Package 'rhdf5client'

April 16, 2019

Title Access HDF5 content from h5serv

Description Provides functionality for reading data from h5serv server from within R.

Version 1.4.1

Suggests knitr, testthat, BiocStyle, DT, reticulate

Imports S4Vectors, httr, R6, rjson, utils

Depends R (>= 3.5), methods, DelayedArray

License Artistic-2.0

LazyLoad yes

BiocViews infrastructure

RoxygenNote 6.1.1

Collate h5serv.R indx.R H5S_Array.R HSDS_Matrix.R 'rhdf5client2.R' 'Source.R' 'File.R' 'Dataset.R' 'RHDF5Array.R'

VignetteBuilder knitr

Encoding UTF-8

biocViews DataImport, Software

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

git_branch RELEASE_3_8

git_last_commit 939791a

git_last_commit_date 2018-12-14

Date/Publication 2019-04-15

Author Samuela Pollack [cre, aut], Shweta Gopaulakrishnan [aut], Vincent Carey [aut]

Maintainer Samuela Pollack <spollack@jimmy.harvard.edu>

R topics documented:

as *coercion for remote array to remote matrix*

Description

coercion for remote array to remote matrix

Coercion method from HSDSMatrix to its superclass HSDSArray

See Also

Other HSDSArray: `HSDSArray`, `HSDSMatrix`

dataset *Find a dataset on source from its name*

Description

Find a dataset on source from its name

Usage

```
dataset(h5s, tag)
```

Arguments

`h5s`	instance of H5S_source
`tag`	character string identifying a dataset

Value

object of type H5S_dataset

dim *HDF Server content is assumed transposed relative to R matrix layout*

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 'H5S_ArraySeed'
dim(x)

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

Arguments

x	instance of H5S_ArraySeed
x	An object of type HSDSArraySeed

Value

integer(2) vector of dimensions corresponding to R's layout, assuming 2-d data

A numeric vector of the dimensions

---

dimnames,H5S_ArraySeed-method *dimnames not stored with H5S_source as of Jan 2018*

Description

(required by DelayedArray seed contract, returns NULL list)

Usage

```
## S4 method for signature 'H5S_ArraySeed'
dimnames(x)

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

Arguments

x	instance of H5S_ArraySeed
x	An object of type HSDSArraySeed

Value

currently returns list(NULL, NULL) as we do not store dimnames in HDF5

A NULL list of length equal to the array dimensionality

---

domains *HSDS server domains accessor*

Description

HSDS server domains accessor

Usage

```
domains(object, ...)
```

Arguments

object	H5S_source instance
...	not used

Value

a data frame with domains name

Examples

```
hsdsCon = H5S_source(URL_hsds()) # hsds server connection
setPath(hsdsCon, "/home/stvjc/")-> hsds
domains(hsds)
```

---

dsmeta *list information about datasets available in an H5S_source*

Description

list information about datasets available in an H5S_source

Usage

```
dsmeta(src)
```

Arguments

src	H5S_source instance

Value

data frame with one row for each group and three columns. The second column has the list of datasets in the group.

Examples

```
## Not run:
bigec2 = H5S_source(URL_h5serv())
dsm <- dsmeta(bigec2)
dst <- unlist(dsm[1,2])    # all dataset candidates in group 1

## End(Not run)
```

extract_array *Access dataset backed by an HSDSArraySeed*

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

fetchDatasets *fetch datasets of a hdf5 file from the hsds server*

Description

fetch datasets of a hdf5 file from the hsds server

Usage

```
fetchDatasets(object)
```

Arguments

object	instance of H5S_source

Value

data.frame with information about the datasets in the file

Examples

```
hsdsCon = H5S_source(URL_hsds()) # hsds server
hsdsCon@folderPath="/home/stvjc/hdf5_mat.h5"
ds = fetchDatasets(hsdsCon)
ds
```

getData *Fetch data from a remote dataset*

Description

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.

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 (default) 'JSON' or 'binary'

Value

an Array containing the data fetched from the server

Examples

