Package 'RNWIS'

June 1, 2011

Version 0.1-0

Date 2011-05-31

Title National Water Information System: R Interface

Author Jason C. Fisher <jfisher@usgs.gov>

Maintainer Jason C. Fisher <jfisher@usgs.gov>

Depends R (>= 2.13.0), tcltk, sp, RODBC, gpclib, rgdal

Suggests RSurvey

SystemRequirements Tcl/Tk (>= 8.5), ODBC to NWIS database

Description This package provides access to water-resources data.

License GPL (>= 2)

R topics documented:

`RNWIS-package` *National Water Information System: R Interface*

Description

This package provides access to water-resources data. It retrieves data from the National Water Information System (NWIS) through an Open Database Connectivity (ODBC) connection.

Details

Package:	RNWIS
Type:	Package
Version:	0.1-0
Date:	2011-05-31
License:	GPL (>= 2)

Note

The **RNWIS** GUI requires R operate as an SDI application, using multiple top-level windows for the console, graphics, and pager.

The following user instructions are provided for accessing NWIS using **RNWIS**. The site administrator must install the Ingres II client on the user's computer. The user must be added to the *nwis_select* Ingres access group (see section 1.5 of the NWIS Security System Documentation). **RNWIS** must be installed on the user's computer. And the user must be provided with the following database connection information:

- The hostname of the NWIS server, e.g. the hostname for the Idaho district NWIS server is *sun2didbse.wr.usgs.gov*
- The name of the NWIS database in Ingres ("nwisxx" where "xx" is the state postal code, e.g. for the Idaho district the database is named "nwisid").
- The name of the data source name (DSN) that contains the connection information to NWIS. An ODBC data source allows the user to connect to an NWIS database using the *nwis_select* Ingres access group. The site administrator enters the data source information using the *ODBC Data Source Administrator*. **RNWIS** will prompt the user for a data source to connect to. To connect to NWIS, the user would select the data source name for the NWIS connection.

The set of standards used for coding **RNWIS** is documented in Google's R Style Guide.

Author(s)

Jason C. Fisher

Maintainer: <<jfisher@usgs.gov>>

Examples

```
library(RNWIS)
OpenRNWIS()
```

ExploreDatabase *Explore Database Connection*

Description

A GUI for exploring the contents of database.

Usage

```
ExploreDatabase(con, parent = NULL)
```

Arguments

con	RODBC; connection to ODBC database.
parent	tkwin; parent window (optional).

Details

Shows database table names and the column structure for each table.

Author(s)

Fisher, J.C.

See Also

`sqlTables`, `sqlColumns`, `sqlPrimaryKeys`

Examples

```
## Not run: con <- odbcConnect("NWIS Idaho", uid = "", pwd = "")
ExploreDatabase(con)
```

`MapSites` *Map Site Locations in Google Maps*

Description

Add site markers and polygon objects to Google Maps.

Usage

```
MapSites(sites, polygons = NULL, map.id = NULL)
```

Arguments

`sites`	data.frame; site information, see 'Details'.
`polygons`	gpc.poly; polygon information.
`map.id`	character; initial part of the temporary file names.

Details

The `sites` data table has components of `lat`, `lng`, `alt`, `site`, `name`, `agency`, and `type`. Where `lat` and `lng` are the latitude and longitude based on the WGS84 datum; `alt` is the altitude of the site referenced to the specified vertical datum (NGVD29 or NAVD 88); `site` is the unique site identification number; `name` is the name of the site; `agency` is the code for the agency reporting the data; and `type` is the hydrologic setting of the site.

Value

Writes three files to a temporary directory. Base names are identical with extensions indicating file type. An HTML file ('`.html`') is loaded into the default web browser and calls on MarkerClusterer, a javascript library ('`.js`') that creates and manages per-zoom-level clusters for large amounts of markers, and a JSON data file ('`.json`') dynamically created from site and polygon information.

Author(s)

Fisher, J.C.

