Package 'GoogleGenomics'

April 14, 2017

Version 1.6.0

Date 2015-07-13

Title R Client for Google Genomics API

Depends R (>= 3.1.0), GenomicAlignments (>= 1.0.1), VariantAnnotation

Imports Biostrings, GenomeInfoDb, GenomicRanges, IRanges, httr, rjson, Rsamtools, S4Vectors (>= 0.9.25)

Suggests BiocStyle, httpuv, knitr, rmarkdown, testthat, ggbio, ggplot2, BSgenome.Hsapiens.UCSC.hg19, org.Hs.eg.db, TxDb.Hsapiens.UCSC.hg19.knownGene

Description Provides an R package to interact with the Google Genomics API.

VignetteBuilder knitr

License Apache License (== 2.0) | file LICENSE

URL https://cloud.google.com/genomics/

BugReports https://github.com/Bioconductor/GoogleGenomics/issues

biocViews DataImport, ThirdPartyClient

NeedsCompilation no

Author Cassie Doll [aut],
Nicole Deflaux [aut],
Siddhartha Bagaria [aut, cre]

Maintainer Siddhartha Bagaria <sidb@google.com>

R topics documented:

authenticate 2
getReads 3
getReadsPage 4
getSearchPage 5
getVariants 5
getVariantsPage 6
GoogleGenomics 7
readsToGAlignments 8
variantsToGRanges 8
variantsToVRanges 9

Index 11

`authenticate` *Configure how to authenticate for Google Genomics API.*

Description

Follow the sign up instructions at https://cloud.google.com/genomics/install-genomics-tools#authenticate to download the client secrets file, or note the clientId and clientSecret pair.

Usage

```
authenticate(file, clientId, clientSecret, invokeBrowser, apiKey)
```

Arguments

`file`	Client secrets file obtained from Google Developer Console. This file could be for a native application or a service account. If this file is not present, clientId and clientSecret must be provided for native application credentials. Service account support needs version 0.1-2 or greater of PKI.
`clientId`	Client ID from Google Developer Console, overridden if file is provided.
`clientSecret`	Client Secret from Google Developer Console, overridden if file is provided.
`invokeBrowser`	If TRUE or not provided, the default browser is invoked with the auth URL iff the `httpuv` package is installed (suggested). If FALSE, a URL is output which needs to be copy pasted in a browser, and the resulting token needs to be pasted back into the R session. With both the options, you will still need to login to your Google account if not logged in already.
`apiKey`	Public API key that can be used to call the Genomics API for public datasets. This method of authentication does not need you to login to your Google account. Providing this key overrides all other arguments.

Value

NULL (silently) if successful.

Examples

```
apiKey <- Sys.getenv("GOOGLE_API_KEY")
if (!is.na(apiKey) && nchar(apiKey)>0) {
  authenticate(apiKey=apiKey)
}
## Not run:
authenticate(file="clientSecrets.json")
authenticate(file="clientSecrets.json", invokeBrowser=FALSE)
authenticate(clientId="abc", clientSecret="xyz", invokeBrowser=FALSE)

## End(Not run)
```

getReads *Get reads from Google Genomics.*

Description

This function will return all of the reads that comprise the requested genomic range, iterating over paginated results as necessary.

Usage

```
getReads(readGroupSetId = "CMvnhpKTFhDnk4_9zcKO3_YB", chromosome = "22",
  start = 16051400, end = 16051500, fields = NULL, converter = c)
```

Arguments

readGroupSetId	The read group set ID.
chromosome	The chromosome.
start	Start position on the chromosome in 0-based coordinates.
end	End position on the chromosome in 0-based coordinates.
fields	A subset of fields to retrieve. The default (NULL) will return all fields.
converter	A function that takes a list of read R objects and returns them converted to the desired type.

Details

By default, this function gets reads for a small genomic region for one sample in 1,000 Genomes.

Optionally pass a converter as appropriate for your use case. By passing it to this method, only the converted objects will be accumulated in memory. The converter function should return an empty container of the desired type if called without any arguments.

Value

By default, the return value is a list of R objects corresponding to the JSON objects returned by the Google Genomics Reads API. If a converter is passed, object(s) of the type returned by the converter will be returned by this function.

