

# Package ‘via’

May 16, 2023

**Type** Package

**Title** Virtual Arrays

**Version** 0.2.0

**Collate** zzz.R data.R generics.R utility.R classes.R  
VirtualArray-base.R VirtualArray-xattrib.R  
VirtualArray-subset.R VirtualArray-replace.R  
VirtualArray-combine.R XArray-base.R XArray-xattrib.R  
XArray-combine.R RasterArray-base.R RasterArray-xattrib.R  
RasterArray-groupgen.R RasterArray-replace.R  
RasterArray-combine.R RasterArray-rast.R SfcArray-base.R  
SfcArray-replace.R SfcArray-sf.R SfArray-base.R  
SfArray-replace.R conversions.R

**Maintainer** Adam T. Kocsis <adam.t.kocsis@gmail.com>

**Description** The base class 'VirtualArray' is defined, which acts as a wrapper around lists allowing users to fold arbitrary sequential data into n-dimensional, R-style virtual arrays. The derived 'XArray' class is defined to be used for homogeneous lists that contain a single class of objects. The 'RasterArray' and 'SfArray' classes enable the use of stacked spatial data instead of lists.

**License** CC BY 4.0

**Date** 2023-05-15

**BugReports** <https://github.com/adamkocsis/via>

**Encoding** UTF-8

**LazyData** false

**Depends** R (>= 4.0.0)

**Imports** methods

**NeedsCompilation** no

**RoxygenNote** 7.2.3

**Suggests** tinytest, knitr, rmarkdown, terra, sp, sf

**Author** Adam T. Kocsis [cre, aut] (<<https://orcid.org/0000-0002-9028-665X>>),  
Deutsche Forschungsgemeinschaft [fnd],  
FAU GeoZentrum Nordbayern [fnd]

**Repository** CRAN

**Date/Publication** 2023-05-16 13:50:02 UTC

**R topics documented:**

aggregate	3
as.XArray	3
c,VirtualArray-method	4
colnames,VirtualArray-method	5
crop	5
dim,VirtualArray-method	6
dimlayer	7
dimnames,VirtualArray-method	7
disagg	8
exemplar	9
ext	9
is.na.RasterArray	10
layers	10
names,VirtualArray-method	11
ncell	12
ncol,VirtualArray-method	12
newbounds	13
nlayers	14
nums	15
nvalues	16
paleocoastlines	16
project	17
proxy	18
RasterArray-class	18
rastex	19
resample	20
rotate	21
rownames,VirtualArray-method	21
SfcArray-class	22
st_bbox	23
st_crs	23
st_transform	24
subset,VirtualArray-method	24
t,VirtualArray-method	25
via	26
XArray-class	26
xres	27
[,VirtualArray,ANY,ANY-method	28
[<-,VirtualArray,ANY,ANY,logical-method	29
[[,VirtualArray,ANY,ANY-method	30
[[<-,VirtualArray,ANY-method	31

---

aggregate	<i>Aggregate raster cells in a 'RasterArray'-class object</i>
-----------	---

---

**Description**

The method is inherited from the 'SpatRaster' class.

**Usage**

```
## S4 method for signature 'RasterArray'
aggregate(x, ...)
```

**Arguments**

x                    a [RasterArray](#)-class object.  
 ...                  arguments passed to the [aggregate](#) function.

**Value**

An aggregated [RasterArray](#)-class object.

**Examples**

```
library(terra)
ex <- rastex()
agg <- aggregate(ex, 30)
```

---

as.XArray	<i>Coerce into an SfcArray or SfArray object</i>
-----------	--

---

**Description**

Coerce into an [SfcArray](#) or [SfArray](#) object

Coerce into an [SfArray](#) or [SfcArray](#) object

**Usage**

```
as.XArray(from)

## S4 method for signature 'SfcArray'
as.XArray(from)

## S4 method for signature 'SfArray'
as.XArray(from)
```

```

as.SfcArray(from)

## S4 method for signature 'XArray'
as.SfcArray(from)

## S4 method for signature 'SfArray'
as.SfcArray(from)

as.SfArray(from)

## S4 method for signature 'XArray'
as.SfArray(from)

## S4 method for signature 'SfcArray'
as.SfArray(from)

```

### Arguments

from                    Either a [SfcArray](#), [SfArray](#) or [XArray](#)-class object

### Value

Either a [SfcArray](#), [SfArray](#) or [XArray](#)-class object

---

c, VirtualArray-method *Combine a one-dimensional 'VirtualArray'-class object with other objects*

---

### Description

NOTE: Sequences that start with an NA do not yet work.

