Package 'influxdbr'

May 8, 2026

Type Package

Title R Interface to InfluxDB

Version 0.14.2

Date 2018-01-10

Description An R interface to the InfluxDB time series database <https://www.influxdata.com>. This package allows you to fetch and write time series data from/to an InfluxDB server. Additionally, handy wrappers for the Influx Query Language (IQL) to manage and explore a remote database are provided.

License GPL-3

ByteCompile TRUE

URL https://github.com/dleutnant/influxdbr

BugReports http://github.com/dleutnant/influxdbr/issues

Imports dplyr (>= 0.7.0), jsonlite, httr, magrittr, purrr (>= 0.2.3), rlang, tibble, tidyr, xts, zoo

RoxygenNote 6.0.1

Suggests testthat

NeedsCompilation no

Author Dominik Leutnant [aut, cre] (ORCID: <https://orcid.org/0000-0003-3293-2315>)

Maintainer Dominik Leutnant <leutnant@fh-muenster.de>

Repository CRAN

Date/Publication 2018-01-10 10:57:53 UTC

Contents

`create_database` *Influx database management*

Description

The folllowing functions are convenient wrappers around `influx_post`.

- `create_database()`: creates a new database
- `drop_database()`: drops an existing database
- `drop_series()`: drops specific series
- `delete()`: deletes all points from a series in a database (supports time intervals)
- `drop_measurement()`: drops an entire measurement
- `create_retention_policy()`: create a new retention policy
- `alter_retention_policy()`: alter a retention policy
- `drop_retention_policy()`: drop a retention policy

Usage

```
create_database(con, db)

drop_database(con, db)

drop_series(con, db, measurement = NULL, where = NULL)

delete(con, db, measurement = NULL, where = NULL)

drop_measurement(con, db, measurement)

create_retention_policy(con, db, rp_name, duration, replication,
  default = FALSE)

alter_retention_policy(con, db, rp_name, duration, replication,
  default = FALSE)

drop_retention_policy(con, db, rp_name)
```

Arguments

con	An influx_connection object (s. influx_connection).
db	Sets the target database for the query.
measurement	Sets a specific measurement.
where	Apply filter on tag key values.
rp_name	The name of the retention policy.
duration	Determines how long InfluxDB keeps the data.
replication	The number of data nodes.
default	logical. If TRUE, the new retention policy is the default retention policy for the database.

Value

A tibble containing post results in case of an error (or message). Otherwise NULL (invisibly).

References

https://docs.influxdata.com/influxdb/

See Also

influx_connection

---

create_user *Influx authentication and authorization*

Description

The folllowing functions are convenient wrappers around influx_post and influx_query (show_users and show grants).

- create_user(): creates a new user
- drop_user(): drops an existing user
- grant_privileges(): grant privileges to an existing users
- revoke_privileges(): revoke privileges to an existing users
- show_users(): show all users
- show_grants(): show grants of an user

Usage

```
create_user(con, username, password)

drop_user(con, username)

grant_privileges(con, privilege = c("READ", "WRITE", "ALL"), db, username)

revoke_privileges(con, privilege = c("READ", "WRITE", "ALL"), db, username)

show_users(con)

show_grants(con, username)
```

Arguments

con	An influx_connection object (s. influx_connection).
username	The username to be used.
password	The password to be used.
privilege	Specifies the user privilege.
db	Sets the target database.

Value

A tibble containing post results in case of an error (or message). Otherwise NULL (invisibly). show_users() and show_grants() return a tibble.

References

https://docs.influxdata.com/influxdb/

See Also

influx_connection

---

influxdbr influxdbr *package*

Description

R Interface for InfluxDB

`influx_connection` *Create an influxdb_connection object*

Description

Create an influxdb_connection object by specifying server connection details. Credentials can also be saved and accessed through a config file.

