

Datafeed Toolbox™ Release Notes



MATLAB®



How to Contact MathWorks



Latest news: www.mathworks.com
Sales and services: www.mathworks.com/sales_and_services
User community: www.mathworks.com/matlabcentral
Technical support: www.mathworks.com/support/contact_us



Phone: 508-647-7000



The MathWorks, Inc.
1 Apple Hill Drive
Natick, MA 01760-2098

Datafeed Toolbox™ Release Notes

© COPYRIGHT 2004–2023 by The MathWorks, Inc.

The software described in this document is furnished under a license agreement. The software may be used or copied only under the terms of the license agreement. No part of this manual may be photocopied or reproduced in any form without prior written consent from The MathWorks, Inc.

FEDERAL ACQUISITION: This provision applies to all acquisitions of the Program and Documentation by, for, or through the federal government of the United States. By accepting delivery of the Program or Documentation, the government hereby agrees that this software or documentation qualifies as commercial computer software or commercial computer software documentation as such terms are used or defined in FAR 12.212, DFARS Part 227.72, and DFARS 252.227-7014. Accordingly, the terms and conditions of this Agreement and only those rights specified in this Agreement, shall pertain to and govern the use, modification, reproduction, release, performance, display, and disclosure of the Program and Documentation by the federal government (or other entity acquiring for or through the federal government) and shall supersede any conflicting contractual terms or conditions. If this License fails to meet the government's needs or is inconsistent in any respect with federal procurement law, the government agrees to return the Program and Documentation, unused, to The MathWorks, Inc.

Trademarks

MATLAB and Simulink are registered trademarks of The MathWorks, Inc. See www.mathworks.com/trademarks for a list of additional trademarks. Other product or brand names may be trademarks or registered trademarks of their respective holders.

Patents

MathWorks products are protected by one or more U.S. patents. Please see www.mathworks.com/patents for more information.

R2023a

Refinitiv Datascope Select: Access historical and current market data	1-2
---	------------

R2022b

No New Features or Changes

R2022a

Bug Fixes

R2021b

Money.Net Web Socket Interface: Access market data and news stories	4-2
---	------------

R2021a

Bloomberg C++ interfaces	5-2
Functionality moving to File Exchange	5-2
Functionality being removed or changed	5-2
datastream object has been removed	5-2
fetch function has been removed	5-3
get function has been removed	5-3
isconnection function has been removed	5-4

close function has been removed	5-4
rdth object has been removed	5-5
fetch function has been removed	5-5
get function has been removed	5-6
isconnection function has been removed	5-7
status function has been removed	5-7
submitftp function has been removed	5-8
rdthloader function has been removed	5-8
close function has been removed	5-9

R2020b

Bug Fixes

R2020a

Functionality being removed or changed	7-2
datastream object will be removed	7-2
fetch function will be removed	7-2
get function will be removed	7-3
isconnection function will be removed	7-3
close function will be removed	7-4

R2019b

Bug Fixes

R2019a

Bug Fixes

R2018b

Quandl Interface: Access historical market data	10-2
--	------

IHS Markit Interface: Access factor data	10-2
Datastream Web Services Interface: Access historical market data	10-2
FRED, Haver Analytics, and RavenPack News Analytics Data Return Formatting: Set new properties to control the output data format . .	10-2
Functionality being removed or changed	10-2
datastream object will be removed	10-2
fetch function will be removed	10-3
get function will be removed	10-3
isconnection function will be removed	10-4
close function will be removed	10-4

R2018a

Bloomberg Data Return Formatting: Set new properties to control output data format	11-2
Thomson Reuters Tick History Interface: Access historical and intraday market data	11-2

R2017b

Twitter Support: Easily connect to Twitter, aggregate data for analysis, and access REST API endpoints	12-2
Functionality Being Removed or Changed	12-2

R2017a

Elektron Interface: Access current and real-time data directly from Elektron	13-2
Yahoo! removed	13-2

R2016b

Money.Net Interface: Access historical, real-time, and news data directly from MATLAB	14-2
STATS.com Interface: Access team, athlete, and event data	14-2
Functionality being removed or changed	14-2

R2016a

Interactive Data removed	15-2
Thomson Reuters Eikon removed	15-2
Functionality being removed or changed	15-2

R2015b

Bloomberg portfolio Function: Access portfolio holding information ..	16-2
RavenPack News Analytics Interface: Access historical and real-time news sentiment data	16-2

R2015a

Bloomberg Data License support	17-2
---	-------------

R2014b

Data access through Thomson Reuters Eikon	18-2
Data access through Bloomberg Server	18-2
Data access through Bloomberg B-PIPE	18-2

R2014a

Lookup functionality for Bloomberg Instrument service	19-2
--	-------------

R2013b

Multiple security request performance improvements for Bloomberg history function	20-2
--	-------------

R2013a

Function for accessing real-time data from FactSet workstations	21-2
X_TRADER functions, originally in Datafeed Toolbox, now in Trading Toolbox	21-2
Bloomberg legacy functions removed	21-2

R2012b

Functions for accessing data through the IQFEED service	22-2
Support for Bloomberg Historical Technical Analysis	22-2
Support for Bloomberg Equity Screening	22-2

R2012a

Support for Trading Technologies X_TRADER	23-2
64-Bit Support for Haver Analytics	23-2
Broker Buy and Sell Codes Added to Bloomberg Intraday Output	23-2
Timeout Argument for blp	23-2

R2011b

Support for SIX Financial Information	24-2
Improved Performance for blp Methods	24-2
blp.history Now Supports Overrides	24-2
Warning and Error ID Changes	24-2

R2011a

New rdth Methods for FTP Requests to Thomson Reuters Tick History	25-2
---	------