### Usage

```
## S4 method for signature 'VirtualArray'
c(x, ...)
```

### Arguments

x                        [VirtualArray](#) object to combine with other objects.  
...                        additional objects to combine.

### Value

A [VirtualArray](#)-class object.

---

colnames,VirtualArray-method

*Column names of two-dimensional 'VirtualArray'-derived class object.*

---

### Description

Get or set the column names of two-dimensional 'VirtualArray'-derived class objects

### Usage

```
## S4 method for signature 'VirtualArray'  
colnames(x)  
  
## S4 replacement method for signature 'VirtualArray'  
colnames(x) <- value
```

### Arguments

x                    [VirtualArray](#)-derived class object.  
value                character vector.

### Value

A character vector of column names or NULL.

### Examples

```
data(paleocoastlines)  
colnames(paleocoastlines)  
colnames(paleocoastlines) <- c("a", "b")
```

---

crop                    *Cropping a 'RasterArray'-class object*

---

### Description

The method is inherited from the '[SpatRaster](#)' class.

### Usage

```
## S4 method for signature 'RasterArray'  
crop(x, y, ...)
```

**Arguments**

- x a [RasterArray](#) class object.
- y an [SpatExtent](#)-class object, or any object from which an extent object can be extracted (see [Details](#))
- ... arguments passed to the [crop](#) function.

**Value**

A cropped [RasterArray](#)-class object.

**Examples**

```
ex <- rastex()
# crop to a specific area
if(requireNamespace("terra", quietly=TRUE)){
  ext <- terra::ext(c(
    xmin = 106.58,
    xmax = 157.82,
    ymin = -45.23,
    ymax = 1.14
  ))
  # cropping all
  au<- crop(ex, ext)
}
```

---

dim, VirtualArray-method

*Dimensions of 'VirtualArray'-derived class objects*

---

**Description**

The function returns the dimensions of the array in which elements are organized.

**Usage**

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

**Arguments**

- x A [VirtualArray](#)-derived class object.

**Value**

A numeric vector.

**Examples**

```
data(exemplar)
dim(exemplar)
```

---

dimlayer	<i>Dimensions of layers in a 'VirtualArray'-class object</i>
----------	--

---

**Description**

The function will return the dimensions 'SpatRaster'-class layers.

**Usage**

```
dimlayer(x, ...)

## S4 method for signature 'RasterArray'
dimlayer(x)
```

**Arguments**

x	A <a href="#">VirtualArray</a> class object.
...	additional arguments passed to class-specific methods.

**Value**

A numeric vector with the number of rows and columns in the [VirtualArrays](#).

---

dimnames,VirtualArray-method	<i>Names of a multidimensional 'VirtualArray'-derived class object.</i>
------------------------------	---

---

**Description**

Get or set the dimnames of multidimensional [VirtualArray](#)-derived class object.

**Usage**

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

## S4 replacement method for signature 'VirtualArray'
dimnames(x) <- value
```

**Arguments**

x                    [RasterArray](#) or [SfArray](#) object.  
value                character vector.

**Value**

A list of character vectors or NULL.

**Examples**

```
ex <- rastex()
dimnames(ex)
data(paleocoastlines)
dimnames(paleocoastlines)
dimnames(paleocoastlines)[[2]] <- c("first", "second")
names(dimnames(paleocoastlines)) <- c("age", "type")
```

---

disagg

*Disaggregate raster cells in a '[RasterArray](#)'-class object*

---

**Description**

The method is inherited from the '[SpatRaster](#)' class.

**Usage**

```
## S4 method for signature 'RasterArray'
disagg(x, ...)
```

**Arguments**

x                    a [RasterArray](#) class object.  
...                  arguments passed to the [disagg](#) function.

**Value**

A disaggregated [RasterArray](#) class object.

**Examples**

```
ex <- rastex()
disagg <- disagg(ex, 3)
```



---

exemplar	<i>Example 'XArray'-class object</i>
----------	--------------------------------------

---

**Description**

A 'XArray'-class objects of data.frames, which were made from a single data.frame with random sampling. The original object had two columns, the first (x) an integer sequence 1:100, the second y a variable produced with  $0.5 * x - 30 + N(0, 10)$ .

**Usage**

```
data(exemplar)
```

**Format**

: XArray with 3 sample sizes (rows), and 4 different seeds (column).

---

ext	<i>Extent of a 'RasterArray'-class object</i>
-----	---

---

**Description**

The method is inherited from the 'SpatRaster' class.

**Usage**

```
## S4 method for signature 'RasterArray'
ext(x, ...)
```

**Arguments**

x                    a RasterArray-class object.  
 ...                  arguments passed to the ext function.

**Value**

An aggregated RasterArray-class object.

**Examples**

