

Package ‘nat.utils’

July 2, 2014

Maintainer Gregory Jefferis <jefferis@gmail.com>

Author Gregory Jefferis

Version 0.4.2

Date 2014-06-13

License GPL-3

Title File System Utility Functions for NeuroAnatomy Toolbox

Description Utility functions to support the NeuroAnatomy Toolbox (nat).
Includes functions to provide a basic make style system to update files
based on timestamp information, file locking and other convenience functions
for working with the filesystem.

Suggests testthat, roxygen2, digest

NeedsCompilation no

Repository CRAN

Date/Publication 2014-06-16 14:09:52

R topics documented:

abs2rel	2
file.hardlink	2
file.swap	3
gzip.crc	4
is.zip	4
makelock	5
nat.utils	6
ncpus	6
RunCmdForNewerInput	7
touch	8
zipinfo	8
zipok	9

Index	10
--------------	-----------

abs2rel	<i>Remove common part of two paths, leaving relative path</i>
---------	---

Description

Remove common part of two paths, leaving relative path

Usage

```
abs2rel(path, stempath = getwd(), StopIfNoCommonPath = FALSE)
```

Arguments

path	Path to make relative
stempath	Root to which path will be made relative
StopIfNoCommonPath	Error if no path in common

Value

Character vector containing relative path

Author(s)

jefferis

See Also

[path.expand](#), [normalizePath](#)

Examples

```
path = "/Volumes/JData/JPeople/Sebastian/images"  
abs2rel(path, '/Volumes/JData')
```

file.hardlink	<i>Make hardlink to file</i>
---------------	------------------------------

Description

Used system call to access 'ln' command line utility. Now deprecated in favour of recently introduced file.link function in base R.

Usage

```
file.hardlink(from, to)
```

Arguments

from Source file
to (New) target hardlink file to create

Value

logical indicating success

Author(s)

jefferis

See Also

[file.symlink](#)

file.swap *Swap names of two files (by renaming first to a temporary file)*

Description

Swap names of two files (by renaming first to a temporary file)

Usage

`file.swap(f1, f2)`

Arguments

f1, f2 Paths to files

Value

logical indicating success

Author(s)

jefferis

See Also

[file.rename](#)

`gzip.crc`*Extract the crc (32 bit hash) of a gzip file*

Description

Assumes that the gzip crc is at the end of the file. Checks for a valid gzip magic number at the start of the file

Usage

```
gzip.crc(f)
```

Arguments

f Path to a gzip file

Value

hexadecimal formatted

Examples

```
rdsfile=system.file('help/aliases.rds')
gzip.crc(rdsfile)
```

`is.zip`*Check if a file is a zip file*

Description

Check if a file is a zip file

Usage

```
is.zip(f)
```

Arguments

f Path to file to test

Value

logical indicating whether f is in zip format (or NA if the file cannot be accessed)

Examples

```
notgzipfile=tempfile()
writeLines('not a gzip', notgzipfile)
is.gzip(notgzipfile)
con=gzfile(gzipfile<-tempfile(),open='wt')
writeLines('This one is gzipped', con)
is.gzip(gzipfile)
unlink(c(notgzipfile,gzipfile))
```

makelock

Make and remove (NFS safe) lock files

Description

Creates a lock file on disk containing a message that should identify the current R session. Will return FALSE if someone else has already made a lockfile. In order to avoid race conditions typical on NFS mounted drives makelock appends a unique message to the lock file and then reads the file back in. Only if the unique message is the first line in the file will makelock return TRUE.

removeLock displays a warning if lockfile does not exist

Usage

```
makelock(lockfile, lockmsg, CreateDirectories = TRUE)
```

```
removeLock(lockfile)
```

Arguments

lockfile	Path to lockfile
lockmsg	Character vector with message to be written to lockfile
CreateDirectories	Recursively create directories implied by lockfile path

Value

logical indicating success

Author(s)

jefferis

Examples

```
makelock(lock<-tempfile())
stopifnot(!makelock(lock))
removeLock(lock)
```

