

# Package ‘anametrix’

February 19, 2015

**Type** Package  
**Title** Bidirectional connector to Anametrix API  
**Version** 1.6  
**Date** 2013-07-18  
**Author** Roman Jugai  
**Maintainer** Roman Jugai <roman@anametrix.com>  
**Depends** R (>= 1.8.0), RCurl, XML, pastecs  
**Description** Bidirectional connector to Anametrix API  
**License** LGPL  
**NeedsCompilation** no  
**Repository** CRAN  
**Date/Publication** 2013-07-22 19:26:51

## R topics documented:

anametrix-package . . . . .	1
getAxDataset . . . . .	2
getAxDatasetFromEncodedQuery . . . . .	3
printTable . . . . .	4
uploadData . . . . .	4

<b>Index</b>	<b>5</b>
--------------	----------

---

anametrix-package	<i>Bidirectional connector to Anametrix API</i>
-------------------	---

---

## Description

Provides function(s) to extract data from Anametrix API into R data structure(s). Provides function(s) to upload data from R into Anametrix table(s).

## Details

Package: anamatrix  
 Type: Package  
 Version: 1.5  
 Date: 2012-12-13  
 License: LGPL

In R install package: `install.package("anamatrix")`. Load package: `library("anamatrix")`

### Author(s)

Author, Roman Jugai

Maintainer: Roman Jugai <[roman@anamatrix.com](mailto:roman@anamatrix.com)>

### References

<http://www.anamatrix.com>

---

getAxDataset	<i>Extracts data from Anamatrix API into R data structure</i>
--------------	---

---

### Description

Function `getAxDataset` extracts data from Anamatrix API and returns R data frame

### Usage

```
getAxDataset(auth, reportSuiteId, tableObject, columns, datestart,
             dateend, count, dateFormat, sortColumn, sortDirection, verbose)
```

### Arguments

<code>auth</code>	Authentication instance.
<code>reportSuiteId</code>	Data Suite ID. Contact your product manager for assistance.
<code>tableObject</code>	Configuration of a table. Use <code>getTableConfiguration(auth, reportsuiteId, tableName)</code> to get it.
<code>columns</code>	Array of column names for the request. Example: <code>c("city", "country", "male_population")</code> Use <code>printTable(tableObject)</code> to see all column names.
<code>datestart</code>	Start date of the request. Use YYYY-MM-DD format. Example: "2012-03-31"
<code>dateend</code>	End date of the request. Use YYYY-MM-DD format. Example: "2012-03-31"
<code>count</code>	Number of rows requested. Example: "1000"
<code>dateFormat</code>	Format of the date column. Example: "day", "week", or custom format like "MM-dd-yy" or "w-yyyy"
<code>sortColumn</code>	Column to sort on. Use any column provided in the columns array
<code>sortDirection</code>	Direction of the sorting. Example: "asc"
<code>verbose</code>	Specifies whether to print out notifications. Example: FALSE

**Format**

R Data frame containing number of observations requested from Anamatrix API

**Value**

D R Data frame containing data

**Author(s)**

Author, Roman Jugai

**References**

<http://www.anamatrix.com>

**Examples**

```
#Example: Extracting 3 columns from Browsers table from Anamatrix API

username = "demo.user@demoanamatrix.com"
password = "demo.user"
auth <- authenticate("https://demo.anamatrix.com/api", username, password)
table <- getTableConfiguration(auth, 10385, "Browsers")
printTable(table)
axdata <- getAxDataset(auth, 10385, table, c("name", "visits", "page_views"), "2012-03-01",
"2012-04-30", 100, "month", "visits", "asc", TRUE)
head(axdata)
```

---

getAxDatasetFromEncodedQuery

*Extracts data from Anamatrix API into R data structure*

---

**Description**

Function `getAxDatasetFromEncodedQuery` extracts data from Anamatrix API using Base64-encoded Anamatrix query, and returns R data frame

**Usage**

```
getAxDatasetFromEncodedQuery(auth, encodedQuery, startDate, endDate, verbose)
```

**Arguments**

<code>auth</code>	Authentication instance.
<code>encodedQuery</code>	Base64-encoded query XML corresponding to Anamatrix API requirements. Contact your product manager for assistance.
<code>startDate</code>	Specifies start date of the query (format: YYYY-MM-DD).
<code>endDate</code>	Specifies end date of the query (format: YYYY-MM-DD).
<code>verbose</code>	Specifies whether to print out notifications. Example: FALSE

**Author(s)**

Author, Roman Jugai

---

printTable                      *Prints Table object configuration*

---

**Description**

Prints configuration of a table object extracted from Anamatrix API

**Usage**

```
printTable(tableObject)
```

**Arguments**

tableObject	Configuration of a table. Use <code>getTableConfiguration(auth, reportsuiteId, table-Name)</code> to get it.
-------------	--

---

uploadData                      *Uploads data into Anamatrix from R data structure*

---

**Description**

Function `uploadData` uploads data from R `data.frame` into Anamatrix table

**Usage**

```
uploadData(auth, reportSuiteId, tableObject, dataframe, columns, tag, truncation)
```

**Arguments**

auth	Authentication instance.
reportSuiteId	Data Suite ID. Contact your product manager for assistance.
tableObject	Configuration of a table. Use <code>getTableConfiguration(auth, reportsuiteId, table-Name)</code> to get it.
dataframe	R Data frame containing data that needs to be uploaded
columns	Array of column names that need to be uploaded. Example: <code>c("city","country","male_population")</code> Use <code>printTable(tableObject)</code> to see all column names.
tag	Tag of the upload.
truncation	Truncation type. If truncation type is 'none', it doesn't truncate anything before the upload. If truncation type is 'tag', all data with that tag will be truncated before the upload.

# Index

## \*Topic **connection**

- anamatrix-package, 1
- getAxDataset, 2
- getAxDatasetFromEncodedQuery, 3
- printTable, 4
- uploadData, 4

## \*Topic **database**

- anamatrix-package, 1
- getAxDataset, 2
- getAxDatasetFromEncodedQuery, 3
- printTable, 4
- uploadData, 4

anamatrix (anamatrix-package), 1

anamatrix-package, 1

authenticate (getAxDataset), 2

constructQueryXML (getAxDataset), 2

constructUploadXML (uploadData), 4

getAxDataset, 2

getAxDatasetFromEncodedQuery, 3

getColumnTypeId (getAxDataset), 2

getDBDateFormat (getAxDataset), 2

getFullTableListXML (getAxDataset), 2

getTableConfiguration (getAxDataset), 2

isdefined (getAxDataset), 2

populateSummaryStatisticsTable  
(uploadData), 4

printTable, 4

queryReader (getAxDataset), 2

randomAlphaNumericUID (getAxDataset), 2

rowToXML (uploadData), 4

uploadData, 4