```
ex <- rastex()
extent <- ext(ex)
```

---

`is.na.RasterArray`      *Positions of missing values in a 'RasterArray'-class object*

---

### Description

The function behaves similar to the regular `is.na()` function applied to the proxy object of a 'RasterArray'.

### Usage

```
## S3 method for class 'RasterArray'
is.na(x)
```

### Arguments

`x`                      A RasterArray class object.

### Value

A logical vector, matrix or array matching the structure of the RasterArray.

### Examples

```
ex <- rastex()
ex[2] <- NA
is.na(ex)
```

---

`layers`                      *Names of layers in the stack of a 'VirtualArray'-class object*

---

### Description

Names of layers in the stack of a 'VirtualArray'-class object

### Usage

```
layers(x, ...)
```

```
## S4 method for signature 'VirtualArray'
layers(x)
```

### Arguments

`x`                      A `VirtualArray`-derived class object.  
`...`                    additional arguments passed to class-specific methods.

**Value**

A character vector of names.

**Examples**

```
# names of layers in the stack
data(exemplar)
layers(exemplar)
```

---

names,VirtualArray-method

*Names of one-dimensional 'VirtualArray'-derived class objects.*

---

**Description**

Get or set the names of one-dimensional 'VirtualArray'-derived class objects

**Usage**

```
## S4 method for signature 'VirtualArray'
names(x)

## S4 replacement method for signature 'VirtualArray'
names(x) <- value
```

**Arguments**

x                    [VirtualArray](#)-derived class object.  
value                character vector.

**Value**

A character vector of names or NULL.

**Examples**

```
ex <- rastex()
names(ex)
names(ex)[4] <- "weirdo"
# NULL
```

---

ncell	<i>Number of cells in a 'RasterArray'-class object</i>
-------	--

---

**Description**

The method is inherited from the 'SpatRaster' class.

**Usage**

```
## S4 method for signature 'RasterArray'
ncell(x)
```

**Arguments**

x                    a [RasterArray](#) class object.

**Value**

A numeric value.

**Examples**

```
ex <- rastex()
ncell(ex)
```

---

ncol, VirtualArray-method	<i>Number of columns and rows of a 'VirtualArray'-derived class object.</i>
---------------------------	---

---

**Description**

Unlike the ncol and nrow functions of the 'terra' package, this function returns the number of columns and rows of the 'VirtualArray'-derived container, rather than the dimensions of the contained 'SpatRaster'-class object.

**Usage**

```
## S4 method for signature 'VirtualArray'
ncol(x)
```

```
## S4 method for signature 'VirtualArray'
nrow(x)
```

**Arguments**

x                    A [VirtualArray](#)-derived class object.

**Value**

A numeric value of the number of columns and rows.

**Examples**

```
data(paleocoastlines)
ncol(paleocoastlines)
nrow(paleocoastlines)
```

---

newbounds

*Redefine bounds of a named matrix*

---

**Description**

The function restructures a `matrix` and extends its current limits to a range defined by a names attribute

**Usage**

```
newbounds(x, cols = NULL, rows = NULL)
```

**Arguments**

<code>x</code>	The matrix to be restructured.
<code>cols</code>	Column names guiding the restructuring.
<code>rows</code>	Row names guiding the restructuring.

**Details**

This is essentially a subsetting function that allows you to subset even when the rownames or colnames vector extends beyond the bounds of a matrix and traditional subsetting methods result in the notorious 'out of bounds' error.

**Value**

A matrix with extended bounds.

**Examples**

```
a<-matrix(1:9, ncol=3)
rownames(a) <- c("a", "c", "d")
newbounds(a, rows=letters[1:5])
```

---

nlayers	<i>Number of elements or layers in a <a href="#">VirtualArray</a>-derived class object</i>
---------	--

---

### Description

Function to return the length of the array in which elements are organized.

### Usage

```
nlayers(x)

## S4 method for signature 'list'
nlayers(x)

## S4 method for signature 'SpatRaster'
nlayers(x)

## S4 method for signature 'VirtualArray'
length(x)

## S4 method for signature 'XArray'
nlayers(x)

## S4 method for signature 'RasterArray'
nlayers(x)
```

### Arguments

x a [VirtualArray](#)-derived class object.

### Details

The `length()` function returns the number elements that should be present based on the array structure itself, and not the total number of values stored in the object. As the object can contain missing values, the number of actual layers can be queried with [nlayers](#).

### Value

A numeric value.

### Examples

```
ex <- rastex()
# omit third element
ex[3] <- NA
# number of elements in the RasterArray
length(ex)
```

