

Package ‘rhdf5client’

May 16, 2024

Title Access HDF5 content from HDF Scalable Data Service

Description This package provides functionality for reading data from HDF Scalable Data Service from within R. The HSDSArray function bridges from HSDS to the user via the DelayedArray interface. Bioconductor manages an open HSDS instance graciously provided by John Readey of the HDF Group.

Version 1.26.0

Suggests knitr, testthat, BiocStyle, DT, rmarkdown

Imports httr, rjson, utils, data.table

Depends R (>= 3.6), methods, DelayedArray

License Artistic-2.0

LazyLoad yes

Collate URL_hds.R indx.R 'rhdf5client2.R' 'Source.R' 'File.R'
'Dataset.R' 'RHDF5Array.R'

VignetteBuilder knitr

Encoding UTF-8

biocViews DataImport, Software, Infrastructure

RoxygenNote 7.2.3

git_url <https://git.bioconductor.org/packages/rhdf5client>

git_branch RELEASE_3_19

git_last_commit aac49be

git_last_commit_date 2024-04-30

Repository Bioconductor 3.19

Date/Publication 2024-05-15

Author Samuela Pollack [aut],
Shweta Gopaulakrishnan [aut],
BJ Stubbs [aut],
Alexey Sergushichev [aut],
Vincent Carey [cre, aut]

Maintainer Vincent Carey <stvjc@channing.harvard.edu>

check_hsd	<i>a test request</i>
-----------	-----------------------

Description

a test request

Usage

```
check_hsd()
```

Value

logical, TRUE if hsd behaving as expected

Examples

```
check_hsd()
```

dim	<i>Obtain dimensions of an object of type HSDSArraySeed</i>
-----	---

Description

(required by DelayedArray seed contract) HDF server content is assumed transposed relative to R matrix layout. This anticipates H5 datasets on the server with rows for experimental samples and columns for *-omic features. The Bioconductor SummarizedExperiment requires *-omic features in rows and samples in columns.

Usage

```
## S4 method for signature 'HSDSArraySeed'  
dim(x)
```

Arguments

x An object of type HSDSArraySeed

Value

A numeric vector of the dimensions

dimnames	<i>Obtain names of dimensions for an object of type HSDSArraySeed</i>
----------	---

Description

(required by DelayedArray seed contract, returns NULL list)

Usage

```
## S4 method for signature 'HSDSArraySeed'
dimnames(x)
```

Arguments

x	An object of type HSDSArraySeed
---	---------------------------------

Value

A NULL list of length equal to the array dimensionality

extractCompoundJSON	<i>compound operation</i>
---------------------	---------------------------

Description

compound operation

Usage

```
extractCompoundJSON(type, value)
```

Arguments

type	type
value	value

extract_array	<i>Access dataset backed by an HSDSArraySeed</i>
---------------	--

Description

Access dataset backed by an HSDSArraySeed

Usage

```
## S4 method for signature 'HSDSArraySeed'
extract_array(x, index)
```

Arguments

x	An object of type HSDSArraySeed
index	A list of numeric vectors to be accessed, one vector for each dimension of the array object. A NULL vector indicates the entire range of indices in that dimension. A zero-length vector indicates no indices in the relevant dimension. (Accordingly, any zero-length vector of indices will result in an empty array being returned.)

Value

An array containing the data elements corresponding to the indices requested

getData	<i>extract elements of a one or two-dimensional HSDSDataset</i>
---------	---

Description

Fetch data from a remote dataset

Usage

```
getData(dataset, indices, transfermode)

## S4 method for signature 'HSDSDataset,character,character'
getData(dataset, indices, transfermode)

## S4 method for signature 'HSDSDataset,character,missing'
getData(dataset, indices)

## S4 method for signature 'HSDSDataset,list,character'
getData(dataset, indices, transfermode)

## S4 method for signature 'HSDSDataset,list,missing'
getData(dataset, indices)
```

Arguments

dataset	An object of type HSDSDataset, the dataset to access.
indices	The indices of the data to fetch
transfermode	Either 'JSON' or 'binary' (default)

Details

The servers require data to be fetched in slices, i.e., in sets of for which the indices of each dimension are of the form start:stop:step. More complex sets of indices will be split into slices and fetched in multiple requests. This is opaque to the user, but may enter into considerations of data access patterns, e.g., for performance-tuning.

