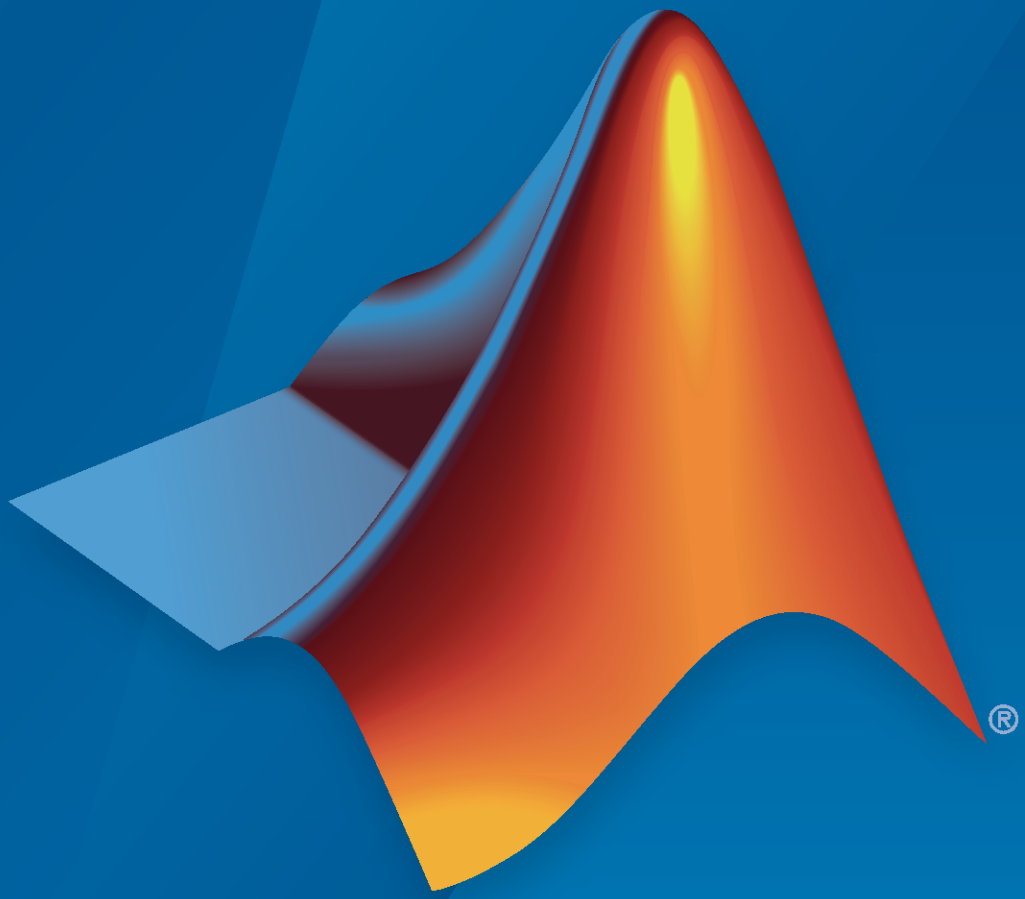
Datafeed Toolbox™ Release Notes



MATLAB®

MathWorks®

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

Contents

R2026a

R2025b

R2025a

R2024b

R2024a

R2023b

R2023a

R2022b

R2022a

R2021b

R2021a

R2026a

★Access historical economic data through FRED REST API

You can access Federal Reserve Economic Data (FRED®) and retrieve historical economic time series data from MATLAB®. Use the `fredrs` function to connect to the FRED server.

Access climate and property data from CoreLogic

CoreLogic® offers a wide range of climate and property data that you can use for climate risk modeling and insurance applications. Use the `corelogic` function to connect to the CoreLogic server and retrieve your data by using the `getData` function.

Access data servers using secure credentials

Save your user credentials in your MATLAB vault by using the `setSecret` function. Then, retrieve your credentials by using the `getSecret` function when you connect to a data server. For more information on making secure data server connections, see “Store Credentials for Data Service Connections”.

R2025b

Version: 25.2

Bug Fixes

Quality and stability improvements

R2025b delivers quality and stability improvements, building on the new features introduced in R2025a.

R2025a

Version: 25.1

New Features

Bug Fixes

Access data from Intercontinental Exchange

You can access reference, pricing, sustainability, and physical risk data from Intercontinental Exchange® (ICE®). Connect to the ICE server and retrieve your data by using the `ice` and `getData` functions.

R2024b

Version: 24.2

Bug Fixes

R2024a

Version: 24.1

New Features

Bug Fixes

★Access data from HaverView and Haver Analytics DLX Direct

You can access cloud-based historical economic data and climate data by using HaverView™ or DLX Direct connection objects.

- Use the `haverview` object to connect to a database on the HaverView cloud platform. This connection uses a valid authentication token supplied by Haver Analytics®.
- Use the `haverdirect` object to connect to a Haver Analytics DLX Direct database file in the cloud. This connection uses two-factor authentication.

★Access climate transition analytics data from Intercontinental Exchange

You can access climate transition analytics data from the Intercontinental Exchange (ICE) API using an `icecta` object with a valid access token. Retrieve ICE data by using the `getData` function.

R2023b

Version: 23.2

New Features

Bug Fixes

Bloomberg Interface: Hypermedia API support

You can access current market, real-time, intraday tick, historical, and security lookup data from the Bloomberg® Hypermedia API using the `bloombergHypermedia` object.

For more information on connecting to Bloomberg Hypermedia and retrieving data, see Retrieve Data Using Bloomberg Hypermedia.

R2023a

Version: 6.4

New Features

Bug Fixes

LSEG Data & Analytics Datascope Select: Access historical and current market data

Create a LSEG Data & Analytics 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

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 LSEG Data & Analytics	https://www.mathworks.com/help/releases/R2020b/datafeed/elektron-1.html
Enterprise Platform from LSEG Data & Analytics	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 LSEG Data & Analytics 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 LSEG Data & Analytics 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 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 LSEG Data & Analytics 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 LSEG Data & Analytics 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 LSEG Data & Analytics 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 LSEG Data & Analytics 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 LSEG Data & Analytics 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.O',{'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 LSEG Data & Analytics 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 LSEG Data & Analytics 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 LSEG Data & Analytics 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 LSEG Data & Analytics 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 LSEG Data & Analytics 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 LSEG Data & Analytics 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.