```
# remaining number values in the stack
length(ex@stack)
# the number of remaining layers in the RasterArray
nlayers(ex)
```

---

nums

*Names as numerics*

---

## Description

The set of functions return names of objects directly cast to numeric values.

## Usage

```
nums(x)
```

```
colnums(x)
```

```
rownums(x)
```

## Arguments

x                    Object with names, colnames or rownames attributes.

## Value

Numeric vector.

## Examples

```
# base R object
a <- 1:10
names(a) <- seq(10, 100, 10)
nums(a)

# XArray
data(exemplar)
colnums(exemplar)
rownums(exemplar)
```

---

nvalues	<i>The total number of values in a 'RasterArray'-class object</i>
---------	---

---

### Description

The total number of values in a 'RasterArray'-class object

### Usage

```
nvalues(x, ...)
```

```
## S4 method for signature 'RasterArray'
```

```
nvalues(x)
```

### Arguments

x            A [RasterArray](#)-class object.

...           additional arguments passed to class-specific methods.

### Value

A numeric value.

### Examples

```
ex <- rastex()
```

```
nvalues(ex)
```

---

paleocoastlines	<i>PaleoMAP PaleoCoastlines (excerpt)</i>
-----------------	---

---

### Description

A dataset containing the coastline reconstructions based on the PaleoMAP PaleoDEMS <https://www.earthbyte.org/paleodem-resource-scotese-and-wright-2018/> and the Paleobiology Database <https://paleobiodb.org> for 0, 10 and 20Ma.

### Usage

```
data(paleocoastlines)
```

### Format

A [SfcArray](#) with 3 continental margin and 3 paleocoastline layers (3 rows and 2 columns).



## Details

This is version v7. The article describing the entire set is under review. Once that is published, the entire dataset will be available.

## Source

Kocsis, A. T., & Scotese, C. R. (2020). PaleoMAP PaleoCoastlines data [Data set]. Zenodo. <https://doi.org/10.5281/zenodo.3903164>

---

project	<i>Project a 'RasterArray'-class object</i>
---------	---

---

## Description

The method implements the [project](#) function for 'RasterArray'-class objects.

The method is inherited from the 'SpatRaster' class. See [project](#) for details.

## Usage

```
project

## S4 method for signature 'RasterArray'
project(x, y, ...)
```

## Arguments

x	A <a href="#">RasterArray</a> object to project.
y	A <a href="#">RasterArray</a> the same options as in <a href="#">project</a> .
...	additional arguments as for <a href="#">project</a> .

## Format

An object of class standardGeneric of length 1.

## Value

A projected [RasterArray](#)-class object.

## Examples

```
# project first three to mollweide
ex <- rastex()
mollEx <- project(ex[1:3], y="ESRI:54009")
```

---

 proxy

*The proxy of an from a class derived from 'VirtualArray'*


---

### Description

This function returns an object that symbolizes the structure of layers in the 'XArray', 'RasterArray' or 'SfArray'.

### Usage

```
proxy(x, ...)
```

```
## S4 method for signature 'VirtualArray'
```

```
proxy(x)
```

### Arguments

x                    XArray, RasterArray or SfArray object.

...                  additional arguments passed to class-specific methods.

### Details

The proxy method wraps the names of layers in the @stack using the @index slot of the 'VirtualArray'.

### Value

A vector, matrix or array of characters representing the VirtualArray structure.

### Examples

```
data(exemplar)
proxy(exemplar)
```

```
data(paleocoastlines)
proxy(paleocoastlines)
```

---

 RasterArray-class

*Array of 'SpatRaster'-class objects*


---

### Description

Array class for easier navigation of multilayer rasters

**Arguments**

stack	A <a href="#">SpatRaster</a> object.
index	A vector, matrix or array type object. Includes either the indices of layers in the stack, or their names.
dim	A numeric vector. Same as for array, creates proxy procedurally.

**Details**

The class implements structures to organize single-layer '[SpatRaster](#)'-class objects that have the same dimensions and coordinate reference system. Subsetting rules were defined using the proxy object in the @index slot. See examples for implementations.

The class has two slots: @stack: A '[SpatRaster](#)'-class object with multiple layers, the actual data.  
index: A proxy object that represents the organization of the layer in the array.

**Value**

A '[RasterArray](#)'-class object.

**Examples**

```
# example data
ex <- rastex()
st <- ex@stack
ind <- 1:6
names(ind) <- letters[1:length(ind)]
ra <- RasterArray(stack=st, index=ind)
```

---

rastex	<i>Procedural example structure to demonstrate the capabilities of the '<a href="#">RasterArray</a>' class</i>
--------	--

---

**Description**

Binary versions of [SpatRaster](#)-class objects are problematic, this function is used to instantiate a [RasterArray](#) example.

**Usage**

