

## Tick History API package for R

Release Notes

Document Version 1.2

Date of Issue: 22/08/2013

Welcome to the Tick History API package for R. This package will integrate the TH services and capabilities into R allowing users to access the data directly.

R is a free software environment for statistical computing and graphics. The TH API package lets you consume TH data within R. This means that you can preview, submit or perform various TH operations within R without the need to log on to the web application.

This document contains the following sections:

- New features and Changes
- Installation & Setup
- Data retrieval and Examples
- Message Type Lists
- Rtime
- Help files

## New Features and Changes

### Version: THAPI\_0.9.4

THAPI\_0.9.4 incorporates updated arguments for the submitRequest and submitFTPRequest functions. The update now allows users to submit requests based on GMT and Local time and also display results in GMT and Local time. An updated help section is also available.

### Version: THAPI\_0.9.3

THAPI\_0.9.3, has an updated and simplified messagetypeinput method and arguments for the submitRequest and submitFTPRequest functions. An updated help section is also available.

## Installation & Setup

After downloading the attached installation file THAPI\_0.9.4, install the package within the R GUI.

You will also need to install R Curl, XML and the base64 packages prior to loading THAPI in R. You can install these from the R GUI in the Packages menu.

Once you have installed the package you will need to call the library function, by typing below command into your workspace, to load the package into R. This will automatically load THAPI and its auxiliary packages.

```
library(THAPI)
```

Before you can use the Tick History service, you will also need to create a login object containing your username and password:

```
rdth <- createCredential(user="username@domain",password="your password")
```

To test if this works, we will make a call to the Tick History service. The following command will make a call to the `getBondTypes` service which will return a list of bond types.

```
getBondTypes(rdth)
```

If you see the following on your screen, you have successfully installed the package.

```
> getBondTypes(rdth)
  value                                     longName
1      1                                [FRN] floating rate note
2      2                                [STR] straights
3      3                                [CNV] convertibles
4      4                                [WNT] warrants
5      5                                [C ] US Corporates, coupon only
6      6                                [R ] US Corporates, registered only
7      7 [CR ] US Corporates, coupon & registered, not interchangeable
8      8 [C+R] US Corporates, coupon & registered, interchangeable
9      9                                [USG] US Government securities
10     10                                [DOM] domestics
11     11                                [FOR] foreign
12     12                                [UKG] UK gilts
13     13                                [UKC] UK Corporates
14     14                                [JDO] Japanese domestics
```

## Data retrieval and Examples

### searchRICs

Similar to the search functionality within the online TRTH application, this function allows searches of the reference data cross-referenced by date range and search criteria, listing all instruments that match.

#### Arguments

<i>rdth</i>	The Tick History credential object
<i>startdate</i>	The start date of the search
<i>enddate</i>	The end date of the search
<i>domain</i>	The domain to search, e.g. "EQU","OPT","FUT", etc
<i>exchange</i>	The exchange to search, e.g. "ASX"
<i>ricregex</i>	The RIC regular expression pattern, e.g. "*.AX\$", all RICs ending in .AX

#### Method

*searchRICs*(*rdth*,*startdate*="character",*enddate*="character",*domain*="character",*exchange*="character",*ricregex*="character")

#### Example

*searchRICs*(*rdth*,"2011-11-11","2011-11-12","EQU","ASX","^BHP.\*")

```
> searchRICs(rdth,"2012-11-22","2012-11-23","EQU","ASX","^BHP.*")
  code startdate enddate status
1   BHP.AX 1996-01-01 2013-08-21 unknown
2  BHPCDta.AX 2010-02-28 2013-03-27 unknown
3   BHPIO1.AX 2011-05-12 2013-08-21 unknown
4  BHPIO1ta.AX 2011-05-12 2013-08-20 unknown
5   BHPIO2.AX 2011-11-04 2013-08-21 unknown
6  BHPIO2ta.AX 2011-11-04 2013-08-20 unknown
7   BHPIOI.AX 2003-11-17 2013-08-21 unknown
8  BHPIOIta.AX 2009-10-30 2013-08-20 unknown
9   BHPIOL.AX 2004-03-23 2013-08-21 unknown
10 BHPIOLta.AX 2010-05-25 2013-08-20 unknown
11  BHPIOM.AX 2004-11-05 2013-08-21 unknown
12 BHPIOMta.AX 2010-05-25 2013-08-20 unknown
13  BHPIOO.AX 2007-12-06 2013-08-21 unknown
14 BHPIOOta.AX 2009-10-30 2013-08-20 unknown
15  BHPIOZ.AX 2008-11-28 2013-03-09 unknown
```