Usage

```
influx_connection(scheme = c("http", "https"), host = "localhost",
  port = 8086, user = "user", pass = "pass", path = "/", group = NULL,
  verbose = FALSE, config_file = "~/.influxdb.cnf", curl_options = NULL)
```

Arguments

`scheme`	The scheme to use, either http or https. Defaults to http.
`host`	Hostname of the InfluxDB server. Defaults to localhost.
`port`	numerical. Port number of the InfluxDB server. Defaults to 8086.
`user`	username The username to use. Defaults to "user".
`pass`	password The password to use. Defaults to "pass".
`path`	The prefix path on which the InfluxDB is running. Can be useful in proxy situations.
`group`	The group to use within the config file.
`verbose`	logical. Provide additional details?
`config_file`	The configuration file to be used if `group` is specified.
`curl_options`	Additional curl arguments created with `config` (e.g. httr::config(verbose = TRUE, timeout = 5, ssl_verifypeer = FALSE)).

structure of configuration file

A configuration file may contain several connection settings. Each setting has the following structure:

```
[group]
scheme=http
host=localhost
port=8086
user=username
pass=password
path=/
```

References

https://influxdb.com/

influx_ping *Ping an influxdb server*

Description

This function pings an influxdb server (e.g. for connection testing)

Usage

```
influx_ping(con)
```

Arguments

con	An influx_connection object (s. influx_connection).

Value

A tibble with server information.

References

https://docs.influxdata.com/influxdb/

See Also

influx_connection

influx_query *Query an InfluxDB server*

Description

This functions queries an InfluxDB server.

Usage

```
influx_query(con, db = NULL, query = "SELECT * FROM measurement",
  timestamp_format = c("n", "u", "ms", "s", "m", "h"), return_xts = TRUE,
  chunked = FALSE, simplifyList = FALSE)
```

Arguments

con	An influx_connection object (s. influx_connection).
db	Sets the target database for the query.
query	The InfluxDB query to be sent.
timestamp_format	Sets the timestamp format ("n", "u", "ms", "s", "m", "h").
return_xts	logical. Sets the return type. If set to TRUE, xts objects are returned, FALSE gives tibbles.
chunked	Either FALSE or an integer. If FALSE, series are not requested in streamed batches. If an integer is provided, responses will be chunked by series or by every chunked points. Chunks are merged internally.
simplifyList	logical. If only one series is returned, the result can be flatten to directly get either a tibble or an xts object (instead of a list) (default is FALSE).

Value

A list of tibble or xts objects.

References

https://docs.influxdata.com/influxdb/

See Also

xts, influx_connection

---

influx_select *influx select helper*

Description

This function is a convenient wrapper for selecting data from a measurement by calling influx_query with the corresponding query.

Usage

```
influx_select(con, db, field_keys, rp = NULL, measurement, where = NULL,
  group_by = NULL, limit = NULL, slimit = FALSE, offset = NULL,
  order_desc = FALSE, return_xts = TRUE, simplifyList = FALSE)
```

Arguments

con	An influx_connection object (s. influx_connection).
db	Sets the target database for the query.
field_keys	Specifies the fields to be selected.
rp	The name of the retention policy.
measurement	Sets the name of the measurement.
where	Apply filter on tag key values.
group_by	The group_by clause in InfluxDB is used not only for grouping by given values, but also for grouping by given time buckets.
limit	Limits the number of the n oldest points to be returned.
slimit	logical. Sets limiting procedure (slimit vs. limit).
offset	Offsets the returned points by the value provided.
order_desc	logical. Change sort order to descending.
return_xts	logical. Sets the return type. If set to TRUE, a list of xts objects is returned, FALSE gives list of tibbles.
simplifyList	logical. If only one series is returned, the result can be flatten to directly get either a tibble or an xts object (instead of a list) (default is FALSE).

Value

A list of xts or tibbles.

References

https://docs.influxdata.com/influxdb/

---

influx_write *Write an xts or data.frame object to an InfluxDB server*

Description

This function writes either an xts object or a data.frame to an InfluxDB server. In case of an xts object, columnnames of the xts object are used as InfluxDB's field keys, xts's coredata represent field values. Attributes are preserved and written as tag keys and values, respectively.