```
rastex()
```

**Value**

A two-dimensional [RasterArray](#)-class object, with three rows and four columns.

## Examples

```
# create example
example <- rastex()

# subset - single bracket
example['b']

# subset - single bracket
example[c(4, 6)]

# subset - double bracket
example[["layer_2"]]
```

---

resample

*Resampling a 'RasterArray'-class object*

---

## Description

The method is inherited from the '[SpatRaster](#)' class.

## Usage

```
## S4 method for signature 'RasterArray,ANY'
resample(x, y, ...)
```

## Arguments

**x** a [RasterArray](#)-class object.

**y** The y argument of the [resample](#) function.

**...** arguments passed to the [resample](#) function.

## Value

A resampled [RasterArray](#)-class object.

## Examples

```
ex <- rastex()
if(requireNamespace("terra", quietly=TRUE)){
  template <- terra::rast(res=5)
  resampled <- resample(ex, template)
}
```

---

rotate	<i>Rotate a 'RasterArray'-class object</i>
--------	--

---

**Description**

The method is inherited from the 'SpatRaster' class.

**Usage**

```
## S4 method for signature 'RasterArray'
rotate(x, ...)
```

**Arguments**

x	(RasterArray) Object.
...	Additional arguments passed to the rotate function.

**Value**

A RasterArray-class object.

---

rownames,VirtualArray-method	<i>Row names of two-dimensional 'VirtualArray'-derived class objects.</i>
------------------------------	---

---

**Description**

Get or set the row names of two-dimensional 'VirtualArray'-derived class object

**Usage**

```
## S4 method for signature 'VirtualArray'
rownames(x)

## S4 replacement method for signature 'VirtualArray'
rownames(x) <- value
```

**Arguments**

x	VirtualArray-class object.
value	character vector.

**Value**

A character vector of row names or NULL.

**Examples**

```
data(paleocoastlines)
rownames(paleocoastlines)
rownames(paleocoastlines) <- paste(rownames(paleocoastlines), "Ma")
```

---

SfcArray-class	Array of 'sf'-derived class data
----------------	----------------------------------

---

**Description**

Array class for easier navigation of vector spatial datasets

**Arguments**

stack	A list of sf-class objects or sfc-class objects.
index	A vector, matrix or array type object. Includes either the indices of layers in the stack, or their names.
dim	A numeric vector. Same as for array, creates proxy procedurally.

**Details**

The class implements structures to organize entire 'sfc' and 'sf' objects that share coordinate reference systems. The 'SfcArray' class is derived from 'XArray' and represents arrays of geometry sets. The 'SfArray' class is derived from 'SfArray', that allows the wrapping of 'sf' objects with attributes. Subsetting rules were defined using the proxy object in the @index slot. See examples for implementations.

The classes have two slots: @stack: A list object with multiple 'sf' class layers, the actual data. @index: A proxy object that represents the organization of the layers.

**Value**

An 'SfcArray' or 'SfArray'-class object.

**Examples**

```
# example data
library(sf)
data(paleocoastlines)
st <- paleocoastlines@stack
ind <- 1:nlayers(st)
dim(ind) <- c(3,2)
dimnames(ind) <- list(age=c(0, 10, 20), c("margin", "coastlines"))
sa <- SfcArray(stack=st, index=ind)
```

---

st_bbox	<i>Bounding box of an 'SfArray'-class object</i>
---------	--

---

**Description**

The method is inherited from the 'sf' class.

**Usage**

```
## S3 method for class 'SfcArray'  
st_bbox(obj, ...)
```

**Arguments**

obj	a <i>sf</i> -class object.
...	arguments passed to the <i>st_bbox</i> function.

**Value**

An *RasterArray* class object.

**Examples**

```
data(paleocoastlines)  
bb<- st_bbox(paleocoastlines)
```

---

st_crs	<i>Coordinate reference system of an 'SfArray'-class object</i>
--------	---

---

**Description**

The method is inherited from the 'sf' class.

**Usage**

```
## S3 method for class 'SfcArray'  
st_crs(x, ...)
```

**Arguments**

x	a <i>sf</i> -class object.
...	arguments passed to the <i>st_crs</i> function.

**Value**

An aggregated *RasterArray* class object.

**Examples**

```
data(paleocoastlines)
crs <- st_crs(paleocoastlines)
```

---

st_transform	<i>Projection change of an 'SfArray'-class object</i>
--------------	---

---

**Description**

The method is inherited from the 'sf' class.

**Usage**

