
  folder = package,
  repos = getOption("repos"),
  verbose = TRUE,
  zip = TRUE,
  ...
)
```

Arguments

`package`	name of package. character.
`folder`	character. folder to place all necessary files in
`repos`	character vector. Addresses of CRAN mirror(s)
`verbose`	logical. If function should be talkative.
`zip`	logical, if folder should be zipped at the end
`...`	additional arguments to functions `install.packages` and `available.packages`

Value

returns nothing, but creates a folder and zip archive

Examples

```
## Not run:
tsd_package_prepare("devtools")

tsd_package_prepare("dplyr", folder = "dplyr_pkg")

tsd_package_prepare("dplyr", repos = "https://cran.uib.no/", verbose = FALSE)


## End(Not run)
```

tsdrtools_install *Constructor for tsdrtools_install*

Description

Constructor for tsdrtools_install

Usage

```
tsdrtools_install(success, error_logs)
```

Arguments

success	data.frame of class tsdrtools_status
error_logs	list of class install_error

tsdrtools_status *Constructor for tsdrtools_status*

Description

Constructor for tsdrtools_status

Usage

```
tsdrtools_status(success, pkg, version)
```

Arguments

success	character vector of "success" or "failed"
pkg	character vector of packages
version	character vector of versions

Index

as_install_error, 2
as_tsdrtools_install, 2
as_tsdrtools_status, 3
available.packages, *4*, *9*

core_pkgs, 3

error_logs, 3

get_dependency_tree, 4

install.packages, *8*, *9*
install_error, 5
is.install_error, 5
is.tsdrtools_status, 5
is_install_error, 6
is_tsdrtools_install, 6
is_tsdrtools_status, 6

package_workarounds, 7

stringi_workaround, 7

tsd_package_install, 8, *9*
tsd_package_prepare, *8*, 9
tsdrtools_install, 10
tsdrtools_status, 10