

# Package ‘envDocument’

August 19, 2019

**Type** Package

**Title** Document the R Working Environment

**Version** 2.4.1

**Date** 2019-08-08

**Description** Prints out information about the R working environment (system, R version,loaded and attached packages and versions) from a single function ``env\_doc()". Optionally adds information on git repository, tags, commits and remotes (if available).

**License** GPL (>= 3.0)

**Repository** CRAN

**URL** <https://github.com/dgJacks0n/envDocument>

**Suggests** git2r(>= 0.22.1), knitr(>= 1.13), stringr(>= 1.2.0), testthat(>= 1.0.2)

**RoxygenNote** 6.1.0

**VignetteBuilder** knitr

**Imports** utils, methods

**NeedsCompilation** no

**Author** Donald Jackson [aut, cre]

**Maintainer** Donald Jackson <dona1d.jackson@bms.com>

**Date/Publication** 2019-08-19 19:00:05 UTC

## R topics documented:

envDocument-package	2
collapseInfo	3
env_doc	3
getDominoInfo	4
getGitInfo	5
getPackageInfo	5
getRepo	6

getScriptInfo . . . . .	6
getScriptPath . . . . .	7
getSysInfo . . . . .	7
get_Rversion . . . . .	8
infoNotFound . . . . .	8
prettyPrintInfo . . . . .	8
remoteInfo . . . . .	9

<b>Index</b>	<b>10</b>
--------------	-----------

---

envDocument-package    *Document the environment under which an analysis was performed*

---

## Description

Provides a function `env_doc` that collects information on the script, environment and system that performed an analysis.

## Details

Returns a data frame with information on:

- R version
- Attached packages and version
- OS, version and user
- Path and modification time of the calling script
- Git commit and tag info (optional, requires `git2r`)

Results are collected in a standardized data frame that can be queried or pretty-printed using `kable` or similar methods.

## Author(s)

**Maintainer:** Donald Jackson <dona1d.jackson@bms.com>

## See Also

Useful links:

- <https://github.com/dgJacks0n/envDocument>

---

collapseInfo	<i>Collapse list of environment information into a single data frame</i>
--------------	--

---

**Description**

Collapse list of environment information into a single data frame

**Usage**

```
collapseInfo(info_list = list())
```

**Arguments**

info_list	List with environment information
-----------	-----------------------------------

---

env_doc	<i>Report the working environment in which an analysis was performed.</i>
---------	---

---

**Description**

env\_doc prints or returns a data frame with the following information: - System version (OS, version, user, working directory)

**Usage**

```
env_doc(output = c("return", "print", "table"), system = TRUE,
        version = TRUE, packages = TRUE, script = TRUE, git = TRUE,
        domino = c("auto", "on", "off"))
```

**Arguments**

output	How should output be handled? Character, values: return - return as a data frame (default); print - print to stdout; table - pretty-print using knitr::kable (requires package knitr)
system	Include OS info from <a href="#">getSysInfo?</a> Boolean, default TRUE
version	Include R version from <a href="#">getRversion?</a> Boolean, default TRUE
packages	Include attached packages with repository and version from <a href="#">getPackageInfo?</a> Boolean, default TRUE
script	Include script path and modification time from <a href="#">getScriptInfo?</a> Boolean, default TRUE
git	Include git repository information from <a href="#">getGitInfo</a> (note: requires git2r)? Boolean, default TRUE
domino	Include Domino Datalab run information from <a href="#">getDominoInfo?</a> Character, values: auto - include if available; on - include with warning if not available; off - do not include evn if available

**Details**

- R version
- Package names and versions
- Top-level script name and modification time
- Git hash, status and tag (if any; requires package git2r)

**Value**

If output = return (default): A data frame with columns for information type, variable name and value. NULL for output = print or output = table

**Examples**

```
env_doc("print") # print information to stdout
info <- env_doc() # return information as a consolidated data frame
```

---

getDominoInfo	<i>Get information about a Domino Datalab run from environment variables in the run environment</i>
---------------	---

---

**Description**

Get information about a Domino Datalab run from environment variables in the run environment

**Usage**

```
getDominoInfo(drop_vars = c("DOMINO_API_HOST",
  "DOMINO_EXECUTOR_HOSTNAME", "DOMINO_USER_API_KEY"))
```

**Arguments**

drop\_vars      Variables that will not be included in results. Character vector; see <https://support.dominodatalab.com/hc/en-us/articles/205536355-Domino-environment-variables> for a list of variables.