```
## S3 method for class 'SfcArray'
st_transform(x, ...)
```

**Arguments**

x                    a *sf*-class object.  
 ...                 arguments passed to the `st_transform` function.

**Value**

An *RasterArray*-class object.

**Examples**

```
data(paleocoastlines)
moll<- st_transform(paleocoastlines, "ESRI:54009")
plot(moll["20", "margin"], col="cyan")
plot(moll["20", "coast"], add=TRUE, col="brown")
```

---

subset,VirtualArray-method	<i>Subset a 'VirtualArray'-class object</i>
----------------------------	---

---

**Description**

Extract subsets of an object from a class derived from 'VirtualArray' similarly to a regular array.

**Usage**

```
## S4 method for signature 'VirtualArray'
subset(x, i, j, ..., oneDim = FALSE, drop = TRUE)
```



**Arguments**

x	VirtualArray-class object.
i	subscript of the first dimension(rows) or vector-like subsetting.
j	subscript of the second dimension (columns).
...	subscript of additional dimensions.
oneDim	logical In case of multidimensional VirtualArrays, setting oneDim to TRUE allows the application of one dimensional subscripts.
drop	logical in case the result of subsetting is a single element, should the \codeVirtualArray wrapper be dropped?

**Value**

Either the same class as x, or the class that forms the element of the VirtualArray.

**Examples**

```
ex <- rastex()
# first 4
subset(ex, i=1:4)
# missing at the end
subset(ex, i=1:12)
# character subscript
subset(ex, i=c("a", "b"))
# logical subscript
subs <- rep(TRUE, length(ex))
subs[1] <- FALSE # remove first
subset(ex, i= subs)
# no drop
subset(ex, i=1, drop=FALSE)
```

---

t,VirtualArray-method *Transpose a 'VirtualArray'-class object*

---

**Description**

Transpose a 'VirtualArray'-class object

**Usage**

```
## S4 method for signature 'VirtualArray'
t(x)
```

**Arguments**

x	A VirtualArray-class object.
---	------------------------------

**Value**

A `VirtualArray`-class object.

**Examples**

```
data(exemplar)
t(exemplar)
data(paleocoastlines)
t(paleocoastlines)
```

---

via

*Virtual Arrays*

---

**Description**

The base class `'VirtualArray'` is defined, which acts as a wrapper around lists allowing users to fold arbitrary sequential data into n-dimensional, R-style virtual arrays. The derived `'XArray'` class is defined to be used for homogeneous lists that contain a single class of objects. The `'RasterArray'` and `'SfArray'` classes enable the use of stacked spatial data instead of lists. # This is still the pre-alpha version. As is R, this is free software and comes with ABSOLUTELY NO WARRANTY. Nevertheless, notes about found bugs and suggestions are more than welcome.

**Author(s)**

Adam T. Kocsis (adam.t.kocsis@gmail.com)

---

XArray-class

*Virtual array of general R objects*

---

**Description**

Template for construction of virtual arrays (`'VirtualArray'`) and a derived class (`'XArray'`) to instantiate it with general objects.

**Arguments**

stack	A list-class object.
index	A vector, matrix or array type object. Includes the indices of layers in the stack.
dim	A numeric vector. Same as for array, creates proxy procedurally.

## Details

The 'VirtualArray' class implements structures to organize objects of the same class in multi-dimensional arrays. Subsetting rules were defined using the proxy object in the index slot. The 'VirtualArray' is the base class for 'XArray' and 'RasterArray' classes. The 'XArray' class derived from VirtualArray allows the instantiation of basic virtual arrays with general R objects, which form a single list in the @stack slot. The 'SfArray' class is derived from the 'XArray' class.

The class has two slots: @stack: A list containing objects of the same class (i.e. layers). @index: A proxy object that represents the structure of the entities.

## Value

An XArray-class object.

## Examples

```
# 2d XArray of vectors
data(exemplar)
st <- exemplar@stack
ind <- 1:nlayers(st)
dim(ind) <- c(3,4)
dimnames(ind) <- list(n = c(10, 20, 30), seed = 1:4)
xa<- XArray(stack=st, index=ind)
```

---

xres

*Resolution of a 'RasterArray'-class object*

---

## Description

The methods are inherited from the 'SpatRaster' class, see [res](#). Replacement is not allowed.

## Usage

```
## S4 method for signature 'RasterArray'
xres(x)
```

```
## S4 method for signature 'RasterArray'
yres(x)
```

```
## S4 method for signature 'RasterArray'
res(x)
```

## Arguments

x                    a RasterArray-class object.

**Value**

A numeric vector.

**Examples**

