

# Package ‘TFX’

February 19, 2015

**Type** Package

**Title** R API to TrueFX(tm)

**Version** 0.1.0

**Date** 2012-11-26

**Author** Garrett See

**Maintainer** Garrett See <gsee000@gmail.com>

**Description** Connects R to TrueFX(tm) for free streaming real-time and historical tick-by-tick market data for dealable interbank foreign exchange rates with millisecond detail.

**License** GPL-3

**URL** <http://www.truefx.com>

**Depends** R(>= 2.15.0)

**Imports** XML

**Collate** 'TrueFXRef.R' 'TFX-package.R' 'TFX.R'

**Repository** CRAN

**Date/Publication** 2012-11-27 06:43:38

**NeedsCompilation** no

## R topics documented:

ConnectTrueFX . . . . .	2
Disconnect . . . . .	4
isActive . . . . .	4
ParseTrueFX . . . . .	5
Reconnect . . . . .	7
TFX . . . . .	8
TrueFXRef . . . . .	9

<b>Index</b>	<b>11</b>
--------------	-----------

---

ConnectTrueFX	<i>Query TrueFX(tm)</i>
---------------	-------------------------

---

### Description

Create a session with TrueFX(tm) and request market data.

### Usage

```
ConnectTrueFX(currency.pairs, username, password,
               qualifier = "default", format, snapshot = FALSE)
```

```
QueryTrueFX(session, parse.response = TRUE,
             pretty = TRUE, reconnect = TRUE)
```

### Arguments

currency.pairs	character vector, or comma delimited string of Symbols (ISO names) of currency pairs. (e.g. "EUR/USD,AUD/USD", or c("EUR/USD", "AUD/USD")). If missing or if nchar(currency.pairs) < 1, the Symbols of all currency pairs for which TrueFX(tm) provides historical data will be used (see references section).
username	character. A registered TrueFX(tm) user name; required to establish an authenticated session.
password	character. A registered TrueFX(tm) password; required to establish an authenticated session
qualifier	any string; required to establish an authenticated session. ("default" by default)
format	One of "default", "csv", or "html". Indicates the format for the HTTP Response.
snapshot	logical. No incremental updates if TRUE
session	a TFXsession object created by ConnectTrueFX.
parse.response	logical. Should the results be passed through ParseTrueFX before returning?
pretty	logical. Passed to ParseTrueFX. Indicates whether to format the parsed results and convert to data.frame. Ignored if parse.response is not TRUE
reconnect	logical. If the TFXsession has timed out, should it be reconnected?

### Details

If no currency.pairs are provided to ConnectTrueFX, the 15 pairs for which TrueFX(tm) offers historical data will be used. Note that only the first 10 of these are returned in an unauthenticated session.

ConnectTrueFX will create a TFXsession classed object that can be used in calls to QueryTrueFX to request market data.

Of the three formats, "default" is the most timely (updates first) and "csv" is the most delayed (updates last)

the "csv" and "html" formats have the "High" and "Low" columns backwards. ("default" does not). This may be corrected for in a future release if the TrueFX(tm) Web service doesn't correct it first.

**Value**

ConnectTrueFX returns a TFXsession object that is a TrueFX(tm) server-generated session ID returned with a successful authenticated session request. It is a colon delimited string with username, password, qualifier, and the time (in milliseconds) that the session was created.

QueryTrueFX returns the results of a TrueFX(tm) request using

**Note**

the formal arguments start with the same lowercase letter as their corresponding TrueFX(tm) Market Data Web Query Parameters

**Author(s)**

Garrett See

**References**

[http://www.truefx.com/dev/data/TrueFX\\_MarketDataWebAPI\\_DeveloperGuide.pdf](http://www.truefx.com/dev/data/TrueFX_MarketDataWebAPI_DeveloperGuide.pdf)

<http://truefx.com/?page=downloads> to see for which pairs TrueFX(tm) provides historical data.

**See Also**

[ParseTrueFX](#), [Reconnect](#), [TrueFXRef](#)

**Examples**

```
## Cannot run these because there may not be an internet connection
## Not run:
QueryTrueFX() #unauthenticated
QueryTrueFX(pretty=FALSE)
QueryTrueFX(parse=FALSE)

## For authenticated session, you must have a username and password (it's free).
## Use your username and password instead of JSTrader and Ou812
id <- ConnectTrueFX('EUR/USD,GBP/USD', username='JSTrader', password='Ou812')
QueryTrueFX(id)
QueryTrueFX(ConnectTrueFX(username='JSTrader', password='Ou812',
                           format='csv'), parse=FALSE)

QueryTrueFX(ConnectTrueFX(username='JSTrader', password='Ou812',
                           format='html'), parse=FALSE)

## If you have shiny installed
## install.packages("shiny", repos="http://rstudio.org/_packages")
library(shiny)
runGist("4122626")
## view the code for this shiny app at
#browseURL("https://gist.github.com/4122626")

## End(Not run)
```

Disconnect

*Disconnect a session*

---

**Description**

Disconnect a session (make it inactive).

