Disclaimer
In order to use this library, users must have a local installation of the *R* programming language and the libraries noted in the **Depends** and **Imports** sections below.

R is open source software, and the creation of this library is neither a statement of affiliation with the developers of, nor the endorsement of the use of, *R* by the U.S. Bureau of Economic Analysis.

Use of this library will result in data being stored on users' local machines. Specifically, local copies of BEA API metadata will be stored and updated in the .libPaths() "/beaR/data" directory in order to improve performance of beaSearch.

Package ‘bea.R’

December 2, 2016

Title Bureau of Economic Analysis API

Version 1.0.0

Author Andrea Julca [aut, cre],
Jeff Chen [ctb],
Walt Kampas [ctb]

Depends R (>= 3.2.1), data.table

Imports httr, DT, shiny, jsonlite, googleVis, shinydashboard, ggplot2, stringr, chron, gtable, scales, htmltools, httpuv, xtable, stringi, magrittr, htmlwidgets, Rcpp, munsell, colorspace, plyr, yaml

Description Provides an R interface for the Bureau of Economic Analysis (BEA) API (see <http://www.bea.gov/API/bea_web_service_api_user_guide.htm> for more information) that serves two core purposes -
1. To Extract/Transform/Load data [beaGet()] from the BEA API as R-friendly formats in the user's work space [transformation done by default in beaGet() can be modified using optional parameters; see, too, bea2List(), bea2Tab()].
2. To enable the search of descriptive meta data [beaSearch()].
Other features of the library exist mainly as intermediate methods or are in early stages of development.
Important Note - You must have an API key to use this library.
Register for a key at <http://www.bea.gov/API/signup/index.cfm> .

URL https://CRAN.R-project.org/package=bea.R

License CC0

LazyData no

RoxygenNote 5.0.1

NeedsCompilation no

Maintainer Andrea Julca <Developers@bea.gov>

Repository CRAN

Date/Publication 2016-12-02 23:20:28

R topics documented:

`bea2List` *Convert BEA API httr response payload to list*

Description

Convert BEA API httr response payload to list

Usage

```
bea2List(beaPayload, isMeta = FALSE)
```

Arguments

`beaPayload`	An object with httr class 'response' from call to BEA API
`isMeta`	Special parameter meant to interact with metadata functions (default: FALSE)

Value

An object of class 'list' of several dimensions. View list structure using 'str(yourList)'.

Examples

```
userSpecList <- list('UserID' = 'yourKey' ,
'Method' = 'GetData',
'datasetname' = 'NIPA',
'Frequency' = 'A',
'TableID' = '68',
'Year' = 'X')
resp <- beaGet(userSpecList, asTable = FALSE)
BL <- bea2List(resp)
```

`bea2Tab` *Convert BEA API httr response or list payload to data.table*

Description

Convert BEA API httr response or list payload to data.table. Also, converts LONG data frame (default API format - see bea2List results) to WIDE data (with years as columns) by default

Usage

```
bea2Tab(beaPayload, asWide = TRUE, iTableStyle = TRUE)
```

Arguments

`beaPayload` An object of class 'list' or httr 'response' returned from beaGet() call to BEA API

`asWide` Return data.table in wide format (default: TRUE)

`iTableStyle` If "asWide = TRUE", setting "iTableStyle = TRUE" will return data.table in same format as shown on BEA website, with dates and attributes as column headers and series as rows; otherwise, results have series codes as column headers (default: TRUE)

Value

An object of class 'data.table' containing data from beaGet(...) with custom attributes(BDT)$params.

Examples

```
userSpecList <- list('UserID' = 'yourKey' ,
'Method' = 'GetData',
'datasetname' = 'NIPA',
'Frequency' = 'A',
'TableID' = '68',
'Year' = 'X')
resp <- beaGet(userSpecList)
BDT <- bea2Tab(resp)
```

`beaGet` *Pass list of user specifications (including API key) to return data from BEA API.*

Description

Pass list of user specifications (including API key) to return data from BEA API.

Usage

```
beaGet(beaSpec, asString = FALSE, asList = FALSE, asTable = TRUE,
  asWide = TRUE, isMeta = FALSE, iTableStyle = TRUE)
```

Arguments

`beaSpec`	A list of user specifications (required). In this example, 'GetData' specifies that we want data values (rather than metadata), 'NIPA' specifies the dataset, 'A' specifies that we want annual data, 'TableID' = '68' gets a specific table, and 'X' gets all years. See BEA API documentation or use metadata methods for complete lists of parameters.
`asString`	Return result body as a string (default: FALSE)
`asList`	Return result body as a list (default: FALSE)
`asTable`	Return result body as a data.table (default: TRUE)
`asWide`	Return data.table in wide format (default: TRUE)
`isMeta`	Special parameter meant to interact with metadata functions (default: FALSE)
`iTableStyle`	If "asWide = TRUE", setting "iTableStyle = TRUE" will return data.table in same format as shown on BEA website, with dates and attributes as column headers and series as rows; otherwise, results have series codes as column headers (default: TRUE)

Value

By default, an object of class 'list' of several dimensions. View list structure using 'str(yourList)'.

