

January 7, 2015

0.0.1 moveDocs

`moveDocs` relocates files by renaming with a new file path. Specifically, it scans for md and html files in the `docs/Rmd` directory and/or pdf files in the `docs/Rnw` directory. If such files are found in the respective locations, they are moved to `docs/md`, `docs/html`, and `docs/pdf`, respectively.

The intent is to clean up the Rmd and Rnw directories after `knitr` has been used to knit documents in place. I do this because I have more success knitting documents with the confluence of `RStudio`, `rmarkdown`, `knitr`, `pandoc`, and `LaTeX` when the knitting occurs all within the directory of the originating files. The process is more prone to throwing errors when trying to specify alternate locations for outputs.

`moveDocs` makes a nominal effort to replace a possible relative path with a full file path before proceeding, if the former is supplied. Default arguments include `move=TRUE` which will call `file.rename` and `copy=FALSE` which, if `TRUE` (and `move=FALSE`), will alternatively call `file.copy`. If both are `TRUE`, any files found are moved.

This function will always overwrite any existing file versions previously moved to the output directories, by way of `file.rename`. To keep the behavior consistent, when `move=FALSE` and `copy=TRUE`, `file.copy` always executes with its argument, `overwrite=TRUE`. This should never cause problems because in the context I intend for this function, the types of files being moved or copied from `docs/Rmd` and `docs/Rnw` are never used as inputs to other files, functions, or processes, nor are they meant to be edited by hand after being generated.

If there are LaTeX-associated files present (`.TeX`, `.aux`, and `.txt` files with the same file names as local pdf files.), these files will be removed if `remove.latex=TRUE` (default). If `FALSE`, the default `latexDir="LaTeX"` means that these files will be moved to the `docs/LaTeX` directory rather than deleted. If this directory does not exist, it will be created. An alternate location can be specified, such as `"pdf"` if you want to keep these files with the related pdf files after those are moved by `moveDocs` as well to `docs/pdf`.

```
# Organization documentation
moveDocs <- function(path.docs, type = c("md", "html", "pdf"), move = TRUE,
  copy = FALSE, remove.latex = TRUE, latexDir = "latex") {
  if (any(!type %in% c("md", "html", "pdf")))
    stop("type must be among 'md', 'html', and 'pdf'")
  stopifnot(move | copy)
  if (path.docs == "." | path.docs == "./")
    path.docs <- getwd()
  if (strsplit(path.docs, "/")[[1]][1] == "..") {
    tmp <- strsplit(path.docs, "/")[[1]][-1]
    if (length(tmp))
      path.docs <- file.path(getwd(), paste0(tmp, collapse = "/")) else stop("Check path.docs argu")
  }
  for (i in 1:length(type)) {
    if (type[i] == "pdf")
      origin <- "Rnw" else origin <- "Rmd"
    path.i <- file.path(path.docs, origin)
    infiles <- list.files(path.i, pattern = paste0("\\\\.", type[i], "$"),
      full = TRUE)
    if (type[i] == "pdf") {
      extensions <- c("tex", "aux", "log")
```

```

all.pdfes <- basename(list.files(path.docs, pattern = ".pdf$", full = T,
    recursive = T))
pat <- paste0("^", rep(gsub("pdf", "", all.pdfes), length(extensions)),
    rep(extensions, each = length(all.pdfes)), "$")
latex.files <- unlist(sapply(1:length(pat), function(p, path, pat) list.files(path,
    pattern = pat[p], full = TRUE), path = path.i, pat = pat))
print(latex.files)
if (length(latex.files)) {
  if (remove.latex) {
    unlink(latex.files)
  } else {
    dir.create(file.path(path.docs, latexDir), showWarnings = FALSE,
        recursive = TRUE)
    file.rename(latex.files, file.path(path.docs, latexDir, basename(latex.files)))
  }
}
}
if (length(infiles)) {
  infiles <- infiles[basename(dirname(infiles)) == origin]
  if (length(infiles)) {
    if (type[i] == "html") {
      html.dirs <- gsub("\\.html", "_files", infiles)
      dirs <- list.dirs(path.i, recursive = FALSE)
      ind <- which(dirs %in% html.dirs)
      if (length(ind)) {
        html.dirs <- dirs[ind]
        html.dirs.recur <- list.dirs(html.dirs)
        for (p in 1:length(html.dirs.recur)) dir.create(gsub("/Rmd",
            "/html", html.dirs.recur[p]), recursive = TRUE, showWarnings = FALSE)
        subfiles <- unique(unlist(lapply(1:length(html.dirs.recur),
            function(p, path) list.files(path[p], full = TRUE), path = html.dirs.recur)))
        subfiles <- subfiles[!(subfiles %in% html.dirs.recur)]
        file.copy(subfiles, gsub("/Rmd", "/html", subfiles), overwrite = TRUE)
        if (move)
          unlink(html.dirs, recursive = TRUE)
      }
    }
    outfiles <- file.path(path.docs, type[i], basename(infiles))
    if (move)
      file.rename(infiles, outfiles) else if (copy)
      file.copy(infiles, outfiles, overwrite = TRUE)
  }
}
}
}

```