R2010b

Support for eSignal Data Retrieval	26-2
New blp Methods Return Information About Bloomberg V3 Connection	26-2
blp.timeseries Method Now Returns Total Value	26-2
Support for Reuters Publishing	26-2
New rmdsconfig Function to Start Reuters Configuration Editor	26-2
Reuters fetch Function Callbacks Now Require Three Inputs	26-2
Proxy Information No Longer Required for yahoo Function	26-2

R2010a

Support for Bloomberg Version 3	27-2
Bloomberg Pre-V3 Access Not Supported	27-2

R2009b

Support for New Access Methods	28-2
---	-------------

R2009a

Support for Retrieving New Types of Data	29-2
---	-------------

R2008b

Support for Retrieving New Types of Data	30-2
---	-------------

R2008a

Datafeed Toolbox Software Support for Hyperfeed Data Service Discontinued	31-2
--	-------------

R2007b

Reuters Market Data System Support	32-2
---	-------------

R2007a

Kx Systems, Inc. kdb+ Database Support	33-2
Haver Analytics Data Support	33-2

R2006b

Federal Reserve Economic Data	34-2
Additional Fields Added for Yahoo! Support	34-2

R2006a

Thomson Datastream Support Added	35-2
Thomson Datastream Function Summary	35-2

R14SP3

Enhanced Capabilities for Bloomberg fetch Command	36-2
--	------

R14SP2

FactSet Data Service Support Added	37-2
FactSet Function Summary	37-2

R2023a

Version: 6.4

New Features

Bug Fixes

Refinitiv Datascope Select: Access historical and current market data

Create a Refinitiv™ Datascope Select connection using the `datascopeSelect` function. You can retrieve the most recent end-of-day, intra-day, and timeseries market data. After retrieving data, you can make investment decisions related to optimization of risk and return of asset portfolios or forecast economic trends related to different markets.

R2022b

Version: 6.3

No New Features or Changes

R2022a

Version: 6.2

Bug Fixes

R2021b

Version: 6.1

New Features

Bug Fixes

Money.Net Web Socket Interface: Access market data and news stories

With the Money.Net web socket interface, you can retrieve current, intraday, historical, and real-time market data from Money.Net. You can also retrieve Money.Net news stories. For details, see Money.Net Web Socket Interface.

R2021a

Version: 6.0

New Features

Bug Fixes

Compatibility Considerations

Bloomberg C++ interfaces

You can connect to Bloomberg® Desktop, Bloomberg Server, Bloomberg B-PIPE®, and Bloomberg EMSX using the corresponding Bloomberg C++ interfaces. For details about each of these interfaces, see:

- Bloomberg Desktop C++ Interface
- Bloomberg B-PIPE C++ Interface
- Bloomberg Server C++ Interface
- Bloomberg EMSX C++ Interface

Functionality moving to File Exchange

Functionality for creating connections to the following data providers has moved to File Exchange on MATLAB® Central™. You can access the documentation from the prior release.

Provider	Archived Documentation
Bloomberg Data License	https://www.mathworks.com/help/releases/R2020b/datafeed/bdl.html
FactSet®	https://www.mathworks.com/help/releases/R2020b/datafeed/factset-1.html
IQFEED®	https://www.mathworks.com/help/releases/R2020b/datafeed/iqfeed.html
Kx Systems®, Inc.	https://www.mathworks.com/help/releases/R2020b/datafeed/kx-systems-inc-.html
RavenPack® News Analytics	https://www.mathworks.com/help/releases/R2020b/datafeed/ravenpack-1.html
Elektron™ from Refinitiv	https://www.mathworks.com/help/releases/R2020b/datafeed/elektron-1.html
Enterprise Platform from Refinitiv	https://www.mathworks.com/help/releases/R2020b/datafeed/enterprise-platform.html
STATS.com	https://www.mathworks.com/help/releases/R2020b/datafeed/stats-com.html
Interactive Brokers®	https://www.mathworks.com/help/releases/R2020b/trading/interactive-brokers.html
FIX Flyer™	https://www.mathworks.com/help/releases/R2020b/trading/fix-flyer.html

Functionality being removed or changed

datastream object has been removed

The `datastream` object has been removed. Use the `datastreamws` object instead.

Some differences between the workflows require updates to your code.

Update Code

Use the `datastreamws` function to create a Datastream™ Web Services from Refinitiv connection.

In prior releases, you created a `datastream` object by writing code similar to the following:

```
username = 'DS:username';  
password = 'password';  
source = 'Datastream';  
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';  
  
c = datastream(username,password,source,url);
```

Now specify only the user name and password with the `datastreamws` function.

```
username = 'ABCDEF';  
password = 'abcdef12345';  
c = datastreamws(username,password);
```

fetch function has been removed

The `fetch` function has been removed. Use the `history` function instead.

Some differences between the workflows require updates to your code.

Update Code

Use the `history` function to retrieve Datastream Web Services from Refinitiv historical data.

In prior releases, you created a `datastream` object and retrieved data by writing code similar to the following:

```
username = 'DS:username';  
password = 'password';  
source = 'Datastream';  
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';  
  
c = datastream(username,password,source,url);  
data = fetch(Connect, 'ICI', {'P', 'P0'}, '09/01/2007');
```

Now use the `datastreamws` and `history` functions instead.

```
username = 'ABCDEF';  
password = 'abcdef12345';  
c = datastreamws(username,password);  
  
sec = 'VOD';  
d = history(c,sec);
```

get function has been removed

The `get` function has been removed without replacement.

Some differences between the workflows require updates to your code.

Update Code

Use the `datastreamws` function to create a Datastream Web Services from Refinitiv connection.

