

Package: multifwf (via r-universe)

September 9, 2024

Type Package

Title Read Fixed Width Format Files Containing Lines of Different Type

Version 0.5.0

Date 2019-02-07

Encoding UTF-8

Author Panos Rontogiannis

Maintainer Panos Rontogiannis <panos@ronto.net>

Description Read a table of fixed width formatted data of different types into a data.frame for each type.

Depends R (>= 3.1.1)

License GPL (>= 2)

URL <https://github.com/prontog/multifwf>

BugReports <https://github.com/prontog/multifwf/issues>

RoxygenNote 6.1.1

Imports utils, readr, tibble

Suggests testthat

Repository <https://prontog.r-universe.dev>

RemoteUrl <https://github.com/prontog/multifwf>

RemoteRef HEAD

RemoteSha fad644cae7c7825a85ac45239bee76f422507b3a

Contents

multifwf	2
read.multi.fwf	2
read_multi_fwf	4

Index	6
--------------	----------

 multifwf

Read Fixed Width Format Files containing lines of different Type

Description

Read a table of fixed width formatted data of different types into a `data.frame` for each type.

Details

The only function you're likely to need from **multifwf** is `read.multi.fwf`.

Author(s)

Panos Rontogiannis <p.g.ronto@gmail.com>

 read.multi.fwf

Read Fixed Width Format Files containing lines of different Type

Description

Read a table of fixed width formatted data of different types into a `data.frame` for each type.

Usage

```
read.multi.fwf(file, multi.specs, select, header = FALSE, sep = "\t",
  skip = 0, n = -1, buffersize = 2000, ...)
```

Arguments

<code>file</code>	the name of the file which the data are to be read from. Alternatively, <code>file</code> can be a connection , which will be opened if necessary, and if so closed at the end of the function call.						
<code>multi.specs</code>	A named list of <code>data.frames</code> containing the following columns: <table style="margin-left: 40px;"> <tr> <td><code>widths</code></td> <td>see read.fwf</td> </tr> <tr> <td><code>col.names</code></td> <td>see read.table</td> </tr> <tr> <td><code>row.names</code></td> <td>see read.table</td> </tr> </table>	<code>widths</code>	see read.fwf	<code>col.names</code>	see read.table	<code>row.names</code>	see read.table
<code>widths</code>	see read.fwf						
<code>col.names</code>	see read.table						
<code>row.names</code>	see read.table						

For more info on these fields see [read.fwf](#).

Note that each list item should have a name. This is important for the `select` function.

<code>select</code>	A function to select the type of a line. This selector should have parameters:
---------------------	--

line the line
 specs the multi.specs list that was passed to read.multi.fwf

The select function should return the name of the spec that matches the line. read.multi.fwf will then use this name to select the a spec from the passed multi.spec. This is why multi.spec should be a named list. If there is no match then NULL can be returned.

header a logical value indicating whether the file contains the names of the variables as its first line. If present, the names must be delimited by sep.

sep character; the separator used internally; should be a character that does not occur in the file (except in the header).

skip number of initial lines to skip; see [read.fwf](#).

n the maximum number of records (lines) to be read, defaulting to no limit.

buffersize Maximum number of lines to read at one time

... further arguments to be passed to [read.fwf](#).

Value

Return value is a named list with an item for each spec in multi.spec. If there was at least one line in file, matching a spec, then the named item will be a [data.frame](#). Otherwise it will be NULL.

Author(s)

Panos Rontogiannis <p.g.ronto@gmail.com>

See Also

[read.fwf](#)

Examples

```
ff <- tempfile()
cat(file = ff, '123456', '287654', '198765', sep = '\n')
specs <- list()
specs[['sp1']] = data.frame(widths = c(1, 2, 3),
                             col.names = c('Col1', 'Col2', 'Col3'))
specs[['sp2']] = data.frame(widths = c(3, 2, 1),
                             col.names = c('C1', 'C2', 'C3'))

myselector <- function(line, specs) {
  s <- substr(line, 1, 1)
  spec_name = ''
  if (s == '1')
    spec_name = 'sp1'
  else if (s == '2')
    spec_name = 'sp2'

  spec_name
}
```

```

}

read_multi_fwf(ff, multi.specs = specs, select = myselector)
#> sp1: 1 23 456 \ 1 98 765, sp2: 287 65 4

unlink(ff)

```

read_multi_fwf

Read Fixed Width Format Files containing lines of different Type

Description

Read a table of fixed width formatted data of different types into a [tibble](#) for each type.

Usage

```
read_multi_fwf(file, multi.specs, select, skip = 0, n = -1, ...)
```

Arguments

file	either a path to a file, a connection, or literal data (either a single string or a raw vector). Files ending in .gz, .bz2, .xz, or .zip will be automatically uncompressed. Files starting with http://, https://, ftp://, or ftps:// will be automatically downloaded. Remote gz files can also be automatically downloaded and decompressed. Literal data is most useful for examples and tests. It must contain at least one new line to be recognised as data (instead of a path) or be a vector of greater than length 1. Using a value of clipboard() will read from the system clipboard.
multi.specs	A named list of data.frames containing the following columns: widths see fwf_widths col_names see fwf_widths

For more info on these fields see [read_fwf](#).

Note that each list item should have a name. This is important for the select function.

select	A function to select the type of a line. This selector should have parameters:
--------	--

line	the line
------	----------

specs	the multi.specs list that was passed to read_multi_fwf
-------	--

The select function should return the name of the spec that matches the line. read_multi_fwf will then use this name to select the a spec from the passed multi.spec. This is why multi.spec should be a named list. If there is no match then NULL can be returned.

skip number of initial lines to skip; see [read_fwf](#).
n the maximum number of records (lines) to be read, defaulting to no limit.
... further arguments to be passed to [read_fwf](#).

Value

Return value is a named list with an item for each spec in multi.spec. If there was at least one line in file, matching a spec, then the named item will be a [tibble](#). Otherwise it will be NULL.

Author(s)

Panos Rontogiannis <panos@ronto.net>

See Also

[read_fwf](#)

Examples

```
ff <- tempfile()
cat(file = ff, '123456', '287654', '198765', sep = '\n')
specs <- list()
specs[['sp1']] = data.frame(widths = c(1, 2, 3),
                             col_names = c('Col1', 'Col2', 'Col3'))
specs[['sp2']] = data.frame(widths = c(3, 2, 1),
                             col_names = c('C1', 'C2', 'C3'))

myselector <- function(line, specs) {
  s <- substr(line, 1, 1)
  spec_name = ''
  if (s == '1')
    spec_name = 'sp1'
  else if (s == '2')
    spec_name = 'sp2'

  spec_name
}

read_multi_fwf(ff, multi.specs = specs, select = myselector)
#> sp1: 1 23 456 \ 1 98 765, sp2: 287 65 4

unlink(ff)
```

Index

connection, [2](#)

data.frame, [2](#), [3](#)

fwf_widths, [4](#)

multifwf, [2](#)

multifwf-package (multifwf), [2](#)

read.fwf, [2](#), [3](#)

read.multi.fwf, [2](#), [2](#)

read.table, [2](#)

read_fwf, [4](#), [5](#)

read_multi_fwf, [4](#)

tibble, [4](#), [5](#)