```
s <- HSDSSource('http://hsdshdflab.hdfgroup.org')
f <- HSDSFile(s, '/shared/bioconductor/tenx_full.h5')
d <- HSDSDataset(f, '/newassay001')
x <- getData(d, c('1:4', '1:27998'), transfermode='JSON')
# x <- getData(d, c(1:4, 1:27998), transfermode='JSON') # method missing?
x <- d[1:4,1:27998]
```

getDatasetAttrs *getDatasetAttrs from hsds server*

Description

getDatasetAttrs from hsds server

Usage

```
getDatasetAttrs(object, duid)
```

Arguments

object	instance of H5S_source(updated object with path to file set)
duid	character string with dataset uuid

Value

list of data obtained

Examples

```
hsdsCon = H5S_source(URL_hsds()) # hsds server
hsdsCon@folderPath="/home/stvjc/hdf5_mat.h5"
ds = fetchDatasets(hsdsCon)# Pick the ID of the dataset you are interested in
getDatasetAttrs(hsdsCon, "d-a9e4b71c-8ea2-11e8-9306-0242ac120022")
```

getDatasetSlice *getDatasetSlice from hsds server*

Description

getDatasetSlice from hsds server

Usage

```
getDatasetSlice(object, dsindex = 1, selectionString, ...)
```

Arguments

object	instance of H5S_source(updated object with path to file set)
dsindex	dataset index
selectionString	character with selectionString
...	unused

Value

list of data obtained

Examples

```
hsdsCon = H5S_source(URL_hsds()) # hsds server
setPath(hsdsCon, "/home/stvjc/hdf5_mat.h5")-> hsds
getDatasetSlice(hsds,dsindex=1,selectionString="[1:2,1:5]")
```

---

getDatasetUUIDs *getDatasetUUIDs from hsds server*

Description

getDatasetUUIDs from hsds server

Usage

```
getDatasetUUIDs(object)
```

Arguments

object	instance of H5S_source(updated object with path to file set)

Value

character of dataset uuid obtained

Examples

```
hsdsCon = H5S_source(URL_hsds()) # hsds server
setPath(hsdsCon, "/home/stvjc/hdf5_mat.h5")-> hsds
getDatasetUUIDs(hsds)
```

---

getDims *getDims from hsds server*

Description

getDims from hsds server

Usage

```
getDims(object, duid)
```

Arguments

object	instance of H5S_source(updated object with path to file set)
duid	character string with dataset uuid

Value

numeric content of dimensions

Examples

```
hsdsCon = H5S_source(URL_hsds()) # hsds server
setPath(hsdsCon, "/home/stvjc/hdf5_mat.h5")-> hsds
duid <- 'd-a9e4b71c-8ea2-11e8-9306-0242ac120022'
getDims(hsds, duid)
```

`getHRDF` *getHRDF from hsds server*

Description

getHRDF from hsds server

Usage

```
getHRDF(object, duid)
```

Arguments

`object`	instance of H5S_source(updated object with path to file set)
`duid`	character string with dataset uuid

Value

DataFrame of data obtained

Examples

```
hsdsCon = H5S_source(URL_hsds()) # hsds server
hsdsCon@folderPath="/home/stvjc/hdf5_mat.h5"
ds = fetchDatasets(hsdsCon) #Pick the ID of the dataset you are interested in
getHRDF(hsdsCon, "d-a9e4b71c-8ea2-11e8-9306-0242ac120022")
```

`getReq` *list information about server content available in an H5S_source hsds instance*

Description

list information about server content available in an H5S_source hsds instance

Usage

```
getReq(src)
```

Arguments

`src`	H5S_source instance

Value

data frame with 5 columns for one row for each user's data

groups *HDF5 server data groups accessor*

Description

HDF5 server data groups accessor

Usage

```
groups(object, index, ...)

