



dxFeed Market Data:  
onDemand Historical Tick Data  
Market Replay and Extraction

Quick Overview and Samples

April 2018

Information in this document is subject to change without notice and does not represent a commitment on the part of dxFeed.

Please visit [www.dxfed.com](http://www.dxfed.com) for latest information updates.

## Contents

1.	Java, JavaScript APIs and web services.....	4
1.1.	dxFeed Java API .....	4
1.2.	dxFeed webservice and Javascript API .....	5
2.	REST web services .....	6
2.1.	IPF web service .....	6
2.2.	onDemand tick data extraction web service.....	7
2.3.	onDemand tick data audit web service .....	8
3.	Documentation.....	9

# 1. Java, JavaScript APIs and web services

## 1.1. dxFeed Java API

dxFeed Java API gives control over the full functionality of dxFeed real-time, historical and aggregated data (including market replay):

<http://www.dxfed.com/downloads/api/java/latest-release/>

Samples package is the best way to quickly get things working:

<https://www.dxfed.com/wp-content/uploads/api/java/dxfed-samples-3.254.zip>

Your basic code to connect to data source and start receiving quotes will be under 10 LOCs:

```
// create subscription for a specific event type on default feed
DXFeedSubscription<Quote> sub = DXFeed.getInstance().createSubscription(Quote.class);

// define listener for events
sub.addListener(new DXFeedEventListener<Quote>() {
    public void eventsReceived(List<Quote> events) {
        for (Quote quote : events)
            System.out.println(quote);
    }
});

// add symbols to start receiving events
sub.addSymbols("GOOG", "AAPL");
```

Historical market replay sample (see `com.dxfed.sample.ondemand`) is not much longer and provides the example of switching to market replay by switching the endpoint to `onDemand` endpoint and setting start datetime and replay speed:

```
public class OnDemandSample {
    public static void main(String[] args) throws ParseException,
        InterruptedException {
        // get on-demand-only data feed
        OnDemandService onDemand = OnDemandService.getInstance();
        DXFeed feed = onDemand.getEndpoint().getFeed();

        // subscribe to Accenture symbol ACN to print its quotes
        DXFeedSubscription<Quote> sub = feed.createSubscription(Quote.class);
        sub.addListener(new DXFeedEventListener<Quote>() {
            public void eventsReceived(List<Quote> events) {
                for (Quote quote : events) {
                    System.out.println(quote.getEventSymbol() +
                        " bid " + quote.getBidPrice() + " /" +
                        " ask " + quote.getAskPrice());
                }
            }
        });
    }
}
```

```
});
sub.addSymbols("ACN");

// Watch Accenture drop under $1 on May 6, 2010 "Flashcrash"
// from 14:47:48 to 14:48:02 EST
SimpleDateFormat format = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss
'EST'");

format.setTimeZone(TimeZone.getTimeZone("America/New_York"));
Date from = format.parse("2010-05-06 14:47:48 EST");
Date to = format.parse("2010-05-06 14:48:02 EST");

// switch into historical on-demand data replay mode
onDemand.replay(from);

// replaying events until end time reached
while (onDemand.getTime().getTime() < to.getTime()) {
    System.out.println("Current state is " +
        onDemand.getEndpoint().getState() + ", " +
        " on-demand time is " + format.format(onDemand.getTime()));
    Thread.sleep(1000);
}

// close endpoint completely to release resources and shutdown JVM
onDemand.getEndpoint().closeAndAwaitTermination();
}
}
```

## 1.2. dxFeed webservice and Javascript API

JavaScript API mirrors dxFeed Java API in JavaScript and provides easy widgets and shortcuts to integrate streaming market data into web apps.

<http://www.dxfed.com/downloads/api/webservice/latest-release/>

Console for dxFeed web service with test data is deployed here: <http://tools.dxfed.com/webservice/>

It provides samples and diagnostic tools for streaming data subscription, building charts, using widgets and using REST services for snapshot queries.

