

# Package ‘infra’

February 20, 2015

**Title** An Infrastructure Proxy Function

**Version** 0.1.2

**Author** Steve Pickering <pickering@penguin.kobe-u.ac.jp>

**Maintainer** Steve Pickering <pickering@penguin.kobe-u.ac.jp>

**Description** Takes a data frame containing latitude and longitude coordinates and downloads images from map servers to determine their file size as a proxy of infrastructure

**Depends** R (>= 3.0.2)

**License** GPL-2

**LazyData** true

**NeedsCompilation** no

**Repository** CRAN

**Date/Publication** 2015-01-31 11:39:26

## R topics documented:

getInfra . . . . .	1
<b>Index</b>	<b>3</b>

---

getInfra	<i>An Infrastructure Proxy Function</i>
----------	---

---

### Description

This function takes latitude and longitude coordinates from a data frame and downloads images from map servers for those coordinates. It will return the file sizes of those map images which can be used as a useful proxy for the level of infrastructure development. Map servers currently used are Google, Bing and Openstreetmap.

### Usage

```
getInfra(dataFrame = NULL, server = "google", zoom = 14)
```

**Arguments**

dataFrame	The name of your data frame
server	The map server you want to use. Options are google, bing, and open. Defaults to google
zoom	The zoom level you want to use. Defaults to 14. Valid values for google are 0 to 22; values for bing are 1 to 19; values for open are 0 to 19

**Examples**

```
# In this example, we'll build a very low resolution (two degrees per grid cell) grid of Japan,  
# retrieve the map values from Bing at zoom level 14, and render a simple map.
```

```
# First, define the latitude and longitude bounds of Japan and create a data frame.  
# (This uses a very restricted version of the territory of Japan for the sake of speed.)
```

```
lats <- rep(rev(seq(32,44, by=2)), each=7)  
lngs <- rep(seq(130,142, by=2), 7)
```

```
coords <- data.frame(lats,lngs)
```

```
infraBing <- getInfra(dataFrame = coords, zoom=14, server="bing")
```

```
# Now that we have the values in the newly-created infraBing  
# variable we can plot them. First, we need to define our colour  
# gamut, running from 0 to the maximum map value retrieved.
```

```
grey <- gray(0:max(infraBing) / max(infraBing))
```

```
# Now we can plot.
```

```
png(filename="japanInfraBing.png", width=480, height=480, units="px")  
plot(lngs, lats, col=grey[infraBing], pch=15, cex=14)  
dev.off()
```

# Index

\*Topic **bing**

getInfra, 1

\*Topic **google**

getInfra, 1

\*Topic **infrastructure**

getInfra, 1

\*Topic **maps**

getInfra, 1

\*Topic **openstreetmap**

getInfra, 1

getInfra, 1