Package 'epivizrData'

October 17, 2017

Type Package

Title Data Management API for epiviz interactive visualization app

Version 1.4.0

URL http://epiviz.github.io

Description Serve data from Bioconductor Objects through a WebSocket connection.

BugReports https://github.com/epiviz/epivizrData/issues

biocViews Infrastructure, Visualization

Depends R (>= 3.4), methods, epivizrServer (>= 1.1.1), Biobase

Imports S4Vectors, GenomicRanges, SummarizedExperiment (>= 0.2.0), OrganismDbi, GenomicFeatures, GenomeInfoDb, IRanges, ensembldb

Suggests testthat, roxygen2, bumphunter, hgu133plus2.db, Mus.musculus, TxDb.Mmusculus.UCSC.mm10.knownGene, rjson, knitr, rmarkdown, BiocStyle, EnsDb.Mmusculus.v79

License MIT + file LICENSE

LazyData true

RoxygenNote 6.0.1

Collate 'epivizrData-package.R' 'EpivizMeasurement-class.R' 'EpivizDataMgr-class.R' 'createMgr.R' 'EpivizData-class.R' 'EpivizTrackData-class.R' 'EpivizBlockData-class.R' 'EpivizBpData-class.R' 'EpivizGeneInfoData-class.R' 'EpivizFeatureData-class.R' 'make_gene_info_gr.R' 'register-methods.R'

VignetteBuilder knitr

NeedsCompilation no

Author Hector Corrada Bravo [aut, cre], Florin Chelaru [aut]

Maintainer Hector Corrada Bravo <hcorrada@gmail.com>

R topics documented:

`as.list,EpivizMeasurement-method` *Convert* `EpivizMeasurement` *object to* `list`

Description

Convert `EpivizMeasurement` object to `list`

Usage

```
## S4 method for signature 'EpivizMeasurement'
as.list(x)
```

Arguments

`x`	`EpivizMeasurement` object to coerce.

Value

a `list` describing measurement object

`createMgr` *Create a data manager for epiviz app*

Description

Create a data manager for epiviz app

Usage

```
createMgr(server = server)
```

Arguments

`server`	An object of class `EpivizServer`

Value

An object of class `EpivizDataMgr`

Examples

```
server <- epivizrServer::createServer(port=7123L)
data_mgr <- epivizrData::createMgr(server)
```

`EpivizBlockData-class` *Data container for interval data.*

Description

Used to serve data for visualizations of genomic regions only. Wraps `GenomicRanges` objects.

Methods

`get_default_chart_type()` Get name of default chart type for this data type

`get_measurements()` Get description of measurements served by this object

See Also

EpivizData

`EpivizBpData-class` *Container for basepair level numeric data*

Description

Used to serve data to genomic line tracks. Wraps `GenomicRanges` objects. Numeric values obtained from `mcols` slot.

Methods

`get_default_chart_type()` Get name of default chart type for this data type

`get_measurements()` Get description of measurements served by this object

See Also

EpivizData

EpivizData-class *Data container for epiviz data server*

Description

Data container for epiviz data server

Methods

`get_default_chart_type()` Get name of default chart type for this data type

`get_id()` Get id provided by manager `EpivizDataMgr-class`

`get_measurements()` Get description of measurements served by this object

`get_name()` Get datasource name, usually set by manager `EpivizDataMgr-class`

`get_rows(query, metadata, useOffset = FALSE)` Get genomic interval information overlapping query <GenomicRanges> region

`get_source_name()` Get original datasource name provided by manager `EpivizDataMgr-class`

`get_values(query, measurement, round = TRUE)` Get measurement values for features overlapping query region <GenomicRanges

`parse_measurement(ms_id = NULL)` Parse a measurement description for data served by this object

`set_id(id)` Set id, used by manager `EpivizDataMgr-class`

`set_limits(ylim)` Set plotting limits for continuous data

`set_mgr(mgr)` Set data manager, `EpivizDataMgr-class`

`set_name(name)` Set datasource name, usually set by manager `EpivizDataMgr-class`

`set_source_name(source_name)` Set original datasource name, used by manager `EpivizDataMgr-class`

`update(new_object, send_request = TRUE)` Update underlying data object with new object

EpivizDataMgr-class *Class providing data manager for epiviz app*

Description

Class providing data manager for epiviz app

Methods

`add_measurements(obj, datasource_name = NULL, datasource_origin_name = deparse(substitute(obj)),` register measurements in data manager