## S4 method for signature 'H5S_source,missing'
groups(object, index, ...)

## S4 method for signature 'H5S_source,numeric'
groups(object, index, ...)
```

Arguments

object	H5S_source instance
index	numeric, if present, extracts metadata about selected group (sequential ordering of groups as returned by server) access for group information for HDF5 server
...	not used

Value

a data frame with group name and number of links for each group

Examples

```
## Not run:
bigec2 = H5S_source(URL_h5serv())
groups(bigec2)

## End(Not run)
```

H5S_Array *create H5S_Array instance given url (filepath) and entity (host) name*

Description

create H5S_Array instance given url (filepath) and entity (host) name

Usage

```
H5S_Array(endpoint, filepath, host)
```

Arguments

endpoint	a character(1) URL to port for HDF Server
filepath	path and name of the H5 file
host	a character(1) name of 'host' in server

Value

an instance of DelayedArray-class

Examples

```
# The true values from yriMulti data element 'banovichSE':
# > assay(banovichSE[c(1:5,329465:329469),c(1:3,63:64)])
#                    NA18498    NA18499    NA18501  |    NA18489     NA18909
# cg00000029      0.47339629  1.2943041 -0.8084735  |  0.6708168 -0.86093022
# cg00000165      1.23640861  0.2099817 -0.2683763  |  0.4446088  0.99868231
# cg00000236     -0.22258183  1.6236857 -0.8654838  |  0.1958195 -0.06090929
# cg00000289      0.65720581  0.5527470 -1.8458295  | -0.4618782  0.34934164
# cg00000363     -0.15063083  0.7498020  0.3254333  |  0.7342878  0.12940774
# #-------------------------------------------------------------------------
# ch.9.98936572R -0.07954958  0.2139431 -0.4719621  |  0.6835012  0.57758798
# ch.9.98937537R  0.04254705  1.0702770  1.7356387  | -0.1531732 -1.52889773
# ch.9.98959675F -1.59253143  0.2982456 -1.1954030  | -1.3703135  0.28974909
# ch.9.98989607R -1.80646652  0.4760022  1.4771808  |  0.9479602  0.49921375
# ch.9.991104F    0.08180195 -0.2434306  1.0281002  | -0.1653721  0.55612215
#
```

H5S_Array-class *extension of DelayedArray for HDF Server content*

Description

extension of DelayedArray for HDF Server content

H5S_ArraySeed-class *H5S_Array for HDF Server content*

Description

H5S_Array for HDF Server content

H5S_dataset *construct H5S_dataset object*

Description

construct H5S_dataset object

Slots

`source` instance of H5S_source instance

`simpleName` character string naming dataset

`shapes` list including dimension information

`hrefs` DataFrame of hrefs as defined in the API

`allatts` list of all attributes

`presel` string prepared for select operation in GET

`transfermode` default "JSON" or "binary" for binary transfer

H5S_dataset2 *H5S_dataset2 for datasets in hsds server*

Description

H5S_dataset2 for datasets in hsds server

Usage

```
H5S_dataset2(object, duid)
```

Arguments

`object`	instance of H5S_source(updated object with path to file set)
`duid`	character vector with dataset uuid of interest

Value

H5S_dataset object

Examples

```
hsdsCon = H5S_source(URL_hsds()) # hsds server
hsdsCon@folderPath="/home/stvjc/hdf5_mat.h5"
ds = fetchDatasets(hsdsCon) #Pick the dataset id of interest
H5S_dataset2(hsdsCon, "d-a9e4b71c-8ea2-11e8-9306-0242ac120022")
```

H5S_Matrix-class *extension of DelayedMatrix for HDF Server content*

Description

extension of DelayedMatrix for HDF Server content

H5S_source *H5S_source identifies an HDF5/HSDS server and manages some metadata about contents*

Description

H5S_source identifies an HDF5/HSDS server and manages some metadata about contents

construct H5S_source

Usage

```
H5S_source(serverURL, domain, ...)

