

Package ‘slackr’

August 29, 2016

Type Package

Title Send Messages, Images, R Objects and Files to 'Slack'
Channels/Users

Version 1.4.2

Date 2016-04-26

Author Bob Rudis [aut, cre], Jay Jacobs [ctb], David Severski [ctb],
Quinn Weber [ctb], Konrad Karczewski [ctb], Shinya Uryu [ctb],
Gregory Jefferis [ctb], Ed Niles [ctb], Rick Saporta [ctb]

Maintainer Bob Rudis <bob@rudis.net>

Description 'Slack' <<http://slack.com/>> provides a service for teams to collaborate by sharing messages, images, links, files and more. Functions are provided that make it possible to interact with the 'Slack' platform 'API'. When you need to share information or data from R, rather than resort to copy/paste in e-mails or other services like 'Skype' <<http://www.skype.com/>>, you can use this package to send well-formatted output from multiple R objects and expressions to all teammates at the same time with little effort. You can also send images from the current graphics device, R objects, and upload files.

URL <http://github.com/hrbrmstr/slackr>

BugReports <https://github.com/hrbrmstr/slackr/issues>

License MIT + file LICENSE

Suggests testthat

Depends R (>= 3.0.0)

Imports httr (>= 0.4.0), jsonlite, ggplot2, utils, methods, dplyr,
graphics, grDevices

RoxygenNote 5.0.1

NeedsCompilation no

Repository CRAN

Date/Publication 2016-07-20 23:19:02

R topics documented:

slackr-package	2
dev_slackr	3
ggslackr	4
save_slackr	5
slackr	6
slackr_bot	8
slackr_channels	9
slackr_chtrans	10
slackr_groups	10
slackr_ims	11
slackr_msg	11
slackr_setup	13
slackr_upload	14
slackr_users	15
text_slackr	15

Index	18
--------------	-----------

slackr-package	<i>slackr-package</i>
----------------	-----------------------

Description

slackr - A package to work with the Slack.com API

Details

Mega thanks to:

- [Jay Jacobs](#)
- [David Severski](#)
- [Quinn Weber](#)
- [Konrad Karczewski](#)
- [Ed Niles](#)
- [Rick Saporta](#)

for their contributions to the package!

Check out:

- the [slackr](#) function to send messages,
- the [dev_slackr](#) function to send images (copies from current graphics device)
- the [ggslackr](#) function to send ggplot objects (without plotting to a device first)
- the [save_slackr](#) function to send R objects (as RData files)
- the [slackr_upload](#) function to send files

Author(s)

Bob Rudis (@hrbrmstr)

Examples

```
## Not run:
slackr_setup()

# send objects
slackr("iris info", head(iris), str(iris))

# send images
library(ggplot2)
qplot(mpg, wt, data=mtcars)
dev.slack("#results")

ggslackr(qplot(mpg, wt, data=mtcars))

## End(Not run)
```

dev_slackr

Send the graphics contents of the current device to a Slack channel

Description

dev.slackr sends the graphics contents of the current device to the specified Slack channel.

Usage

```
dev_slackr(channels = Sys.getenv("SLACK_CHANNEL"), ...,
  api_token = Sys.getenv("SLACK_API_TOKEN"), file = "plot")

dev.slackr(channels = Sys.getenv("SLACK_CHANNEL"), ...,
  api_token = Sys.getenv("SLACK_API_TOKEN"), file = "plot")
```

Arguments

channels	list of channels to post image to
...	other arguments passed into png device
api_token	the slack.com full API token (chr)
file	prefix for filenames (defaults to plot)

Value

httr response object from POST call

Note

You can pass in `add_user=TRUE` as part of the `...` parameters and the Slack API will post the message as your logged-in user account (this will override anything set in `username`)

Author(s)

Konrad Karczewski [ctb], Bob Rudis [aut]

References

<https://github.com/hrbrmstr/slackr/pull/12/files>

See Also

[slackrSetup](#), [save.slackr](#), [slackrUpload](#)

Examples

```
## Not run:
slackr_setup()

# base
library(maps)
map("usa")
dev_slackr("#results", file='map')

# base
barplot(VADeaths)
dev_slackr("@jayjacobs")

## End(Not run)
```

ggslackr

Post a ggplot to a Slack channel

Description

Unlike the `dev_slackr` function, this one takes a `ggplot` object, eliminating the need to have a graphics device (think use in scripts).