In prior releases, you created a `datastream` object and retrieved properties by writing code similar to the following:

```
username = 'DS:username';  
password = 'password';  
source = 'Datastream';  
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';  
  
c = datastream(username,password,source,url);  
value = get(c);
```

Now specify only the user name and password with the `datastreamws` function.

```
username = 'ABCDEF';  
password = 'abcdef12345';  
c = datastreamws(username,password);
```

There is no replacement functionality for the `get` function. To access the properties of the `datastreamws` object, use dot notation.

isconnection function has been removed

The `isconnection` function has been removed without replacement.

Some differences between the workflows require updates to your code.

Update Code

Use the `datastreamws` function to create a Datastream Web Services from Refinitiv connection.

In prior releases, you created a `datastream` object and verified the connection by writing code similar to the following:

```
username = 'DS:username';  
password = 'password';  
source = 'Datastream';  
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';  
  
c = datastream(username,password,source,url);  
x = isconnection(c);
```

Now specify only the user name and password with the `datastreamws` function.

```
username = 'ABCDEF';  
password = 'abcdef12345';  
c = datastreamws(username,password);
```

There is no replacement functionality for the `isconnection` function.

close function has been removed

The `close` function has been removed. There is no replacement for the `close` function.

Some differences between the workflows require updates to your code.

Update Code

Use the `datastreamws` function to create a Datastream Web Services from Refinitiv connection.

In prior releases, you created a `datastream` object and closed the Datastream connection by writing code similar to the following:

```
username = 'DS:username';
password = 'password';
source = 'Datastream';
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';

c = datastream(username,password,source,url);
close(c)
```

Now specify only the user name and password with the `datastreamws` function.

```
username = 'ABCDEF';
password = 'abcdef12345';
c = datastreamws(username,password);
```

There is no replacement functionality for the `close` function.

rdth object has been removed

The `rdth` object has been removed. Use the `trth` object instead.

Some differences between the workflows require updates to your code.

Update Code

Use the `trth` function to create a Tick History from Refinitiv connection.

In prior releases, you created a `rdth` object and closed the Datastream connection by writing code similar to the following:

```
username = 'user@company.com';
password = 'mypassword';

c = rdth(username,password);
close(c)
```

Now specify only the user name and password with the `trth` function.

```
username = 'username';
password = 'password';
c = trth(username,password);
```

fetch function has been removed

The `fetch` function has been removed. Use the `history` or `timeseries` functions instead.

Some differences between the workflows require updates to your code.

Update Code

Use the `history` or `timeseries` functions to retrieve data using a Tick History from Refinitiv connection.

In prior releases, you retrieved data using the `fetch` function by writing code similar to the following:

```
username = 'user@company.com';
password = 'mypassword';

c = rdth(username,password);
x = fetch(c,'ABCD.0',{ 'Exchange ID','Price','Volume'}, ...
        {'09/05/2008 12:00:06','09/05/2008 12:00:10'}, ...
        'TimeAndSales','Trade','NSQ','EQU');
close(c)
```

Now use the `history` function to retrieve historical data.

```
username = 'username';
password = 'password';
c = trth(username,password);

sec = ["IBM.N","Ric"];
fields = ["Open","Last"];
startdate = datetime('yesterday');
enddate = datetime('today');
d = history(c,sec,fields,startdate,enddate);
```

Or, use the `timeseries` function to retrieve intraday data.

```
username = 'username';
password = 'password';
c = trth(username,password);

sec = ["IBM.N","Ric"];
fields = ["Trade - Exchange Time","Trade - Price","Trade - Volume"];
startdate = datetime('11/06/2017','InputFormat','MM/dd/yyyy');
enddate = datetime('11/07/2017','InputFormat','MM/dd/yyyy');
d = timeseries(c,sec,fields,startdate,enddate);
```

get function has been removed

The `get` function has been removed without replacement.

Some differences between the workflows require updates to your code.

Update Code

Use the `trth` function to create a Tick History from Refinitiv connection.

In prior releases, you retrieved the properties of the connection by writing code similar to the following:

```
username = 'user@company.com';
password = 'mypassword';
```

```
c = rdth(username,password);
v = get(c);
close(c)
```

Now specify only the user name and password with the `trth` function.

```
username = 'username';
password = 'password';
c = trth(username,password);
```

There is no replacement functionality for the `get` function. To access the properties of the `trth` object, use dot notation.

isconnection function has been removed

The `isconnection` function has been removed without replacement.

Some differences between the workflows require updates to your code.

Update Code

Use the `trth` function to create a Tick History from Refinitiv connection.

In prior releases, you verified the connection by writing code similar to the following:

```
username = 'user@company.com';
password = 'mypassword';

c = rdth(username,password);
x = isconnection(c);
close(c)
```

Now specify only the user name and password with the `trth` function.

```
username = 'username';
password = 'password';
c = trth(username,password);
```

There is no replacement functionality for the `isconnection` function.

status function has been removed

The `status` function has been removed without replacement.

Some differences between the workflows require updates to your code.

Update Code

Use the `trth` function to create a Tick History from Refinitiv connection.

In prior releases, you checked the status of your FTP request by writing code similar to the following:

```
username = 'user@company.com';
password = 'mypassword';

c = rdth(username,password);
```

```
x = submitftp(c, 'GOOG.O', {'Exchange ID', 'Price', 'Volume'}, ...
    {(floor(now))-10, (floor(now))}, 'TimeAndSales', 'Trade', ...
    'NSQ', 'EQU')
s = [];
while ~strcmp(s, 'Complete')
[s, qp] = status(c, x);
end
close(c)
```

Now specify only the user name and password with the `trth` function to create a connection.

```
username = 'username';
password = 'password';
c = trth(username, password);
```

There is no replacement functionality for the `status` function.

submitftp function has been removed

The `submitftp` function has been removed without replacement.

Some differences between the workflows require updates to your code.

Update Code

Use the `trth` function to create a Tick History from Refinitiv connection.