## submitRequest

Request data for a single instrument over a single day. The request will return a unique request id – the operation `getRequestResult` can be used to retrieve the data or query the status of the request.

For larger requests which include multiple instruments over multiple days, see `submitFTPRequest` function.

### Arguments

<i>rdth</i>	The Tick History credential object
<i>friendlyname</i>	Give the request a name
<i>instrument</i>	The RIC code, e.g. "BHP.AX"
<i>date</i>	The request date, e.g. "2011-11-12"
<i>starttime</i>	Request start time in GMT, e.g. "00:00:00:000000"
<i>endtime</i>	Request end time in GMT, e.g. "23:59:59:999999"
<i>reqtype</i>	The request type, i.e. "TimeAndSales", "MarketDepth", "NasdaqLevel2", "Intraday", "EndOfDay", "RTCE", "RTCEMarketDepth" and "Any"
<i>mktdepth</i>	Specify a numeric string to indicate whether this is a market depth request e.g. not a market depth request "()"
<i>messagelist</i>	The message type and field string for the request, in this format "MessageType1:FieldName1,FieldName2;MessageType2:FieldName1,FieldName2", i.e. "Trade:Price,Volume;Quote:Bid Price,Ask Price"
<i>disInGMT</i>	Display time in GMT or Local time, i.e. display in GMT "true"
<i>reqInGMT</i>	The request time in GMT or Local time, i.e. request time in GMT "true"

### Method

```
signature(r = "rdth", friendlyname = "character", instrument = "character", date = "character", starttime = "character", endtime = "character", reqtype = "character", mktdepth = "character", messagelist = "character", disInGMT = "character", reqInGMT = "character")
```

### Example

```
j <- submitRequest(rdth, "test", "BHP.AX", "2012-11-22", "00:00:00", "23:59:59.999", "TimeAndSales", mktdepth="0", messagelist="Trade:Price,Volume;Quote:Bid Price,Ask Price", reqInGMT="false", disInGMT="false")
```

```
> j <- submitRequest(rdth, "test", "BHP.AX", "2012-11-22", "00:00:00", "23:59:59.999",
"TimeAndSales", mktdepth="0", messagetype="Trade:Price,volume;Quote:Bid Price,Ask Price",
reqInGMT="false", disInGMT="false")
[1] "Submitted request: kikai.kobayashi@sirca.org.au-test-N56334508"
[1] "Polling InFlightStatus to check if request is ready "
[1] "Polling InFlightStatus to check if request is ready "
[1] "Polling InFlightStatus to check if request is ready "
[1] "Request is completed"
[1] "Getting request result"
```

Once you have the data stored in a variable e.g. 'j' above, you can select the columns and rows to display. Below outputs the RIC, Price, Volume, Bid price and Ask price of the first 10 rows of data in Local time.

```
> j[1:10,]
      RIC      RTime Current.RIC Date.L.      Time.L. Type Price volume Bid.Price Ask.Price
1 BHP.AX 2012-11-22 06:10:01      NA 20121122 06:10:01.307511 Quote  NA      NA      33.33      NA
2 BHP.AX 2012-11-22 06:10:01      NA 20121122 06:10:01.307511 Quote  NA      NA      NA      33.42
3 BHP.AX 2012-11-22 07:00:20      NA 20121122 07:00:20.637208 Quote  NA      NA      33.37      NA
4 BHP.AX 2012-11-22 07:00:28      NA 20121122 07:00:28.835146 Quote  NA      NA      NA      33.39
5 BHP.AX 2012-11-22 07:00:30      NA 20121122 07:00:30.756973 Quote  NA      NA      NA      33.20
6 BHP.AX 2012-11-22 07:01:12      NA 20121122 07:01:12.629390 Quote  NA      NA      33.38      NA
7 BHP.AX 2012-11-22 07:01:15      NA 20121122 07:01:15.969108 Quote  NA      NA      33.40      NA
8 BHP.AX 2012-11-22 07:09:07      NA 20121122 07:09:07.114851 Quote  NA      NA      35.88      NA
9 BHP.AX 2012-11-22 08:16:14      NA 20121122 08:16:14.260902 Quote  NA      NA      NA      32.70
10 BHP.AX 2012-11-22 08:21:42      NA 20121122 08:21:42.699427 Quote  NA      NA      NA      31.66
```