Usage

```
ggslackr(plot = last_plot(), channels = Sys.getenv("SLACK_CHANNEL"),
  scale = 1, width = par("din")[1], height = par("din")[2],
  units = c("in", "cm", "mm"), dpi = 300, limitsize = TRUE,
  api_token = Sys.getenv("SLACK_API_TOKEN"), file = "ggplot", ...)
```

Arguments

plot	ggplot object to save, defaults to last plot displayed
channels	list of channels to post image to
scale	scaling factor
width	width (defaults to the width of current plotting window)
height	height (defaults to the height of current plotting window)
units	units for width and height when either one is explicitly specified (in, cm, or mm)
dpi	dpi to use for raster graphics
limitsize	when TRUE (the default), ggsave will not save images larger than 50x50 inches, to prevent the common error of specifying dimensions in pixels.
api_token	the slack.com full API token (chr)
file	prefix for filenames (defaults to ggplot)
...	other arguments passed to graphics device

Value

httr response object (invisibly)

Note

You need to setup a full API token (i.e. not a webhook & not OAuth) for this to work Also, you can pass in `add_user=TRUE` as part of the `...` parameters and the Slack API will post the message as your logged-in user account (this will override anything set in `username`)

Examples

```
## Not run:  
slackr_setup()  
ggslackr(qplot(mpg, wt, data=mtcars))  
  
## End(Not run)
```

save_slackr

Save R objects to an RData file on Slack

Description

save_slackr enables you upload R objects (as an R data file) to Slack and (optionally) post them to one or more channels (if `channels` is not empty).

Usage

```
save_slackr(..., channels = "", file = "slackr",
  api_token = Sys.getenv("SLACK_API_TOKEN"))

save_slackr(..., channels = "", file = "slackr",
  api_token = Sys.getenv("SLACK_API_TOKEN"))
```

Arguments

...	objects to store in the R data file
channels	slack.com channels to save to (optional)
file	filename (without extension) to use
api_token	full API token

Value

httr response object from POST call

Note

You can pass in `add_user=TRUE` as part of the ... parameters and the Slack API will post the message as your logged-in user account (this will override anything set in `username`)

See Also

[slackr_setup](#), [dev_slackr](#), [slackr_upload](#)

Examples

```
## Not run:
slackr_setup()
save_slackr(mtcars, channels="#slackr", file="mtcars")

## End(Not run)
```

slackr

Output R expressions to a Slack channel/user

Description

Takes an `expr`, evaluates it and sends the output to a Slack chat destination. Useful for logging, messaging on long compute tasks or general information sharing.

Usage

```
slackr(..., channel = Sys.getenv("SLACK_CHANNEL"),
        username = Sys.getenv("SLACK_USERNAME"),
        icon_emoji = Sys.getenv("SLACK_ICON_EMOJI"),
        api_token = Sys.getenv("SLACK_API_TOKEN"))
```

Arguments

<code>...</code>	expressions to be sent to Slack.com
<code>channel</code>	which channel to post the message to (chr)
<code>username</code>	what user should the bot be named as (chr)
<code>icon_emoji</code>	what emoji to use (chr) "" will mean use the default
<code>api_token</code>	your full slack.com API token

Details

By default, everything but `expr` will be looked for in a "SLACK_" environment variable. You can override or just specify these values directly instead, but it's probably better to call [slackr_setup](#) first.

Note

You need a <https://www.slack.com> account and will also need to setup an API token <https://api.slack.com/> Also, you can pass in `add_user=TRUE` as part of the `...` parameters and the Slack API will post the message as your logged-in user account (this will override anything set in `username`)

See Also

[slackr_setup](#), [slackr_bot](#), [dev_slackr](#), [save_slackr](#), [slackr_upload](#)

Examples

```
## Not run:
slackr_setup()
slackr("iris info", head(iris), str(iris))

## End(Not run)
```

 slackr_bot

Send result of R expressions to a Slack channel via webhook API

Description

Takes an `expr`, evaluates it and sends the output to a Slack chat destination via the webhook API. Useful for logging, messaging on long compute tasks or general information sharing.