**Usage**

```
Disconnect(x, ...)
```

```
## S3 method for class 'TFXsession'  
Disconnect(x, ...)
```

**Arguments**

x                    an object to disconnect  
...                   other arguments for methods

**Value**

a disconnected/inactive TFXsession object

**See Also**

[ConnectTrueFX](#), [Reconnect](#)

**Examples**

```
## Not run:  
sess <- ConnectTrueFX(username='JSTrader', password='0u812')  
isActive(sess) #TRUE  
Disconnect(sess)  
isActive(sess) #FALSE  
  
## End(Not run)
```

---

isActive*Is a session active?*

---

**Description**

Test to see if a session is active

**Usage**

```
isActive(x, ...)  
  
## S3 method for class 'TFXsession'  
isActive(x, ...)
```

**Arguments**

x	an object to test
...	other arguments for methods

**Details**

In order for a TFXsession to be active, it must have been authenticated less than 1 minute ago. If it was created with snapshot=TRUE it will become inactive after it is used once.

**Value**

logical

**Note**

This function assumes that if the session has not been used in 60 seconds is not active, even though TrueFX(tm) sessions actually stay active for roughly 70 seconds.

**Examples**

```
## Not run:  
sess <- ConnectTrueFX("GBP/JPY", username='JSTrader', password='0u812')  
isActive(sess) #TRUE  
  
## End(Not run)
```

---

ParseTrueFX

*Parse TrueFX(tm) response*

---

**Description**

Parse the results of a TrueFX(tm) query.

**Usage**

```
ParseTrueFX(x, pretty = TRUE)
```

**Arguments**

<code>x</code>	The response from a TrueFX(tm) request. Can be any of the three formats: <code>default</code> , <code>csv</code> or <code>html</code>
<code>pretty</code>	logical. If TRUE (Default), output will be converted to a <code>data.frame</code> and columns will be converted from character to the appropriate classes and combined.

**Details**

This function will parse the results of a call to [QueryTrueFX](#). It can handle any of the three TrueFX(tm) response formats: “default”, “csv”, or “html”. By default, it will convert the results into a nicely formatted `data.frame`. If, called with `pretty=FALSE`, a list of strings will be returned.

All times are in GMT

**Value**

By default, a `data.frame` is returned that has columns “Bid.Price”, “Ask.Price”, “High”, “Low”, and “TimeStamp”. If called with `pretty=FALSE`, a list of character vectors – named “Symbol”, “BidBigNumber”, “BidPip”, “OfferBigNumber”, “OfferPip”, “High”, “Low”, “TimeStamp” – will be returned.

If the format is “html”, there will also be an “Open” column

**Note**

Although the TrueFX(tm) Market Data Web API Developer Guide indicates that both the “csv” and “html” formats include values for “Open”, only the “html” format actually does.

**Author(s)**

Garrett See

**References**

[http://www.truefx.com/dev/data/TrueFX\\_MarketDataWebAPI\\_DeveloperGuide.pdf](http://www.truefx.com/dev/data/TrueFX_MarketDataWebAPI_DeveloperGuide.pdf)

**See Also**

[QueryTrueFX](#), [TrueFXRef](#)

**Examples**

```
# x <- QueryTrueFX() #Cannot run this if no internet connection
x <- paste0("EUR/USDUSD/JPY1.31#81.9085661.31#81.9435941.31990#81.6421.3182",
           "1#81.50413351311514701335131150004")
ParseTrueFX(x)
ParseTrueFX(x, pretty=FALSE)
```

---

**Reconnect***Reconnect a session that is no longer active*

---

**Description**

Reconnect will create a new session and update the 'id' to the new authenticated id returned by the TrueFX(tm) server.

**Usage**

```
Reconnect(x, ...)
```

```
## S3 method for class 'TFXsession'  
Reconnect(x, ...)
```

**Arguments**

x	an object to be re-connected
...	other args for methods

**Details**

After roughly 70 seconds, an authenticated TrueFX(tm) session will time-out. Also, a connection made with snapshot=FALSE will be disconnected after it is used once.

A non-active TrueFX(tm) session id is treated like an unauthenticated session.

**Value**

a TFXsession object of an active/authenticated session.