## submitFTPRequest

Request data for multiple instruments over multiple days. The request will return a unique request id – the operation `getRequestResult` can be used to retrieve the data or query the status of the request.

### Arguments

<i>rdth</i>	The Tick History credential object
<i>friendlyname</i>	Give the request a name
<i>instrumentList</i> "BHP.AX;RIO.AX;CBA.AX"	A string of RIC codes separated by semi-colons, e.g.
<i>startdate</i>	The request start date, e.g. "2011-11-1"
<i>enddate</i>	The request end date, e.g. "2011-11-30"
<i>starttime</i>	Request start time in GMT, e.g. "00:00:00:000000"
<i>endtime</i>	Request end time in GMT, e.g. "23:59:59:999999"
<i>reqtype</i>	The request type, i.e. "TimeAndSales", "MarketDepth", "NasdaqLevel2", "Intraday", "EndOfDay", "RTCE", "RTCEMarketDepth" and "Any"
<i>mktdepth</i>	Specify a numeric string to indicate whether this is a market depth request e.g. not a market depth request "()"
<i>messagetype</i> <i>list</i>	The message type and field string for the request, in this format "MessageType1:FieldName1,FieldName2;MessageType2:FieldName1,FieldName2", i.e. "End Of Day:Open,High,Low,Last"
<i>disInGMT</i>	Display time in GMT or Local time, i.e. display in GMT "true"
<i>reqInGMT</i>	The request time in GMT or Local time, i.e. request time in GMT "true"

### Method

```
signature(r = "rdth", friendlyname = "character", instrumentList = "character", startdate = "character", enddate = "character", starttime = "character", endtime = "character", reqtype = "character", mktdepth = "character", messagetype = "character", disInGMT = "character", reqInGMT = "character")
```

## Example

```
j<-submitFTPRequest(rdth, "test", instrumentList="BHP.AX;ASX.AX", "2012-11-22", "2011-12-22", "00:00:00", "23:59:59.999", "TimeAndSales", mktdepth="0",
messagetypeList="Trade:Price,Volume;Quote:Bid Price,Ask Price", reqInGMT="false",
disInGMT="false")

> j <- submitFTPRequest(rdth, "test", instrumentList="BHP.AX;ASX.AX", "2012-11-22", "2012-12-22", "00:00:00",
"23:59:59.999", "TimeAndSales", mktdepth="0", messagetypeList="Trade:Price,Volume;Quote:Bid Price,Ask Price",
reqInGMT="false", disInGMT="false")
[1] "Your request id is: kikai.kobayashi@sirca.org.au-test-N56335047"
```

To check the status of your request, use the `getRequestResult` function. E.g. for the given request id above after the request has been completed.

```
> getRequestResult(rdth, "kikai.kobayashi@sirca.org.au-test-N56335047")
[1] "complete"
```

You can retrieve completed FTP requests via the TRTH HTTP Pull Server link on the Thomson Reuters Tick History homepage after you have logged into the product, underneath Support.

For more details on the above commands as well as a full list of all functions, documentation and examples, please view the online Help link through R.



## Message Type Lists

The submitRequest and submitFTPRequest functions use the messagetypeplist argument to recognise the message type and fields being requested.

