

Package ‘RcppRedis’

July 2, 2014

Type Package

Title Rcpp bindings for Redis using the hiredis library

Version 0.1.1

Date 2014-06-09

Author Dirk Eddelbuettel <edd@debian.org>

Maintainer Dirk Eddelbuettel <edd@debian.org>

Description RcppRedis connects R to the Redis key/value store using the C client library hiredis

SystemRequirements The hiredis library (eg via package libhiredis-dev on Debian/Ubuntu, hiredis-devel on Fedora/RedHat, or directly from source from <https://github.com/redis/hiredis>)

License GPL (>= 2)

Imports methods, Rcpp (>= 0.11.0), RApiSerialize

LinkingTo Rcpp, RApiSerialize

Suggests rredis, RUnit

NeedsCompilation yes

Repository CRAN

Date/Publication 2014-06-10 01:08:09

R topics documented:

RcppRedis 2

Index 3

RcppRedis

Rcpp module using hiredis library to connect R to Redis

Description

The Redis module is created using Rcpp modules and wraps a minimal class Redis around an Redis connection context object which permits bi-directional communication with a Redis in-memory database.

New instances can be created using either a default constructor (using localhost and the default port) and either host and port, or just the host.

Currently, the module has just one worker command which sends a string to the Redis instance and returns a string.

The helper functions `serializeToChar()` and `unserializeFromChar` convert R objects to/from a character representation (and internalize the conversion from raw to char representation at the compiled level). The helper functions `serializeToRaw()` and `unserializeFromRaw` convert R objects to/from raw vectors.

Details

Please consult the Redis documentation for details on the available commands. See the Rcpp-modules vignette for details on Rcpp modules.

Author(s)

Dirk Eddelbuettel <edd@debian.org>

References

Redis: <http://redis.io/commands>

Index

*Topic **package**

RcppRedis, [2](#)

Rcpp_Redis (RcppRedis), [2](#)

Rcpp_Redis-class (RcppRedis), [2](#)

RcppRedis, [2](#)

Redis (RcppRedis), [2](#)

serializeToChar (RcppRedis), [2](#)

serializeToRaw (RcppRedis), [2](#)

unserializeFromChar (RcppRedis), [2](#)

unserializeFromRaw (RcppRedis), [2](#)