

# Package: odbc32 (via r-universe)

September 2, 2024

**Type** Package

**Title** 32-bit ODBC drivers for 64-bit R sessions

**Version** 0.2.9.9002

**Author** Vaclav Hausenblas

**Maintainer** Vaclav Hausenblas <vaclav.hausenblas@gmail.com>

**Description** Tools for using 32-bit ODBC drivers from 64-bit R sessions. Based on socket connection between 64-bit R and host 32-bit R.

**License** MIT

**Encoding** UTF-8

**LazyData** true

**RoxygenNote** 7.0.2

**Imports** r2r, RODBC

**Suggests** testthat

**Remotes** github::vh-d/r2r

**Repository** <https://hsonne.r-universe.dev>

**RemoteUrl** <https://github.com/hsonne/odbc32>

**RemoteRef** HEAD

**RemoteSha** 1ea1dfe70df6a31a162616ae54ef915e33e60894

## Contents

odbc32-package . . . . .	2
.append_con . . . . .	2
close.odbc32 . . . . .	3
connect_remotely . . . . .	4
create_odbc32_object . . . . .	4
data2access . . . . .	5
default_socket . . . . .	5
getSqlTypeInfo . . . . .	6

odbcClose . . . . .	6
odbcConnect . . . . .	7
odbcConnectAccess . . . . .	7
odbcConnectAccess2007 . . . . .	8
odbcDataSources . . . . .	8
odbcDriverConnect . . . . .	9
odbcEndTran . . . . .	9
odbcGetErrMsg . . . . .	9
odbcGetInfo . . . . .	10
package_file . . . . .	10
print.odbc32 . . . . .	11
remote_access_query . . . . .	11
rscript_path . . . . .	12
sqlClear . . . . .	12
sqlDrop . . . . .	13
sqlFetch . . . . .	13
sqlQuery . . . . .	14
sqlSave . . . . .	14
sqlTables . . . . .	15
sqlUpdate . . . . .	15
start_server . . . . .	16
start_server_access2007 . . . . .	17
stop_server . . . . .	17
write_access_db . . . . .	18

## Index 19

---

odbc32-package	<i>odbc32: 32-bit ODBC drivers for 64-bit R sessions</i>
----------------	--

---

### Description

Tools for using 32-bit ODBC drivers from 64-bit R sessions. Based on socket connection between 64-bit R and host 32-bit R.

---

.append_con	<i>Append a (new) connection to a global list of connections</i>
-------------	--

---

### Description

Append a (new) connection to a global list of connections

### Usage

.append\_con(con)

**Arguments**

con

**Value**

a reference (index) number to the list of all connections

**Examples**

#TBA

---

<code>close.odbc32</code>	<i>Closes odbc32 connection</i>
---------------------------	---------------------------------

---

**Description**

Closes odbc32 connection

**Usage**

```
## S3 method for class 'odbc32'  
close(con)
```

**Arguments**

con                   odbc32 connection object

**Value**

logical

**Examples**

TBA

---

connect\_remotely      *Connect to Database Via ODBC using RODB*

---

### Description

This helper function is used by

- [odbcConnectAccess](#)
- [odbcConnectAccess2007](#)
- [odbcConnect](#)
- [odbcDriverConnect](#)

### Usage

```
connect_remotely(socket, FUN, ns = "RODBC", ...)
```

### Arguments

socket	socket
FUN	function to be called remotely
...	arguments to be passed to FUN

### Value

odbc32 object

---

create\_odbc32\_object      *Create Object of Class "odbc32"*

---

### Description

Create Object of Class "odbc32"

### Usage

```
create_odbc32_object(socket, ref)
```

### Arguments

socket	socket
ref	reference returned by <a href="#">eval_remote</a>

---

data2access	<i>writing data into MS Access via 32bit R session script</i>
-------------	---

---

### Description

Procedure for easier exporting of data from an R into MS Access database via separate 32-bit R session script

### Usage

```
data2access(
  data,
  accdb_file,
  overwrite = FALSE,
  fast = TRUE,
  safer = TRUE,
  rownames = FALSE,
  ...
)
```

### Arguments

data	a named list of tables
accdb_file	path to accdb file
overwrite	logical; TRUE for an attempt to drop each table first.
fast	passed to RDOBC::sqlSave
safer	passed to RDOBC::sqlSave
rownames	passed to RDOBC::sqlSave
...	passed to RDOBC::sqlSave

### Details

**overwrite** logical; Attempt to drop any existing table of the name before writing data?

**convert\_dates** character; Convert dates as text before writing into database.

---

default_socket	<i>Default Socket</i>
----------------	-----------------------

---

### Description

Currently, this is just a shortcut to `.GlobalEnv$.r2r_socket`

### Usage

```
default_socket()
```

---

getSqlTypeInfo	<i>Specify or retrieve a mapping of R types to DBMS datatypes.</i>
----------------	--

---

**Description**

Specify or retrieve a mapping of R types to DBMS datatypes.