You can use the getMessageTypes function to retrieve a full list of message types and fields for specific asset class domains. For example, the below table lists the full set of message types and fields available for the Equity domain for Time And Sales data.

```
getMessageTypes(rdth,c("EQU"),"TimeAndSales")
```

	MessageType	FieldList
1	Nominal Value	Nominal Value,Currency,Reference ID Issue Level,Reference PILC,Delete Marker,No Par Value,Corporate Action ID,Nominal Value Date,Nominal Value Scaling Factor
2	Short Sale	Short Price,Short Volume,Short Traded Volume,Short Turnover,Short Weighting,Short Limit,Loan Ask Volume,Loan Ask Amount Trading Price,Percentage Short Volume vs Traded Volume,Percentage Short Price vs Traded Price,Qualifiers
3	Indices and Market Statistics	Exchange Identification,Description,Currency Code,Trading Status,Ticker Symbol,Issue Long Name,ISIN Code,Asset Category,CFI Code,Place of Listing,Primary Reference Market Quote,Market Segment Name,Market MIC,CESR EEA Regulated,Reuters Classification Scheme
4	Issue Level Share Types	Shares Change Date,Thousands of Shares,Shares Amount Scaling Factor,Shares Amount Event ID,PILC,Delete Marker,Shares Amount,Default Share Type,Corporate Action ID
5	Earning	Ann. Date,EPS Currency,EPS Amount,EPS Scaling Factor,EPS General Marker,EPS Gross Marker,EPS Period End Date,EPS Period Length,EPS Period Units,EPS ID,EPS PILC,Delete Marker,EPS Calculation Basis,EPS Basis Number of Shares,Accounting Standard
6	Dividend	Announcement Date,Period End Date,Ex Dividend Date,Registered Payment Date,Div Payment Date,Currency,Dividend Amount,Type Marker,Feature,Frequency,Tax Rate,Tax Marker,Special Tax Rules,Period Length,Period Units,Indicated Annual Amount,Qualified Income Eligibility,Qualified Income Percentage,Foreign Investor Tax Rate,Franking Percentage,Description,Dividend Market Level ID,Delete Marker,Mandatory Voluntary Indicator,Payment Status,Reinvestment Plan Available,Reinvestment Plan Deadline,Reinvestment Price,Source of Fund,Payment Type,Corporate Action ID

7	CapitalChange	Deal Date,Expiration Date,Ex Date,Announcement Date,Market Level ID,Event Type,Terms New Shares,Terms Old Shares,Nominal Value of New Shares,Nominal Value New Scaling Factor,Nominal Value New Shares Currency,Nominal Value Old Shares,Nominal Value Old Scaling Factor,Nominal Value Old Shares Currency,Offer Price,Offer Price Scaling Factor,Offer Price Currency,New Security Identifier,Adjustment Factor,Capital Additive Adj. Factor,Capital Change Issue Level ID,Capital PILC,Cap Delete Marker,Subscription Start,Rights Trading Start,Rights Trading End,Shares Offered,Payment Amount,Payment Currency,Mandatory Voluntary Indicator,Scheme of Arrangement,Effective Date,Rights Identifier,Offer Type,Event Status,Amalgamated description,Issue Renounceable,Stock Different,Odd Lot Size,Acquire Odd Lot,Sell Odd Lot,Retain Odd Lot,Corporate Action ID
8	Mkt. Condition	Qualifiers
9	Trade	Exchange ID,Price,Volume,Market VWAP,Accumulative Volume,Turnover,Buyer ID,Seller ID,Qualifiers,Sequence Number,Exchange Time,Block Trade,Floor Trade,PE Ratio,12 Months EPS,Yield,Implied Volatility,Trade Date,Tick Direction,Dividend Code,Adjusted Close Price,Price Trade-Through-Exempt Flag,Irregular Trade-Through-Exempt Flag,TRF Price Sub Market ID,TRF Irregular Price Sub Market ID,Imputed Close,Volatility,Strike,Premium,Auction Price (unused),Auction Volume (unused),Mid Price,Final Evaluation Price,Provisional Evaluation Price,Percentage Change,Open Price,High Price,Low Price,Ex Dividend Date,Dividend Payment Date,Dividend Amount,Base Price,Original Price,Original Volume,Correction Qualifiers,Contract Physical Units,Minimum quantity of a contract,Number of Physicals
10	Auction	Auction Price,Auction Volume,Qualifiers,Exchange Time
11	Mkt. Statistic	Qualifiers,Exchange Time,Turnover,Advancing Issues,Declining Issues,Unchanged Issues>Total Issues,Advancing Volume,Declining Volume,Unchanged Volume>Total Volume,New Highs,New Lows>Total Moves,Advancing Moves,Declining Moves,Unchanged Moves,Strong Market,Weak Market,Changed Market,Market Volatility,30 Day ATM IV Call,60 Day ATM IV Call,90 Day ATM IV Call,30 Day ATM IV Put,60 Day ATM IV Put,90 Day ATM IV Put
12	Stock Split	Split Factor,Adjust Volume,Split Date,Corporate Action ID
13	Correction	Exchange ID,Original Price,Original Volume,Market VWAP,Buyer ID,Seller ID,Qualifiers,Sequence Number,Exchange Time,PE Ratio,Yield,New Price,New Volume,New Sequence Number,Financial Status,Last,Last Date,Accumulative Volume,Turnover,Original Date
14	Quote	Buyer ID / Exchange ID,Bid Price,Bid Size,Number of Buyers,Seller ID / Exchange ID,Ask Price,Ask Size,Number of Sellers,Qualifiers,Sequence Number,Bid Implied Volatility,Ask Implied Volatility,Quote Date,Quote Time,Bid Tic