Usage

```
slackr_bot(..., channel = Sys.getenv("SLACK_CHANNEL"),
           username = Sys.getenv("SLACK_USERNAME"),
           icon_emoji = Sys.getenv("SLACK_ICON_EMOJI"),
           incoming_webhook_url = Sys.getenv("SLACK_INCOMING_URL_PREFIX"))
```

```
slackrBot(..., channel = Sys.getenv("SLACK_CHANNEL"),
          username = Sys.getenv("SLACK_USERNAME"),
          icon_emoji = Sys.getenv("SLACK_ICON_EMOJI"),
          incoming_webhook_url = Sys.getenv("SLACK_INCOMING_URL_PREFIX"))
```

Arguments

<code>...</code>	expressions to be sent to Slack.com
<code>channel</code>	which channel to post the message to (chr)
<code>username</code>	what user should the bot be named as (chr)
<code>icon_emoji</code>	what emoji to use (chr) "" will mean use the default
<code>incoming_webhook_url</code>	which slack.com API endpoint URL to use (see section Webhook URLs for details)

Details

By default, everything but `expr` will be looked for in a "SLACK_" environment variable. You can override or just specify these values directly instead, but it's probably better to call [slackr_setup](#) first.

This function uses the incoming webhook API. The webhook will have a default channel, username, icon etc, but these can be overridden.

Webhook URLs

Webhook URLs look like:

- <https://hooks.slack.com/services/XXXXX/XXXXX/XXXXX>

OLD STYLE WEBHOOKS ARE NO LONGER SUPPORTED

Note

You need a <https://www.slack.com> account and will also need to setup an incoming webhook: <https://api.slack.com/>. Old style webhooks are no longer supported.

See Also

[slackrSetup](#), [slackr](#), [dev_slackr](#), [save_slackr](#), [slackr_upload](#)

Examples

```
## Not run:
slackr_setup()
slackr_bot("iris info", head(iris), str(iris))

# or directly
slackr_bot("Test message", username="slackr", channel="#random",
  incoming_webhook_url="https://hooks.slack.com/services/XXXXX/XXXXX/XXXXX")

## End(Not run)
```

slackr_channels

Get a data frame of Slack channels

Description

Get a data frame of Slack channels

Usage

```
slackr_channels(api_token = Sys.getenv("SLACK_API_TOKEN"))
```

```
slackrChannels(api_token = Sys.getenv("SLACK_API_TOKEN"))
```

Arguments

api_token the Slack full API token (chr)

Value

data.table of channels

slackr_chtrans	<i>Translate vector of channel names to channel ID's for API</i>
----------------	--

Description

Given a vector of one or more channel names, it will retrieve list of active channels and try to replace channels that begin with "#" or "@" with the channel ID for that channel. Also incorporates groups.

Usage

```
slackr_chtrans(channels, api_token = Sys.getenv("SLACK_API_TOKEN"))
```

```
slackrChtrans(channels, api_token = Sys.getenv("SLACK_API_TOKEN"))
```

```
slackrChTrans(channels, api_token = Sys.getenv("SLACK_API_TOKEN"))
```

Arguments

channels	vector of channel names to parse
api_token	the slack.com full API token (chr)

Value

character vector - original channel list with # or @ channels replaced with ID's.

Author(s)

Quinn Weber [ctb], Bob Rudis [aut]

slackr_groups	<i>Get a data frame of Slack groups</i>
---------------	---

Description

Get a data frame of Slack groups

Usage

```
slackr_groups(api_token = Sys.getenv("SLACK_API_TOKEN"))
```

```
slackrGroups(api_token = Sys.getenv("SLACK_API_TOKEN"))
```

Arguments

api_token	the Slackfull API token (chr)
-----------	-------------------------------

Value

data.frame of channels

slackr_ims	<i>Get a data frame of Slack IM ids</i>
------------	---

Description

Get a data frame of Slack IM ids

Usage

```
slackr_ims(api_token = Sys.getenv("SLACK_API_TOKEN"))
```

```
slackrImS(api_token = Sys.getenv("SLACK_API_TOKEN"))
```

Arguments

api_token the Slack full API token (chr)

Value

data.frame of im ids and user names

Author(s)

Quinn Weber [aut], Bob Rudis [ctb]

References

<https://github.com/hrbrmstr/slackr/pull/13>

slackr_msg	<i>Output R expressions to a Slack channel/user</i>
------------	---

Description

Takes an expr, evaluates it and sends the output to a Slack chat destination. Useful for logging, messaging on long compute tasks or general information sharing.