**Examples**

```
## Cannot run because there may not be an internet connection  
## Not run:  
## You must use your username and password instead of JSTrader and Ou812  
sess <- ConnectTrueFX("USD/JPY", username='JSTrader', password='Ou812')  
Disconnect(sess)  
isActive(sess) #FALSE  
Reconnect(sess)  
isActive(sess) #TRUE  
  
## End(Not run)
```

---

TFX

*TFX package: An R interface to the TrueFX(tm) Market Data Web API*

---

## Description

*TrueFX(tm) is a brand name that is owned by Integral Development Corp. This software is in no way affiliated, endorsed, or approved by TrueFX(tm), Integral Development Corp, or any of its affiliates. The **TFX** package provides functions to access the free, real-time, top-of-book, dealable, interbank foreign exchange rate quotes provided by TrueFX(tm).*

## Details

The Web API can be accessed with an authenticated or an unauthenticated session. The following is from TrueFX(tm) Market Data Web API Developer Guide: "Unauthenticated requests ... return a default response in the form of a snapshot session that includes data for all currency pairs... Unauthenticated requests do not support the more powerful query-based functionality, including incremental responses containing only the changed currency pairs."

A registered TrueFX(tm) account with a confirmed username is required to create an authenticated session (TrueFX(tm) accounts are free).

There are typically three steps to creating an authenticated TrueFX(tm) session and requesting data:

[ConnectTrueFX](#) is used to request a `TFXsession` object which has a session ID needed to make a data request.

[QueryTrueFX](#) will request market data from TrueFX(tm) using a `TFXsession` object that was created by [ConnectTrueFX](#)

[ParseTrueFX](#) will parse the results returned by [QueryTrueFX](#) into something that is easier to work with.

There are also functions to [Disconnect](#) and [Reconnect](#) a `TFXsession`

This package does not yet have explicit support streaming data, but it can be accomplished a few different ways, one of which would be to use a `while` loop.

In addition to real time data, TrueFX(tm) also offers historical data since 2009.

From the TrueFX(tm) website, "TrueFX is the first service that brings you real, dealable prices from real market participants from all the major market makers, with absolutely no intermediary. As a technology company, we can offer you historical tick-by-tick market data, at zero cost to you.

This data is top-of-the-book, tick-by-tick market data, with fractional pip spreads in millisecond detail. All timestamps are based on GMT."

The data can be downloaded from <http://www.truefx.com/?page=downloads>

There is a script in the `/inst/parser` directory of the `FinancialInstrument` package (<http://tinyurl.com/DownloadTrueFX>) that will download all available data, convert it to `xts` and save it to disk in binary files that are split by day such that `FinancialInstrument::getSymbols.FI` can easily read them.

Some version of that script may make its way into a future release of this package.



**Author(s)**

Garrett See <gsee000@gmail.com>

**References**

<http://www.truefx.com>

[http://www.truefx.com/dev/data/TrueFX\\_MarketDataWebAPI\\_DeveloperGuide.pdf](http://www.truefx.com/dev/data/TrueFX_MarketDataWebAPI_DeveloperGuide.pdf)

**Examples**

```
## Not run:
## Unauthenticated
QueryTrueFX()
QueryTrueFX(pretty=FALSE)

## Must have a TrueFX(tm) account to run the following (Membership is free)
## Replace JSTrader and Ou812 with your username and password, respectively
id <- ConnectTrueFX("EUR/USD,AUD/JPY", u='JSTrader', p='Ou812', f='html')
QueryTrueFX(id)

browseURL(paste0("http://webrates.truefx.com/rates/connect.html?id=", id))

#view the Web API Developer Guide:
TrueFXRef()

## End(Not run)
```

---

TrueFXRef

*Download and view the TrueFX(tm) Market Data Web API Developer Guide*

---

**Description**

Download and view the TrueFX(tm) Market Data Web API Developer Guide

**Usage**

```
TrueFXRef(show = c("pdf", "webpage"))
```

**Arguments**

show	What to show. Either “pdf” or “webpage” (“web” and “url” will also be recognized as “webpage”). Alternatively, can be numeric: 1 for “pdf”, 2 for “webpage”)
------	--

**Value**

called for side-effect. Opens the TrueFX(tm) Market Data Web API Developer Guide in a pdf-viewer or web browser.

**Note**

the idea for this function came from the **IBrokers** function IBrokersRef written by Jeff Ryan.  
Thanks Jeff!

**Author(s)**

Garrett See

**References**

[http://www.truefx.com/dev/data/TrueFX\\_MarketDataWebAPI\\_DeveloperGuide.pdf](http://www.truefx.com/dev/data/TrueFX_MarketDataWebAPI_DeveloperGuide.pdf)

**Examples**

```
## Not run:  
TrueFXRef()  
TrueFXRef("web")  
  
## End(Not run)
```

# Index

\*Topic **IO**

TFX, 8

\*Topic **package**

TFX, 8

\*Topic **programming**

TFX, 8

ConnectTrueFX, 2, 4, 8

Disconnect, 4, 8

isActive, 4

ParseTrueFX, 3, 5, 8

QueryTrueFX, 6, 8

QueryTrueFX (ConnectTrueFX), 2

Reconnect, 3, 4, 7, 8

TFX, 8

TFX-package (TFX), 8

TrueFXRef, 3, 6, 9