Package 'PDSR'

April 21, 2013

Type Package

Title An R package for NASA's Planetary Data System

Version 0.13.04

Date 2013-04-21

Author
George G. Vega <g.vegayon@gmail.com> [aut], Joshua B. Kunst <jbkunst@gmail.com> [aut]

Maintainer George G. Vega <g.vegayon@gmail.com>

Description PDSR use NASA's FTP service to smoothly access PDS. Through metadata file parsing, it helps the data-scientist downloading and describing data without using the web-browser.

URL http://github.com/gvegayon/PDSR

Depends XML, RCurl, plyr, ggplot2, GGally, reshape2,scales

License GPL (>= 3)

LazyLoad yes

R topics documented:

PDSR-package *An R package for NASA's Planetary Data System*

Description

PDSR use NASA's FTP service to smoothly access PDS. Through metadata file parsing, it helps the data-scientist downloading and describing data without using the web-browser.

This package was designed during the 2013 version of the NASA's International SpaceApps Challenge

Please visit the project home for more information: `https://github.org/gvegayon/PDSR`.

or its SpaceApps repo

`https://github.org/spaceappcl/team20`.

Details

Package:	PDSR
Type:	Package
Version:	0.13.04
Date:	2013-04-21
License:	GPL version 2 or later

Author(s)

George G. Vega <g.vegayon@gmail.com>,

Joshua B. Kunst <jbkunst@gmail.com>

dirMissions *Returns a data.frame of the existing missions that suit a keyword*

Description

Given a single or a set of keywords, this function lists NASA's missions and returns a two column data-frame including its name and ID (for PDS)

Usage

```
dirMissions(keywords=NULL, missions=NULL)
```

Arguments

keywords	An optional vector character with pattern to search in the list of NASA's missions.
missions	An optional data.frame of missions.

Details

If no keywords is given, then the full set of missions will be returned, otherwise a regex match will be made against missions' names.

If mission is NULL, then dirMissions will retrive an updated set of missions from http://pds.jpl.nasa.gov/tools/dsstatus/

Value

Two-column data.frame containing missions PDS Id and Name.

Author(s)

George G. Vega

See Also

getFolderStructure

Examples

```
## Not run:
marcians <- dirMissions("mars")

## End(Not run)
```

---

exploreMission *Explore missions datasets*

Description

This function parser the labels files extracting information about the asociated data.

Usage

```
exploreMission(dataid, fullMissions=NULL, maxdep = 1)
```

Arguments

dataid	A character indicating the path of a .lbl object.
fullMissions	A character indicating the path of a .lbl object.
maxdep	Integer of the maximum number of recursions.

Details

This is a recursive function which use regex to perform the extraction of the information.

Value

A nested list with the mission's corresponding data (folder) FTP tree.

Author(s)

George G. Vega

Examples

```
## Not run:
parseLbl("apollo12_sws_28s_19760325")

## End(Not run)
```

---

fullMissionsList *Nested List containing NASA Missions*

Description

This list is used to lookup specific datasets IDs in order to get access to their FTP path.

Usage

```
data(fullMissionsList)
```

Format

A nested list containing NASA's PDS FTP paths.

Source

- Atmospheres Node `ftp://pds-atmospheres.nmsu.edu/`
- Geosciences Node `ftp://pds-geosciences.wustl.edu/`
- Planetary Plasma Interaction (PPI) Node `ftp://pds-ppi.igpp.ucla.edu/`
- Planetary Rings Node `ftp://pds-rings.seti.org/`

Examples

```
data(fullMissionsList)
```

getColnames *Retrieve information of the variables from a .lbl file*

Description

Given a path to a .lbl file this function get the column names of the asociated .tab file.

Usage

```
getColnames(x)
```

Arguments

x	A string indicating the path of a .lbl file.

Details

this and that

Value

A character vector of names.