15	Warrants	<p>Description,CUSIP Code,SEDOL,Common Code,ISIN Code,Issue Classification,Exchange Identification,Currency Code,Trading Status,Directory Company Name,Industrial Classification Code,Organisation ID ORGID,Organisation Display Name,Organisation Geographical Unit,Australia Code,Austria Code,Belgium Code,France Code Sicovam,Wertpapier WM,Japan Code SICC,Netherlands Code,Rio De Janeiro Code,Sao Paulo Code,Switzerland Code VALOREN,Taiwan Code,Hong Kong Code,Malaysia Code,Singapore Code,Primary Issue Level Code PILC,Finsbury Company Code,ISO CFI Code,Primary RIC Edcoid,New MSCI Industrial Classification Code GICS,Dow Jones Stoxx Industrial Classification Code,FTSE Industrial Classification Code,PE Code,Quotron Symbol,New Belgian Code,Market Identifier Code MIC,Official Place Of Listing OPOL,Asset Categories,Primary Trading RIC,Ticker Symbol,Issue Long Name,RBSS Code,MiFID Indicator,CFI Code,Place Of Listing,Primary Reference Market Quote,Primary Execution Venue,Market Segment Name,Market MIC,Organizational Identifier of the Competent Regulatory Authority,CESR EEA Regulated,CESR Most Relevant Market,CESR Average Daily Turnover,CESR Average Daily Turnover Currency,CRA Average Daily Turnover,CRA Average Daily Turnover Currency,CESR Average Value Of Orders Executed,CESR Average Value Of Orders Executed Currency,CRA Average Value Of Orders Executed,CRA Average Value Of Orders Executed Currency,CESR Free Float,CESR Free Float Currency,CRA_Free Float,CRA Free Float Currency,CESR Standard Market Size,CESR Standard Market Size Currency,Exercise Style,Expiry Date,Put Call indicator,Strike Price,New Netherlands Code,Round Lot Size,Underlying RIC,Reuters Classification Scheme</p>
16	Exchange Level Share Types	<p>Shares Change Date,Share Type,Default Share Type Flag,Shares Amount,Shares Amount Event ID,PILC&gt;Delete Marker,Corporate Action ID</p>