**Value**

data frame of Domino environment variables (renamed) and their values

---

`getGitInfo`*Get git repository information for a script*

---

**Description**

`getGitInfo` locates the git repository for the calling script (if any) and retrieves relevant information such as last commit, status (e.g. modified since last commit) and tag (if any).

**Usage**

```
getGitInfo(scriptpath = NA)
```

**Arguments**

`scriptpath` Path to script (optional, defaults to calling script from `get_scriptPath()`)

**Details**

Requires that `git2r` be installed.

**Examples**

```
git_info <- getGitInfo()
```

---

`getPackageInfo`*Get information on attached packages*

---

**Description**

`getPackageInfo` returns information on attached packages including name, version and source.

**Usage**

```
getPackageInfo()
```

**Examples**

```
packages <- getPackageInfo()
```

---

getRepo	<i>Locate the git repository where a file is tracked</i>
---------	--

---

### Description

getRepo looks the up the repository for a file and makes sure path is tracked in repo. Repositories are searched from the file directory up using [discover\\_repository](#)

### Usage

```
getRepo(testPath = NA)
```

### Arguments

testPath	Path to a file to find/check repo
----------	-----------------------------------

---

getScriptInfo	<i>Get information about the calling script</i>
---------------	---

---

### Description

getScriptInfo retrieves the file path and modification time for the calling script.

### Usage

```
getScriptInfo()
```

### Details

Note: Currently this works if the script was called via `source()`, `knitr::spin()`, Rstudio's `_compile` notebook\_ (ctrl-shift-K), `Rscript` or `R -f`.

### Examples

```
script_info <- getScriptInfo()
```

---

getScriptPath	<i>Get the path of the calling script</i>
---------------	---

---

**Description**

getScriptPath returns the full path of the script that called this function (if any) or NULL if path is not available

**Usage**

```
getScriptPath(absolute = TRUE)
```

**Arguments**

absolute	Return absolute path (absolute = TRUE, default) or path relative to working directory (FALSE)
----------	---

**Examples**

```
## Not run:  
mypath <- getScriptPath()  
  
## End(Not run)
```

---

getSysInfo	<i>Get system information.</i>
------------	--------------------------------

---

**Description**

getSysInfo returns the OS, hostname, userid and working directory for the current analysis as a data frame

**Usage**

```
getSysInfo()
```

**Examples**

```
getSysInfo()
```

---

get_Rversion	<i>Get information on R version</i>
--------------	-------------------------------------

---

**Description**

Get information on R version

**Usage**

```
get_Rversion()
```

**Examples**

```
Rversion <- getRversion()
```

---

infoNotFound	<i>Return a consistent failure result when something is not available</i>
--------------	---

---

**Description**

Return a consistent failure result when something is not available

**Usage**

```
infoNotFound()
```

---

prettyPrintInfo	<i>Output environment information as individual tables per section</i>
-----------------	--

---

**Description**

Uses `knitr::kable` to pretty-print separate tables for each 'section' (list element) in the environment information.

**Usage**

```
prettyPrintInfo(info_list = list())
```

**Arguments**

info_list	R list with environment information
-----------	-------------------------------------



---

remoteInfo	<i>Look up remotes for repository</i>
------------	---------------------------------------

---

**Description**

Look up remotes for repository

**Usage**

remoteInfo(repo)

**Arguments**

repo            A git2r repository object

# Index

`collapseInfo`, 3

`discover_repository`, 6

`env_doc`, 2, 3

`envDocument (envDocument-package)`, 2

`envDocument-package`, 2

`get_Rversion`, 8

`get_scriptpath (getScriptPath)`, 7

`getDominoInfo`, 3, 4

`getGitInfo`, 3, 5

`getPackageInfo`, 3, 5

`getRepo`, 6

`getRversion`, 3

`getScriptInfo`, 3, 6

`getScriptPath`, 7

`getSysInfo`, 3, 7

`infoNotFound`, 8

`kable`, 2

`prettyPrintInfo`, 8

`remoteInfo`, 9