

# Package ‘webglobe’

September 16, 2020

**Type** Package

**Title** 3D Interactive Globes

**Version** 1.0.3

**Date** 2020-09-13

**Maintainer** Richard Barnes <rijard.barnes@gmail.com>

**NeedsCompilation** no

**Depends** R (>= 3.4.0), gejsonio (>= 0.3.2), jsonlite (>= 1.4), httpuv (>= 1.3.3)

**Suggests** knitr, R.rsp (>= 0.41.0), rmarkdown (>= 1.0)

**License** MIT + file LICENCE

**Description** Displays geospatial data on an interactive 3D globe in the web browser.

**URL** <https://github.com/r-barnes/webglobe/>

**BugReports** <https://github.com/r-barnes/webglobe/>

**RoxygenNote** 7.1.1

**VignetteBuilder** knitr, R.rsp

**Author** Richard Barnes [aut, cre]

**Repository** CRAN

**Date/Publication** 2020-09-15 22:20:03 UTC

## R topics documented:

|                             |   |
|-----------------------------|---|
| <code>+.webglobe</code>     | 2 |
| <code>is.webglobe</code>    | 3 |
| <code>print.webglobe</code> | 3 |
| <code>webglobe</code>       | 4 |
| <code>wgbar</code>          | 5 |
| <code>wgcamcenter</code>    | 5 |
| <code>wgcamhome</code>      | 6 |
| <code>wgclear</code>        | 7 |
| <code>wgimmediate</code>    | 7 |

|                           |    |
|---------------------------|----|
| wgimmediate_set . . . . . | 8  |
| wgpoints . . . . .        | 8  |
| wgpolygondf . . . . .     | 9  |
| wgport . . . . .          | 10 |
| wgtitle . . . . .         | 10 |
| wgunimmediate . . . . .   | 11 |

|              |           |
|--------------|-----------|
| <b>Index</b> | <b>12</b> |
|--------------|-----------|

---

|            |                     |
|------------|---------------------|
| +.webglobe | <i>Send command</i> |
|------------|---------------------|

---

## Description

Send a command to a webglobe

## Usage

```
## S3 method for class 'webglobe'
wg + x
```

## Arguments

|    |                 |
|----|-----------------|
| wg | Webglobe        |
| x  | Command to send |

## Value

The same webglobe

## Examples

```
## Not run:
library(webglobe)
wg<-webglobe(immediate=TRUE)
wg + wgcLEAR()

## End(Not run)
```

---

|             |                          |
|-------------|--------------------------|
| is.webglobe | <i>Is it a webglobe?</i> |
|-------------|--------------------------|

---

**Description**

Determine if an object is a webglobe

**Usage**

```
is.webglobe(x)
```

**Arguments**

|   |                                     |
|---|-------------------------------------|
| x | The object that might be a webglobe |
|---|-------------------------------------|

**Value**

TRUE or FALSE

**Examples**

```
## Not run:  
library(webglobe)  
wg<-webglobe(immediate=TRUE)  
is.webglobe(wg)  
  
## End(Not run)
```

---

|                |                           |
|----------------|---------------------------|
| print.webglobe | <i>Display a webglobe</i> |
|----------------|---------------------------|

---

**Description**

Displays a webglobe. If the webglobe is immediate, then a browser window containing it should already be open; in this case, the webglobe's address is returned. If the webglobe is not immediate then a new browser is open and the cached pipeline is sent to it.

**Usage**

```
## S3 method for class 'webglobe'  
print(x, ...)
```

**Arguments**

|     |              |
|-----|--------------|
| x   | The webglobe |
| ... | Ignored      |

**Value**

NA

**Examples**

```
## Not run:  
library(webglobe)  
wg<-webglobe()  
wg  
  
## End(Not run)
```

---

webglobe

*Make a new webglobe*

---

**Description**

Constructs a new webglobe and starts its server

**Usage**

```
webglobe(immediate = FALSE)
```

**Arguments**

`immediate` Whether the webglobe should immediately show the results of graphics commands or additively cache them. 'immediate' mode can be used to experimentally build up a pipeline. Once established this can be stored in a non-immediate webglobe for easy access later

**Value**

A webglobe object

**Examples**

```
## Not run:  
library(webglobe)  
wg<-webglobe(immediate=TRUE)  
  
## End(Not run)
```

---

wgbars *Plot bars from the surface*

---

### Description

Plots bars rising upwards from points on the Earth's surface

### Usage

```
wgbars(lat, lon, alt = 3e+06, colour = "blue", width = 3)
```

### Arguments

|        |   |
|--------|---|
| lat    | Latitude of the bars' bases, in degrees               |
| lon    | Longitude of the bars' bases, in degrees              |
| alt    | Altitude of the bars' tops, may be one or many values |
| colour | Colour of the bars, may be one or many values         |
| width  | Width of bars, may be one or many values              |

### Value

A webglobe command

### Examples

```
## Not run:
library(webglobe)
data(quakes) #Load up some data
wg <- webglobe(immediate=FALSE) #Make a webglobe
wg <- wg + wgbars(quakes$lat, quakes$lon, alt=1.5e6*quakes$mag/10) #Plot quakes
wg <- wg + wgcamcenter(-33.35, 142.96, 8000) #Move camera
wg

## End(Not run)
```

---

wgcamcenter *Camera: Center on a point*

---

### Description

Centers the camera on a point

### Usage

```
wgcamcenter(lat, lon, alt = NA)
```

**Arguments**

|     |   |
|-----|---|
| lat | Latitude of the center point, in degrees    |
| lon | Longitude of the center point, in degrees   |
| alt | Altitude of the center point, in kilometres |

