

Package ‘Rparadox’

September 22, 2025

Title Read Paradox Database Files into R

Version 0.1.2

Description Provides a simple and efficient way to read data from Paradox database files (.db) directly into R as modern 'tibble' data frames. It uses the underlying 'pxlib' C library to handle the low-level file format details and provides a clean, user-friendly R interface.

License GPL (>= 2)

Encoding UTF-8

RoxygenNote 7.3.3

Imports blob, hms, tibble

Suggests rmarkdown, devtools, knitr, testthat (>= 3.0.0), usethis

Config/testthat/edition 3

URL <https://github.com/celebithil/Rparadox>

BugReports <https://github.com/celebithil/Rparadox/issues>

VignetteBuilder knitr, rmarkdown

NeedsCompilation yes

Author Daniil Popov [aut, cre]

Maintainer Daniil Popov <popov.daniil@gmail.com>

Depends R (>= 3.5.0)

Repository CRAN

Date/Publication 2025-09-22 08:10:08 UTC

Contents

pxlib_close_file	2
pxlib_get_data	2
pxlib_open_file	3
Index	5

pxlib_close_file	<i>Close a Paradox database file</i>
------------------	--------------------------------------

Description

This function explicitly closes a Paradox database file associated with a pxdoc_t external pointer and releases its resources.

Usage

```
pxlib_close_file(pxdoc)
```

Arguments

pxdoc An external pointer of class 'pxdoc_t' obtained from pxlib_open_file().

Value

Invisible NULL.

pxlib_get_data	<i>Read Data from a Paradox File</i>
----------------	--------------------------------------

Description

Retrieves all records from an open Paradox database file and returns them as a tibble, ready for use in R.

Usage

```
pxlib_get_data(pxdoc)
```

Arguments

pxdoc An object of class pxdoc_t, representing an open Paradox file connection. This object is obtained from pxlib_open_file().

Details

This function provides a high-level interface for reading Paradox data. The heavy lifting is done by a C-level function (R_pxlib_get_data) which efficiently reads the raw data into memory. This R function then acts as a wrapper to perform crucial post-processing steps:

- It identifies columns containing binary (BLOB) data and correctly converts them into blob objects.
- It ensures that date/time columns are properly classed for seamless integration with other R functions.

The result is a clean, modern tibble that is fully compatible with the tidyverse ecosystem.

Value

A tibble containing the data from the Paradox file. Each row represents a record and each column represents a field. If the file contains no records, an empty tibble is returned.

Examples

```
# Define the path to the demo database included with the package
db_path <- system.file("extdata", "biolife.db", package = "Rparadox")

# Open the file handle
pxdoc <- pxlib_open_file(db_path)

if (!is.null(pxdoc)) {
  # Read all data into a tibble
  biolife_data <- pxlib_get_data(pxdoc)

  # Always close the file handle when finished
  pxlib_close_file(pxdoc)

  # Work with the data
  print(biolife_data)
}
```

pxlib_open_file	<i>Open a Paradox Database File</i>
-----------------	-------------------------------------

Description

Opens a Paradox database (.db) file and prepares it for reading. This function serves as the entry point for interacting with a Paradox database.

Usage

```
pxlib_open_file(path, encoding = NULL)
```

Arguments

path	A character string specifying the path to the Paradox (.db) file.
encoding	An optional character string specifying the input encoding of the data (e.g., "cp866", "cp1252"). If NULL (the default), the encoding is determined from the file header.

Details

This function initializes a connection to a Paradox file via the underlying C library. It automatically performs two key setup tasks:

1. **Encoding Override:** It allows the user to specify the character encoding of the source file via the encoding parameter. This is crucial for legacy files where the encoding stored in the header may be incorrect. If encoding is NULL, the function will attempt to use the codepage from the file header.
2. **BLOB File Attachment:** It automatically searches for an associated BLOB file (with a .mb extension, case-insensitively) in the same directory and, if found, attaches it to the database handle.

Value

An external pointer of class 'pxdoc_t' if the file is successfully opened, or NULL if an error occurs (e.g., file not found).

Examples

```
# Example 1: Open a bundled demo file (biolife.db)
db_path <- system.file("extdata", "biolife.db", package = "Rparadox")
pxdoc <- pxlib_open_file(db_path)
if (!is.null(pxdoc)) {
  # normally you'd read data here
  pxlib_close_file(pxdoc)
}

# Example 2: Open a file with overridden encoding (of_cp866.db)
db_path2 <- system.file("extdata", "of_cp866.db", package = "Rparadox")
pxdoc2 <- pxlib_open_file(db_path2, encoding = "cp866")
if (!is.null(pxdoc2)) {
  # read some data ...
  pxlib_close_file(pxdoc2)
}
```

Index

`pxlib_close_file`, [2](#)
`pxlib_get_data`, [2](#)
`pxlib_open_file`, [3](#)