In case of a data.frame, columns may represent times and both tag and field values. Columnnames of the data.frame object are used as InfluxDB's tag and field keys. Times and tags are optional. Use parameter time_col and tag_col to define the interpretation. By specifiying one of the arguments measurement or measurement_col, a data.frame may contain data from one measurement or multiple measurements, respectively.

Usage

```
influx_write(x, con, db, measurement, rp = NULL, precision = c("s", "ns",
  "u", "ms", "m", "h"), consistency = c(NULL, "one", "quroum", "all", "any"),
  max_points = 5000, use_integers = FALSE, ...)

## S3 method for class 'xts'
influx_write(x, con, db, measurement, rp = NULL,
  precision = c("s", "ns", "u", "ms", "m", "h"), consistency = c(NULL,
  "one", "quroum", "all", "any"), max_points = 5000, use_integers = FALSE,
  ...)

## S3 method for class 'data.frame'
influx_write(x, con, db, measurement = NULL, rp = NULL,
  precision = c("s", "ns", "u", "ms", "m", "h"), consistency = c(NULL,
  "one", "quroum", "all", "any"), max_points = 5000, use_integers = FALSE,
  time_col = NULL, tag_cols = NULL, measurement_col = NULL, ...)
```

Arguments

x	The object to write to an InfluxDB server (either of class `xts` or `data.frame`).
con	An `influx_connection` object (s. `influx_connection`).
db	Sets the target database for the query.
measurement	Sets the name of the measurement (data.frame has data to write to one measurement only). If both arguments `measurement` and `measurement_col` are given, `measurement` gets overridden.
rp	Sets the target retention policy for the write. If not present the default retention policy is used.
precision	Sets the precision of the supplied Unix time values ("s", "ns", "u", "ms", "m", "h"). If not present timestamps are assumed to be in seconds.
consistency	Set the number of nodes that must confirm the write. If the requirement is not met the return value will be partial write if some points in the batch fail, or write failure if all points in the batch fail.
max_points	Defines the maximum points per batch (defaults to 5000).
use_integers	Should integers (instead of doubles) be written if present?
...	Arguments to be passed to methods.
time_col	A character scalar naming the time index column.
tag_cols	A character vector naming tag columns.
measurement_col	A character scalar naming the measurement column (data.frame has data to write to multiple measurements). Overrides `measurement` argument.

Value

A list of server responses.

References

https://docs.influxdata.com/influxdb/

See Also

xts, influx_connection

---

show_databases *Influx schema exploration*

Description

The folllowing functions are convenient wrappers around `influx_query`.

- `show_databases()`: returns database names
- `show_measurements()`: returns measurement names
- `show_series()`: returns unambiguous series
- `show_tag_keys()`: returns tag keys
- `show_tag_values()`: returns tag values
- `show_field_keys()`: returns field keys
- `show_retentions_policies()`: returns retention policies

Usage

```
show_databases(con)

show_measurements(con, db, where = NULL)

show_series(con, db, measurement = NULL, where = NULL)

show_tag_keys(con, db, measurement = NULL)

show_tag_values(con, db, measurement = NULL, key)

show_field_keys(con, db, measurement = NULL)

show_retention_policies(con, db)
```

Arguments

con	An `influx_connection` object (s. `influx_connection`).
db	Sets the target database for the query.
where	Apply filter on tag key values.
measurement	Query a specific measurement.
key	The key to be queried.

Value

A tibble containing query results.

References

https://docs.influxdata.com/influxdb/

See Also

influx_connection

---

show_diagnostics *Show diagnostics*

Description

This function calls `influx_query` to receive some diagnostics.

Usage

```
show_diagnostics(con)
```

Arguments

con	An `influx_connection` object (s. `influx_connection`).

Value

A tibble with diagnostics.

See Also

influx_connection

show_stats *Show stats*

Description

This function calls `influx_query` to receive some stats.

Usage

```
show_stats(con)
```

Arguments

con	An `influx_connection` object (s. `influx_connection`).

Value

A tibble.

Warning

InfluxDB response might take some time.

See Also

`influx_connection`

Index