

# Package ‘GCSFilesystem’

April 12, 2022

**Type** Package

**Title** Mounting a Google Cloud bucket to a local directory

**Version** 1.4.0

**Date** 2020-09-16

**Description** Mounting a Google Cloud bucket to a local directory.

The files in the bucket can be viewed and read as if they are locally stored.

For using the package, you need to install GCSDokan on Windows or gcsfuse on Linux and MacOS.

**License** GPL (>= 2)

**Depends** R (>= 4.0.0)

**Imports** stats

**Suggests** testthat, knitr, rmarkdown, BiocStyle, GCSCConnection

**SystemRequirements** GCSDokan for Windows, gcsfuse for Linux and macOS

**biocViews** Infrastructure

**RoxygenNote** 7.1.1

**Encoding** UTF-8

**VignetteBuilder** knitr

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

**git\_branch** RELEASE\_3\_14

**git\_last\_commit** 0b3a288

**git\_last\_commit\_date** 2021-10-26

**Date/Publication** 2022-04-12

**Author** Jiefei Wang [aut, cre]

**Maintainer** Jiefei Wang <szwjf08@gmail.com>

## R topics documented:

<code>gcs_list_mountpoints</code> . . . . .	2
<code>gcs_mount</code> . . . . .	2
<code>gcs_unmount</code> . . . . .	3

**Index****5**

---

`gcs_list_mountpoints` *List all GCS mountpoints*

---

**Description**

List all GCS mountpoints. The function uses ‘GCSDokan‘ on Windows or the command ‘df‘ on Linux to show all mountpoints. Due to the system differences, the function is only able to show the bucket name on Linux and can show the full remote path on Windows.

**Usage**

```
gcs_list_mountpoints()
```

**Value**

a data.frame object with the first column named ‘remote‘ and second named ‘mountpoint‘

**Examples**

```
gcs_list_mountpoints()
```

---

`gcs_mount` *Mount a Google Cloud Storage(GCS) file system*

---

**Description**

The function uses the command-line program ‘GCSDokan‘ on Windows or ‘gcsfuse‘ on Linux system to mount a google cloud bucket path to your local file system.

**Usage**

```
gcs_mount(  
  remote,  
  mountpoint,  
  mode = c("r", "rw"),  
  cache_type = c("disk", "memory", "none"),  
  cache_arg = NULL,  
  billing = NULL,  
  refresh = 60,  
  implicit_dirs = TRUE,  
  key_file = NULL,  
  additional_args = NULL  
)
```

**Arguments**

remote	the remote path to a Google Cloud Storage bucket. It can be either the bucket itself or a directory in the bucket.
mountpoint	The mount point where the GCS file system will be mounted to.
mode	the permission of the mounted directory. The write permission is only available on Linux.
cache_type	The location where the file cache will be stored. The cache type 'none' and 'memory' are only available on Windows
cache_arg	The argument of a cache type. If 'cache_type = "memory"', the argument is the limit of the memory usage in MB. If 'cache_type = "disk"', the argument is the path to a cache directory.
billing	The billing project ID.
refresh	The refresh rate of the cloud files.
implicit_dirs	Implicitly define directories based on content. This argument is only available on Linux.
key_file	The service account credentials file
additional_args	The additional argument that will be passed to the command-line program.

**Value**

no return value

**Examples**

```
bucket <- "genomics-public-data"
mountpoint <- paste0(tempdir(), "/GCSFilesystemTest")
## You must have a credentials on Linux and macOS
## To run this code
if(Sys.getenv("GOOGLE_APPLICATION_CREDENTIALS")!=""){
  gcs_mount(bucket, mountpoint)
  gcs_unmount(mountpoint)
}
```

---

gcs\_unmount

*Unmount a mounted GCS file system*


---

**Description**

Unmount a mounted GCS file system

**Usage**

```
gcs_unmount(mountpoint)
```

**Arguments**

mountpoint      The path to the mounted GCS file system.

**Value**

No return value

**Examples**

```
## Unmount a directory
## No operation will be done
## if the directory does not exist or
## not a mounted directory
gcs_unmount("path_to_your_mounted_directory")
```

# Index

`gcs_list_mountpoints`, 2  
`gcs_mount`, 2  
`gcs_unmount`, 3