Examples

```
userSpecList <- list('UserID' = 'yourAPIKey' ,
'Method' = 'GetData',
'datasetname' = 'NIPA',
'Frequency' = 'A',
'TableID' = '68',
'Year' = 'X')
BDT <- beaGet(userSpecList, asTable = TRUE)
```

beaParams *Gives list of parameters possible for a given dataset*

Description

Gives list of parameters possible for a given dataset

Usage

```
beaParams(beaKey, setName)
```

Arguments

`beaKey`	Your API key
`setName`	Name of BEA dataset (e.g., ’NIPA’)

Value

A metadata object of class ’list’ of several dimensions. View list structure using ’str(yourList)’.

Examples

```
beaParams('yourAPIkey', 'RegionalData')
```

`beaParamVals` *Gives list of values possible for a given dataset’s parameters*

Description

Gives list of values possible for a given dataset’s parameters

Usage

```
beaParamVals(beaKey, setName, paramName)
```

Arguments

`beaKey`	Your API key
`setName`	Name of BEA dataset (e.g., NIPA)
`paramName`	Name of BEA dataset parameter (e.g., TableID)

Value

A metadata object of class ’list’ of several dimensions. View list structure using ’str(yourList)’.

Examples

```
beaParamVals('yourAPIkey', 'RegionalData', 'keycode')
```

beaSearch *Search a selection of indexed BEA data table names, series labels, and series codes.*

Description

Searches indexed dataset table name, label, and series codes. CAUTION: Currently only works with NATIONAL datasets (NIPA, NIUnderlyingDetail), temporarily excluding FixedAssets, and REGIONAL datasets (RegionalData, RegionalProduct, RegionalIncome)

Usage

```
beaSearch(searchTerm, beaKey = NULL, asHtml = FALSE)
```

Arguments

`searchTerm`	A word or phrase of class 'character' to be found in BEA datasets
`beaKey`	Character string representation of user API key. Necessary for first time use and updates; recommended for anything beyond one-off searches from the console.
`asHtml`	Option to return results as DT markup, viewable in browser. Allows search WITHIN YOUR ALREADY-FILTERED RESULTS ONLY. Requires package 'DT' to be installed.

Value

An object of class 'data.table' with information about all indexed sets in which the search term was found.

Examples

```
beaSearch('gross domestic product', asHtml = TRUE)
```

beaSets *Returns a list of all datasets*

Description

Returns a list of all datasets

Usage

```
beaSets(beaKey)
```

Arguments

`beaKey`	Your API key

Value

A metadata object of class ’list’ of several dimensions. View list structure using ’str(yourList)’.

Examples

```
beaSets('yourAPIkey')
```

beaUpdateMetadata *Download BEA metadata into library/data folder if needed*

Description

Download BEA metadata into library/data folder if needed

Usage

```
beaUpdateMetadata(datasetList, beaKey)
```

Arguments

`datasetList`	list of BEA datasets to update local metadata file for (e.g., list(’NIPA’, ’FixedAssets’))
`beaKey`	Your API key

Value

Nothing. This updates local .RData files to be used in beaSearch.

Examples

```
beaUpdateMetadata(list('RegionalData', 'NIPA'), beaKey = 'yourAPIkey')
```

beaViz *Visualize BEA API response payload*

Description

When entered into the R console, the function below starts an interactive dashboard. CAUTION: Currently only works with NATIONAL datasets (NIPA, NIUnderlyingDetail, FixedAs-sets). R Studio users must opt to "show in browser" for this method to be fully functional.

Usage

```
beaViz(beaPayload = NULL, beaKey = NULL)
```

Arguments

`beaPayload`	An httr response from call to BEA API
`beaKey`	Your 36-digit BEA API key

Examples

```
userSpecList <- list('UserID' = 'yourKey' ,
'Method' = 'GetData',
'datasetname' = 'NIPA',
'Frequency' = 'A',
'TableID' = '68',
'Year' = 'X')
resp <- beaGet(userSpecList)
BDF <- beaViz(resp)
userSpecList <- list('UserID' = 'yourKey' ,
'Method' = 'GetData',
'datasetname' = 'NIPA',
'Frequency' = 'A',
'TableID' = '68',
'Year' = 'X')
```

Index