`nat.utils`*nat.utils: File System Utility Functions for NeuroAnatomy Toolbox*

Description

Utility functions to support the NeuroAnatomy Toolbox (nat). Includes functions to provide a basic make style system to update files based on timestamp information, file locking and other convenience functions for working with the filesystem

See Also`nat`

`ncpus`*Return number of cpus (or a default on failure)*

Description

Return number of cpus (or a default on failure)

Usage`ncpus(default = 1)`**Arguments**

<code>default</code>	Number of cores to assume if <code>detectCores</code> fails
----------------------	---

Value

Integer number of cores

Author(s)`jefferis`**See Also**[detectCores](#)**Examples**`ncpus()`

RunCmdForNewerInput *Run a command if input files are newer than outputs*

Description

Run a command if input files are newer than outputs

Usage

```
RunCmdForNewerInput(cmd, infiles, outfiles, Verbose = FALSE,  
  UseLock = FALSE, Force = FALSE, ReturnInputTimes = FALSE, ...)
```

Arguments

cmd	An expression , a string or NA/NULL
infiles	Character vector of path to one or more input files
outfiles	Character vector of path to one or more output files
Verbose	Write information to console (Default FALSE)
UseLock	Stop other processes working on this task (Default FALSE)
Force	Ignore file modification times and always produce output if input files exist.
ReturnInputTimes	Return mtimes of input files (default FALSE)
...	additional parameters passed to system call.

Details

cmd can be an R expression, which is [evaluated](#) if necessary in the environment calling RunCmdForNewerInput, a string to be passed to [system](#) or NULL/NA in which cases the files are checked and TRUE or FALSE is returned depending on whether action is required.

When UseLock=TRUE, the lock file created is called outfiles[1].lock

When ReturnInputTimes=TRUE, the input mtimes are returned as an attribute of a logical value (if available).

Value

logical indicating if cmd was run or for an R expression, eval(cmd)

See Also

[makelock](#), [eval](#), [expression](#)

Examples

```
## Not run:  
RunCmdForNewerInput(expression(myfunc("somefile")))  
  
## End(Not run)
```

touch	<i>Use unix touch utility to change file's timestamp</i>
-------	--

Description

If neither a time or a reference file is provided then the current time is used. If the file does not already exist, it is created unless Create=FALSE.

Usage

```
touch(file, time, reference, timestouupdate = c("access", "modification"),
      Create = TRUE)
```

Arguments

file	Path to file to modify
time	Absolute time in POSIXct format
reference	Path to a reference file
timestouupdate	"access" or "modification" (default both)
Create	Logical indicating whether to create file (default TRUE)

Value

TRUE or FALSE according to success

Author(s)

jefferis

zipinfo	<i>Return information about a zip archive using system unzip command</i>
---------	--

Description

Return information about a zip archive using system unzip command

Usage

```
zipinfo(f)
```

Arguments

f	Path to one (or more) files
---	-----------------------------

Details

Uses system unzip command.

Value

dataframe of information

Author(s)

jefferis

See Also

[zip](#)

Other ziputils: [zipok](#)

zipok

Verify integrity of one or more zip files

Description

Verify integrity of one or more zip files

Usage

```
zipok(f, Verbose = FALSE)
```

Arguments

f	Path to one (or more) files
Verbose	Whether to be Verbose (default FALSE)

Details

Uses system unzip command.

Value

TRUE when file OK, FALSE otherwise

Author(s)

jefferis

See Also

Other ziputils: [zipinfo](#)

Index

abs2rel, [2](#)

detectCores, [6](#)

eval, [7](#)
expression, [7](#)

file.hardlink, [2](#)
file.rename, [3](#)
file.swap, [3](#)
file.symlink, [3](#)

gzip.crc, [4](#)

is.gzip, [4](#)

makelock, [5](#), [7](#)

nat.utils, [6](#)
nat.utils-package (nat.utils), [6](#)
ncpus, [6](#)
normalizePath, [2](#)

path.expand, [2](#)

removelock (makelock), [5](#)
RunCmdForNewerInput, [7](#)

system, [7](#)

touch, [8](#)

zip, [9](#)
zipinfo, [8](#), [9](#)
zipok, [9](#), [9](#)