## 2. REST web services

### 2.1. IPF web service

Instrument profile web service providing essential symbol attributes and symbol discovery:

URL: <http://tools.dxfed.com/ipf>

```
Usage: /ipf?<parameters>
Available parameters to filter instrument profiles output, returning instrument
profiles only for:
  mode          'batch' (default) for bulk retrieval, 'ui' for symbol lookup
  date          ipf for the business day specified as YYYYMMDD (not available in 'ui'
mode)
  types         list of type patterns, e.g. INDEX, STOCK
  symbols       list of symbol patterns, e.g. IBM or *DJI*
  underlyings   list of underlyings patterns
  products      list of products patterns, e.g. /ESZ3 is returned for /ES
  lists         list of list name patterns, e.g. index constituents of SP500
  text          text for full-text search (only available and required in 'ui' mode)
  limit         limit result to <n> rows (only available and required in 'ui' mode)
  compression  'plain' (default) for uncompressed, 'zip' or 'gzip' for compressed
output

Note: 'list of patterns' denotes a comma-separated list of glob patterns like * or
/ES,/ES??

Additional help and listing commands:
  ?help        prints this message
  ?help&lists  list of available lists in CSV format

Examples:
  /ipf?date=20131210          ipf for 2013-12-20
  /ipf?types=INDEX,STOCK     output only INDEX and STOCK types
  /ipf?symbols=IBM,AAPL,SPY  instrument profiles for IBM,AAPL,SPY only
  /ipf?underlyings=/*        instrument profiles for futures options only (underlyings
starting with "/" are futures)
  /ipf?products=/ES         instrument profiles for futures series for /ES - S&P500
futures on CME
  /ipf?lists=SP500,DJ*      instrument profiles for constituents of S&P500 and all
DowJones indexes
```

e.g. getting S&P500 e-mini series: <http://tools.dxfed.com/ipf?symbols=/ES??>

All futures: <http://tools.dxfed.com/ipf?types=FUTURE>

## 2.2. onDemand tick data extraction web service

Extracts tick data for given symbols, time and event types from our cloud storage.

All instrument types are available for you to test.

URL: <http://tools.dxfed.com/onDemand/data>

Usage: /onDemand/data?categories=<category-list>&symbols=<symbol-list>&start=<start-time>&end=<end-time>

Optional parameters: &fields=<field-list>&conflate=<conflate-period>&format=csv

<category-list> is a comma-separated list of 3-letter categories like F-M or E-Q,E-T,E-H

1st letter: E=Equities, F=Futures, O=Options, Q=Future Options, X=Forex

2nd letter: -=Composite, .=Regional

3rd letter: Q=Quote, T=Trade, S=Summary, P=Profile, H=TradeHistory, M=MarketMaker

Glob patterns with \* and ? can be used.

<symbol-list> is a comma-separated list of symbol glob patterns like \* or /ES,/ES?? or IBM,AAPL,SPY

<start-time> is a timestamp when to start data extraction like 20110509-0830

<end-time> is a timestamp when to stop data extraction like 20110509-1600  
timestamps are specified in Eastern Standard Time timezone

<field-list> is a comma-separated list of globs like Symbol,EventTime,\*Price

<conflate-period> is a time-period for conflation like 1s or 1m

It is possible and recommended to launch separate instances of a tool to extract data for different categories or different category groups (e.g. equities, futures).

It is also recommended to use separate launches for extraction of large amounts of data.

Note that extracted data is reordered in chunks, for original order sorting by timestamp is required.

e.g.

get S&P500 e-mini quotes and trades in 2014 between 2014-08-12 9:00 and 9:10 am ET:

<http://tools.dxfed.com/onDemand/data?categories=F-Q,F-H&symbols=/ES??&start=20140812-0900-0500&end=20140812-0910-0500>

get S&P500 e-mini quotes and trades in 2013 between 2013-08-12 9:00 and 9:10 am ET:

<http://tools.dxfed.com/onDemand/data?categories=F-Q,F-H&symbols=/ES??&start=20130812-0900-0500&end=20130812-0910-0500>

get WTI, Natural Gas, S&P500 e-mini in 2012 between 2013-08-12 9:00 and 9:10 am ET:

