RMySQL-package {RMySQL} R Documentation

R interface to the MySQL database

Description

The functions in this package allow you interact with one or more MySQL databases from R.

Overview

A typical usage of the R-MySQL interface is:

1. Connect and authenticate to one or more MySQL databases:

```
con <- dbConnect(MySQL(), group = "lasers")
con2 <- dbConnect(MySQL(), user="opto", password="pure-light",
                  dbname="lasers", host="merced")
```

2. List tables and fields in a table:

```
dbListTables(con)
dbListFields(con, "table\_name")
```

3. Import and export data.frames:

```
d <- dbReadTable(con, "WL")
dbWriteTable(con, "WL2", a.data.frame)         ## table from a data.frame
dbWriteTable(con, "test2", "~/data/test2.csv") ## table from a file
```

4. Run an arbitrary SQL statement and extract all its output (returns a data.frame):

```
dbGetQuery(con, "select count(*) from a\_table")
dbGetQuery(con, "select * from a\_table")
```

5. Run an SQL statement and extract its output in pieces (returns a result set):

```
rs <- dbSendQuery(con, "select * from WL where width\_nm between 0.5 and 1")
d1 <- fetch(rs, n = 10000)
d2 <- fetch(rs, n = -1
```

6. Run multiple SQL statements and process the various result sets (note the `client.flag` value in the `dbConnect` call):

```
con <- dbConnection(MySQL(), dbname = "rs-dbi",
          client.flag = CLIENT\_MULTI\_STATEMENTS)
script  <- paste("select * from WL where width\_nm between 0.5 and 1"
                 "select * from lasers\_id  where id LIKE 'AL100
                 sep = ";")
rs1 <- dbSendQuery(con, script)
d1 <- fetch(rs1, n = -1)
if(dbMoreResults(con)){
   rs2 <- dbNextResult(con)
   d2 <- fetch(rs2, n=-1)
}
```

7. Get meta-information on a connection (thread-id, etc.):

```
summary(MySQL(), verbose = TRUE)
summary(con, verbose = TRUE)
summary(rs, verbose = TRUE)
```

```
dbListConnections(MySQL())
dbListResultSets(con)
dbHasCompleted(rs)
```

8. Close connections:

```
dbDisconnect(con)
dbDisconnect(con2)
```

Data mappings between MySQL and R

MySQL tables are read into R as data.frames, but without coercing character or logical data into factors. Similarly while exporting data.frames, factors are exported as character vectors.

Integer columns are usually imported as R integer vectors, except for cases such as `BIGINT` or `UNSIGNED INTEGER` which are coerced to R's `double` precision vectors to avoid truncation (currently R's integers are signed 32-bit quantities).

Time variables are imported/exported as character data, so you need to convert these to your favorite date/time representation.

Currently there are no facilities to import/export `BLOBs`.

RDBMS tables, data.frames, and data types

Tables in a relational database are only superficially similar to R's data.frames (e.g., tables as unordered sets of rows compared to data.frames as ordered sets, tables having referential constraints, indexes, and so on.)

User authentication

Although you can specify user authentication parameters (user, password, database, and host) in the call to `dbConnect`, the preferred method to pass these parameters to the server is through a MySQL `default.file`, e.g., '`$HOME/.my.cnf`' (or '`c:/my.cnf`' under Windows). The MySQL `dbConnect` method parses the `default.file=\$HOME/.my.cnf` to initialize connections to MySQL databases. This file consists of zero or more named sections each starting with a line of the form `[section-name]`; each section includes zero or more MySQL variable declaration per line, such as, `user=`, `password=`, `host=`, etc. For instance,

```
$ cat $HOME/.my.cnf
# this is a comment
; this is also a comment
[client]
user = dj
host = localhost

[rs-dbi]
database = s-data

[lasers]
user = opto
database = opto
password = pure-light
host = merced
...
[iptraffic]
host = data
database = iptraffic
```

This file should be readable only by you. `RMySQL` always initializes connection values from the `[client]` and `[rs-dbi]` sections, but you may define you own project-specific sections (as in the example above) to tailor its environment; if the same parameter appears in multiple sections (e.g., in `client` and `rs-dbi`), the last (closer to the bottom) occurrence is used.

If you define a section, for instance, `[iptraffic]`, then instead of including all these parameters in the call to `dbConnect`, you simply supply the name of the `group`, e.g., `dbConnect(MySQL(), group = "iptraffic")`.

In addition to `user`, `password`, `host`, and `dbname`, you may specify any other connection parameters, e.g., `port`, `socket`. See the MySQL documentation for details.

Lastly, you may specify an alternate `default.file`, e.g., `dbConnect(MySQL(), group="iptraffic", default.file="router\_shield")`.

References

See stat.bell-labs.com/RS-DBI for more details on the R/S-Plus database interface.

See the documentation at the MySQL Web site http://www.mysql.com for details.

Author(s)

David A. James <dj@bell-labs.com> Saikat DebRoy <saikat@stat.wisc.edu>

See Also

On database managers:

`DBI` `dbDriver` `dbUnloadDriver`

On connections, SQL statements and resultSets:

`dbConnect` `dbDisconnect` `dbSendQuery` `dbGetQuery` `fetch` `dbClearResult`

On transaction management:

`dbCommit` `dbRollback`

On meta-data:

`summary` `dbGetInfo` `dbGetDBIVersion` `dbListTables` `dbListConnections` `dbListResults` `dbColumnInfo` `dbGetException` `dbGetStatement` `dbHasCompleted` `dbGetRowCount` `dbGetAffectedRows`

Examples

```
## Not run:
# create a MySQL instance and create one connection.
> m <- dbDriver("MySQL")  ## or MySQL()
<MySQLDriver:(4378)>

# open the connection using user, passsword, etc., as
# specified in the "[iptraffic]" section of the
# configuration file \file{\$HOME/.my.cnf}
```

```
> con <- dbConnect(m, group = "iptraffic")
> rs <- dbSendQuery(con, "select * from HTTP_ACCESS where IP_ADDRESS = '127.0.0.1'")
> df <- fetch(rs, n = 50)
> dbHasCompleted(rs)
[1] FALSE
> df2 <- fetch(rs, n = -1)
> dbHasCompleted(rs)
[1] TRUE
> dbClearResult(rs)
> dim(dbGetQuery(con, "show tables"))
[1] 74   1
> dbListTables(con)
## End(Not run)
```

[Package *RMySQL* version 0.6-0 Index]