In prior releases, you submitted an FTP request by writing code similar to the following:

```
username = 'user@company.com';
password = 'mypassword';

c = rdth(username, password);
x = submitftp(r, 'GOOG.O', {'Exchange ID', 'Price', 'Volume'}, ...
    {(floor(now))-10, (floor(now))}, 'TimeAndSales', 'Trade', ...
    'NSQ', 'EQU')
close(c)
```

Now specify only the user name and password with the `trth` function to create a connection.

```
username = 'username';
password = 'password';
c = trth(username, password);
```

There is no replacement functionality for the `submitftp` function.

rdthloader function has been removed

The `rdthloader` function has been removed without replacement.

Some differences between the workflows require updates to your code.

Update Code

Use the `trth` function to create a Tick History from Refinitiv connection.

In prior releases, you retrieved data from a Tick History file by writing code similar to the following:

```
x = rdthloader('file.csv','date',{ '02/02/2007' });
```

Now use the `history` function to retrieve historical data.

```
username = 'username';
password = 'password';
c = trth(username,password);

sec = ["IBM.N","Ric"];
fields = ["Open","Last"];
startdate = datetime('yesterday');
enddate = datetime('today');
d = history(c,sec,fields,startdate,enddate);
```

Or, use the `timeseries` function to retrieve intraday data.

```
username = 'username';
password = 'password';
c = trth(username,password);

sec = ["IBM.N","Ric"];
fields = ["Trade - Exchange Time","Trade - Price","Trade - Volume"];
startdate = datetime('11/06/2017','InputFormat','MM/dd/yyyy');
enddate = datetime('11/07/2017','InputFormat','MM/dd/yyyy');
d = timeseries(c,sec,fields,startdate,enddate);
```

There is no replacement functionality for the `rdthloader` function.

close function has been removed

The `close` function has been removed without replacement.

Some differences between the workflows require updates to your code.

Update Code

Use the `trth` function to create a Tick History from Refinitiv connection.

In prior releases, you created a `rdth` object and closed the Datastream connection by writing code similar to the following:

```
username = 'user@company.com';
password = 'mypassword';

c = rdth(username,password);
close(c)
```

Now specify only the user name and password with the `trth` function.

```
username = 'username';
password = 'password';
c = trth(username,password);
```

There is no replacement functionality for the `close` function.

R2020b

Version: 5.9.2

Bug Fixes

R2020a

Version: 5.9.1

Bug Fixes

Compatibility Considerations

Functionality being removed or changed

datastream object will be removed

Warns

The `datastream` object will be removed in a future release. Use the `datastreamws` object instead.

Some differences between the workflows require updates to your code.

Update Code

Use the `datastreamws` function to create a Datastream Web Services from Refinitiv connection.

In prior releases, you created a `datastream` object by writing code similar to the following:

```
username = 'DS:username';  
password = 'password';  
source = 'Datastream';  
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';  
  
c = datastream(username,password,source,url);
```

Now specify only the user name and password with the `datastreamws` function.

```
username = 'ABCDEF';  
password = 'abcdef12345';  
c = datastreamws(username,password);
```

fetch function will be removed

Warns

The `fetch` function will be removed in a future release. Use the `history` function instead.

Some differences between the workflows require updates to your code.

Update Code

Use the `history` function to retrieve Datastream Web Services from Refinitiv historical data.

In prior releases, you created a `datastream` object and retrieved data by writing code similar to the following:

```
username = 'DS:username';  
password = 'password';  
source = 'Datastream';  
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';  
  
c = datastream(username,password,source,url);  
data = fetch(Connect, 'ICI', {'P', 'P0'}, '09/01/2007');
```

Now use the `datastreamws` and `history` functions instead.

```
username = 'ABCDEF';  
password = 'abcdef12345';  
c = datastreamws(username,password);  
  
sec = 'VOD';  
d = history(c,sec);
```

get function will be removed

Warns

The `get` function will be removed in a future release.

Some differences between the workflows require updates to your code.

Update Code

Use the `datastreamws` function to create a Datastream Web Services from Refinitiv connection.

In prior releases, you created a `datastream` object and retrieved properties by writing code similar to the following:

```
username = 'DS:username';
password = 'password';
source = 'Datastream';
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';

c = datastream(username,password,source,url);
value = get(c);
```

Now specify only the user name and password with the `datastreamws` function.

```
username = 'ABCDEF';
password = 'abcdef12345';
c = datastreamws(username,password);
```

There is no replacement functionality for the `get` function. To access the properties of the `datastreamws` object, use dot notation.

isconnection function will be removed

Warns

The `isconnection` function will be removed in a future release without replacement.

Some differences between the workflows require updates to your code.

Update Code

Use the `datastreamws` function to create a Datastream Web Services from Refinitiv connection.

In prior releases, you created a `datastream` object and verified the connection by writing code similar to the following:

```
username = 'DS:username';
password = 'password';
source = 'Datastream';
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';

c = datastream(username,password,source,url);
x = isconnection(c);
```

Now specify only the user name and password with the `datastreamws` function.

```
username = 'ABCDEF';
password = 'abcdef12345';
c = datastreamws(username,password);
```

There is no replacement functionality for the `isconnection` function.

close function will be removed

Warns

The `close` function will be removed in a future release. There is no replacement for the `close` function.

Some differences between the workflows require updates to your code.

Update Code

Use the `datastreamws` function to create a Datastream Web Services from Refinitiv connection.

In prior releases, you created a `datastream` object and closed the Datastream connection by writing code similar to the following:

```
username = 'DS:username';  
password = 'password';  
source = 'Datastream';  
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';  
  
c = datastream(username,password,source,url);  
close(c)
```

Now specify only the user name and password with the `datastreamws` function.

```
username = 'ABCDEF';  
password = 'abcdef12345';  
c = datastreamws(username,password);
```

There is no replacement functionality for the `close` function.