17	Equities	Description,CUSIP Code,SEDOL,Common Code,ISIN Code,Issue Classification,Exchange Identification,Currency Code,Trading Status,Directory Company Name,Industrial Classification Code,Organisation ID ORGID,Organisation Display Name,Organisation Geographical Unit,Australia Code,Austria Code,Belgium Code,France Code Sicovam,Wertpapier WM,Japan Code SICC,Netherlands Code,Rio De Janeiro Code,Sao Paulo Code,Switzerland Code VALOREN,Taiwan Code,Hong Kong Code,Malaysia Code,Singapore Code,Primary Issue Level Code PILC,Finsbury Company Code,ISO CFI Code,Primary RIC Edcoid,New MSCI Industrial Classification Code GICS,Dow Jones Stoxx Industrial Classification Code,FTSE Industrial Classification Code,PE Code,Quotron Symbol,New Belgian Code,Market Identifier Code MIC,Official Place Of Listing OPOL,Asset Categories,Primary Trading RIC,Ticker Symbol,Issue Long Name,RBSS Code,MiFID Indicator,CFI Code,Place Of Listing,Primary Reference Market Quote,Primary Execution Venue,Market Segment Name,Market MIC,Organizational Identifier of the Competent Regulatory Authority,CESR EEA Regulated,CESR Most Relevant Market,CESR Average Daily Turnover,CESR Average Daily Turnover Currency,CRA Average Daily Turnover,CRA Average Daily Turnover Currency,CESR Average Value Of Orders Executed,CESR Average Value Of Orders Executed Currency,CRA Average Value Of Orders Executed,CRA Average Value Of Orders Executed Currency,CESR Free Float,CESR Free Float Currency,CRA Free Float,CRA Free Float Currency,CESR Standard Market Size,CESR Standard Market Size Currency,New Netherlands Code,Round Lot Size,Reuters Classification Scheme
18	Symbology Change	Change Type,Old Value,New Value
19	Reference Change	Change Type,Old Value,New Value
20	Order Imbalance	Reference Price,Paired Quantity,Qualifiers,Closing Reference Price,Imbalance Quantity,Far Clearing Price,Near Clearing Price
21	Raw	All Transactions

Refer to the `getMessageTypes` help section for more details and a full list of asset class domains.

## RTime

You will notice that there is an additional column RTime. The RTime is a POSIXlt date-time object which allows you to perform addition, subtraction and comparison of different date-time objects, which otherwise would be very difficult to do with the Time.G character string field.

By default, the RTime column only displays precision to the second, however to display millisecond or microseconds you can set the digits.secs option to indicate the number of digits to display after the second.

*options(digits.secs=6)*

RTime displaying millisecond precision (3 decimal places):

```
> options(digits.secs=3)
> j[1:2,]
      RIC      RTime Current.RIC Date.L.      Time.L. Type Price Volume Bid.Price Ask.Price
1 BHP.AX 2012-11-22 06:10:01.307      NA 20121122 06:10:01.307511 Quote      NA      NA      33.33      NA
2 BHP.AX 2012-11-22 06:10:01.307      NA 20121122 06:10:01.307511 Quote      NA      NA      NA      33.42
```

RTime displaying microsecond precision (6 decimal places):

```
> options(digits.secs=6)
> j[1:2,]
      RIC      RTime Current.RIC Date.L.      Time.L. Type Price Volume Bid.Price Ask.Price
1 BHP.AX 2012-11-22 06:10:01.307511      NA 20121122 06:10:01.307511 Quote      NA      NA      33.33      NA
2 BHP.AX 2012-11-22 06:10:01.307511      NA 20121122 06:10:01.307511 Quote      NA      NA      NA      33.42
```

An example of comparison between RTimes, which will allow you to manipulate the data more easily.

```
> j[1000:1001,]
      RIC      RTime Current.RIC Date.L.      Time.L. Type Price Volume Bid.Price Ask.Price
1000 BHP.AX 2012-11-22 10:10:49.976484      NA 20121122 10:10:49.976484 Quote      NA      NA      NA      33.61
1001 BHP.AX 2012-11-22 10:10:51.835766      NA 20121122 10:10:51.835766 Trade 33.59      225      NA      NA
> j[1000,"RTime"]<j[1001,"RTime"]
[1] TRUE
> j[1000,"RTime"]>j[1001,"RTime"]
[1] FALSE
```

## Help Files

Each of the Tick History functions is documented in R's help file. To see the list of functions, type the following: (Make sure you have loaded the THAPI library first by calling the library function, see above)

```
??rdth
```

This gives you the list of the functions available in the THAPI package.

You can get more help from each function by typing question mark before the function name. For example, if you would like to get more information on the `getBondTypes` function, type:

```
?getBondTypes
```