

# Package ‘BrowserViz’

August 20, 2025

**Type** Package

**Title** BrowserViz: interactive R/browser graphics using websockets and JSON

**Version** 2.30.0

**Date** 2023-09-12

**Author** Paul Shannon

**Maintainer** Arkadiusz Gladki <gladki.arkadiusz@gmail.com>

**Depends** R (>= 3.5.0), jsonlite (>= 1.5), httpuv(>= 1.5.0)

**Imports** methods, BiocGenerics

**Suggests** RUnit, BiocStyle, knitr, rmarkdown

**Description** Interactive graphics in a web browser from R, using websockets and JSON.

**License** GPL-2

**URL** <https://gladkia.github.io/BrowserViz/>

**BugReports** <https://github.com/gladkia/BrowserViz/issues>

**LazyLoad** yes

**biocViews** Visualization, ThirdPartyClient

**NeedsCompilation** no

**VignetteBuilder** knitr

**Encoding** UTF-8

**RoxygenNote** 7.2.1

**git\_url** <https://git.bioconductor.org/packages/BrowserViz>

**git\_branch** RELEASE\_3\_21

**git\_last\_commit** a71a830

**git\_last\_commit\_date** 2025-04-15

**Repository** Bioconductor 3.21

**Date/Publication** 2025-08-20

Contents

|                                                   |           |
|---------------------------------------------------|-----------|
| addRMessageHandler . . . . .                      | 2         |
| browserResponseReady,BrowserViz-method . . . . .  | 3         |
| BrowserViz constructor . . . . .                  | 3         |
| BrowserViz-class . . . . .                        | 4         |
| closeWebSocket,BrowserViz-method . . . . .        | 5         |
| dispatchMessage . . . . .                         | 5         |
| displayHTMLInDiv,BrowserViz-method . . . . .      | 6         |
| fromJSON . . . . .                                | 6         |
| getBrowserInfo,BrowserViz-method . . . . .        | 7         |
| getBrowserResponse,BrowserViz-method . . . . .    | 7         |
| getBrowserWindowSize,BrowserViz-method . . . . .  | 8         |
| getBrowserWindowTitle,BrowserViz-method . . . . . | 8         |
| handleResponse . . . . .                          | 9         |
| port,BrowserViz-method . . . . .                  | 9         |
| ready,BrowserViz-method . . . . .                 | 10        |
| roundTripTest,BrowserViz-method . . . . .         | 10        |
| send,BrowserViz-method . . . . .                  | 11        |
| setBrowserWindowTitle,BrowserViz-method . . . . . | 11        |
| show,BrowserViz-method . . . . .                  | 12        |
| toJSON . . . . .                                  | 12        |
| wait,BrowserViz-method . . . . .                  | 13        |
| webBrowserAvailableForTesting . . . . .           | 13        |
| <b>Index</b>                                      | <b>14</b> |

---

|                    |                                                                     |
|--------------------|---------------------------------------------------------------------|
| addRMessageHandler | <i>Supply the name of a function to call, identified by its key</i> |
|--------------------|---------------------------------------------------------------------|

---

Description

Supply the name of a function to call, identified by its key

Usage

addRMessageHandler(key, functionName)

Arguments

- |              |                    |
|--------------|--------------------|
| key          | A character string |
| functionName | A character string |

---

```
browserResponseReady,BrowserViz-method
browserResponseReady
```

---

**Description**

browserResponseReady

**Usage**

```
## S4 method for signature 'BrowserViz'
browserResponseReady(obj)
```

**Arguments**

obj                    An object of class BrowserViz

---

```
BrowserViz constructor
Constructor for BrowserViz
```

---

**Description**

This constructor function:

- creates the BrowserViz object
- initializes the httpuv web server
- prepares that web server to additionally handle websocket traffic
- loads a "browserFile" - an html/javascript/css web page to communicate with in your web browser
- opens websocket communication between your R session and your browser
- installs an optional "httpQueryProcessingFunction" to handle http (non-websocket) requests.