**Usage**

```
getSqlTypeInfo(con, driver)
```

```
setSqlTypeInfo(con, driver, value)
```

```
setSqlTypeInfo2(con, driver, value)
```

**Arguments**

con	odbc32 object
-----	---------------

value	passed to RODBC::setSqlTypeInfo
-------	---------------------------------

**Details**

setSqlTypeInfo2 is a custom implementation for cases when setSqlTypeInfo fails on "locked environment" error

**Value**

For setSqlTypeInfo none. For getSqlTypeInfo with an argument, a named list. Without an argument, a data frame.

---

odbcClose	<i>close odbc32 connection</i>
-----------	--------------------------------

---

**Description**

close odbc32 connection

**Usage**

```
odbcClose(con)
```

**Arguments**

con	odbc32 connection object
-----	--------------------------

---

odbcConnect	<i>establish ODBC connection defined by DSN name</i>
-------------	--

---

**Description**

establish ODBC connection defined by DSN name

**Usage**

```
odbcConnect(dsn, uid = "", pwd = "", socket = default_socket())
```

**Arguments**

dsn	dsn
uid	user name
pwd	password
socket	socket

---

odbcConnectAccess	<i>Connect to an Access database (*.mdb)</i>
-------------------	--

---

**Description**

Connect to an Access database (\*.mdb)

**Usage**

```
odbcConnectAccess(access.file, uid = "", pwd = "", socket = default_socket())
```

**Arguments**

access.file	path to .mdb file
uid	user name
pwd	password

**Value**

odbc32 object

**Examples**

```
#TBA
```

---

odbcConnectAccess2007 *Connect to an Access 2007 database*

---

**Description**

Connect to an Access 2007 database

**Usage**

```
odbcConnectAccess2007(
  access.file,
  uid = "",
  pwd = "",
  socket = default_socket()
)
```

**Arguments**

access.file	path to .accdb file
uid	user name
pwd	password

**Value**

odbc32 object

**Examples**

```
#TBA
```

---

odbcDataSources *list data sources*

---

**Description**

list data sources

**Usage**

```
odbcDataSources(type = c("all", "user", "system"), socket = default_socket())
```

**Arguments**

type	see type param of RODBC::odbcDataSources()
------	--



---

odbcDriverConnect	<i>establish ODBC connection based on a connection string</i>
-------------------	---

---

**Description**

establish ODBC connection based on a connection string

**Usage**

```
odbcDriverConnect(..., socket = default_socket())
```

**Arguments**

...	other args passed to RODBC::odbcDriverConnect()
socket	

---

odbcEndTran	<i>clear/empty a db table</i>
-------------	-------------------------------

---

**Description**

clear/empty a db table

**Usage**

```
odbcEndTran(con, commit)
```

**Arguments**

con	odbc32 connection object
commit	logical

---

odbcGetErrMsg	<i>Get error message</i>
---------------	--------------------------

---

**Description**

Get error message

**Usage**

```
odbcGetErrMsg(con)
```

**Arguments**

con	odbc32 object
-----	---------------

---

odbcGetInfo	<i>Information on an ODBC connection.</i>
-------------	---

---

**Description**

see RODBC::odbcGetInfo()

**Usage**

```
odbcGetInfo(con)
```

**Arguments**

con	odbc32 connection object
-----	--------------------------

---

package_file	<i>Path to File in This Package</i>
--------------	-------------------------------------

---

**Description**

Path to File in This Package

**Usage**

```
package_file(..., mustWork = FALSE)
```

**Arguments**

...	passed to <a href="#">system.file</a>
mustWork	passed to <a href="#">system.file</a>

**Value**

absolute file path to a file in the installation folder of this package

---

print.odbc32	<i>print method for odbc32</i>
--------------	--------------------------------

---

**Description**

print method for odbc32

**Usage**

```
## S3 method for class 'odbc32'  
print(x)
```

**Arguments**

x                    odbc32 object

**Value**

printed odbc32 object

**Examples**

```
#TBA
```

---

remote_access_query	<i>Query over remote connection</i>
---------------------	-------------------------------------

---

**Description**

Query over remote connection

**Usage**

```
remote_access_query(...)
```

**Arguments**

...                    is passed to r2r::do.call\_remote

**Value**

whatever the call on the remote R session returns

**Examples**

```
#TBA
```

---

rscript_path	<i>Path to Rscript.exe (on Windows)</i>
--------------	---

---

**Description**

Path to Rscript.exe (on Windows)

**Usage**

```
rscript_path(arch = "i386")
```

**Arguments**

arch                    architecture: "i386" for 32 bit or "x64" for 64 bit

**Value**

absolute file path to "Rscript.exe"

---

sqlClear	<i>clear/empty a db table</i>
----------	-------------------------------

---

**Description**

clear/empty a db table

**Usage**

```
sqlClear(con, name)
```

**Arguments**

con                    odbc32 connection object  
name                   name of db table to be dropped

---

sqlDrop	<i>drop a db table</i>
---------	------------------------

---

**Description**

drop a db table

**Usage**

sqlDrop(con, name)

**Arguments**

con	odbc32 connection object
name	name of db table to be dropped

---

sqlFetch	<i>fetch db table</i>
----------	-----------------------

---

**Description**

fetch db table

**Usage**

sqlFetch(con, name, ...)

sqlFetchMore(con, ...)