R2019b

Version: 5.9

Bug Fixes

R2019a

Version: 5.8.1

Bug Fixes

R2018b

Version: 5.8

New Features

Bug Fixes

Compatibility Considerations

Quandl Interface: Access historical market data

Create a Quandl® connection using the `quandl` function. You can retrieve historical market data. After retrieving data, you can make investment decisions. For details, see [Quandl](#) .

IHS Markit Interface: Access factor data

Create an IHS Markit® connection using the `ihsmarkitrs` function, and then retrieve factor, security, universe, and signals data. After retrieving data, you can use it for portfolio selection or in your existing models. For details, see [IHS Markit](#) .

Datastream Web Services Interface: Access historical market data

Create a Datastream Web Services connection using the `datastreamws` function. You can retrieve historical market data. After retrieving data, you can make investment decisions. For details, see [Datastream Web Services](#).

FRED, Haver Analytics, and RavenPack News Analytics Data Return Formatting: Set new properties to control the output data format

The `fred` and `haver` connection objects have two new properties, `DataReturnFormat` and `DatetimeType`. The `ravenpack` connection object has the new `DataReturnFormat` property. When you execute [FRED](#)®, [Haver Analytics](#)®, and [RavenPack News Analytics](#) functions, use the `DataReturnFormat` property to control the data type of the returned data. The `DatetimeType` property controls the data type for dates and times in the returned data. For details about these properties, see the `fred`, `haver`, and `ravenpack` objects.

Functionality being removed or changed

datastream object will be removed

Still runs

The `datastream` object will be removed in a future release. Use the `datastreamws` object instead.

Some differences between the workflows require updates to your code.

Update Code

Use the `datastreamws` function to create a Datastream web services connection.

In prior releases, you created a `datastream` object by writing code similar to the following:

```
username = 'DS:username';  
password = 'password';  
source = 'Datastream';  
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';  
  
c = datastream(username,password,source,url);
```

Now specify only the user name and password with the `datastreamws` function.

```
username = 'ABCDEF';
password = 'abcdef12345';
c = datastreamws(username,password);
```

fetch function will be removed

Still runs

The `fetch` function will be removed in a future release. Use the `history` function instead.

Some differences between the workflows require updates to your code.

Update Code

Use the `history` function to retrieve Datastream web services historical data.

In prior releases, you created a `datastream` object and retrieved data by writing code similar to the following:

```
username = 'DS:username';
password = 'password';
source = 'Datastream';
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';

c = datastream(username,password,source,url);
data = fetch(Connect, 'ICI', {'P', 'PO'}, '09/01/2007');
```

Now use the `datastreamws` and `history` functions instead.

```
username = 'ABCDEF';
password = 'abcdef12345';
c = datastreamws(username,password);

sec = 'VOD';
d = history(c,sec);
```

get function will be removed

Still runs

The `get` function will be removed in a future release.

Some differences between the workflows require updates to your code.

Update Code

Use the `datastreamws` function to create a Datastream web services connection.

In prior releases, you created a `datastream` object and retrieved properties by writing code similar to the following:

```
username = 'DS:username';
password = 'password';
source = 'Datastream';
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';

c = datastream(username,password,source,url);
value = get(c);
```

Now specify only the user name and password with the `datastreamws` function.

```
username = 'ABCDEF';  
password = 'abcdef12345';  
c = datastreamws(username,password);
```

There is no replacement functionality for the `get` function. To access the properties of the `datastreamws` object, use dot notation.

isconnection function will be removed

Still runs

The `isconnection` function will be removed in a future release without replacement.

Some differences between the workflows require updates to your code.

Update Code

Use the `datastreamws` function to create a Datastream web services connection.

In prior releases, you created a `datastream` object and verified the connection by writing code similar to the following:

```
username = 'DS:username';  
password = 'password';  
source = 'Datastream';  
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';  
  
c = datastream(username,password,source,url);  
x = isconnection(c);
```

Now specify only the user name and password with the `datastreamws` function.

```
username = 'ABCDEF';  
password = 'abcdef12345';  
c = datastreamws(username,password);
```

There is no replacement functionality for the `isconnection` function.

close function will be removed

Still runs

The `close` function will be removed in a future release. There is no replacement for the `close` function.

Some differences between the workflows require updates to your code.

Update Code

Use the `datastreamws` function to create a Datastream web services connection.

In prior releases, you created a `datastream` object and closed the Thomson Reuters® Datastream connection by writing code similar to the following:

```
username = 'DS:username';  
password = 'password';  
source = 'Datastream';  
url = 'http://dataworks.thomson.com/Dataworks/Enterprise/1.0';  
  
c = datastream(username,password,source,url);  
close(c)
```

Now specify only the user name and password with the `datastreamws` function.

```
username = 'ABCDEF';  
password = 'abcdef12345';  
c = datastreamws(username,password);
```

There is no replacement functionality for the `close` function.

R2018a

Version: 5.7

New Features

Compatibility Considerations

Bloomberg Data Return Formatting: Set new properties to control output data format

The Bloomberg connection objects have two new properties, `DataReturnFormat` and `DatetimeType`. When you execute a Bloomberg function, use the `DataReturnFormat` property to control the data type of the returned data. The `DatetimeType` property controls the data type for dates in the returned data. For details, see these properties in the `blp`, `blpsrv`, and `bpipe` functions.

Thomson Reuters Tick History Interface: Access historical and intraday market data

Create a Thomson Reuters Tick History connection using the `trth` function, and then retrieve historical and intraday market data. After retrieving data, you can make investment decisions. For details, see Thomson Reuters Tick History.