Value

an Array containing the data fetched from the server

Examples

```
if (check_hsd()) {
  s <- HSDSource(URL_hsd())
  f <- HSDFile(s, '/shared/bioconductor/patelGBMSC.h5')
  d <- HSDDataset(f, '/assay001')
  x <- getData(d, c('1:4', '1:27998'), transfermode='JSON')
  xb <- getData(d, c('1:4', '1:27998'), transfermode='binary')
  # x <- getData(d, c(1:4, 1:27998), transfermode='JSON') # method missing?
  x
  xb
}
```

HSDSArray

A DelayedArray backend for accessing a remote HDF5 server.

Description

A DelayedArray backend for accessing a remote HDF5 server.

Construct an object of type HSDSArray directly from the data members of its seed

Usage

```
HSDSArray(endpoint, svrtype, domain, dsetname)
```

Arguments

endpoint	URL of remote server
svrtype	type of server, must be either 'hds' or 'h5serv'
domain	HDF5 domain of H5 file on server
dsetname	complete internal path to dataset in H5 file

Value

An initialized object of type HSDSArray

See Also

Other HSDSArray: [HSDSMatrix](#), [as\(\)](#)

Examples

```
if (check_hsd()) {
  HSDSArray(URL_hsd(),
            "hsds", "/shared/bioconductor/darmgcls.h5", "/assay001")
}
```

HSDSArraySeed

HSDSArraySeed for HSDSArray backend to DelayedArray

Description

HSDSArraySeed for HSDSArray backend to DelayedArray

Construct an object of type HSDSArraySeed

Usage

```
HSDSArraySeed(endpoint, svrtype, domain, dsetname)
```

Arguments

endpoint	URL of remote server
svrtype	type of server, must be either 'hsds' or 'h5serv'
domain	HDF5 domain of H5 file on server
dsetname	complete internal path to dataset in H5 file

Value

An initialized object of type HSDSArraySeed

Slots

endpoint	URL of remote server
svrtype	type of server, must be either 'hsds' or 'h5serv'
domain	HDF5 domain of H5 file on server
dsetname	complete internal path to dataset in H5 file
dataset	object of type HSDSDataset for access to the H5 dataset

HSDSDataset	<i>Construct an object of type HSDSDataset A HSDSDataset is a representation of a dataset in a HDF5 file.</i>
-------------	---

Description

Construct an object of type HSDSDataset A HSDSDataset is a representation of a dataset in a HDF5 file.

Usage

```
HSDSDataset(file, path)
```

Arguments

file	An object of type HSDSFile which hosts the dataset
path	The complete intrafile path to the dataset

Value

An initialized object of type HSDSDataset

Examples

```
if (check_hsd()) {
  src <- HSDSource(URL_hsd())
  f <- HSDSFile(src, '/shared/bioconductor/patelGBMSC.h5')
  d <- HSDSDataset(f, '/assay001')
}
```

HSDSDataset-class	<i>An S4 class to represent a dataset in a HDF5 file.</i>
-------------------	---

Description

An S4 class to represent a dataset in a HDF5 file.

Slots

file	An object of type HSDSFile; the file in which the dataset is resident.
path	The dataset's path in the internal HDF5 hierarchy.
uuid	The unique unit ID by which the dataset is accessed in the server database system.
shape	The dimensions of the dataset
type	The dataset's HDF5 datatype

HSDSFile	<i>Construct an object of type HSDSFile</i>
----------	---

Description

A HSDSFile is a representation of an HDF5 file the contents of which are accessible exposed by a HDF5 server.

Usage

```
HSDSFile(src, domain)
```

Arguments

src	an object of type HSDSSource, the server which exposes the file
domain	the domain string; the file's location on the server's file system.

Details

This function is deprecated and will be defunct in the next release.

