Package ‘influxdbclient’

October 13, 2022

Title  'InfluxDB' 2.x Client
Version  0.1.2
Author  Ales Pour <ales.pour@bonitoo.io>
Maintainer  Ales Pour <ales.pour@bonitoo.io>

URL  https://github.com/influxdata/influxdb-client-r

BugReports  https://github.com/influxdata/influxdb-client-r/issues

Depends  R (>= 3.4)
Encoding  UTF-8
License  MIT License + file LICENSE

Imports  R6, httr, jsonlite, bit64, nanotime (>= 0.3.0), plyr

RoxygenNote  7.1.1
Suggests  testthat, httptest

NeedsCompilation  no

Repository  CRAN

Date/Publication  2022-09-01 09:10:42 UTC

R topics documented:

| InfluxDBClient | 2 |
| RetryOptions   | 4 |

Index

1
Description

Client for querying from and writing to InfluxDB 2.x.

Format

An R6Class object

Public fields

- url Database URL
- token Authentication token
- org Organization name
- dialect Flux dialect
- retryOptions Retry options

Methods

Public methods:

- `InfluxDBClient$new()`
- `InfluxDBClient$health()`
- `InfluxDBClient$query()`
- `InfluxDBClient$ready()`
- `InfluxDBClient$write()`
- `InfluxDBClient$clone()`

Method `new()`: Creates instance of `InfluxDBClient`.

Usage:

```
InfluxDBClient$new(url, token, org, retryOptions = NULL)
```

Arguments:

- `url` InfluxDB instance URL
- `token` Authentication token
- `org` Organization name
- `retryOptions` Retry options. See `RetryOptions` for details. Set to `TRUE` for default retry options. Default is `NULL` which disables retries.

Method `health()`: Gets health info of the InfluxDB instance.

Usage:

```
InfluxDBClient$health()
```

Returns: Named list with name, message, status, version, commit elements or error
**Method** `query()`: Queries data in the InfluxDB instance.

Usage:
```
InfluxDBClient$query(
  text,
  POSIXctCol = c('_time' = "time"),
  flatSingleResult = TRUE
)
```

Arguments:
- **text**  Flux query
- **POSIXctCol** Flux time to (new) POSIXct column mapping (named list). Default is c("_time"="time"). Use NULL to skip it.
- **flatSingleResult** Whether to return simple list when response contains only one result. Default is TRUE.

Returns: List of data frames. Data frame represents Flux table. It can be a named list of nested lists of data frames when query response contains multiple results (see Flux yield), or a simple list of data frames for single result response.

**Method** `ready()`: Gets readiness status of the InfluxDB instance.

Usage:
```
InfluxDBClient$ready()
```

Returns: Named list with `status`, `started` and `up` elements or error

**Method** `write()`: Writes data to the InfluxDB instance.

Usage:
```
InfluxDBClient$write(
  x,
  bucket,
  batchSize = 5000,
  precision = c("ns", "us", "ms", "s"),
  measurementCol = "_measurement",
  tagCols = NULL,
  fieldCols = c('_field' = '_value'),
  timeCol = '_time',
  object = NULL,
  ...
)
```

Arguments:
- **x** Data as (list of) data.frame
- **bucket** Target bucket name
- **batchSize** Batch size. Positive number or FALSE to disable. Default is 5000.
- **precision** Time precision
- **measurementCol** Name of measurement column. Default is "_measurement".
- **tagCols** Names of tag (index) columns
fieldCols  Names of field columns. In case of unpivoted data previously retrieved from InfluxDB, use default value \texttt{ie. named list c("_field"="_value")}. For all other cases, just use simple vector of column names (see Examples).

timeCol  Name of time column. The column values should be either of \texttt{nanotime} or \texttt{POSIXct} type. Default is \texttt{"_time"}.

object  Output object name. For dry-run operation, specify the name of the object to receive the output. Default is \texttt{NULL}. For debugging purposes.

Method \texttt{clone()}: The objects of this class are cloneable with this method.

\textit{Usage:}  
\texttt{InfluxDBClient$clone(deep = FALSE)}

\textit{Arguments:}  
deep  Whether to make a deep clone.

Examples

```r
## Not run:
# Instantiation
client <- InfluxDBClient$new(url = "http://localhost:8086",
  token = "my-token",
  org = "my-org")

# Query
data <- client$query('from(bucket: "my-bucket") |> range(start: -1h)')

# Write
data <- data.frame(...)  
client$write(data, bucket = "my-bucket", precision = "us",  
  measurementCol = "name",  
  tagCols = c("location", "id"),  
  fieldCols = c("altitude", "temperature"),  
  timeCol = "time")

# Ready status
ready <- client$ready()

# Health info
ready <- client$health()

## End(Not run)
```

Description

Retry options may be specified as optional argument to \texttt{write}. 

---

RetryOptions

Description

Retry options may be specified as optional argument to \texttt{write}. 

---
**Format**

An R6Class object

**Public fields**

- retryJitter: Maximum number of random milliseconds included in delay. Default is 0.
- retryInterval: First retry delay in seconds. Default is 5.
- maxDelay: Maximum delay between retries in seconds. Default is 125.
- maxRetryTime: Maximum time to spend retrying in seconds. Default is 180.
- maxAttempts: Number of retry attempts. Default is 5.
- exponentialBase: Base for exponential backoff strategy. Default is 2.

**Methods**

**Public methods:**

- `RetryOptions$new()`
- `RetryOptions$clone()`

**Method new():** Creates instance of RetryOptions.

*Usage:*

```r
RetryOptions$new(
    retryJitter = 0,
    retryInterval = 5,
    maxDelay = 125,
    maxRetryTime = 180,
    maxAttempts = 5,
    exponentialBase = 2,
    ...
)
```

**Method clone():** The objects of this class are cloneable with this method.

*Usage:*

```r
RetryOptions$clone(deep = FALSE)
```

*Arguments:*

- `deep`: Whether to make a deep clone.
Index

InfluxDBClient, 2
RetryOptions, 4