Package 'alabaster.sfe'

March 18, 2025

Type Package

Title Language agnostic on disk serialization of SpatialFeatureExperiment

Version 0.99.4

Description Builds upon the existing ArtifactDB project, expending alabaster.spatial for language agnostic on disk serialization of SpatialFeatureExperiment.

Depends SpatialFeatureExperiment (>= 1.9.3), alabaster.base

Imports alabaster.sce, alabaster.spatial (>= 1.5.2), EBImage, jsonlite, methods, RBioFormats, S4Vectors, sfarrow, SingleCellExperiment, spatialreg, spdep, SummarizedExperiment, terra, xml2

License MIT + file LICENSE

Encoding UTF-8

LazyData true

RoxygenNote 7.3.2

Suggests BiocStyle, fs, knitr, rmarkdown, scater, sf, SFEData, testthat (>= 3.0.0), Voyager (>= 1.9.1)

Config/testthat/edition 3

VignetteBuilder knitr

URL https://pachterlab.github.io/alabaster.sfe/

biocViews DataRepresentation, Spatial

BugReports https://github.com/pachterlab/alabaster.sfe/issues

git_url https://git.bioconductor.org/packages/alabaster.sfe

git_branch devel

git_last_commit e344d90

git_last_commit_date 2025-02-06

Repository Bioconductor 3.21

Date/Publication 2025-03-18

Author Lambda Moses [aut, cre] (ORCID:
<`https://orcid.org/0000-0002-7092-9427`>)

Maintainer Lambda Moses <`dl3764@columbia.edu`>

Contents

`readBioFormatsImage` *Read* `BioFormatsImage` *from* `alabaster` *on disk representation*

Description

Reads the metadata and `imgSource` will point to the file within the on disk representation of SFE. The image itself should not be moved or the `BioFormatsImabe` object will no longer work.

Usage

```
readBioFormatsImage(path, metadata = NULL, ...)
```

Arguments

`path`	String containing a path to a directory, itself created with a `saveObject` method.
`metadata`	Named list containing metadata for the object - most importantly, the `type` field that controls dispatch to the correct loading function. If `NULL`, this is automatically read by `readObjectFile(path)`.
`...`	Ignored, but used for other methods.

Value

A `BioFormatsImage` object for SFE.

See Also

Other readObject-SFE-image: `readExtImage()`, `readSpatRaster()`

Examples

```
library(SFEData)
fp <- tempfile()
x1 <- XeniumOutput(dataset = "v1", file_path = file.path(fp, "xenium1"))
x2 <- XeniumOutput("v2", file_path = file.path(fp, "xenium2"))

# Single file OME-TIFF
fsave <- file.path(fp, "bfi1")
sfe <- readXenium(x1)
bfi <- getImg(sfe)
bfi <- affineImg(bfi, M = matrix(c(cos(pi/6), sin(pi/6), -sin(pi/6), cos(pi/6)), nrow = 2),
                 v = c(0,0))
saveObject(bfi, fsave)
bfi2 <- readObject(fsave)

unlink(fsave, recursive = TRUE)

# Multi file OME-TIFF
fsave <- file.path(fp, "bfi2")
sfe <- readXenium(x2)
bfi <- getImg(sfe)
saveObject(bfi, fsave)
bfi2 <- readObject(fsave)
unlink(fsave, recursive = TRUE)
```

`readExtImage` *Read* `ExtImage` *from disk*

Description

Read `ExtImage` from disk

Usage

```
readExtImage(path, metadata = NULL, ...)
```

Arguments

`path`	String containing a path to a directory, itself created with a `saveObject` method.
`metadata`	Named list containing metadata for the object - most importantly, the `type` field that controls dispatch to the correct loading function. If `NULL`, this is automatically read by `readObjectFile(path)`.
`...`	Arguments passed to `readImage`.

Value

A `ExtImage` object for SFE.

See Also

Other readObject-SFE-image: `readBioFormatsImage()`, `readSpatRaster()`

Examples

```
library(SFEData)
fp <- tempfile()
fsave <- file.path(fp, "exi")
x1 <- XeniumOutput(dataset = "v1", file_path = file.path(fp, "xenium1"))
sfe <- readXenium(x1)
bfi <- getImg(sfe)
exi <- toExtImage(bfi)
saveObject(exi, fsave)
exi2 <- readObject(fsave)
unlink(fsave, recursive = TRUE)
```

`readSF` *Read* `sf` *from* `alabaster` *on disk representation*

Description

Read the GeoParquet file into R. GeoParquet should also work in any programming language that supports `arrow`. Newer version of GDAL with Parquet driver is not strictly necessary if the geometry's WKB can be converted to whichever language specific object such as `sfc` in R.

