Package 'ctf'

July 22, 2025

Type Package

Title Read and Write Column Text Format (CTF)

Version 0.1.0

Date 2021-06-21

Maintainer Clark Fitzgerald <fitzgerald@csus.edu>

Depends R (>= 3.1.0)

Imports jsonlite, iotools

Suggests roxygen2, knitr, rmarkdown, testthat

Description

Column Text Format (CTF) is a new tabular data format designed for simplicity and performance. CTF is the simplest column store you can imagine: plain text files for each column in a table, and a metadata file.
The underlying plain text means the data is human readable and familiar to programmers, unlike specialized binary formats.
CTF is faster than row oriented formats like CSV when loading a subset of the columns in a table.
This package provides functions to read and write CTF data from R.

License MIT + file LICENSE

URL https://github.com/julianofhernandez/ctf

BugReports https://github.com/julianofhernandez/ctf

VignetteBuilder knitr

RoxygenNote 7.1.1

NeedsCompilation no

Author Clark Fitzgerald [aut, cre] (ORCID:
<https://orcid.org/0000-0003-3446-6389>)

Repository CRAN

Date/Publication 2021-07-07 09:00:05 UTC

Contents

`read.ctf` *Read CTF data*

Description

Read external CTF data into the corresponding R data frame.

Usage

```
read.ctf(location, columns, nrows)
```

Arguments

`location`	location of the CTF data, either a file path to a CTF metadata JSON file, or a directory containing a single CTF metadata JSON file.
`columns`	names of the columns to read. If missing, then read in all columns.
`nrows`	integer, the maximum number of rows to read in. If missing, then read in all rows.

Value

data frame

See Also

`write.ctf` to write CTF

Examples

```
# An example CTF metadata file included in this package
d <- system.file("extdata", "vgsales", "vgsales-metadata.json", package = "ctf")

# Read all the rows and columns
vgsales <- read.ctf(d)

# Read 10 rows of two columns, Name and Rank
vgsales2 <- read.ctf(d, columns = c("Name", "Rank"), nrows = 10)
```

write.ctf *Write Data Frame To CTF*

Description

Save a data frame using Column Text Format

Usage

```
write.ctf(x, datadir = name, name = deparse(substitute(x)), ...)
```

Arguments

x	data frame to write
datadir	directory to write the metadata and CTF columns
name	table name
...	further arguments to write.table.raw

Value

NULL, used for its side effect

See Also

read.ctf to read CTF, write.table.raw for the underlying functionality, and save for writing any R objects.

Examples

```
d <- file.path(tempdir(), "iris_ctf_data")
write.ctf(iris, d)

# Same object as iris, but carries around some extra metadata
iris2 <- read.ctf(d)

# This directory contains plain text files for each column in iris
list.files(d)

# Clean up
unlink(d, recursive = TRUE)
```

Index