## S4 method for signature 'H5S_source,character'
x[[i, j]]
```

Arguments

serverURL	a URL for a port for HDF5Server
domain	character string with path to file for HSDS
...	not used
x	instance of H5S_source
i	character string intended to identify dataset on server
j	not used

Value

an initialized object of type H5S_source

Slots

serverURL character string with a URL

dsmeta DataFrame instance with metadata about content of h5serv server

dmains DataFrame instance with metadata about the content of hsds server

getReq DataFrame instance with metadata about hsds server

folderPath character string with path to user's folder/file on hsds server

Note

The dsmeta slot holds a DataFrame with a column dsnames that is a list with ith element a character vector of all dsnames available for the ith group. There is no effort at present to search all groups for candidate datasets.

If the domain for the HSDS server is known, pass the domain path as a character string along with ther serverURL

Examples

```
## Not run:
bigec2 = H5S_source(URL_h5serv()) # h5serv
bigec2
dsmeta(bigec2)[1:2,]       # two groups
dsmeta(bigec2)[1,2][[1]]   # all dataset candidates in group 1

## End(Not run)
hsdsCon = H5S_source(URL_hsds()) # hsds server connection
hsdsCon
getReq(hsdsCon)
setPath(hsdsCon,"/home/stvjc/hdf5_mat.h5") -> hsds
fetchDatasets(hsds)     # grab the dataset id of interest
H5S_dataset2(hsds, "d-a9e4b71c-8ea2-11e8-9306-0242ac120022")
```

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 '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 HSDSArray

See Also

Other HSDSArray: HSDSMatrix, as

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 *Construct an object of type HSDSDataset*

Description

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

```
src <- HSDSSource('http://hsdshdflab.hdfgroup.org')
f <- HSDSFile(src, '/home/spollack/testzero.h5')
d <- HSDSDataset(f, '/grpA/grpAB/dsetX')
```

HSDSDataset-class *An S4 class to represent a dataset in a HDF5 file.*

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 hiearchy.

`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 *Construct an object of type HSDSFile*

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.

Value

an initialized object of type HSDSFile

Examples

```
src <- HSDSSource('http://hsdshdflab.hdfgroup.org')
f10x <- HSDSFile(src, '/shared/bioconductor/tenx_full.h5')
```

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

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

hsdsInfo *HSDS server get request accessor*

Description

HSDS server get request accessor

Usage

```
hsdsInfo(object)
```

Arguments

`object`	H5S_source instance

Value

a data frame with response

Examples

```
hsdsCon = H5S_source(URL_hsds()) # hsds server connection
hsdsInfo(hsdsCon)
```

HSDSMatrix *DelayedMatrix subclass for a two-dimensional HSDSArray*

Description

DelayedMatrix subclass for a two-dimensional HSDSArray

See Also

Other HSDSArray: `HSDSArray`, `as`

HSDSSource — *Construct an object of type HSDSSource.*

Description

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

Usage

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

Arguments

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

Value

An object of type HSDSSource

Examples

```
src.hsds <- HSDSSource('http://hsdshdflab.hdfgroup.org')
```

HSDSSource-class — *An S4 class to represent a HDF5 server listening on a port.*

Description

An S4 class to represent a HDF5 server listening on a port.

Slots

endpoint URL for server

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

`HSDS_Matrix` *simplify construction of DelayedMatrix from url and path in HSDS*

Description

simplify construction of DelayedMatrix from url and path in HSDS

Usage

```
HSDS_Matrix(url, path, title)
```

Arguments

`url`	character(1) URL for HSDS object store with port
`path`	character(1) path from root defining HDF Cloud resource
`title`	character(1) name of dataset to use

Value

instance of DelayedArray

Examples

```
HSDS_Matrix(URL_hsds(), "/shared/bioconductor/darmgcls.h5")
```

`HSDS_Matrix_OLD` *simplify construction of DelayedMatrix from url and path in HSDS*

Description

simplify construction of DelayedMatrix from url and path in HSDS

Usage

```
HSDS_Matrix_OLD(url, path)
```

Arguments

`url`	character(1) URL for HSDS object store with port
`path`	character(1) path from root defining HDF Cloud resource

Value

instance of DelayedArray

Examples

```
HSDS_Matrix
```

internalDim *acquire internal HDF5 dimension information for matrix*

Description

acquire internal HDF5 dimension information for matrix

Usage

```
internalDim(h5d)
```

Arguments

h5d	instance of H5S_dataset

Value

vector with dimensions of dataset

Examples