Usage

```
slackr_msg(txt = "", channel = Sys.getenv("SLACK_CHANNEL"),
           username = Sys.getenv("SLACK_USERNAME"),
           icon_emoji = Sys.getenv("SLACK_ICON_EMOJI"),
           api_token = Sys.getenv("SLACK_API_TOKEN"), ...)

slackrMsg(txt = "", channel = Sys.getenv("SLACK_CHANNEL"),
          username = Sys.getenv("SLACK_USERNAME"),
          icon_emoji = Sys.getenv("SLACK_ICON_EMOJI"),
          api_token = Sys.getenv("SLACK_API_TOKEN"), ...)
```

Arguments

txt	text message to send to Slack. If a character vector of length > 1 is passed in, they will be combined and separated by newlines.
channel	which channel to post the message to (chr)
username	what user should the bot be named as (chr)
icon_emoji	what emoji to use (chr) "" will mean use the default
api_token	your full slack.com API token
...	other arguments passed to the Slack API chat.postMessage call

Details

By default, everything but expr will be looked for in a "SLACK_" environment variable. You can override or just specify these values directly instead, but it's probably better to call [slackrSetup](#) first.

Note

You need a <https://www.slack.com> account and will also need to setup an API token <https://api.slack.com/> Also, you can pass in add_user=TRUE as part of the ... parameters and the Slack API will post the message as your logged-in user account (this will override anything set in username)

See Also

[slackr_setup](#), [slackr_bot](#), [dev_slackr](#), [save_slackr](#), [slackr_upload](#)

Examples

```
## Not run:
slackr_setup()
slackr_msg("Hi")

## End(Not run)
```

slackr_setup	<i>Setup environment variables for Slack API access</i>
--------------	---

Description

Initialize all the environment variables `slackr` will need to use to work properly.

Usage

```
slackr_setup(channel = "#general", username = "slackr", icon_emoji = "",
  incoming_webhook_url = "", api_token = "", config_file = "~/.slackr",
  echo = FALSE)
```

```
slackrSetup(channel = "#general", username = "slackr", icon_emoji = "",
  incoming_webhook_url = "", api_token = "", config_file = "~/.slackr",
  echo = FALSE)
```

Arguments

<code>channel</code>	default channel to send the output to (chr) defaults to #general
<code>username</code>	the username output will appear from (chr) defaults to slackr
<code>icon_emoji</code>	which emoji picture to use (chr) defaults to none (can be left blank in config file as well)
<code>incoming_webhook_url</code>	the slack.com URL prefix to use (chr) defaults to none
<code>api_token</code>	the Slack full API token (chr)
<code>config_file</code>	a configuration file (DCF) - see read.dcf - format with the config values.
<code>echo</code>	display the configuration variables (bool) initially FALSE

Details

By default, `slackr` (and other functions) will use the #general room and a username of `slackr()` with no emoji.

If a valid file is found at the locaiton pointed to by `config_file`, the values there will be used. The fields should be specified as such in the file:

```
api_token: YOUR_FULL_API_TOKEN
channel: #general
username: slackr
incoming_webhook_url: https://hooks.slack.com/services/XXXXX/XXXXX/XXXXX
```

Note

You need a **Slack** account and all your API URLs & tokens setup to use this package.

See Also

[slackr](#), [dev_slackr](#), [save_slackr](#), [slackr_upload](#)

Examples

```
## Not run:
# reads from default file (i.e. ~/.slackr)
slackr_setup()

# reads from alternate config
slackr_setup(config_file="/path/to/my/slackrconfig)

# the hard way
slackr_setup(channel="#code",
             incoming_webhook_url="https://hooks.slack.com/services/XXXXX/XXXXX/XXXXX")

## End(Not run)
```

slackr_upload	<i>Send a file to Slack</i>
---------------	-----------------------------

Description

slackrUpload enables you upload files to Slack and (optionally) post them to one or more channels (if channels is not empty).