Value

an initialized object of type HSDSFile

Examples

```
if (check_hsd()) {
  src <- HSDSSource(URL_hsd())
  f10x <- HSDSFile(src, '/shared/bioconductor/patelGBMSC.h5')
}
```

HSDSFile-class	<i>An S4 class to represent an HDF5 file accessible from a server.</i>
----------------	--

Description

An S4 class to represent an HDF5 file accessible from a server.

Slots

HSDSSource	an object of type HSDSSource
domain	the file's domain on the server; more or less, an alias for its location in the external server file system
dsetdf	a data.frame that caches often-used information about the file

HSDSMatrix	<i>DelayedMatrix subclass for a two-dimensional HSDSArray</i>
------------	---

Description

DelayedMatrix subclass for a two-dimensional HSDSArray

See Also

Other HSDSArray: [HSDSArray](#), [as\(\)](#)

HSDSSource	<i>Construct an object of type HSDSSource.</i>
------------	--

Description

A HSDSSource is a representation of a URL which provides access to a HDF5 server (either h5serv or hds.)

Usage

```
HSDSSource(endpoint, type = "hds")
```

Arguments

endpoint	URL for server
type	Type of server software at the source; must be

Details

This function is deprecated and will be defunct in the next release.

Value

An object of type HSDSSource

Examples

```
if (check_hds()) {
  src.hds <- HSDSSource(URL_hds())
}
```

HSDSSource-class	<i>An S4 class to represent a HDF5 server listening on a port.</i>
------------------	--

Description

This class is deprecated and will be defunct in the next release.

Slots

endpoint URL for server

type Type of server software at the source; must be either 'h5serv' or (default) 'hds'

isplit	<i>This function is deprecated and will be defunct in the next release.</i>
--------	---

Description

isplit converts a numeric vector into a list of sequences for compact reexpression

Usage

```
isplit(x)
```

```
sproc(spl)
```

Arguments

x a numeric vector (should be integers)

spl output of isplit

Value

list of vectors of integers which can be expressed as initial/final/stride triplets

list of colon-delimited strings each with initial/final/stride triplet

Examples

```
inds = c(1:10, seq(25,50,2), seq(200,150,-2))
sproc(isplit(inds))
```

listDatasets	<i>Search inner file hierarchy for datasets</i>
--------------	---

Description

The datasets in an HDF5 file are organized internally by groups. This routine traverses the internal group hierarchy, locates all datasets and prints a list of them. Note that if the file's group hierarchy is complex, this could be time-consuming.

Usage

```
listDatasets(file)
```

Arguments

file an object of type HSDSFile to be searched

Details

This function is deprecated and will be defunct in the next release.

Value

a list of inner-paths

Examples

```
if (check_hds()) {  
  src <- HSDSource(URL_hds())  
  f <- HSDSFile(src, '/shared/bioconductor/patelGBMSC.h5')  
  listDatasets(f)  
}
```

listDomains	<i>List files and subdirectories of a domain</i>
-------------	--

Description

The user needs to give the domain to start in. The search will be non-recursive. I.e., output for domain '/home/jreadey/' will not return the files in '/home/jreadey/HDFLabTutorial/'

Usage

```
listDomains(object, rootdir)

## S4 method for signature 'HSDSSource,character'
listDomains(object, rootdir)

## S4 method for signature 'HSDSSource,missing'
listDomains(object)
```

Arguments

object An object of type HSDSSource

rootdir A slash-separated directory in the HSDSSource file system.

Details

This function is deprecated and will be defunct in the next release.

Value

a vector of domains in the rootdir

Examples

```
src.hsds <- HSDSSource(URL_hsds())
listDomains(src.hsds, '/shared')
```

rhdf5client	<i>rhdf5client: A package for accessing HDFGroup HDF5 servers from R.</i>
-------------	---

Description

The rhdf5client package provides read-only access to HDF5 files maintained on a server. The HDFGroup provides two servers, an obsolescent one called ‘h5serv’ and the newer prototype called ‘hsds’.

rhdf5client-deprecated

Deprecated functions in package ‘rhdf5client’

Description

These functions are provided for compatibility with older versions of ‘rhdf5client’ only, and will be defunct at the next release.

