Package: srcr (via r-universe)

September 10, 2024

Type Package

Title Simplify Connections to Database Sources

Version 1.1.1

Maintainer Charles Bailey <baileyc@chop.edu>

Description Connecting to databases requires boilerplate code to specify connection parameters and to set up sessions properly with the DBMS. This package provides a simple tool to fill two purposes: abstracting connection details, including secret credentials, out of your source code and managing configuration for frequently-used database connections in a persistent and flexible way, while minimizing requirements on the runtime environment.

License Artistic-2.0

Encoding UTF-8

ByteCompile TRUE

Roxygen list(markdown = TRUE)

Imports DBI, dplyr, jsonlite, utils

Suggests knitr, rmarkdown, RSQLite, withr

VignetteBuilder knitr

RoxygenNote 7.3.1

URL https://github.com/baileych/srcr

BugReports https://github.com/baileych/srcr/issues

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

RemoteUrl https://github.com/baileych/srcr

RemoteRef HEAD

RemoteSha 3883ee3fc69e7aa859259f6e2c9ce7ed49f8d1a9

Contents

`find_config_files` *Locate candidate configuration files*

Description

Given vectors of directories, basenames, and suffices, combine them to find existing files.

Usage

```
find_config_files(
  basenames = .basename.defaults(),
  dirs = .dir.defaults(),
  suffices = .suffix.defaults()
)
```

Arguments

`basenames`	A vector of file names (without directory or file type) to use in searching for configuration files.
`dirs`	A vector of directory names to use in searching for configuration files.
`suffices`	A vector of suffices (file "type"s) to use in searching for the configuration file.

Details

This function is intended to support a variety of installation patterns, so it attempts to be flexible in looking for configuration files. First, environment variables of the form *basename*`_CONFIG`, where *basename* is the uppercase form of each candidate basename, are examined to see whether any translate to a file path.

Following this, the path name parts supplied as arguments are used to build potential file names. If `dirs` is not specified, the following directories are checked by default:

1. the user's `$HOME` directory
2. the directory named `.srcr` (no leading `.` on Windows) under `$HOME`
3. the directory in which the executing script is located
4. the directory in which the calling function's calling function's source file is located (typically an application-level library). For example, if the function `my_setup()` calls `srcr()`, which in turn calls `find_config_files()`, then the directory of the file containing `my_setup()` will be tried.
5. the directory in which the calling function's source file is located (typically a utility function, such as `srcr()`)

Note that the current working directory is not part of the search by default. This is done to limit the potential for accidentally introducing (potentially harmful) configuration files by setting the working directory.

In each location, the file names given in `basenames` are checked; if none are specified, several default file names are tried:

1. the name of the calling function's source file
2. the name of the executing script
3. the directory in which the calling function's calling function's source file is located (typically an application-level library). For example, if the function `my_setup()` calls `srcr()`, which in turn calls `find_config_files()`, then the name of the file containing `my_setup()` will be tried.

The suffices (file "type"s) of `.json`, `.conf`, and nothing, are tried with each candidate path; you may override this default by using the `suffices` parameter. Finally, in order to accommodate the Unix tradition of "hidden" configuration files, each basename is prefixed with a period before trying the basename alone.

Value

A vector of path specifications, or an empty vector if none are found.

Examples

```
## Not run:
find_config_files() # All defaults
find_config_files(dirs = c(file.path(Sys.getenv('HOME'),'etc'),
                           '/usr/local/etc', '/etc'),
                  basenames = c('my_app'),
                  suffices = c('.conf', '.rc'))

## End(Not run)
```

Index