Usage

```
readSF(path, metadata = NULL)
```

Arguments

`path`	String containing a path to a directory, itself created with a `saveObject` method.
`metadata`	Named list containing metadata for the object - most importantly, the `type` field that controls dispatch to the correct loading function. If `NULL`, this is automatically read by `readObjectFile(path)`.

Value

An `sf` data frame

Examples

```
library(sf)
fp <- tempfile()
df <- as.data.frame(matrix(rnorm(10), ncol = 2))
df <- st_as_sf(df, coords = names(df), crs = NA)
saveObject(df, path = fp)
df2 <- readObject(fp)
```

`readSpatialFeatureExperiment` *Read SFE object from alabaster on disk representation*

Description

Read SFE object from alabaster on disk representation

Usage

```
readSpatialFeatureExperiment(path, metadata = NULL, ...)
```

Arguments

`path`	String containing a path to a directory, itself created with a `saveObject` method.
`metadata`	Named list containing metadata for the object - most importantly, the `type` field that controls dispatch to the correct loading function. If `NULL`, this is automatically read by `readObjectFile(path)`.
`...`	Further arguments to pass to individual methods.

Value

A `SpatialFeatureExperiment` object

Examples

```
library(SFEData)
fp <- tempfile()
fn <- file.path(fp, "vizgen")
d <- VizgenOutput(dataset = "cellpose", file_path = fn)
suppressWarnings(sfe <- readVizgen(d))
fsave <- file.path(fp, "sfe_save")
saveObject(sfe, fsave)
sfe2 <- readObject(fsave)
```

`readSpatRaster` *Read* `SpatRaster` *from* `alabaster` *on disk representation*

Description

Not the same as `terra::rast`; here this function also reads the metadata file.

Usage

```
readSpatRaster(path, metadata = NULL, ...)
```

Arguments

`path`	String containing a path to a directory, itself created with a `saveObject` method.
`metadata`	Named list containing metadata for the object - most importantly, the `type` field that controls dispatch to the correct loading function. If `NULL`, this is automatically read by `readObjectFile(path)`.
`...`	Arguments passed to `rast`, character method.

Value

A `SpatRasterImage` object for SFE.

See Also

Other readObject-SFE-image: `readBioFormatsImage()`, `readExtImage()`

Examples

```
library(SFEData)
fp <- tempfile()
fn <- file.path(fp, "vizgen")
d <- VizgenOutput(dataset = "cellpose", file_path = fn)
suppressWarnings(sfe <- readVizgen(d))
img <- getImg(sfe)
class(img)
fsave <- file.path(fp, "img")
saveObject(img, fsave)
img2 <- readObject(fsave)
```

`saveObject,BioFormatsImage-method`

Save `BioFormatsImage` *for* `alabaster`

Description

This function copies the original file to a pre-defined location within the directory that stores the on disk representation of the SFE object for data sharing. Since `BioFormatsImage` is essentially just some metadata in memory and it never loads the image into memory (once the image is loaded into memory it becomes `ExtImage`), once the original image is moved, the `BioFormatsImage` object will no longer work, which is why the pre-defined location is important. This function also saves the metadata, which includes spatial extent and affine transformations.

Usage

```
## S4 method for signature 'BioFormatsImage'
saveObject(x, path, ...)
```

Arguments

`x`	A Bioconductor object of the specified class.
`path`	String containing the path to a directory in which to save `x`.
`...`	Ignored

Value

x is saved into `path` and `NULL` is invisibly returned.

See Also

Other saveObject-SFE-image: `saveObject,ExtImage-method`, `saveObject,SpatRaster-method`

Examples

```
library(SFEData)
fp <- tempfile()
fsave <- file.path(fp, "bfi2")
x1 <- XeniumOutput(dataset = "v1", file_path = file.path(fp, "xenium1"))
sfe <- readXenium(x1)
bfi <- getImg(sfe)
saveObject(bfi, fsave)
bfi2 <- readObject(fsave)
unlink(fsave, recursive = TRUE)
```

`saveObject,ExtImage-method`

Save `ExtImage` *to disk for* `alabaster`

Description

Save `ExtImage` to disk for `alabaster`

Usage

```
## S4 method for signature 'ExtImage'
saveObject(x, path, ...)
```

Arguments

`x`	A Bioconductor object of the specified class.
`path`	String containing the path to a directory in which to save `x`.
`...`	Extra arguments passed to `writeImage`.

Value

x is saved into `path` and `NULL` is invisibly returned.

See Also

Other saveObject-SFE-image: `saveObject,BioFormatsImage-method`, `saveObject,SpatRaster-method`

Examples

```
library(SFEData)
fp <- tempfile()
fsave <- file.path(fp, "exi")
x1 <- XeniumOutput(dataset = "v1", file_path = file.path(fp, "xenium1"))
sfe <- readXenium(x1)
bfi <- getImg(sfe)
exi <- toExtImage(bfi)
saveObject(exi, fsave)
exi2 <- readObject(fsave)
unlink(fsave, recursive = TRUE)
```

`saveObject,sf-method` *Save* `sf` *to disk for* `alabaster`

Description

`sf` data frames are saved as GeoParquet.