**Usage**

```
BrowserViz(
  host = "localhost",
  portRange = 10000:10100,
  title = "BrowserViz",
  browserFile,
  quiet = TRUE,
  httpQueryProcessingFunction = NULL
)
```

**Arguments**

|                             |                                                                                                                      |
|-----------------------------|----------------------------------------------------------------------------------------------------------------------|
| host                        | character The constructor will open an http/websocket port here for web browsers to connect to. localhost by default |
| portRange                   | The constructor looks for a free websocket port in this range. 10000:10100 by default                                |
| title                       | Used for the web browser window, "igvR" by default                                                                   |
| browserFile                 | The full path to the bundled html, js and libraries, and css which constitute the browser app                        |
| quiet                       | A logical variable controlling verbosity during execution                                                            |
| httpQueryProcessingFunction | a function, default NULL, provides subclasses with the opportunity to execute code on the http server created here.  |

**Value**

An object of the BrowserViZ class

---

|                  |                                                                                                                                                               |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| BrowserViz-class | <i>BrowserViz: a base class providing simple, extensible message passing between your R session and your web browser, for interactive data visualization.</i> |
|------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|

---

**Description**

Many of the best interactive graphics capabilities available today are written in Javascript and run in a web browser. BrowserViz makes these capabilities available in R, using websockets for message passing back and forth between R and the browser. This class connects your R session to your web browser via websockets, using the R httpuv library, which in turn uses the Rook webserver.

BrowserViz is a concrete base class, in that instances can be constructed and run - which we do for testing. The primary use of this BrowserViz is to be subclassed: to facilitate the creation of new browser-based, R-connected interactive graphics capabilities embodied in R packages, written by programmers with some skill in both R and Javascript. Two examples of this can be found in these Bioconductor packages <https://bioconductor.org/packages/devel/bioc/html/igvR.html> and <https://bioconductor.org/packages/devel/bioc/html/RCyjs.html>.

**Slots**

|                     |                                                            |
|---------------------|------------------------------------------------------------|
| uri                 | The http location at which this modest webserver runs      |
| port                | An integer port number for the http connection             |
| websocketConnection | An environment managed by the httpuv library on our behalf |
| quiet               | Logical variable controlling verbosity during execution    |

**See Also**[BrowserViz](#)

An S4 class to create and manage a modest webserver for websocket message passing between R and Javascript running in your web browser

---

 closeWebSocket,BrowserViz-method

*Close the websocket connection - between your R session and your web browser.*

---

**Description**

Close the websocket connection - between your R session and your web browser.

**Usage**

```
## S4 method for signature 'BrowserViz'
closeWebSocket(obj)
```

**Arguments**

obj                      An object of class BrowserViz

---

|                 |                                                                                    |
|-----------------|------------------------------------------------------------------------------------|
| dispatchMessage | <i>Route the message coming in from the browser to the appropriate R function.</i> |
|-----------------|------------------------------------------------------------------------------------|

---

**Description**

Route the message coming in from the browser to the appropriate R function.

**Usage**

```
dispatchMessage(ws, msg, quiet)
```

**Arguments**

|       |                                           |
|-------|-------------------------------------------|
| ws    | a websocket connectin                     |
| msg   | the JSON-encoded message from the browser |
| quiet | logical TRUE or FALSE                     |

displayHTMLInDiv, BrowserViz-method

*Ask the browser to display html markup in the specified div*

---

### Description

Ask the browser to display html markup in the specified div

### Usage

```
## S4 method for signature 'BrowserViz'  
displayHTMLInDiv(obj, htmlText, div.id)
```

### Arguments

|          |                                     |
|----------|-------------------------------------|
| obj      | An object of class BrowserViz       |
| htmlText | A character string with HTML markup |
| div.id   | A character string                  |

---

fromJSON

*Transform JSON string into a native R object*

---

### Description

Transform JSON string into a native R object

### Usage

```
fromJSON(...)
```

### Arguments

|     |                                         |
|-----|-----------------------------------------|
| ... | Extra arguments passed to this function |
|-----|-----------------------------------------|

### Value

a native R data structure

### Examples

```
fromJSON(toJSON(data.frame(a=8:10, b=LETTERS[8:10], stringsAsFactors=FALSE)))
```

---

getBrowserInfo,BrowserViz-method

*Retrieve basic attributes of the attached web browser.*

---

### **Description**

Retrieve basic attributes of the attached web browser.