Compatibility Considerations

This table lists all Thomson Reuters Tick History functionality that will be removed in a future release. Use the functionality of the new interface instead.

Functionality	Result	Use Instead	Compatibility Considerations
<code>rdth</code>	Still runs	<code>trth</code>	Replace all instances of the <code>rdth</code> function with the <code>trth</code> function.
<code>fetch</code>	Still runs	<code>history</code> or <code>timeseries</code>	Replace all instances of the <code>fetch</code> function with either the <code>history</code> or <code>timeseries</code> functions.
<code>get</code>	Still runs	Nothing	No replacement
<code>isconnection</code>	Still runs	Nothing	No replacement
<code>status</code>	Still runs	Nothing	No replacement
<code>submitftp</code>	Still runs	Nothing	No replacement
<code>rdthloader</code>	Still runs	Nothing	No replacement
<code>close</code>	Still runs	Nothing	No replacement

R2017b

Version: 5.6

New Features

Compatibility Considerations

Twitter Support: Easily connect to Twitter, aggregate data for analysis, and access REST API endpoints

Create a Twitter® connection using the `twitter` function. You can retrieve historical Tweets and other data, perform sentiment analysis, and post data to Twitter. Also, you can access any Twitter REST API endpoint. For details, see [Twitter](#) .

Functionality Being Removed or Changed

This table lists all Thomson Reuters Tick History functionality that will be removed in a future release.

Functionality	Result	Use Instead	Compatibility Considerations
<code>rdth</code>	Still runs	Nothing	No replacement
<code>fetch</code>	Still runs	Nothing	No replacement
<code>get</code>	Still runs	Nothing	No replacement
<code>isconnection</code>	Still runs	Nothing	No replacement
<code>status</code>	Still runs	Nothing	No replacement
<code>submitftp</code>	Still runs	Nothing	No replacement
<code>rdthloader</code>	Still runs	Nothing	No replacement
<code>close</code>	Still runs	Nothing	No replacement

R2017a

Version: 5.5

New Features

Bug Fixes

Compatibility Considerations

Elektron Interface: Access current and real-time data directly from Elektron

Create a connection to Elektron using the `elektron` function. Retrieve current and real-time data using MATLAB functionality. For details, see <https://www.mathworks.com/help/releases/R2017a/datafeed/thomson-reuters-elektron.html>.

Yahoo! removed

All Yahoo!® functionality and the Datafeed dialog box have been removed.

Compatibility Considerations

These functions have no replacement: `yahoo`, `builduniverse`, `fetch`, `isconnection`, and `close`.

R2016b

Version: 5.4

New Features

Bug Fixes

Compatibility Considerations

Money.Net Interface: Access historical, real-time, and news data directly from MATLAB

Create a connection to Money.Net using `money.net`. Retrieve current, intraday, historical, real-time, and news data using MATLAB functionality. For details, see Money.Net.

STATS.com Interface: Access team, athlete, and event data

Retrieve current and historical team, athlete, and event data from STATS.com using MATLAB functionality. For details, see STATS.com.

Functionality being removed or changed

This table lists all eSignal® functionality that has been removed.

Functionality	What Happens When You Use It?	Use This Instead	Compatibility Considerations
<code>esig</code>	Errors	Nothing	No replacement
<code>getdata</code>	Errors	Nothing	No replacement
<code>getfundamentaldata</code>	Errors	Nothing	No replacement
<code>history</code>	Errors	Nothing	No replacement
<code>timeseries</code>	Errors	Nothing	No replacement
<code>close</code>	Errors	Nothing	No replacement

R2016a

Version: 5.3

Bug Fixes

Compatibility Considerations

Interactive Data removed

All Interactive Data™ functionality has been removed because MATLAB no longer supports 32-bit Windows®. Accessing the Interactive Data RemotePlus^(SM) data using the Datafeed dialog box is no longer supported.

Compatibility Considerations

These functions have no replacement: `idc`, `fetch`, `get`, `isconnection`, and `close`.

Thomson Reuters Eikon removed

All Thomson Reuters Eikon® functionality has been removed because MATLAB no longer supports 32-bit Windows.

Compatibility Considerations

These functions have no replacement: `treikon`, `getdata`, `history`, `realtime`, `start`, `stop`, and `chain`.

Functionality being removed or changed

This table lists all eSignal functionality that is being removed in a future release.

Functionality	What Happens When You Use It?	Use This Instead	Compatibility Considerations
<code>esig</code>	Warns	Nothing	No replacement
<code>getdata</code>	Warns	Nothing	No replacement
<code>getfundamentaldata</code>	Warns	Nothing	No replacement
<code>history</code>	Warns	Nothing	No replacement
<code>timeseries</code>	Warns	Nothing	No replacement
<code>close</code>	Warns	Nothing	No replacement

R2015b

Version: 5.2

New Features

Bug Fixes

Bloomberg portfolio Function: Access portfolio holding information

Request current portfolio data for Bloomberg using portfolio.

RavenPack News Analytics Interface: Access historical and real-time news sentiment data

Retrieve intraday, historical, and real-time news event data using RavenPack News Analytics.

Function	Purpose
ravenpack	Connect to RavenPack News Analytics.
entitlements	Retrieve RavenPack News Analytics Data Gateway entitlements.
timeseries	Retrieve RavenPack News Analytics intraday and historical data.
realtime	Retrieve RavenPack News Analytics real-time data.
rploader	Read the RavenPack News Analytics data file.
close	Close the RavenPack News Analytics connection.

R2015a

Version: 5.1

New Features

Bug Fixes

Bloomberg Data License support

Connect to Bloomberg Data License using `bdl`. Upload data request files and download the output files.

R2014b

Version: 5.0

New Features

Bug Fixes

Data access through Thomson Reuters Eikon

Retrieve current market, real-time, and historical data using Thomson Reuters Eikon.