See Also

browseURL, tempfile

Examples

```
s <- data.frame(lat = c(43.5757402325, 43.5803046219, 43.8642317971),
                lng = c(-112.934719019, -112.876178537, -112.742880208),
                site = c(433433112560201, 433449112523101, 435152112443101),
                name = c("CPP 1", "NPR Test", "ANP 6"),
                agency = "USGS", type = "GW", stringsAsFactors = FALSE)
MapSites(sites = s, map.id = "Example1")

f <- system.file("extdata/ExamplePolygon.ply", package="RNWIS")
p <- read.polyfile(f, nohole=FALSE)
MapSites(sites = s, polygons = p, map.id = "Example2")
```

OpenRNWIS *Open Main Graphical User Interface*

Description

This function activates the main GUI for **RNWIS**.

Usage

```
OpenRNWIS()
```

Details

All functions within **RNWIS** are accessible in this GUI.

Author(s)

Fisher, J.C.

See Also

odbcConnect, writeOGR, write.table, point.in.polygon

Examples

```
OpenRNWIS()
```

`QueryDatabase` *Query the National Water Information System Database*

Description

Acquire data from a single database table using a site selection criteria to constrain the number of sites selected.

Usage

```
QueryDatabase(con, sqtable, sqvars = "*",
          site.no.var = NULL, site.no = NULL,
          site.tp.cd.var = NULL, site.tp.cd = NULL,
          agency.cd.var = NULL, agency.cd = NULL,
          lat.var = NULL, lat.lim = c(NA, NA),
          lng.var = NULL, lng.lim = c(NA, NA),
          alt.var = NULL, alt.lim = c(NA, NA),
          d.t.var = NULL, d.t.lim = c(NA, NA))
```

Arguments

`con`	RODBC; a connection to a ODBC database.
`sqtable`	character; name of the table from which data is to be retrieved.
`sqvars`	character; vector of column names in queried table to be included in the final query results; its default is an asterisk '`*`' and specifies that all columns will be returned.
`site.no.var`	character; column name of table which shows the site identification number.
`site.no`	numeric; vector of site identification numbers.
`site.tp.cd.var`	character; column name of queried table which shows the site type.
`site.tp.cd`	character; vector of site type codes.
`agency.cd.var`	character; column name of queried table which shows the agency code.
`agency.cd`	character; vector of agency codes.
`lat.var`	character; column name of queried table which shows the latitude.
`lat.lim`	numeric; vector of minimum and maximum latitude values (WGS84).
`lng.var`	character; column name of queried table which shows the longitude.
`lng.lim`	numeric; vector of minimum and maximum longitude values (WGS84).
`alt.var`	character; column name of queried table which shows the altitude.
`alt.lim`	numeric; vector of minimum and maximum altitude values (NGVD29 or NAVD 88).
`d.t.var`	character; column name of queried table which shows a date and time variable.
`d.t.lim`	POSIXt; vector of minimum and maximum date values.

Value

On success, a data frame (possibly with 0 rows) or character string. On error, a character vector of error message(s).

Author(s)

Fisher, J.C.

See Also

`sqlQuery`

Examples

```
## Not run: con <- odbcConnect("NWIS Idaho", uid = "", pwd = "")

d <- QueryDatabase(con, sqtable = "sitefile_01",
                   sqvars = c("site_no", "alt_va"),
                   site.no.var = "site_no",
                   site.no = c(432700112470801, 435038112453401))

d <- QueryDatabase(con, sqtable = "sitefile_01",
                   sqvars = c("site_no", "dec_lat_va", "dec_long_va"),
                   site.tp.cd.var = "site_tp_cd", site.tp.cd = "GW",
                   agency.cd.var = "agency_cd", agency.cd = c("USGS", "USEPA"),
                   lng.var = "dec_long_va", lng.lim = c(-114.00, -112.00),
                   lat.var = "dec_lat_va", lat.lim = c(43.00, 44.00))

d <- QueryDatabase(con, sqtable = "gw_lev_01",
                   sqvars = c("site_no", "lev_dt", "lev_va"),
                   site.no.var = "site_no", site.no = 432700112470801)

close(con)
```

Index