### **Usage**

```
## S4 method for signature 'BrowserViz'  
getBrowserInfo(obj)
```

### **Arguments**

obj                    An object of class BrowserViz

---

getBrowserResponse,BrowserViz-method

*Retrieve the response sent by the browser*

---

### **Description**

Retrieve the response sent by the browser

### **Usage**

```
## S4 method for signature 'BrowserViz'  
getBrowserResponse(obj)
```

### **Arguments**

obj                    An object of class BrowserViz

---

getBrowserWindowSize,BrowserViz-method

*Supply the name of a function to call, identified by its key*

---

### Description

Supply the name of a function to call, identified by its key

### Usage

```
## S4 method for signature 'BrowserViz'  
getBrowserWindowSize(obj)
```

### Arguments

|     |                               |
|-----|-------------------------------|
| obj | An object of class BrowserViz |
|-----|-------------------------------|

---

getBrowserWindowTitle,BrowserViz-method

*Supply the name of a function to call, identified by its key*

---

### Description

Supply the name of a function to call, identified by its key

### Usage

```
## S4 method for signature 'BrowserViz'  
getBrowserWindowTitle(obj)
```

### Arguments

|     |                               |
|-----|-------------------------------|
| obj | An object of class BrowserViz |
|-----|-------------------------------|

---

|                |                       |
|----------------|-----------------------|
| handleResponse | <i>handleResponse</i> |
|----------------|-----------------------|

---

**Description**

handleResponse

**Usage**

handleResponse(ws, msg)

**Arguments**

- |     |                                                           |
|-----|-----------------------------------------------------------|
| ws  | websocket connection                                      |
| msg | the JSON-encoded character string returned by the browser |

---

|                            |
|----------------------------|
| port,BrowserViz-method     |
| <i>Get the port number</i> |

---

**Description**

Get the port number

**Usage**

```
## S4 method for signature 'BrowserViz'  
port(obj)
```

**Arguments**

- |     |                               |
|-----|-------------------------------|
| obj | An object of class BrowserViz |
|-----|-------------------------------|

**Value**

the port number use in the websocket connection, a numeric value.

---

ready,BrowserViz-method

*Is the websocket connection to the browser ready for use?*

---

### Description

Is the websocket connection to the browser ready for use?

### Usage

```
## S4 method for signature 'BrowserViz'  
ready(obj)
```

### Arguments

obj                    An object of class BrowserViz

---

roundTripTest,BrowserViz-method

*Send data to the browser, ensure that it is returned accurately.*

---

### Description

Send data to the browser, ensure that it is returned accurately.

### Usage

```
## S4 method for signature 'BrowserViz'  
roundTripTest(obj, ...)
```

### Arguments

obj                    An object of class BrowserViz  
...                    other arguments

---

`send,BrowserViz-method`*Send the specified message to the browser*

---

**Description**

Send the specified message to the browser

**Usage**

```
## S4 method for signature 'BrowserViz'  
send(obj, msg)
```

**Arguments**

|                  |                                                                                                                                                                       |
|------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>obj</code> | An object of class <code>BrowserViz</code>                                                                                                                            |
| <code>msg</code> | A list with four fields: <code>cmd</code> : "someCommand", <code>status</code> : "request", <code>callback</code> : "someFunction", <code>payload</code> : "someData" |

---

`setBrowserWindowTitle,BrowserViz-method`*Supply the name of a function to call, identified by its key*

---

**Description**

Supply the name of a function to call, identified by its key

**Usage**

```
## S4 method for signature 'BrowserViz'  
setBrowserWindowTitle(obj, newTitle)
```

**Arguments**

|                       |                                            |
|-----------------------|--------------------------------------------|
| <code>obj</code>      | An object of class <code>BrowserViz</code> |
| <code>newTitle</code> | A character string                         |