`is_ms_connected(ms_obj_or_id)` check if measurement object was properly added to JS app

`is_server_closed()` Check if underlying server is closed, <logical>

`list_measurements()` make a printable list of registered measurements

`rm_all_measurements()` remove all registered measurements

`rm_measurements(ms_obj_or_id)` remove registered measurments from a given data object

`update_measurements(ms_obj_or_id, new_object, send_request = TRUE)` update the underlying data object for a registered measurement (given by object or id)

EpivizFeatureData-class

Data container for RangedSummarizedExperiment objects

Description

Used to serve general data (used in e.g., scatter plots and heatmaps). Wraps `RangedSummarizedExperiment` objects. Numeric values obtained from `assays` slot

Methods

`get_default_chart_type()` Get name of default chart type for this data type

`get_measurements()` Get description of measurements served by this object

See Also

EpivizData

EpivizGeneInfoData-class

Container for gene annotation data

Description

Used to serve data to gene annotation tracks. Wraps `GenomicRanges` objects. Annotation obtained from columns `Gene` (gene symbols) and `Exons` (exon start and end locations).

Methods

`get_default_chart_type()` Get name of default chart type for this data type

`get_measurements()` Get description of measurements served by this object

`get_rows(query, metadata, useOffset = FALSE)` Get genomic interval information overlapping query <`GenomicRanges`> region

See Also

EpivizData

register,OrganismDb

EpivizMeasurement-class

Class encapsulating a measurement description for epiviz app.

Description

Class encapsulating a measurement description for epiviz app.

`epivizrData` *epivizrData*

Description

epivizrData

`register` *Generic method to register data to the data server*

Description

Generic method to register data to the data server

Usage

```
register(object, columns = NULL, ...)

## S4 method for signature 'GenomicRanges'
register(object, columns, type = c("block", "bp",
  "gene_info"), ...)

## S4 method for signature 'RangedSummarizedExperiment'
register(object, columns = NULL,
  assay = 1, metadata = NULL)

## S4 method for signature 'ExpressionSet'
register(object, columns, annotation = NULL,
  assay = "exprs")

## S4 method for signature 'OrganismDb'
register(object, kind = c("gene", "tx"),
  keepSeqlevels = NULL, ...)

## S4 method for signature 'TxDb'
register(object, kind = c("gene", "tx"),
  keepSeqlevels = NULL, ...)

## S4 method for signature 'EnsDb'
register(object, kind = c("gene", "tx"),
  keepSeqlevels = NULL, ...)
```

Arguments

`object`	The object to register to data server
`columns`	Name of columns containing data to register
`...`	Additonal arguments passed to object constructors

type	Which type of data object to register for a GenomicRanges object. block: only region data, bp base-pair resolution quantitative data (see columns argument), geneInfo information about gene location.
assay	Which assay in object to register
metadata	Additional metadata about features
annotation	Character string indicating platform annotation (only hgu133plus2 supported for now)
kind	Make gene or transcript annotation (only gene supported for now)
keepSeqlevels	character vector indicating seqlevels in object to keep

Value

Object inheriting from EpivizData class

Methods (by class)

- GenomicRanges: Register a GenomicRanges object
- RangedSummarizedExperiment: Register a RangedSummarizedExperiment object
- ExpressionSet: Register an ExpressionSet object
- OrganismDb: Register an OrganismDb object
- TxDb: Register a TxDb object
- EnsDb: Register an EnsDb object

Examples

```
library(GenomicRanges)
# create an example GRanges object
gr <- GRanges("chr10", IRanges(start=1:1000, width=100), score=rnorm(1000))
# this returns an EpivizData object without adding to data manager
# this is not the preferred way of creating these object, but is shown
# here for completeness.
ms_obj <- epivizrData:::register(gr, type="bp", columns="score")

server <- epivizrServer::createServer(port=7123L)
data_mgr <- epivizrData::createMgr(server)

# This adds a data object to the data manager
data_mgr$add_measurements(gr, "example_gr", type="bp", columns="score")
```

---

show,EpivizMeasurement-method

Display measurement datasourceId and id

Description

Display measurement datasourceId and id

Usage

```
## S4 method for signature 'EpivizMeasurement'
show(object)
```

Arguments

object	a EpivizMeasurement to display

Value

A string describing measurement

Index