Package: provisionr (via r-universe)

October 5, 2024

Title Provision a Library

Author Rich FitzJohn

Maintainer Rich FitzJohn <rich.fitzjohn@gmail.com>

Version 0.1.14

Description Provision a set of packages into a new library, possibly for use on a different architecture.

License MIT + file LICENSE

Imports R6, curl, drat, prettyunits, progress (>= 1.2.0), remotes, rversions, storr (>= 1.1.0), withr, yaml

Suggests testthat, zip

RoxygenNote 6.1.1

Encoding UTF-8

Repository https://mrc-ide.r-universe.dev

RemoteUrl https://github.com/mrc-ide/provisionr

RemoteRef master

RemoteSha 3350a3872b175d678622fd5b4a392e861aac969c

Contents

check_r_version *Check R version*

Description

Utility function for checking an R version

Usage

```
check_r_version(version = NULL)
```

Arguments

version	Something to coerce into an R version. Valid values are NULL (the current session's R version), a character string of the form x.y.z, a character string oldrel or release or a numeric_version object.

download_cran *Download some of CRAN*

Description

Download a fraction of CRAN to serve locally.

Usage

```
download_cran(packages, path, r_version = NULL, target = "windows",
  suggests = FALSE, package_sources = NULL,
  missing_index_is_error = TRUE, progress = NULL)
```

Arguments

packages	Character vector of packages to download
path	Destination
r_version	Target R version
target	Target platform (use "ALL" for all platforms)
suggests	Include suggested packages too?
package_sources	A provisionr::package_sources object
missing_index_is_error	Is a missing PACKAGES index an error? Set this to FALSE when downloading from incomplete repositories (e.g., a drat repo that contains only source files if you are trying to download binaries)
progress	Control progress bar printing

`download_files` *Download multiple files*

Description

Download multiple files.

Usage

```
download_files(urls, dest_dir, ..., labels = NULL, overwrite = FALSE,
  count = TRUE, dest_file = NULL, copy_file_urls = TRUE,
  progress = NULL, report = TRUE, headers = NULL)
```

Arguments

`urls`	A character vector of urls
`dest_dir`	A single existing directory to download files into
`...`	Currently ignored
`labels`	A character vector of labels to use to describe the files being downloaded when printing (defaults to `basename(urls)`)
`overwrite`	Overwrite files that exist already? If `FALSE` (the default) then existing files are skipped.
`count`	Logical, indicating if a count of progress across the urls should be included.
`dest_file`	If the files should be renamed as they are downloaded, include a vector of filenames here the same length as `urls`. Directory components will be created, within `dest_dir`.
`copy_file_urls`	Logical, indicating if `file:///` urls should be copied into `dest_dir`
`progress`	Print a progress bar?
`report`	Print a summary?
`headers`	Named character vector of HTTP headers (optional)

Value

A character vector, the same length as `urls`, with the destination file paths (even if no downloading was done). A failure to download a file (e.g., a 403 forbidden, 404 not found, or general network error) will result in an R error.

package_sources *Collect information on package sources*

Description

Collect information on package sources

Usage

```
package_sources(cran = NULL, repos = NULL, github = NULL,
  local = NULL, expire = NULL, local_drat = NULL, data = NULL,
  spec = NULL)
```

Arguments

cran	A single URL for the CRAN repo to use. If NULL, then this will use the rstudio mirror.
repos	A character vector of additional repositories to use. Repositories of the form drat://username will be translated to a drat repository hosted on github (e.g., https://username.github.io/drat)
github	A vector of github package specifications (e.g., username/password). The full syntax as implemented in parse_github_repo_spec is supported (packages in subdirectories, or referencing a particular branch, commit or tag).
local	A character vector of local files to include. Can be directories or built packages (source or binary)
expire	Optional period, in days, to expire the local copy of the package. If specified, then if a package was downloaded more thanexpire days ago when the local copy is checked, a new version will be downloaded. Can be a fractional value (e.g., expire = 0.04 for an expiry of around an hour).
local_drat	Optional location to cache downloaded packages, when github or local packages are used. If not given, this can be set at any time by setting the local_drat element of the returned element. If not given by the time that the packages need to be downloaded then a session-specific temporary directory will be used.
data	An object of class package_sources_list, created by running the as_list() method of a package_sources object. This is useful for serialisation without saving references to provisionr.
spec	Raw entries of the form <type>::<user>/<repo>...

provision_dependencies *Provision dependencies for a package*

Description

Provision dependencies for a package. The provision_dependencies_bootstrap function writes out a bootstrap script.

Usage

```
provision_dependencies(lib, path_description = ".", ..., src = NULL,
  read_travis = FALSE)

provision_dependencies_bootstrap(lib = ".packages", src = NULL,
  read_travis = FALSE, strict_lib = TRUE)
```

Arguments

lib	Library to provision
path_description	Path to the DESCRIPTION file
...	Additional arguments to provision_library
src	An optional description of additional packages, using package_sources.
read_travis	Logical, indicating if the .travis.yml should be read (if present). If TRUE then packages listed within r_github_packages will be used in provisioning.
strict_lib	Should the bootstrap script install provisionr within the local library too?

provision_library *Create or update a library*

Description

Create or update a library of packages.

Usage

```
provision_library(packages, lib, platform = NULL, version = NULL,
  src = NULL, check_dependencies = TRUE,
  installed_action = "upgrade", allow_missing = FALSE,
  refresh_drat = FALSE, quiet = FALSE, progress = NULL)
```

Arguments

packages	A character vector of packages to include
lib	A path to the library; if it does not exist, it will be created. If given as a vector of libraries (i.e., with more than one element) then packages will be installed into the first library, but subsequent libraries will be checked to make sure that dependencies are satisfied.
platform	The platform to create the library for. If NULL then we build for the current platform (using install_packages if version is NULL or compatible with our current version). Otherwise this can be one of "windows", "macosx", "macosx/mavericks" or "linux", correspinding to the different directories that binaries live in (in the case of "linux" there are no binaries and things are a little more complicated).
version	The version of R to install packages for. By default, we use the same version (major.minor) as the current running R version. Otherwise provide the desired version number as a string or numeric_version object.
src	An optional description of additional packages, using package_sources. It will only be rebuilt (fetching packages) if it has *never* been built, or of packages listed in the spec element are not found in the repository.
check_dependencies	Logical, indicating if dependencies of packges should be checked. If TRUE, then any missing dependencies (packages in Depends, Imports or LinkingTo of the requested packages) will be installed.
installed_action	The behaviour when some packages are already installed. Options are "replace" (will re-install the package and, for cross-installation, its dependencies), "upgrade_all" (upgrade all packages that lag behind the versions in repositories), "upgrade" (upgrade packages listed in "packages" only, but not their dependencies) and "skip" (do not install or upgrade any package that is already installed).
allow_missing	For cross-installation (via cross_install when platform is non-NULL), allow packages to be missing that need to be compiled? The interface here is going to change a bunch, so watch out...
refresh_drat	Logical indicating if the cache of packages pointed to by src should be refreshed. The other way of forcing this would be to pass expire = 0 through to the package_sources argument but this is likely to be tidier.
quiet	Passed through to to install.packages, indicating if package installation should be done quietly. With this as FALSE (the default) rather a lot of output can be generated.
progress	Passed through to the package downloading to control printing of the progress bar.

Details

Cross installation of binary files is difficult and I need to come up with a way of making that work nicely.

Author(s)

Rich FitzJohn

Index