Package ‘RGreenplum’

October 12, 2022

Title  Interface to 'Greenplum' Database
Version  0.1.2
Description  Fully 'DBI'-compliant interface to 'Greenplum' <https://greenplum.org/>, an open-source parallel database. This is an extension of the 'RPostgres' package <https://github.com/r-dbi/RPostgres>.
License  GPL-2
URL  https://github.com/mwillumz/RGreenplum
BugReports  https://github.com/mwillumz/RGreenplum/issues
Encoding  UTF-8
LazyData  true
Depends  R (>= 3.1.0)
Imports  DBI, RPostgres, methods
Suggests  DBItest, testthat
RoxygenNote  6.0.1
Collate  'GreenplumDriver.R' 'GreenplumConnection.R' 'table-create.R'
'tables.R'
NeedsCompilation  no
Author  Michael Williams [aut, cre]
Maintainer  Michael Williams <mw8765999@gmail.com>
Repository  CRAN
Date/Publication  2018-06-24 11:44:18 UTC

R topics documented:

  dbConnect,GreenplumDriver-method  ........................................  2
dbIsValid,GreenplumDriver-method  ........................................  3
dbWriteTable,GreenplumConnection,character,data.frame-method  ..........  3
Greenplum  .................................................................  4
sqlCreateTable,GreenplumConnection-method  ............................  5

Index  6
Connect to a Greenplum database.

Description

Manually disconnecting a connection is not necessary with RGreenplum, but still recommended; if you delete the object containing the connection, it will be automatically disconnected during the next GC with a warning.

Usage

```r
## S4 method for signature 'GreenplumDriver'
dbConnect(drv, dbname = NULL, host = NULL, port = NULL, password = NULL, user = NULL, service = NULL, ..., bigint = c("integer64", "integer", "numeric", "character"))
```

Arguments

- `drv` 'RGreenplum::Greenplum()

- `dbname` Database name. If 'NULL', defaults to the user name. Note that this argument can only contain the database name, it will not be parsed as a connection string (internally, `expand_dbname` is set to `false` in the call to `['PQconnectdbParams()']`(https://www.postgresql.org/docs/9.6/static/libpq-connect.html)).

- `host, port` Host and port. If 'NULL', will be retrieved from `PGHOST` and `PGPORT` env vars.

- `user, password` User name and password. If 'NULL', will be retrieved from `PGUSER` and `PGPASSWORD` envvars, or from the appropriate line in `~/.pgpass`. See <http://www.postgresql.org/docs/9.6/static/libpq-pgpass.html> for more details.

- `service` Name of service to connect as. If 'NULL', will be ignored. Otherwise, connection parameters will be loaded from the `pg_service.conf` file and used. See <http://www.postgresql.org/docs/9.6/static/libpq-pgservice.html> for details on this file and syntax.

- `...` Other name-value pairs that describe additional connection options as described at <http://www.postgresql.org/docs/9.6/static/libpq-connect.html#LIBPQ-PARAMKEYWORDS>

- `bigint` The R type that 64-bit integer types should be mapped to, default is [bit64::integer64], which allows the full range of 64 bit integers.
**Description**

Greenplum dbIsValid method

**Usage**

```r
## S4 method for signature 'GreenplumDriver'
dbIsValid(dbObj, ...)
```

**Arguments**

- `dbObj` database object
- `...` other arguments for method

---

**Description**

Greenplum dbWriteTable method

**Usage**

```r
## S4 method for signature 'GreenplumConnection,character,data.frame'
dbWriteTable(conn, name, value, ..., row.names = FALSE, overwrite = FALSE, append = FALSE, field.types = NULL, temporary = FALSE, distributed_by = NULL, copy = TRUE)
```

```r
## S4 method for signature 'GreenplumConnection,character'
dbExistsTable(conn, name, ...)
```
Arguments

- **conn**: a [GreenplumConnection-class] object
- **name**: a character string specifying a table name. Names will be automatically quoted so you can use any sequence of characters, not just any valid bare table name.
- **value**: A data.frame to write to the database.
- **...**: Other arguments used by individual methods.
- **row.names**: Either TRUE, FALSE, NA or a string
- **overwrite**: a logical specifying whether to overwrite an existing table or not. Its default is FALSE.
- **append**: a logical specifying whether to append to an existing table in the DBMS. Its default is FALSE.
- **field.types**: character vector of named SQL field types where the names are the names of new table's columns.
- **temporary**: If TRUE, will generate a temporary table statement.
- **distributed_by**: Distribution columns for new table. NULL for random distribution.
- **copy**: If TRUE, data will be copied to remote database

---

**Greenplum**  
*Greenplum driver*

Description

This driver never needs to be unloaded and hence `dbUnload()` is a null-op.

Usage

Greenplum()

Examples

library(RPostgres)  
RGreenplum::Greenplum()
**sqlCreateTable, GreenplumConnection-method**

*Greenplum sqlCreateTable method*

**Description**

Greenplum sqlCreateTable method

**Usage**

```r
## S4 method for signature 'GreenplumConnection'
sqlCreateTable(con, table, fields,
    row.names = NA, temporary = FALSE, distributed_by = NULL, ...)
```

**Arguments**

- **con**: A database connection.
- **table**: Name of the table. Escaped with `dbQuoteIdentifier()`.
- **fields**: Either a character vector or a data frame.
  - A named character vector: Names are column names, values are types. Names are escaped with `dbQuoteIdentifier()`. Field types are unescaped.
  - A data frame: field types are generated using `dbDataType()`.
- **row.names**: Either `TRUE`, `FALSE`, `NA` or a string.
  - If `TRUE`, always translate row names to a column called "row_names". If `FALSE`, never translate row names. If `NA`, translate rownames only if they’re a character vector.
  - A string is equivalent to `TRUE`, but allows you to override the default name.
  - For backward compatibility, `NULL` is equivalent to `FALSE`.
- **temporary**: If `TRUE`, will generate a temporary table statement.
- **distributed_by**: Distribution columns for new table. `NULL` for random distribution.
- **...**: Other arguments used by individual methods.
Index

dbConnect, GreenplumDriver-method, 2

dbExistsTable, GreenplumConnection, character-method
    (dbWriteTable, GreenplumConnection, character, data.frame-method), 3

dbIsValid, GreenplumDriver-method, 3

dbWriteTable, GreenplumConnection, character, data.frame-method, 3

Greenplum, 4

sqlCreateTable, GreenplumConnection-method, 5