Function	Purpose
treikon	Connect to Thomson Reuters Eikon.
getdata	Retrieve Thomson Reuters Eikon current market data.
history	Retrieve Thomson Reuters Eikon historical data.
realtime	Retrieve Thomson Reuters Eikon real-time data.
start	Resume Thomson Reuters Eikon real-time data retrieval.
stop	Stop Thomson Reuters Eikon real-time data retrieval.
chain	Retrieve Thomson Reuters Eikon chain data.

Data access through Bloomberg Server

Connect to Bloomberg Server using `blpsrv`. You can use the existing functions to retrieve current market, real-time, intraday tick, historical, and security lookup data using Bloomberg Server.

Data access through Bloomberg B-PIPE

Connect to Bloomberg B-PIPE using `bpipe`. You can use the existing functions to retrieve current market, real-time, intraday tick, historical, and security lookup data using Bloomberg B-PIPE.

R2014a

Version: 4.7

New Features

Bug Fixes

Lookup functionality for Bloomberg Instrument service

The lookup function enables finding information about a security, a curve, or a government security.

R2013b

Version: 4.6

New Features

Bug Fixes

Multiple security request performance improvements for Bloomberg history function

The history function has improved performance.

R2013a

Version: 4.5

New Features

Bug Fixes

Compatibility Considerations

Function for accessing real-time data from FactSet workstations

Support for FactSet Data Server real-time data using `fds`, `close`, `stop`, and `realtime`.

X_TRADER functions, originally in Datafeed Toolbox, now in Trading Toolbox

The following X_TRADER® functions are removed from Datafeed Toolbox and migrated to Trading Toolbox™: `xtrdr`, `close`, `createInstrument`, `createNotifier`, `createOrderProfile`, `createOrderSet`, and `getData`.

Compatibility Considerations

Contact your account representative.

Bloomberg legacy functions removed

The following legacy Bloomberg functions are removed from Datafeed Toolbox: `showtrades`, `pricevol`, `stockticker`, `history`, `realtime`, `getdata`, `timeseries`, `getdata`, and `stop`.

Compatibility Considerations

Use the Datafeed Toolbox V3 `blp` functions for Bloomberg: `getdata`, `history`, `realtime`, and `timeseries`.

R2012b

Version: 4.4

New Features

Bug Fixes

Functions for accessing data through the IQFEED service

Support for IQFEED data using `iqf`, `iqf.close`, `iqf.history`, `iqf.marketdepth`, `iqf.news`, `iqf.realtime`, and `iqf.timeseries`.

Support for Bloomberg Historical Technical Analysis

Support for Bloomberg V3 API for historical technical analysis using `blp.tahistory`.

Support for Bloomberg Equity Screening

Support for Bloomberg V3 API for equity screening service (EQS) using `blp.eqs`.

R2012a

Version: 4.3

New Features

Bug Fixes

Support for Trading Technologies X_TRADER

The new `xtrdr` function and related methods provide access to the Trading Technologies® X_TRADER API for 32-bit Windows.

Three examples demonstrating `xtrdr` and its methods are included:

- `TTPriceUpdateExample.m`
- `TTPriceUpdateDepthExample.m`
- `TTOrderSubmitExample.m`

64-Bit Support for Haver Analytics

The `haver` function and related methods are now supported on 64-bit Windows.

Broker Buy and Sell Codes Added to Bloomberg Intraday Output

You can now request the broker buy and sell codes of each trade as part of an intraday tick request. These codes identify the market makers.

Timeout Argument for `blp`

The `blp` function has a new timeout mechanism. You can now specify how long to attempt connection before timing out if the connection cannot be made.

R2011b

Version: 4.2

New Features

Bug Fixes

Compatibility Considerations

Support for SIX Financial Information

The new `tlkrs` function and related methods provide access to current, historical, and intraday data from SIX Financial Information.

Improved Performance for blp Methods

The `blp.history` and `blp.timeseries` methods now have improved performance.

blp.history Now Supports Overrides

The `blp.history` method now includes an `'overrideOption'` argument.

Warning and Error ID Changes

Many warning and error IDs have changed from their previous versions. These warnings or errors typically appear during a function call.

Compatibility Considerations

If you use warning or error IDs, you might need to change the strings you use. For example, if you turned off a warning for a certain ID, the warning might now appear under a different ID. If you use a `try/catch` statement in your code, replace the old identifier with the new identifier. There is no definitive list of the differences, or of the IDs that changed.

R2011a

Version: 4.1

New Features

Bug Fixes

New rdth Methods for FTP Requests to Thomson Reuters Tick History

The new `rdth.submitftp` method submits FTP requests for Thomson Reuters Tick History data.

The `rdth.status` method returns the status and queue position of the FTP request handle.

R2010b

Version: 4.0

New Features

Bug Fixes

Compatibility Considerations

Support for eSignal Data Retrieval

Datafeed Toolbox software now supports retrieving data from eSignal.

New blp Methods Return Information About Bloomberg V3 Connection

Three new methods for the `blp` function return information about the Bloomberg V3 connection:

- `blp.category` returns category information for a specific search string.
- `blp.fieldinfo` returns field information for a specific field mnemonic.
- `blp.fieldsearch` returns field information for a specific search string.

blp.timeseries Method Now Returns Total Value

The `blp.timeseries` method now returns an additional field, Total Tick Value, when returning Bloomberg V3 intraday tick data.

Support for Reuters Publishing

You can now contribute data to a Reuters® datafeed with the new `reuters.contrib` method.

New rmdsconfig Function to Start Reuters Configuration Editor

You can now use the `rmdsconfig` function to start the Reuters Configuration Editor.