Usage

```
slackr_upload(filename, title = basename(filename),
             initial_comment = basename(filename), channels = "",
             api_token = Sys.getenv("SLACK_API_TOKEN"))

slackrUpload(filename, title = basename(filename),
             initial_comment = basename(filename), channels = "",
             api_token = Sys.getenv("SLACK_API_TOKEN"))
```

Arguments

filename	path to file
title	title on slack (optional - defaults to filename)
initial_comment	comment for file on slack (optional - defaults to filename)
channels	slack.com channels to save to (optional)
api_token	full API token

Value

httr response object from POST call (invisibly)

Author(s)

Quinn Weber [ctb], Bob Rudis [aut]

References

<https://github.com/hrbrmstr/slackr/pull/15/files>

See Also

[slackr_setup](#), [dev_slackr](#), [save_slackr](#)

slackr_users	<i>Get a data frame of Slack users</i>
--------------	--

Description

Get a data frame of Slack users

Usage

```
slackr_users(api_token = Sys.getenv("SLACK_API_TOKEN"))
```

```
slackrUsers(api_token = Sys.getenv("SLACK_API_TOKEN"))
```

Arguments

api_token the Slack full API token (chr)

Value

data.frame of users

text_slackr	<i>Sends basic text to a slack channel. Calls the chat.postMessage method on the Slack Web API. Information on this method can be found here: https://api.slack.com/methods/chat.postMessage</i>
-------------	---

Description

Sends basic text to a slack channel. Calls the chat.postMessage method on the Slack Web API. Information on this method can be found here: <https://api.slack.com/methods/chat.postMessage>

Usage

```
text_slackr(text, ..., preformatted = TRUE,
            channel = Sys.getenv("SLACK_CHANNEL"),
            username = Sys.getenv("SLACK_USERNAME"),
            icon_emoji = Sys.getenv("SLACK_ICON_EMOJI"),
            api_token = Sys.getenv("SLACK_API_TOKEN"))
```

```
textSlackr(text, ..., preformatted = TRUE,
            channel = Sys.getenv("SLACK_CHANNEL"),
            username = Sys.getenv("SLACK_USERNAME"),
            icon_emoji = Sys.getenv("SLACK_ICON_EMOJI"),
            api_token = Sys.getenv("SLACK_API_TOKEN"))
```

Arguments

text	The character vector to be posted
...	Optional arguments such as: as_user, parse, unfurl_links, etc.
preformatted	Should the text be sent as preformatted text. Defaults to TRUE
channel	The name of the channels to which the DataTable should be sent. Prepend channel names with a hashtag. Prepend private-groups with nothing. Prepend direct messages with an @
username	what user should the bot be named as (chr)
icon_emoji	what emoji to use (chr) "" will mean use the default
api_token	your full slack.com API token

Value

httr response object (invisibly)

Note

You can pass in add_user=TRUE as part of the ... parameters and the Slack API will post the message as your logged-in user account (this will override anything set in username)

Author(s)

Quinn Weber [aut], Bob Rudis [ctb]

References

<https://github.com/hrbrmstr/slackr/pull/11>

See Also

<https://api.slack.com/methods/chat.postMessage>

Examples

```
## Not run:  
slackr_setup()  
text_slackr('hello world', as_user=TRUE)  
  
## End(Not run)
```

Index

`dev.slackr` (`dev_slackr`), 3
`dev_slackr`, 2, 3, 4, 6, 7, 9, 12, 14, 15

`ggslackr`, 2, 4

`read.dcf`, 13

`save.slackr`, 4
`save.slackr` (`save_slackr`), 5
`save_slackr`, 2, 5, 7, 9, 12, 14, 15
`slackr`, 2, 6, 9, 13, 14
`slackr-package`, 2
`slackr_bot`, 7, 8, 12
`slackr_channels`, 9
`slackr_chtrans`, 10
`slackr_groups`, 10
`slackr_ims`, 11
`slackr_msg`, 11
`slackr_setup`, 6–8, 12, 13, 15
`slackr_upload`, 2, 6, 7, 9, 12, 14, 14
`slackr_users`, 15
`slackrBot` (`slackr_bot`), 8
`slackrChannels` (`slackr_channels`), 9
`slackrChTrans` (`slackr_chtrans`), 10
`slackrChtrans` (`slackr_chtrans`), 10
`slackrGroups` (`slackr_groups`), 10
`slackrIms` (`slackr_ims`), 11
`slackrMsg` (`slackr_msg`), 11
`slackrSetup`, 4, 9, 12
`slackrSetup` (`slackr_setup`), 13
`slackrUpload`, 4
`slackrUpload` (`slackr_upload`), 14
`slackrUsers` (`slackr_users`), 15

`text_slackr`, 15
`textSlackr` (`text_slackr`), 15