```
## Not run:
bigec2 = H5S_source(URL_h5serv())
tex <- bigec2[["tenx_100k_sorted"]]
internalDim(tex)

## End(Not run)
```

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

Description

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

sproc makes vector of type character of triplets initial:final:stride in R-conventions

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))
```

`links` *access for link metadata for HDF5 server groups*

Description

access for link metadata for HDF5 server groups

Usage

```
links(object, index, ...)
```

Arguments

`object`	H5S_source instance
`index`	numeric group index
`...`	not used

Value

an object of type H5S_linkset with the linkset of the group

Examples

```
## Not run:
bigec2 = H5S_source(URL_h5serv())
lks <- links(bigec2, 1)    # linkset for root group
urls <- targets(lks)       # URLs of datasets in linkset

## End(Not run)
```

listDatasets *Search inner file hierarchy for datasets*

Description

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

Usage

```
listDatasets(file)
```

Arguments

file	an object of type HSDSFile to be searched

Value

a list of inner-paths

Examples

```
src <- HSDSSource('http://hsdshdflab.hdfgroup.org')
f <- HSDSFile(src, '/home/spollack/testzero.h5')
listDatasets(f)
```

listDomains *List files and subdirectories of a domain*

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.

Value

a vector of domains in the rootdir

Examples

```
src.hsds <- HSDSSource('http://hsdshdflab.hdfgroup.org')
src.chan <- HSDSSource('http://h5s.channingremotedata.org:5000', 'h5serv')
listDomains(src.chan)
listDomains(src.hsds, '/home/jreadey')
```

---

rhdf5client *rhdf5client: A package for accessing HDFGroup HDF5 servers from R.*

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‘.

---

setPath *set path for hsds server resource*

Description

set path for hsds server resource

Usage

```
setPath(object, folderPath, ...)
```

Arguments

object	H5S_source instance
folderPath	character string with path to user’s folder on hsds server
...	not used

Value

an updated object with folderPath set

Examples

```
hsdsCon = H5S_source(URL_hsds()) # hsds server connection
setPath(hsdsCon, "/home/stvjc/hdf5_mat.h5")-> hsds
```

targets *provide the full URLs for link members*

Description

provide the full URLs for link members

Usage

```
targets(h5linkset, index)
```

Arguments

h5linkset	instance of H5S_linkset
index	numeric index into link vector - ignored

Value

a vector of dataset tags

Examples

```
## Not run:
bigec2 = H5S_source(URL_h5serv())
lks <- links(bigec2, 1)    # linkset for root group
urls <- targets(lks)       # URLs of datasets in linkset

## End(Not run)
```

transfermode<- *replace transfer mode*

Description

replace transfer mode

Usage

```
transfermode(object) <- value
```

Arguments

object	instance of H5S_linkset
value	either "JSON" (default) or "binary"

Value

updated object of type H5S_dataset

URL_h5serv *manage h5serv URL*

Description

manage h5serv URL

Usage

```
URL_h5serv()
```

Value

URL of h5serv server

Examples

```
URL_h5serv()
```

URL_hsds *manage hsds URL*

Description

manage hsds URL

Usage

```
URL_hsds()
```

Value

URL of hsds server

Examples

```
URL_hsds()
```

`[,H5S_dataset,numeric,numeric-method`
extract elements from H5S_dataset

Description

extract elements from H5S_dataset

extract elements from H5S_dataset

extract elements of a one or two-dimensional HSDSDataset

Usage

```
## S4 method for signature 'H5S_dataset,numeric,numeric'
x[i, j, ..., drop = FALSE]

## S4 method for signature 'H5S_dataset,character,character'
x[i, j, ..., drop = FALSE]

## S4 method for signature 'HSDSDataset,numeric,ANY'
x[i]
```

Arguments

`x`	instance of H5S_dataset
`i`	select option for first matrix index in HDF5 server value API
`j`	select option for second matrix index in HDF5 server value API
`...`	unused
`drop`	logical defaults to FALSE
`x`	object of type HSDSDataset
`i`	vector of indices (first dimension)

Value

matrix of data obtained

an array with the elements requested from the HSDSDataset

Index