

Package: IRkernel (via r-universe)

August 27, 2024

Title Native R Kernel for the 'Jupyter Notebook'

Description The R kernel for the 'Jupyter' environment executes R code which the front-end ('Jupyter Notebook' or other front-ends) submits to the kernel via the network.

Version 1.3.2.9000

URL <https://irkernel.github.io>

BugReports <https://github.com/IRkernel/IRkernel/issues/>

Depends R (>= 3.2.0)

Suggests testthat, roxygen2

SystemRequirements jupyter, jupyter_kernel_test (Python package for testing)

License MIT + file LICENSE

Encoding UTF-8

Imports repr (>= 0.4.99), methods, evaluate (>= 0.10), IRdisplay (>= 0.3.0.9999), pbdZMQ (>= 0.2-1), crayon, jsonlite (>= 0.9.6), uuid, digest

Collate 'class_unions.r' 'logging.r' 'comm_manager.r' 'compat.r' 'completion.r' 'environment_runtime.r' 'environment_shadow.r' 'options.r' 'execution.r' 'handlers.r' 'help.r' 'installspec.r' 'utils.r' 'kernel.r' 'main.r' 'onload.r'

RoxygenNote 7.2.3

Repository <https://irkernel.r-universe.dev>

RemoteUrl <https://github.com/irkernel/irkernel>

RemoteRef HEAD

RemoteSha 124f2347f15cfadeabc6738868e05e6087f4c456

Contents

Comm-class	2
CommManager-class	2

comm_manager	2
installspec	3
jupyter_option_defaults	4
log	5
main	5

Index	6
--------------	----------

Comm-class	<i>The Comm</i>
------------	-----------------

Description

Has methods able to register and handle message callbacks

CommManager-class	<i>The CommManager</i>
-------------------	------------------------

Description

Has methods able to register comms/targets and process comm messages

comm_manager	<i>Get global CommManager instance</i>
--------------	--

Description

Get global CommManager instance

Usage

comm_manager()

Value

[CommManager](#) instance if a kernel is running, else NULL

`installspec`*Install the kernelspec to tell Jupyter about IRkernel.*

Description

This can be called multiple times for different R interpreter, but you have to give a different name (and displayname to see a difference in the notebook UI). If the same name is give, it will overwrite older versions of the kernel spec with that name!

Usage

```
installspec(  
  user = NULL,  
  name = "ir",  
  displayname = "R",  
  rprofile = NULL,  
  prefix = NULL,  
  sys_prefix = NULL,  
  verbose = getOption("verbose"),  
  env = NULL  
)
```

Arguments

<code>user</code>	Install into user directory (<code>\$XDG_DATA_HOME/jupyter/kernels</code>) or globally? (default: NULL but treated as TRUE if "prefix" is not specified)
<code>name</code>	The name of the kernel (default "ir")
<code>displayname</code>	The name which is displayed in the notebook (default: "R")
<code>rprofile</code>	(optional) Path to kernel-specific Rprofile (defaults to system-level settings)
<code>prefix</code>	(optional) Path to alternate directory to install kernelspec into (default: NULL)
<code>sys_prefix</code>	(optional) Install kernelspec using the <code>--sys-prefix</code> option of the currently detected jupyter (default: NULL)
<code>verbose</code>	(optional) If FALSE, silence output of <code>install</code>
<code>env</code>	(optional) Named list of environment variables to set in the kernel (default: NULL)

Value

Exit code of the jupyter kernelspec `install` call.

jupyter_option_defaults

An R kernel for Jupyter.

Description

Jupyter speaks a JSON+ZMQ protocol to a 'kernel' which is responsible for executing code. This package is a kernel for the R language.

Usage

```
jupyter_option_defaults
```

Format

An object of class `list` of length 7.

Options

The following can be set/read via `options(opt.name = ...) /getOption('opt.name')`

`jupyter.log_level` 1L (errors), 2L (warnings), or 3L (debug). 1L is the default.

`jupyter.pager_classes` Classes to use the pager for instead of displaying them inline. Default: `help pages`

`jupyter.in_kernel` TRUE if this code is executed in a running kernel. Set to pretend being/not being in a kernel

`jupyter.rich_display` Use more than just text display

`jupyter.display_mimetypes` The formats emitted when any return value is to be displayed (default: all mimetypes listed [here](#))

`jupyter.plot_mimetypes` The plot formats emitted to the frontend when a plot is displayed. (default: `image/png` and `application/pdf`)

`jupyter.plot_scale` The ratio (notebook PPI / `repr.plot.res`). E.g.: With the defaults `repr.plot.res=120` px/in (PPI) and `jupyter.plot_scale=2`, a `1in×1in` image will be displayed as a `0.5in×0.5in`, 240 PPI image. (default: 2, fit for retina displays)

See Also

[installspec](#)

log	<i>Kernel logging functions</i>
-----	---------------------------------

Description

A set of exported logging utilities that have the capability to be used in upstream projects. Log level and log file can be set via R package options e.g. `options(jupyter.log_level = 2L)` or from the environment variables `JUPYTER_LOG_LEVEL` and `JUPYTER_LOGFILE`.

Usage

```
log_debug(...)
```

```
log_info(...)
```

```
log_error(...)
```

Arguments

```
...          message to log
```

main	<i>Initialise and run the kernel</i>
------	--------------------------------------

Description

Initialise and run the kernel

Usage

```
main(connection_file = "")
```

Arguments

```
connection_file  
                The path to the Jupyter connection file, written by the frontend
```

Index

* datasets

jupyter_option_defaults, 4

Comm (Comm-class), 2

Comm-class, 2

comm_manager, 2

CommManager, 2

CommManager (CommManager-class), 2

CommManager-class, 2

installspec, 3, 4

IRkernel (jupyter_option_defaults), 4

IRkernel-options

(jupyter_option_defaults), 4

IRkernel-package

(jupyter_option_defaults), 4

jupyter_option_defaults, 4

log, 5

log_debug (log), 5

log_error (log), 5

log_info (log), 5

main, 5