**Value**

A webglobe command

**Examples**

```
## Not run:  
library(webglobe)  
wg<-webglobe(immediate=TRUE)  
wg+wgcamcenter(45,-93,5000)  
  
## End(Not run)
```

---

wgcamhome

*Camera: Send home*

---

**Description**

Send the camera to its home location

**Usage**

```
wgcamhome()
```

**Value**

A webglobe command

**Examples**

```
## Not run:  
library(webglobe)  
wg<-webglobe(immediate=TRUE)  
wg+wgcamhome()  
  
## End(Not run)
```

---

|         |                        |
|---------|------------------------|
| wgclear | <i>Clear the scene</i> |
|---------|------------------------|

---

**Description**

Clears everything from the map

**Usage**

```
wgclear()
```

**Value**

A webglobe command

**Examples**

```
## Not run:  
library(webglobe)  
wg<-webglobe(immediate=TRUE)  
wg+wgclear()  
  
## End(Not run)
```

---

|             |                           |
|-------------|---------------------------|
| wgimmediate | <i>Immediate mode: On</i> |
|-------------|---------------------------|

---

**Description**

Turns on immediate mode

**Usage**

```
wgimmediate()
```

**Value**

A webglobe command

**Examples**

```
## Not run:  
library(webglobe)  
wg<-webglobe(immediate=FALSE)  
wg + wgimmediate() #wg is now immediate  
  
## End(Not run)
```

wgimmediate\_set      *Immediate mode: Set*

---

**Description**

Set immediate mode by value

**Usage**

```
wgimmediate_set(mode)
```

**Arguments**

mode                    TRUE or FALSE: TRUE implies immediate mode on, FALSE implies off

**Value**

A webglobe command

**Examples**

```
## Not run:  
library(webglobe)  
wg<-webglobe(immediate=TRUE)  
wg + wgimmediate_set(FALSE) #wg is now unimmediate  
  
## End(Not run)
```

---

wgpoints                *Plot points*

---

**Description**

Plots latitude-longitude points

**Usage**

```
wgpoints(lat, lon, label = NA, alt = 0, colour = "yellow", size = 10)
```

**Arguments**

lat                    One or more latitude values  
lon                    One or more longitude values  
label                  Label to put next to point  
alt                    Altitude of the points, can be single value or vector  
colour                 Colour name of the points, can be single value or vector  
size                   Size of the points, can be single value or vector

**Value**

A webglobe command

**Examples**

```
## Not run:
library(webglobe)
wg <- webglobe(immediate=FALSE)
wg <- wg + wgpoints(c(45,20),c(-93,127),alt=3,colour=c("blue","red"))
wg <- wg + wgpoints(51.5074,-0.1278,label="London",alt=0,colour="blue")
wg

## End(Not run)
```

---

wgpolygondf

*Plot long-frame polygons*


---

**Description**

Plot polygons defined by long-style data frame

**Usage**

```
wgpolygondf(df, fill = NA, alpha = 1, extrude_height = 0,
  stroke = "yellow", stroke_width = 10)
```

**Arguments**

|                |  |
|----------------|--|
| df             | Data frame to plot   |
| fill           | Fill colour name   |
| alpha          | Alpha (transparency value)                                     |
| extrude_height | Height of the polygon above the surrounding landscape, in TODO |
| stroke         | Outline colour (TODO)  |
| stroke_width   | Outline width (TODO)   |

**Value**

A webglobe command

**Examples**

```
## Not run:
library(webglobe)
wg<-webglobe(immediate=TRUE)
wg+wgpolygondf(ggplot2::map_data("usa"),fill="blue",extrude_height=1000)

## End(Not run)
```

wgport *Get webglobe's port*

---

**Description**

Determine which port a webglobe is running on

**Usage**

```
wgport(wg)
```

**Arguments**

wg                    Webglobe whose port should be determined

**Value**

A number representing the webglobe's port

**Examples**

```
## Not run:  
library(webglobe)  
wg<-webglobe(immediate=TRUE)  
wgport(webglobe)  
  
## End(Not run)
```

---

wgttitle *Title of webglobe browser window*

---

**Description**

Changes the tab/window title of the webglobe's browser view

**Usage**

```
wgttitle(title)
```

**Arguments**

title                The title to use

**Value**

A webglobe command

**Examples**

```
## Not run:  
library(webglobe)  
wg<-webglobe(immediate=TRUE)  
wg+wgtitle("I am the new title!")  
  
## End(Not run)
```

---

|               |                            |
|---------------|----------------------------|
| wgunimmediate | <i>Immediate mode: Off</i> |
|---------------|----------------------------|

---

**Description**

Turns off immediate mode

**Usage**

```
wgunimmediate()
```

**Value**

A webglobe command

**Examples**

```
## Not run:  
library(webglobe)  
wg<-webglobe(immediate=TRUE)  
wg + wgunimmediate() #wg is now unimmediate  
  
## End(Not run)
```

# Index

`+.webglobe`, 2

`is.webglobe`, 3

`print.webglobe`, 3

`webglobe`, 4

`wgbar`, 5

`wgcamcenter`, 5

`wgcamhome`, 6

`wgclear`, 7

`wgimmediate`, 7

`wgimmediate_set`, 8

`wgpoints`, 8

`wgpolygondf`, 9

`wgport`, 10

`wgtitle`, 10

`wgunimmediate`, 11