Author(s)

George Vega

Examples

```
getColnames("data/apollo12_sws_28s_19760325.lbl")
```

getFolderStructure *Get the Folder Structure of an ftp*

Description

This function can be used to get the folder structure given a the URL of an FTP in a list form.

Usage

```
getFolderStructure(url, maxdep=-1, dep=0)
```

Arguments

url	A character string naming the URL of a FTP.
maxdep	Integer of the maximum number of recursions to run.
dep	Integer of the current level of the recursion.

Details

this and that

Value

Nested list containing folder and their folder and files.

Author(s)

Joshua B. Kunst

Examples

```
## Not run: getFolderStructure("ftp://pds-geosciences.wustl.edu/earth/grsfe/")
```

---

getMissionsList *Listing all the missions URL*

Description

Retrieves NASA's PDS FTP structure

Usage

```
getMissionsList(maxdep=1)
```

Arguments

maxdep	Integer of the maximum number of recursions

Details

Matches `dataid` with `fullMissions` looking for its

Value

A nested list of PDS FTP structure.

Author(s)

George G. Vega

Examples

```
## Not run:
getMissionsList()

## End(Not run)
```

getMissionURL *What is this mission URL?*

Description

Given a dataid, returns the corresponding root FTP path of the mission

Usage

```
getMissionURL(dataid, fullMissions=NULL)
```

Arguments

dataid	A character indicating the path of a .lbl object.
fullMissions	A character indicating the path of a .lbl object.

Details

Matches `dataid` with `fullMissions` looking for its

Value

A list with the mission's corresponding FTP URL.

Author(s)

George G. Vega

Examples

```
## Not run:
getMissionURL("apollo12_sws_28s_19760325")

## End(Not run)
```

nasaMissions *Two column data-frame of NASA's PDS missions*

Description

This data-frame provides a full list of NASA's missions which are supported in PDS.

Usage

```
data(nasaMissions)
```

Format

A data-frame with 2103 rows

`dataid` PDS Dataset ID

`mission` Short Mission Description

Source

NASA's Planetary Data System `http://pds.jpl.nasa.gov/`

Examples

```
data(nasaMissions)
```

`parseLbl` *Parser for .lbl files.*

Description

This function parser the labels files extracting information about the asociated data.

Usage

```
parseLbl(x)
```

Arguments

`x`	A character indicating the path of a .lbl object.

Details

This is a recursive function which use regex to perform the extraction of the information.

Value

A nested list with the information.

Author(s)

George Vega

Examples

```
parseLbl("data/apollo12_sws_28s_19760325.lbl")
```

`plot.PDS` *Ploting a PDS object*

Description

This function make a plot of a PDS object.

Usage

```
## S3 method for class 'PDS'
plot(x, variables=sample(names(x$table), size =
                min(c(4, ncol(x$table)))), type = c("ggpairs",
                "plotmatrix", "time"), ...)
```

Arguments

`x`	A PDS Object.
`variables`	A character vector with names of variables in the object$table data.frame.
`type`	A string indicating the type of plot between "ggpairs", "plotmatrix" and "time".
`...`	Other arguments of plot (unused)

Details

ggpairs is

Value

Depending on the type of plot, this function can return a ggplot oject or ggpairs object (from ggplot2 and ggpairs packages respectively)

Author(s)

Joshua B. Kunst

Examples

```
## Not run:
object <- readPDStable("data/apollo12_sws_1hr_1976c9388")
plot.PDS(object, type= "time")

## End(Not run)
```

`readPDStable` *Retrieve data into a PDS object.*

Description

Given a data id download the data from the ftp://pds-geosciences.wustl.edu/ and it convert into a PDS object

Usage

```
readPDStable(dataid)
```

Arguments

`dataid`	String of PDS data id

Details

Given a header search for two files, a plain-text table (.tab) and plain text metadata file (.lbl) which describes table data.

Value

A PDS object.

Author(s)

George G. Vega