Details

The following functions are deprecated and will be made defunct in the next release:

- URL_h5serv
- URL_hsd
- dsmeta
- getReq
- groups
- setPath
- links
- transfermode
- dataset
- internalDim
- hsdInfo
- domains
- getDatasetUUIDs
- getDatasetAttrs
- getDims
- getHRDF
- H5S_dataset2
- getDatasetSlice
- fetchDatasets
- isplit
- sproc
- listDomains
- listDatasets
- getData

The following classes are deprecated and will be made defunct in the next release:

- H5S_source

- H5S_dataset
- H5S_Array
- H5S_Matrix
- HSDSSource
- HSDSFile
- HSDSDataset

 URL_hsd

manage hsd URL

Description

manage hsd URL

Usage

URL_hsd()

Value

URL of hsd server

Examples

URL_hsd()

 [,HSDSDataset,numeric,ANY,ANY-method

bracket method for 1d request from HSDSDataset

Description

bracket method for 1d request from HSDSDataset

Usage

```
## S4 method for signature 'HSDSDataset,numeric,ANY,ANY'
x[i, j, ..., drop = TRUE]
```

Arguments

x	object of type HSDSDataset
i	vector of indices (first dimension)
j	not used
...	not used
drop	logical(1) if TRUE return has no array character

Value

an array with the elements requested from the HSDSDataset

[,HSDSDataset,numeric,numeric,ANY-method
bracket method for 2d request from HSDSDataset

Description

bracket method for 2d request from HSDSDataset

Usage

```
## S4 method for signature 'HSDSDataset,numeric,numeric,ANY'
x[i, j, ..., drop = TRUE]
```

Arguments

x	object of type HSDSDataset
i	vector of indices (first dimension)
j	vector of indices (second dimension)
...	not used
drop	logical(1) if TRUE return has no array character

Value

an array with the elements requested from the HSDSDataset

Index

- * **HSDSArray**
 - as, [2](#)
 - HSDSArray, [6](#)
 - HSDSMatrix, [10](#)
- [,HSDSDataset,numeric,ANY,ANY-method,[15](#)
- [,HSDSDataset,numeric,numeric,ANY-method,[16](#)

- as, [2](#), [7](#), [10](#)

- check_hsd, [3](#)

- dim, [3](#)
- dim,HSDSArraySeed-method (dim), [3](#)
- dimnames, [4](#)
- dimnames,HSDSArraySeed-method (dimnames), [4](#)

- extract_array, [5](#)
- extract_array,HSDSArraySeed-method (extract_array), [5](#)
- extractCompoundJSON, [4](#)

- getData, [5](#)
- getData,HSDSDataset,character,character-method (getData), [5](#)
- getData,HSDSDataset,character,missing-method (getData), [5](#)
- getData,HSDSDataset,list,missing-method (getData), [5](#)
- getData,HSDSDataset,list,character-method (getData), [5](#)
- getData,HSDSDataset,list,missing-method (getData), [5](#)

- HSDSArray, [2](#), [6](#), [10](#)
- HSDSArray-class (HSDSArray), [6](#)
- HSDSArraySeed, [7](#)
- HSDSArraySeed-class (HSDSArraySeed), [7](#)
- HSDSDataset, [8](#)

- HSDSDataset-class, [8](#)
- HSDSFile, [9](#)
- HSDSFile-class, [9](#)
- HSDSMatrix, [2](#), [7](#), [10](#)
- HSDSMatrix-class (HSDSMatrix), [10](#)
- HSDSSource, [10](#)
- HSDSSource-class, [11](#)

- isplit, [11](#)

- listDatasets, [12](#)
- listDomains, [12](#)
- listDomains,HSDSSource,character-method (listDomains), [12](#)
- listDomains,HSDSSource,missing-method (listDomains), [12](#)

- rhdf5client, [13](#)
- rhdf5client-deprecated, [14](#)

- sproc (isplit), [11](#)

- URL_hsd, [15](#)