**Arguments**

con	odbc32 connection object
name	name of db table to be dropped
...	other args passed to RODBC::sqlFetch() on remote

**Value**

data.frame

---

sqlQuery	<i>send query to a db and get results</i>
----------	---

---

**Description**

send query to a db and get results

**Usage**

```
sqlQuery(con, query, errors = NULL, ...)
```

**Arguments**

con	odbc32 connection object
query	text of the query
...	other args passed to RODBC::sqlQuery() on remote

---

sqlSave	<i>upload table to a db</i>
---------	-----------------------------

---

**Description**

upload table to a db

**Usage**

```
sqlSave(con, data, name = NULL, ...)
```

**Arguments**

con	odbc32 object
data	tabular data
name	name of the resulting db table
...	other args passed to RODBC::sqlSave() on remote

---

sqlTables	<i>list tables in DB</i>
-----------	--------------------------

---

**Description**

list tables in DB

**Usage**

```
sqlTables(con, errors = NULL, ...)
```

**Arguments**

con	odbc32 connection object
errors	ignored
...	other args passed to RODBC::sqlTables() on remote

---

sqlUpdate	<i>update db table</i>
-----------	------------------------

---

**Description**

update db table

**Usage**

```
sqlUpdate(con, data, name = NULL, ...)
```

**Arguments**

con	odbc32 connection object
data	tabular data object
name	name of the resulting db table
...	other args passed to RODBC::sqlUpdate() on remote

---

start_server	<i>Start a new R session with a running r2r server</i>
--------------	--

---

### Description

Start a new R session with a running r2r server

### Usage

```
start_server(  
  address = "tcp://localhost",  
  port = pbdZMQ::random_open_port(),  
  arch = "i386",  
  Rbin = rscript_path(arch),  
  r_args = NULL,  
  invisible = FALSE,  
  wait = FALSE,  
  global = TRUE,  
  debug = FALSE  
)
```

### Arguments

address	host address
port	port
arch	architecture
Rbin	path to Rscript binary
r_args	further arguments to Rscript command
invisible	logical; visibility of the new session window

### Value

socket connection (invisible)

### Examples

```
#TBA
```



---

`start_server_access2007`

*Start r2r server connected to an Access database over 32-bit ODBC driver*

---

**Description**

Start r2r server connected to an Access database over 32-bit ODBC driver

**Usage**

```
start_server_access2007(  
  accdb_fp,  
  invisible = FALSE,  
  wait = FALSE,  
  address = "tcp://*",  
  port = pbdZMQ::random_open_port(),  
  arch = "i386",  
  Rbin = rscript_path(arch)  
)
```

---

`stop_server`

*Stop r2r server running on a remote R session*

---

**Description**

Stop r2r server running on a remote R session

**Usage**

```
stop_server(...)
```

**Arguments**

```
...          args passed to r2r::stop_remote()
```

**Value**

logical

**Examples**

```
#TBA
```

---

write_access_db	<i>export data from Rdata file into MS Access db via separate 32-bit Rscript</i>
-----------------	--

---

**Description**

export data from Rdata file into MS Access db via separate 32-bit Rscript

**Usage**

```
write_access_db(rdata_file)
```

# Index

.append\_con, 2

close.odbc32, 3

connect\_remotely, 4

create\_odbc32\_object, 4

data2access, 5

default\_socket, 5

eval\_remote, 4

getSqlTypeInfo, 6

odbc32 (odbc32-package), 2

odbc32-package, 2

odbcClose, 6

odbcConnect, 4, 7

odbcConnectAccess, 4, 7

odbcConnectAccess2007, 4, 8

odbcDataSources, 8

odbcDriverConnect, 4, 9

odbcEndTran, 9

odbcGetErrMsg, 9

odbcGetInfo, 10

package\_file, 10

print.odbc32, 11

remote\_access\_query, 11

rscript\_path, 12

setSqlTypeInfo (getSqlTypeInfo), 6

setSqlTypeInfo2 (getSqlTypeInfo), 6

sqlClear, 12

sqlDrop, 13

sqlFetch, 13

sqlFetchMore (sqlFetch), 13

sqlQuery, 14

sqlSave, 14

sqlTables, 15

sqlUpdate, 15

start\_server, 16

start\_server\_access2007, 17

stop\_server, 17

system.file, 10

write\_access\_db, 18