Package ‘alabaster.spatial’

June 1, 2023

Title Save and Load Spatial 'Omics Data to/from File

Description Save SpatialExperiment objects and their images into file artifacts, and load them back into memory.
This is a more portable alternative to serialization of such objects into RDS files.
Each artifact is associated with metadata for further interpretation;
downstream applications can enrich this metadata with context-specific properties.

Version 1.1.0

Date 2023-02-28

License MIT + file LICENSE

Depends SpatialExperiment, alabaster.base

Imports methods, utils, grDevices, S4Vectors, SummarizedExperiment, jsonlite, alabaster.sce

Suggests testthat, knitr, rmarkdown, BiocStyle, magick, png, digest

VignetteBuilder knitr

RoxygenNote 7.2.1

biocViews DataImport, DataRepresentation

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

git_branch devel

git_last_commit ebaf837

git_last_commit_date 2023-04-25

Date/Publication 2023-06-01

Author Aaron Lun [aut, cre]

Maintainer Aaron Lun <`infinite.monkeys.with.keyboards@gmail.com`>

R topics documented:

`loadSpatialExperiment` *Load a spatial experiment*

Description

Load a SpatialExperiment object from its constituent files in DataSetDB.

Usage

```
loadSpatialExperiment(exp.info, project)
```

Arguments

`exp.info`	Named list of metadata for a spatial 'omics experiment.
`project`	Any argument accepted by the acquisition functions, see ?`acquireFile`. By default, this should be a string containing the path to a staging directory.

Value

A SpatialExperiment object.

Author(s)

Aaron Lun

Examples

```
library(SpatialExperiment)
example(read10xVisium, echo=FALSE)
colnames(spe) <- make.unique(colnames(spe)) # forcing unique column names.

tmp <- tempfile()
dir.create(tmp)
meta <- stageObject(spe, tmp, "experiment-1")

meta$path <- "experiment-1/experiment.json"
loadSpatialExperiment(meta, tmp)
```

`loadSpatialImage` *Load a spatial image*

Description

Load an image as a SpatialImage or subclass thereof.

Usage

```
loadSpatialImage(img.info, project)
```

Arguments

`img.info`	Named list containing the metadata for this assay.
`project`	Any argument accepted by the acquisition functions, see ?`acquireFile`. By default, this should be a string containing the path to a staging directory.

Value

A SpatialImage containing the image data (or a reference to it).

Author(s)

Aaron Lun

Examples

```
example(read10xVisium, echo=FALSE)
img <- imgData(spe)$data[[1]]

tmp <- tempfile()
dir.create(tmp)
meta <- stageObject(img, tmp, "whee")

out <- loadSpatialImage(meta, tmp)
```

`stageObject,SpatialExperiment-method` *Stage a spatial experiment*

Description

Stage a SpatialExperiment object.

Usage

```
## S4 method for signature 'SpatialExperiment'
stageObject(x, dir, path, child = FALSE, ...)
```

Arguments

`x`	A SpatialExperiment object.
`dir`	String containing the path to the staging directory.
`path`	String containing a prefix of the relative path inside `dir` where `x` is to be saved. The actual path used to save `x` may include additional components, see Details.
`child`	Logical scalar indicating whether `x` is a child of a larger object.
`...`	Further arguments to pass to specific methods.

Value

A named list of the same form as that returned by the `stageObject` method for a SingleCellExperiment, but containing additional fields for the spatial data. A directory is created at `path` inside `dir` and is populated with the contents of `x`.

Author(s)

Aaron Lun

Examples

```
library(SpatialExperiment)
example(read10xVisium, echo=FALSE)
colnames(spe) <- make.unique(colnames(spe)) # forcing unique column names.

tmp <- tempfile()
dir.create(tmp)
stageObject(spe, tmp, "experiment-1")
list.files(tmp, recursive=TRUE)
```

`stageSpatialImage` *Stage images for upload to DataSetDB*

Description

Stage images from a variety of sources in preparation for upload to DataSetDB.

Usage

```
## S4 method for signature 'VirtualSpatialImage'
stageObject(x, dir, path, child = FALSE, ...)

## S4 method for signature 'StoredSpatialImage'
stageObject(x, dir, path, child = FALSE, ...)

## S4 method for signature 'RemoteSpatialImage'
stageObject(x, dir, path, child = FALSE, ...)
```

Arguments

`x`	A SpatialImage object.
`dir`	String containing the path to the staging directory.
`path`	String containing a prefix of the relative path inside `dir` where `x` is to be saved. The actual path used to save `x` may include additional components, see Details.
`child`	Logical scalar indicating whether `x` is a child of a larger object.
`...`	Further arguments to pass to specific methods.

Details

Each of the different methods will take advantage of any existing files to avoid an actual save. For example, the RemoteSpatialImage method will download the file directly to `path`, while the StoredSpatialImage method will create a link or copy the file. The SpatialImage method will fall back to saving the raster directly as a PNG.

Value

An image file is created at `file.path(dir, path)`, possibly after appending an appropriate file extension.

The return value should be a named list containing at least:

- `$schema`, a string specifying the schema to use to validate the metadata. This may have a `package` attribute to specify the package where the schema lives (in its `inst/schemas` directory).
- `path`, a string containing the path to the file containing the assay contents. This should start with the input `path` but can be followed by any necessary file extensions.
- `child`, whether this is a child resource of a larger object.

Other fields can be provided and will be included in the metadata, provided that they are recognized by the specified schema.

Author(s)

Aaron Lun

Examples

```
example(read10xVisium, echo=FALSE)
(img <- imgData(spe)$data[[1]])

# Doing a local run:
tmp <- tempfile()
dir.create(tmp)
stageObject(img, tmp, "whee")

# Forcing a re-save:
Y <- as(img, "LoadedSpatialImage")
stageObject(Y, tmp, "foo")
```

Index