```
ex <- rastex()
res(ex)
yres(ex)
xres(ex)
```

---

[,VirtualArray,ANY,ANY-method

*Indexing to extract subsets of a 'codeVirtualArray'-class object*

---

**Description**

Single bracket '[' refers to indices and names within the 'VirtualArray'-class object. Use double brackets to extract layers based on their names (in the @stack).

**Usage**

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

**Arguments**

x	An object from a <a href="#">VirtualArray</a> -derived class.
i	subscript of the first dimension(rows) or vector-like subsetting.
j	subscript of the second dimension (columns).
...	subscript of additional dimensions.
drop	logical in case the result of subsetting is a single element, should the <a href="#">VirtualArray</a> -derived wrapper be dropped?

**Value**

An object from either the same class as x or the class of its elements.

**Examples**

```
ex <- rastex()
# numeric subsetting
firstThree <- ex[1:3]
# character subsetting
second <- ex["d"]
# logical subsetting
subscript <- rep(FALSE, length(ex))
subscript[2] <- TRUE
```

```
second2 <- ex[subscript]
data(paleocoastlines)
present<- paleocoastlines["0", ]
allMargin <- paleocoastlines[, "margin"]
```

---

```
[<-,VirtualArray,ANY,ANY,logical-method
```

*Replace layers in an object that is of a class derived from 'VirtualArray'.*

---

## Description

Single bracket '[' refers to indices and names within the 'VirtualArray'-class object. Use double brackets to replace layers based on their names (in the @stack). Object types of the same kind class can be used to replace values in 'XArray'-class objects. 'SpatRaster'-class objects can be used to replace values in 'RasterArray'-class objects. Classes inheriting from 'sf' can be used with 'SfArray'-class objects.

## Usage

```
## S4 replacement method for signature 'VirtualArray,ANY,ANY,logical'
x[i, j, ...] <- value

## S4 replacement method for signature 'RasterArray,ANY,ANY,SpatRaster'
x[i, j, ...] <- value

## S4 replacement method for signature 'SfcArray,ANY,ANY,sfc'
x[i, j, ...] <- value

## S4 replacement method for signature 'SfArray,ANY,ANY,sf'
x[i, j, ...] <- value
```

## Arguments

x	VirtualArray-class object.
i	subscript of the first dimension(rows) or vector-like subsetting.
j	subscript of the second dimension (columns).
...	subscript of additional dimensions.
value	A same class object as x.

## Value

The function has no return value.

**Examples**

```

ex <- rastex()
# replace third element with missing value
ex[3] <- NA
# duplicate first element and make it the second too
ex[2] <- ex[1]
ex

```

---

```
[[,VirtualArray,ANY,ANY-method
```

*Indexing to extract the elements of a 'VirtualArray'-derived class object.*

---

**Description**

Double bracket '[' refers to elements'/layers' name in the @stack of the 'VirtualArray'-derived object. Use single brackets to extract elements based on their position in the 'VirtualArray'.

**Usage**

```

## S4 method for signature 'VirtualArray,ANY,ANY'
x[[i, drop = TRUE]]

```

**Arguments**

x	VirtualArray
i	subscript of the first dimension(rows) or vector-like subsetting.
drop	logical should the VirtualArray be dropped and the element be reduced to the element class?

**Value**

A VirtualArray-derived class object, or an object of the class that makes up the VirtualArray

**Examples**

```

data(exemplar)
# finds a layer
exemplar[["sample1"]]
# returns a stack
exemplar[[c("sample1", "sample2")]]
# replaces a layer values, but not the attributes of the layer
exemplar2 <- exemplar
exemplar2[["sample1"]] <- exemplar2[["sample2"]]
# compare every value in the they are all the same
exemplar2[["sample1"]]$x == exemplar2[["sample2"]]$x

```

---

`[[<- ,VirtualArray,ANY-method`*Replace elements of 'VirtualArray'-class objects.*

---

**Description**

Double bracket '[' refers to layers' name in the names of the @stack member of the 'VirtualArray'.  
Use single brackets to replace elements based on their position in the 'VirtualArray'-class object.

**Usage**

