Package 'liftr'

August 29, 2016

Type Package

Title Dockerize R Markdown Documents

Version 0.4

Maintainer Nan Xiao <me@nanx.me>

Description Dockerize R Markdown documents with support for Rabix.

License GPL

SystemRequirements Docker (see
<https://docs.docker.com/engine/installation/>), Rabix (see
<https://github.com/rabix/rabix>)

VignetteBuilder knitr

URL http://liftr.me

BugReports https://github.com/road2stat/liftr/issues

Depends R (>= 3.0.2)

Imports yaml, knitr, rmarkdown, stringr

RoxygenNote 5.0.1

NeedsCompilation no

Author Miaozhu Li [ctb],
Tengfei Yin [ctb],
Nan Xiao [aut, cre]

Repository CRAN

Date/Publication 2016-08-05 12:52:07

R topics documented:

liftr-package *Dockerize R Markdown Documents*

Description

liftr dockerizes your R Markdown documents.

Details

The vignette can be opened with `vignette('liftr-intro')`.

Package:	liftr
Type:	Package
License:	GPL

Author(s)

Nan Xiao <<me@nanx.me>> Miaozhu Li <<miaozhu.li@duke.edu>> Tengfei Yin <<tengfei.yin@sbgenomics.com>>

drender *Render Dockerized R Markdown Documents*

Description

Render dockerized R Markdown documents using Docker containers. Rabix tools/workflows will be ran if there is a `Rabixfile` generated by lift under the same directory.

Usage

```
drender(input = NULL, tag = NULL, build_args = NULL,
  container_name = NULL, reset = TRUE, ...)
```

Arguments

`input`	Input file to render in Docker container.
`tag`	Docker image name to build, sent as docker argument `-t`. If not specified, it will use the same name as the input file.
`build_args`	A character string specifying additional `docker build` arguments. For example, `--pull=true -m="1024m" --memory-swap="-1"`.
`container_name`	Docker container name to run. If not specified, we will generate and use a random name.
`reset`	Should we cleanup the Docker container and Docker image after getting the rendered result?
`...`	Additional arguments passed to `render`.

Details

Before using `drender()`, run lift on the document first to generate `Dockerfile`. See `vignette('liftr-intro')` for details about the extended YAML front-matter metadata format and system requirements for rendering dockerized R Markdown documents.

Value

Rendered file is written to the same directory of the input file. A character vector with the image name and container name will be returned. You will be able to manage them with `docker` commands later if `reset = FALSE`.

Examples

```
# 1. Dockerized R Markdown document
# Docker is required to run the example,
# so make sure we can use `docker` in shell.
dir_docker = paste0(tempdir(), '/drender_docker/')
dir.create(dir_docker)
file.copy(system.file("docker.Rmd", package = "liftr"), dir_docker)
docker_input = paste0(dir_docker, "docker.Rmd")
lift(docker_input)
## Not run:
drender(docker_input)
# view rendered document
browseURL(paste0(dir_docker, "docker.html"))
## End(Not run)

# 2. Dockerized R Markdown document with Rabix options
# Docker and Rabix are required to run the example,
# so make sure we can use `docker` and `rabix` in shell.
dir_rabix  = paste0(tempdir(), '/drender_rabix/')
dir.create(dir_rabix)
file.copy(system.file("rabix.Rmd", package = "liftr"), dir_rabix)
rabix_input = paste0(dir_rabix, "rabix.Rmd")
lift(rabix_input)
## Not run:
drender(rabix_input)
# view rendered document
browseURL(paste0(dir_rabix, "rabix.html"))
## End(Not run)
```

`lift` *Dockerize R Markdown Documents*

Description

Generate `Dockerfile` for R Markdown documents. Rabix is supported if there is certain metadata in the R Markdown document: the function will generate a `Rabixfile` containing the parsed running parameters under the output directory.

Usage

```
lift(input = NULL, output_dir = NULL)
```

Arguments

input	Input (R Markdown) file.
output_dir	Directory to output Dockerfile. If not provided, will be the same directory as input.

Details

After running lift, run drender on the document to render the Dockerized R Markdown document using Docker containers. See vignette('liftr-intro') for details about the extended YAML front-matter metadata format used by liftr.

Value

Dockerfile (and Rabixfile if possible).

Examples

```
# 1. Dockerized R Markdown document
dir_docker = paste0(tempdir(), '/lift_docker/')
dir.create(dir_docker)
file.copy(system.file("docker.Rmd", package = "liftr"), dir_docker)
# use lift() to parse Rmd and generate Dockerfile
lift(paste0(dir_docker, "docker.Rmd"))
# view generated Dockerfile
readLines(paste0(dir_docker, "Dockerfile"))

# 2. Dockerized R Markdown document with Rabix options
dir_rabix = paste0(tempdir(), '/lift_rabix/')
dir.create(dir_rabix)
file.copy(system.file("rabix.Rmd", package = "liftr"), dir_rabix)
lift(input = paste0(dir_rabix, "rabix.Rmd"))
# view generated Dockerfile
readLines(paste0(dir_rabix, "Dockerfile"))
# view generated Rabixfile
readLines(paste0(dir_rabix, "Rabixfile"))
```

Index