<http://tools.dxfed.com/onDemand/data?categories=F-Q,F-H&symbols=/CL?/,/NG?/,/ES??&start=20120807-0900-0500&end=20120807-0910-0500>

...conflated to 1 minute:

<http://tools.dxfed.com/onDemand/data?categories=F-Q,F-H&symbols=/CL?/,/NG?/,/ES??&start=20120807-0900-0500&end=20120807-0910-0500&conflate=1m>

...in CSV format instead of QDS text dump:

<http://tools.dxfed.com/onDemand/data?categories=F-Q,F-H&symbols=/CL?/,/NG?/,/ES??&start=20120807-0900-0500&end=20120807-0910-0500&conflate=1m&format=csv>

get Apple and Google ticks in 2014 between 2014-04-14 10:00 and 10:10 am ET:

<http://tools.dxfed.com/onDemand/data?categories=E-Q,E-H,E-S&symbols=AAPL,GOOG&start=20140414-1000-0500&end=20140414-1010-0500>

...get Apple and Google options quotes and trades in the same period:

[http://tools.dxfed.com/onDemand/data?categories=O-Q,O-H,O-S&symbols=.AAPL\\*,.GOOG\\*&start=20140414-1000-0500&end=20140414-1010-0500](http://tools.dxfed.com/onDemand/data?categories=O-Q,O-H,O-S&symbols=.AAPL*,.GOOG*&start=20140414-1000-0500&end=20140414-1010-0500)

## 2.3. onDemand tick data audit web service

Extracts data from onDemand cloud storage for given symbols, events and timeframes keeping the order. Used for trade audit purposes primarily, currently provides access only to 1 month of data on all symbols except OPRA regional options data where the limitation is 1 week. Limitations are configurable.

URL: <http://tools.dxfed.com/tickdata>

```
Usage: /tickdata?help
Prints this help screen.

Usage: /tickdata?<parameters>
Extracts data for the specified parameters as a tape in the specified format.
Available parameters:
records          record names pattern (e.g. Quote,Trade)
symbols          comma-separated list of symbols (e.g. AAPL,GOOG)
start            start time as YYYYMMDD-HHMMSS[.sss][zone]
stop            stop time as YYYYMMDD-HHMMSS[.sss][zone]
format          output format - text, csv or binary (optional, text by default)
compression     output compression - none, zip or gzip (optional, none by default)
```

e.g.

get quotes and trades for WTI between 2014-08-11 11:00 – 11:20 am ET:

<http://tools.dxfed.com/tickdata?records=Quote,Trade,TimeAndSale&symbols=/CLU4&start=20140811-1100-0500&stop=20140811-1120-0500&compression=none>

...same in CSV:

<http://tools.dxfed.com/tickdata?records=Quote,Trade,TimeAndSale&symbols=/CLU4&start=20140811-1100-0500&stop=20140811-1120-0500&compression=none&format=csv>

...same in CSV zipped:

<http://tools.dxfed.com/tickdata?records=Quote,Trade,TimeAndSale&symbols=/CLU4&start=20140811-1100-0500&stop=20140811-1120-0500&compression=zip&format=csv>

mix CLU4, CLV4, AAPL and GOOG in one query:

<http://tools.dxfed.com/tickdata?records=Quote,Trade,TimeAndSale&symbols=/CLU4,/CLV4,AAPL,GOOG&start=20140811-1100-0500&stop=20140811-1120-0500&compression=none>



### 3. Documentation

Please follow the links below for online docs on dxFeed API and supporting information:

General documentation section: <https://www.dxfed.com/specifications-and-guides/>

Javadocs online: <http://docs.dxfed.com/>

...specifically for dxFeed API: <http://docs.dxfed.com/dxfed/api/index.html>

A couple of our blog articles on API structure and usage:

<https://www.dxfed.com/dxfed-api-tutorial-part-1-basics/>

<https://www.dxfed.com/dxfed-api-tutorial-part-2-quotes-and-trades/>