See Also

getVariants

Examples

```
# Authenticated on package load from the env variable GOOGLE_API_KEY.
reads <- getReads()
summary(reads)
summary(reads[[1]])
```

`getReadsPage` *Get one page of reads from Google Genomics.*

Description

In general, use the getReads method instead. It calls this method, returning reads from all of the pages that comprise the requested genomic range.

Usage

```
getReadsPage(readGroupSetId = "CMvnhpKTFhDnk4_9zcKO3_YB", chromosome = "22",
  start = 16051400, end = 16051500, fields = NULL, pageToken = NULL)
```

Arguments

`readGroupSetId`	The read group set ID.
`chromosome`	The chromosome.
`start`	Start position on the chromosome in 0-based coordinates.
`end`	End position on the chromosome in 0-based coordinates.
`fields`	A subset of fields to retrieve. The default (NULL) will return all fields.
`pageToken`	The page token. This can be NULL (default) for the first page.

Details

By default, this function gets reads for a small genomic region for one sample in 1,000 Genomes.

Note that the Global Alliance for Genomics and Health API uses a 0-based coordinate system. For more detail, please see GA4GH discussions such as the following:

- https://github.com/ga4gh/schemas/issues/168
- https://github.com/ga4gh/schemas/issues/121

Value

A two-element list is returned by the function.

reads: A list of R objects corresponding to the JSON objects returned by the Google Genomics Reads API.

nextPageToken: The token to be used to retrieve the next page of results, if applicable.

See Also

Other page fetch functions: `getSearchPage`; `getVariantsPage`

Examples

```
# Authenticated on package load from the env variable GOOGLE_API_KEY.
readsPage <- getReadsPage()
summary(readsPage)
summary(readsPage$reads[[1]])
```

getSearchPage — *Get one page of search results for a particular entity type from Google Genomics.*

Description

In general, use higher level methods such as getReads and getVariants instead.

Usage

```
getSearchPage(entityType, body, fields, pageToken)
```

Arguments

entityType	Entities with a search API such as reads, variants, variantSets, etc...
body	The body of the message to POST to the search endpoint.
fields	The fields to be returned in the search response.
pageToken	The page token. This can be NULL for the first page.

Value

The raw response converted from JSON to an R object.

See Also

Other page fetch functions: getReadsPage; getVariantsPage

Examples

```
# Authenticated on package load from the env variable GOOGLE_API_KEY.
body <- list(readGroupSetIds=list("CMvnhpKTFhDnk4_9zcKO3_YB"),
             referenceName="22",
             start=16051400, end=16051500, pageToken=NULL)
reads <- getSearchPage("reads", body, NULL, NULL)
summary(reads)
```

getVariants — *Get variants from Google Genomics.*

Description

This function will return all of the variants that comprise the requested genomic range, iterating over paginated results as necessary.

Usage

```
getVariants(datasetId = "10473108253681171589", chromosome = "22",
  start = 16051400, end = 16051500, fields = NULL, converter = c)
```

Arguments

datasetId	The dataset ID.
chromosome	The chromosome.
start	Start position on the chromosome in 0-based coordinates.
end	End position on the chromosome in 0-based coordinates.
fields	A subset of fields to retrieve. The default (NULL) will return all fields.
converter	A function that takes a list of variant R objects and returns them converted to the desired type.

Details

By default, this function gets variants from a small section of 1000 Genomes phase 1 variants.

Optionally pass a converter as appropriate for your use case. By passing it to this method, only the converted objects will be accumulated in memory. The converter function should return an empty container of the desired type if called without any arguments.

Value

By default, the return value is a list of R objects corresponding to the JSON objects returned by the Google Genomics Variants API. If a converter is passed, object(s) of the type returned by the converter will be returned by this function.

See Also

getReads for equivalent function for reads, and variantsToVRanges for a converter function.

Examples

```
# Authenticated on package load from the env variable GOOGLE_API_KEY.
variants <- getVariants()
summary(variants)
summary(variants[[1]])
```

---

getVariantsPage *Get one page of variants from Google Genomics.*

Description

In general, use the getVariants method instead. It calls this method, returning variants from all of the pages that comprise the requested genomic range.

Usage

```
getVariantsPage(datasetId = "10473108253681171589", chromosome = "22",
  start = 16051400, end = 16051500, fields = NULL, pageToken = NULL)
```

Arguments

datasetId	The dataset ID.
chromosome	The chromosome.
start	Start position on the chromosome in 0-based coordinates.
end	End position on the chromosome in 0-based coordinates.
fields	A subset of fields to retrieve. The default (NULL) will return all fields.
pageToken	The page token. This can be NULL (default) for the first page.