Reuters fetch Function Callbacks Now Require Three Inputs

In previous releases, the `reuters.fetch` function callbacks required two inputs. They now require three. The third input argument is the `serviceName`.

Compatibility Considerations

If you wrote code with the `reuters.fetch` function in R2010a, it will no longer work. Modify your callbacks to take a third input. Type `help rtdemo` at the command line to see an example.

Proxy Information No Longer Required for yahoo Function

In previous releases, if you wanted to connect to Yahoo! through a proxy server, you needed to supply your IP address and port number. This is no longer necessary. The `yahoo` function now uses the proxy information supplied in the MATLAB Web Preferences. To view or update your information, select **File > Preferences > Web**.

R2010a

Version: 3.5

New Features

Bug Fixes

Compatibility Considerations

Support for Bloomberg Version 3

Bloomberg has released a new version of their API. The new `blp` class supports the Bloomberg V3 interface. Support for Bloomberg V3 has been implemented using their Java® API which makes the toolbox platform independent for Bloomberg users allowing them to run on 64-bit Windows machines.

Bloomberg Pre-V3 Access Not Supported

Bloomberg has released a new version of their API. They are not disabling the older API but no longer support it. The existing `bloomberg` methods will display warnings notifying users that they should move to the `blp` methods. The ActiveX® methods for `bloomberg` will also generate warnings. The `bloomberg` class and its accompanying methods will be removed in a future release.

Compatibility Considerations

Update your code to use the new `blp` class and its accompanying methods. For example:

```
b = bloomberg;  
d = fetch(b, 'FOOB US Equity', 'GETDATA', 'LAST_PRICE');  
d = fetch(b, 'FOOB US Equity', 'TIMESERIES', '10/30/2009');  
d = fetch(b, 'FOOB US Equity', 'HISTORY', 'LAST_PRICE', ...  
    '10/01/2009', '10/31/2009');
```

becomes

```
b = blp;  
d = getdata(b, 'FOOB US Equity', 'LAST_PRICE');  
d = timeseries(b, 'FOOB US Equity', '10/30/2009');  
d = history(b, 'FOOB US Equity', 'LAST_PRICE', ...  
    '10/01/2009', '10/31/2009');
```

See the function reference page for `blp` for more information.

R2009b

Version: 3.4

New Features

Bug Fixes

Support for New Access Methods

- Reuters Time Series One allows you to access historical end-of-day data from Reuters.
- New Bloomberg methods use the Bloomberg ActiveX interface.

R2009a

Version: 3.3

New Features

Bug Fixes

Support for Retrieving New Types of Data

Datafeed Toolbox software now supports retrieving data from the following sources:

- Reuters Datascope Tick History software via the Web
- Enhanced Kx Systems, Inc. kdb+ data type support
- RFA 6 for Reuters Market Data System
- Haver Analytics data aggregation

For more information, see Thomson Reuters Tick History, Kx Systems, and Haver Analytics.

R2008b

Version: 3.2

New Features

Bug Fixes

Support for Retrieving New Types of Data

The Datafeed Toolbox software now supports retrieving data from the following sources:

- Reuters Datascope Tick History software
- Reuters Knowledge Direct software
- Reuters Newscope software

For more information, see Thomson Reuters Tick History and Reuters Newscope.

R2008a

Version: 3.1

New Features

Bug Fixes

Datafeed Toolbox Software Support for Hyperfeed Data Service Discontinued

The Datafeed Toolbox software no longer supports Hyperfeed® data service.

R2007b

Version: 3.0

New Features

Bug Fixes

Reuters Market Data System Support

Datafeed Toolbox support has been added for Reuters Market Data System. For more information, see [reuters](#).

R2007a

Version: 2.0

New Features

Kx Systems, Inc. kdb+ Database Support

Support has been added for Kx Systems, Inc. kdb+ database.

Haver Analytics Data Support

Support has been added for Haver Analytics financial data.

R2006b

Version: 1.9

New Features

Federal Reserve Economic Data

Support has been added for Federal Reserve Economic Data (FRED).

Additional Fields Added for Yahoo! Support

The Yahoo! `fetch` command supports additional values for *Fields*. For a complete list of supported values for market and historical data, see `matlabroot/toolbox/datafeed/datafeed/@yahoo/yhfields.mat`.

R2006a

Version: 1.8

New Features

Thomson Datastream Support Added

Version 1.8 of the toolbox adds support for Thomson® Datastream data service. Version 1.8 provides functions to obtain price, profile, historical, and tick data from Thomson Datastream data servers.

Thomson Datastream Function Summary

Function	Purpose
<code>datastream.close</code>	Close connection to data server
<code>datastream</code>	Connect to the Thomson Datastream API
<code>datastream.fetch</code>	Request data from data server
<code>datastream.get</code>	Get connection object properties
<code>datastream.isconnection</code>	Verify whether connection to data server is valid

R14SP3

Version: 1.7

New Features

Enhanced Capabilities for Bloomberg fetch Command

The following capabilities have been added to the `bloomberg.fetch` command:

- Currency conversion
- Security type identifier support
- Real-time support

The real-time support capability effectively replaces the monitoring capability that had been added to the Bloomberg `fetch` command in Release 1.5. Although the monitoring capability has been removed from the documentation, the capability remains in the product for backward compatibility.

R14SP2

Version: 1.6

New Features

FactSet Data Service Support Added

Version 1.6 of the toolbox adds support for FactSet data service. Version 1.6 provides functions to obtain price, profile, historical, and tick data from FactSet data servers.

FactSet Function Summary

Function	Purpose
<code>factset.close</code>	Close connection to data server
<code>factset.fetch</code>	Request data from data server
<code>factset.fetch</code>	Get connection object properties
<code>factset</code>	Connect to FactSet data server
<code>factset.isconnection</code>	Verify whether connection to data server is valid