---

`show, BrowserViz-method`

*Display the core attributes of the BrowserViz object to stdout*

---

### Description

Display the core attributes of the BrowserViz object to stdout

### Usage

```
## S4 method for signature 'BrowserViz'
show(object)
```

### Arguments

`object`                      An object of class BrowserViz

---

`toJSON`

*Transform an R data structure into JSON*

---

### Description

The semantics of toJSON changed between RJSONIO and jsonlite: in the latter, scalars are promoted to arrays of length 1. rather than change our javascript code, and since such promotion – while sensible in the context of R – strikes me as gratuitous, I follow jeroen ooms suggestion, creating this wrapper

### Usage

```
toJSON(..., auto_unbox = TRUE)
```

### Arguments

`...`                      Extra arguments passed to this function  
`auto_unbox`              Logical

### Value

a character string with the JSON representation of the R object

### Examples

```
toJSON(data.frame(a=8:10, b=LETTERS[8:10], stringsAsFactors=FALSE))
```

---

`wait,BrowserViz-method`*Pause for the specified number of milliseconds*

---

**Description**

Pause for the specified number of milliseconds

**Usage**

```
## S4 method for signature 'BrowserViz'  
wait(obj, msec)
```

**Arguments**

|                   |                                            |
|-------------------|--------------------------------------------|
| <code>obj</code>  | An object of class <code>BrowserViz</code> |
| <code>msec</code> | Numeric                                    |

---

`webBrowserAvailableForTesting`*Is there a web browser available for testing?*

---

**Description**

This package's unit tests require a web browser to connect to. our heuristic, though not bullet proof, is that one of three conditions must be met Supply the name of a function to call, identified by its key

**Usage**

```
webBrowserAvailableForTesting()
```

**Value**

Logical TRUE or FALSE

# Index

.BrowserViz (BrowserViz-class), [4](#)  
addRMessageHandler, [2](#)  
browserResponseReady  
    (browserResponseReady, BrowserViz-method),  
    [3](#)  
browserResponseReady, BrowserViz-method,  
    [3](#)  
BrowserViz, [5](#)  
BrowserViz (BrowserViz constructor), [3](#)  
BrowserViz constructor, [3](#)  
BrowserViz-class, [4](#)  
closeWebSocket  
    (closeWebSocket, BrowserViz-method),  
    [5](#)  
closeWebSocket, BrowserViz-method, [5](#)  
dispatchMessage, [5](#)  
displayHTMLInDiv  
    (displayHTMLInDiv, BrowserViz-method),  
    [6](#)  
displayHTMLInDiv, BrowserViz-method, [6](#)  
fromJSON, [6](#)  
getBrowserInfo  
    (getBrowserInfo, BrowserViz-method),  
    [7](#)  
getBrowserInfo, BrowserViz-method, [7](#)  
getBrowserResponse  
    (getBrowserResponse, BrowserViz-method),  
    [7](#)  
getBrowserResponse, BrowserViz-method,  
    [7](#)  
getBrowserWindowSize  
    (getBrowserWindowSize, BrowserViz-method),  
    [8](#)  
getBrowserWindowSize, BrowserViz-method,  
    [8](#)  
getBrowserWindowTitle  
    (getBrowserWindowTitle, BrowserViz-method),  
    [8](#)  
getBrowserWindowTitle, BrowserViz-method,  
    [8](#)  
handleResponse, [9](#)  
port (port, BrowserViz-method), [9](#)  
port, BrowserViz-method, [9](#)  
ready (ready, BrowserViz-method), [10](#)  
ready, BrowserViz-method, [10](#)  
roundTripTest  
    (roundTripTest, BrowserViz-method),  
    [10](#)  
roundTripTest, BrowserViz-method, [10](#)  
send (send, BrowserViz-method), [11](#)  
send, BrowserViz-method, [11](#)  
setBrowserWindowTitle  
    (setBrowserWindowTitle, BrowserViz-method),  
    [11](#)  
setBrowserWindowTitle, BrowserViz-method,  
    [11](#)  
show (show, BrowserViz-method), [12](#)  
show, BrowserViz-method, [12](#)  
toJSON, [12](#)  
wait (wait, BrowserViz-method), [13](#)  
wait, BrowserViz-method, [13](#)  
webBrowserAvailableForTesting, [13](#)