Details

By default, this function gets variants from a small section of 1000 Genomes phase 1 variants.

Value

A two-element list is returned by the function.

variants: A list of R objects corresponding to the JSON objects returned by the Google Genomics Variants API.

nextPageToken: The token to be used to retrieve the next page of results, if applicable.

See Also

Other page fetch functions: getReadsPage; getSearchPage

Examples

```
# Authenticated on package load from the env variable GOOGLE_API_KEY.
variantsPage <- getVariantsPage()
summary(variantsPage)
summary(variantsPage$variants[[1]])
```

GoogleGenomics *A basic R package for Google Genomics API.*

Description

This project is in active development - the current code is very minimal and a lot of work is left.

Details

For more details, read the package README.

readsToGAlignments *Convert reads to GAlignments.*

Description

Note that the Global Alliance for Genomics and Health API uses a 0-based coordinate system. For more detail, please see GA4GH discussions such as the following:

- https://github.com/ga4gh/schemas/issues/168
- https://github.com/ga4gh/schemas/issues/121

Usage

```
readsToGAlignments(reads, oneBasedCoord = TRUE, slStyle = "UCSC")
```

Arguments

reads	A list of R objects corresponding to the JSON objects returned by the Google Genomics Reads API.
oneBasedCoord	Convert genomic positions to 1-based coordinates.
slStyle	The style for seqnames (chrN or N or...). Default is UCSC.

Value

GAlignments

Examples

```
# Authenticated on package load from the env variable GOOGLE_API_KEY.
alignments1 <- getReads(converter=readsToGAlignments)
summary(alignments1)
alignments2 <- readsToGAlignments(getReads())
print(identical(alignments1, alignments2))
```

variantsToGRanges *Convert variants to GRanges.*

Description

Note that the Global Alliance for Genomics and Health API uses a 0-based coordinate system. For more detail, please see GA4GH discussions such as the following:

- https://github.com/ga4gh/schemas/issues/168
- https://github.com/ga4gh/schemas/issues/121

Usage

```
variantsToGRanges(variants, oneBasedCoord = TRUE, slStyle = "UCSC")
```

Arguments

variants	A list of R objects corresponding to the JSON objects returned by the Google Genomics Variants API.
oneBasedCoord	Convert genomic positions to 1-based coordinates.
slStyle	The style for seqnames (chrN or N or...). Default is UCSC.

Value

GRanges

See Also

Other variants converter functions: variantsToVRanges

Examples

```
# Authenticated on package load from the env variable GOOGLE_API_KEY.
variants1 <- getVariants(converter=variantsToGRanges)
summary(variants1)
variants2 <- variantsToGRanges(getVariants())
print(identical(variants1, variants2))
```

---

variantsToVRanges *Convert variants to VRanges.*

Description

Note that the Global Alliance for Genomics and Health API uses a 0-based coordinate system. For more detail, please see GA4GH discussions such as the following:

- https://github.com/ga4gh/schemas/issues/168
- https://github.com/ga4gh/schemas/issues/121

Usage

```
variantsToVRanges(variants, oneBasedCoord = TRUE, slStyle = "UCSC")
```

Arguments

variants	A list of R objects corresponding to the JSON objects returned by the Google Genomics Variants API.
oneBasedCoord	Convert genomic positions to 1-based coordinates.
slStyle	The style for seqnames (chrN or N or...). Default is UCSC.

Value

VRanges

See Also

Other variants converter functions: variantsToGRanges

Examples

```
# Authenticated on package load from the env variable GOOGLE_API_KEY.
variants1 <- getVariants(converter=variantsToVRanges)
summary(variants1)
variants2 <- variantsToVRanges(getVariants())
print(identical(variants1, variants2))
```

Index

authenticate, 2

GAlignments, *8*
getReads, 3, *6*
getReadsPage, 4, *5*, *7*
getSearchPage, *4*, 5, *7*
getVariants, *3*, 5
getVariantsPage, *4, 5*, 6
GoogleGenomics, 7
GoogleGenomics-package
        *(*GoogleGenomics*)*, 7
GRanges, *9*

httpuv, *2*

readsToGAlignments, 8

variantsToGRanges, 8, *9*
variantsToVRanges, *6*, *9*, 9
VRanges, *9*