```
## S4 replacement method for signature 'VirtualArray,ANY'  
x[[i]] <- value
```

**Arguments**

x	Object from a class derived from <a href="#">VirtualArray</a> .
i	subscript of layers to replace.
value	character vector.

**Value**

The function has no return value.

# Index

\* **datasets**  
  exemplar, 9  
  paleocoastlines, 16  
  project, 17  
[, VirtualArray, ANY, ANY-method, 28  
[, VirtualArray-method  
  ([, VirtualArray, ANY, ANY-method),  
  28  
[<-, VirtualArray, ANY, ANY, logical-method,  
  29  
[<-, RasterArray, ANY, ANY, SpatRaster-method  
  ([<-, VirtualArray, ANY, ANY, logical-method),  
  29  
[<-, SfcArray, ANY, ANY, sf-method  
  ([<-, VirtualArray, ANY, ANY, logical-method),  
  29  
[<-, SfcArray, ANY, ANY, sfc-method  
  ([<-, VirtualArray, ANY, ANY, logical-method),  
  29  
[<-, VirtualArray-method  
  ([<-, VirtualArray, ANY, ANY, logical-method),  
  29  
[[, VirtualArray, ANY, ANY-method, 30  
[[, VirtualArray-method  
  ([[, VirtualArray, ANY, ANY-method),  
  30  
[[<-, VirtualArray, ANY-method, 31  
[[<-, VirtualArray, ANY, ANY-method  
  ([[<-, VirtualArray, ANY-method),  
  31  
[[<-, VirtualArray-method  
  ([[<-, VirtualArray, ANY-method),  
  31  
  
aggregate, 3, 3  
aggregate, RasterArray-method  
  (aggregate), 3  
as (as.XArray), 3  
as.XArray, 3  
  
c, VirtualArray-method, 4  
colnames, VirtualArray-method, 5  
colnames<-, VirtualArray-method  
  (colnames, VirtualArray-method),  
  5  
colnums (nums), 15  
crop, 5, 6  
crop, RasterArray-method (crop), 5  
  
dim, VirtualArray-method, 6  
dimlayer, 7  
dimlayer, RasterArray-method (dimlayer),  
  7  
dimnames, VirtualArray-method, 7  
dimnames<-, VirtualArray-method  
  (dimnames, VirtualArray-method),  
  7  
disagg, 8, 8  
disagg, RasterArray-method (disagg), 8  
  
exemplar, 9  
ext, 9, 9  
ext, RasterArray-method (ext), 9  
  
is.na.RasterArray, 10  
  
layers, 10  
layers, VirtualArray-method (layers), 10  
length, VirtualArray-method (nlayers), 14  
  
matrix, 13  
  
names, VirtualArray-method, 11  
names<-, VirtualArray-method  
  (names, VirtualArray-method), 11  
ncell, 12  
ncell, RasterArray-method (ncell), 12  
ncol, VirtualArray-method, 12  
newbounds, 13  
nlayers, 14, 14  
nlayers, list-method (nlayers), 14



- nlayers, RasterArray-method (nlayers), 14
- nlayers, SpatRaster-method (nlayers), 14
- nlayers, XArray-method (nlayers), 14
- nrow, VirtualArray-method
  - (ncol, VirtualArray-method), 12
- nums, 15
- nvalues, 16
- nvalues, RasterArray-method (nvalues), 16
  
- paleocoastlines, 16
- project, 17, 17
- project, RasterArray-method (project), 17
- proxy, 18
- proxy, VirtualArray-method (proxy), 18
  
- RasterArray, 3, 5, 6, 8–10, 12, 16–21, 23, 24, 26, 27, 29
- RasterArray (RasterArray-class), 18
- RasterArray-class, 18
- rastex, 19
- res, 27
- res (xres), 27
- res, RasterArray-method (xres), 27
- resample, 20, 20
- resample, RasterArray, ANY-method
  - (resample), 20
- resample, RasterArray-method (resample), 20
- rotate, 21, 21
- rotate, RasterArray-method (rotate), 21
- rownames, VirtualArray-method, 21
- rownames<-, VirtualArray-method
  - (rownames, VirtualArray-method), 21
- rownums (nums), 15
  
- sf, 22–24, 29
- SfArray, 3, 4, 8, 18, 22–24, 26, 27, 29
- SfArray (SfcArray-class), 22
- SfArray-class (SfcArray-class), 22
- sfc, 22
- SfcArray, 3, 4, 16, 22
- SfcArray (SfcArray-class), 22
- SfcArray-class, 22
- SpatExtent, 6
- SpatRaster, 3, 5, 7–9, 12, 17–21, 27, 29
- st\_bbox, 23, 23
- st\_crs, 23, 23
- st\_transform, 24, 24
  
- subset, VirtualArray-method, 24
- t, VirtualArray-method, 25
- terra, 12
  
- via, 26
- VirtualArray, 4–7, 10–12, 14, 18, 21, 24–26, 28–31
  
- XArray, 4, 9, 18, 22, 26, 29
- XArray (XArray-class), 26
- XArray-class, 26
- xres, 27
- xres, RasterArray-method (xres), 27
  
- yres (xres), 27
- yres, RasterArray-method (xres), 27