Usage

```
## S4 method for signature 'sf'
saveObject(x, path)
```

Arguments

`x`	A Bioconductor object of the specified class.
`path`	String containing the path to a directory in which to save `x`.

Value

x is saved into `path` and `NULL` is invisibly returned.

Examples

```
library(sf)
fp <- tempfile()
df <- as.data.frame(matrix(rnorm(10), ncol = 2))
df <- st_as_sf(df, coords = names(df), crs = NA)
saveObject(df, path = fp)
```

`saveObject,SpatialFeatureExperiment-method`
Save a SpatialFeatureExperiment object

Description

Save SFE objects to disk in an interoperable, language agnostic format that may also facilitate out of memory operations via HDF5 (non-spatial, inherited from `alabaster.sce`) and Apache Parquet (geometries).

Usage

```
## S4 method for signature 'SpatialFeatureExperiment'
saveObject(x, path, ...)
```

Arguments

`x`	A Bioconductor object of the specified class.
`path`	String containing the path to a directory in which to save `x`.
`...`	Additional named arguments to pass to specific methods.

Details

There's no new arguments for `...` for the SFE, `SpatialExperiment`, and `SingleCellExperiment` methods, but there is an argument that can be specified for `...` in the `SummarizedExperiment` method of `saveObject`.

At present, spatial results in `featureData` for geometries and dimension reductions (see `colFeatureData`) and parameters of spatial analyses (see `getParams`) are not save because those parts of the SFE object are more experimental and are subject to change. `colFeatureData` where global spatial results are stored for columns of `colData(x)` is saved by the `alabaster` `saveObject` method for `DataFrame`.

Value

x is saved into `path` and `NULL` is invisibly returned.

Examples

```
library(SpatialFeatureExperiment)
library(Voyager)
library(SFEData)
library(SingleCellExperiment)
library(scater)

fp <- tempfile()
fn <- file.path(fp, "vizgen")
d <- VizgenOutput(dataset = "cellpose", file_path = fn)
```

```
suppressWarnings(sfe1 <- readVizgen(d, add_molecules = TRUE))

colGraph(sfe1, "knn5") <- findSpatialNeighbors(sfe1, method = "knearneigh", k = 5)
SpatialFeatureExperiment::centroids(sfe1)$foo <- rnorm(ncol(sfe1))
sfe1 <- logNormCounts(sfe1)
sfe1 <- runMoransI(sfe1, colGraphName = "knn5")
sfe1 <- colDataMoransI(sfe1, features = c("transcript_count", "anisotropy",
                                          "perimeter_area_ratio", "solidity"))
sfe1 <- colGeometryMoransI(sfe1, colGeometryName = "centroids", features = "foo")
sfe1 <- runPCA(sfe1, ncomponents = 10)
sfe1 <- reducedDimMoransI(sfe1, components = 1:10)

sfe1 <- runUnivariate(sfe1, type = "localmoran", features = rownames(sfe1)[1])

fsave <- file.path(fp, "sfe_vizgen")
saveObject(sfe1, fsave)
sfe2 <- readObject(fsave)

unlink(fsave, recursive = TRUE)
```

`saveObject,SpatRaster-method`

Save `SpatRaster` *to disk for* `alabaster`

Description

Intended for `SpatRasterImage` which really is `SpatRaster` that also inherits from SPE's `SpatialImage`. Besides `writeRaster`, this function also writes a metadata file in the `alabaster` framework. If the image is not loaded into memory and the original file is already spatially registered, e.g. it has a spatial extent, then the original file is copied to a pre-defined place in the on-disk representation of SFE and `writeRaster` is not called.

Usage

```
## S4 method for signature 'SpatRaster'
saveObject(x, path, ...)
```

Arguments

`x`	A Bioconductor object of the specified class.
`path`	String containing the path to a directory in which to save `x`.
`...`	Extra parameters passed to `writeRaster`.

Value

`x` is saved into `path` and `NULL` is invisibly returned.

See Also

Other saveObject-SFE-image: `saveObject,BioFormatsImage-method`, `saveObject,ExtImage-method`

Examples

```
library(SFEData)
fp <- tempfile()
fn <- file.path(fp, "vizgen")
d <- VizgenOutput(dataset = "cellpose", file_path = fn)
suppressWarnings(sfe <- readVizgen(d))
img <- getImg(sfe)
class(img)
fsave <- file.path(fp, "img")
saveObject(img, fsave)
img2 <- readObject(fsave)
```

Index