



# BATS Europe Secure Web API

Version 1.4

August 29, 2011

BATS Trading Limited is authorized and regulated by the Financial Services Authority. BATS Trading Limited is a wholly-owned subsidiary of BATS Global Markets, Inc. and is a company registered in England and Wales with Company Number 6547680 and registered office at 25 Cophall Avenue, London EC2R 7BP. This document has been established for information purposes only. None of the information concerning the services or products described in this document constitutes advice or a recommendation of any product or service. To the extent that the information provided in this document constitutes a financial promotion as defined by section 21 of the Financial Services and Markets Act 2000, it is only directed at persons who qualify as a Professional Client or Eligible Counterparty. Persons who do not qualify should not act or rely upon it.

## Contents

|          |                                 |          |
|----------|---------------------------------|----------|
| <b>1</b> | <b>Introduction</b> .....       | <b>3</b> |
| 1.1      | Overview.....                   | 3        |
| 1.2      | Requirements .....              | 3        |
| 1.3      | Access.....                     | 3        |
| 1.4      | Authentication.....             | 3        |
| 1.5      | Testing.....                    | 4        |
| <b>2</b> | <b>Request Structure</b> .....  | <b>4</b> |
| <b>3</b> | <b>Response Structure</b> ..... | <b>5</b> |
| <b>4</b> | <b>Services</b> .....           | <b>6</b> |
| 4.1      | Sponsored Access.....           | 6        |
| 4.2      | Market Analytics Service.....   | 6        |
| 4.3      | Liquidity Provider .....        | 6        |
| <b>5</b> | <b>Support</b> .....            | <b>6</b> |

## 1 Introduction

### 1.1 Overview

The BATS Europe Secure Web API allows client applications to view and update BATS Europe data using the HTTPS protocol over the Internet. For example, if a Participant sponsors other Participants, they can use the BATS Europe Web Portal to manage these relationships. The API exposes this same functionality in a programmatic way allowing Participants to write programs to automate these tasks. You could, for example, use the BATS Europe API to limit the max share size allowed on all FIX sessions used by a particular Sponsored Participant.

### 1.2 Requirements

#### Secure services:

- A BATS Europe API Developer Key
- A BATS Europe API Developer Secret
- Ability to send HTTPS requests and receive HTTPS responses over the Internet

### 1.3 Access

The API is accessed via a URL using the HTTPS protocol. The API is served from the `api.batstrading.co.uk` domain.

```
https://api.batstrading.co.uk/service/
```

Where `service` is the path to a particular service you want to reach. The available services are detailed later in this document. More services will be added over time.

Access to our UAT environment is through the `api.certification.batstrading.co.uk` domain.

```
https://api.certification.batstrading.co.uk/service/
```

### 1.4 Authentication

The Secure Web API uses HTTP Basic Auth over SSL to authenticate your login credentials. You use your BATS Europe API Developer Key as your username and your BATS Europe API Developer Secret as your password. If you already have an API Key for the BATS Europe Public Web API, you can use this same key. Contact the BATS Europe Trade Desk to receive your API Key and Secret.

Never share your Secret with anyone. BATS Associates will never ask you for your Secret.

## 1.5 Testing

In order to facilitate testing of the API, BATS Europe provides the following two Python files. They are provided on an as-is basis and no guarantee as to their suitability for use in a production environment is made.

- [General base API class](#)
- [API test script](#)

## 2 Request Structure

Requests that retrieve data without making any modifications can be made using an HTTP GET. Requests that are expected to modify data must use an HTTP POST. Each service may have several commands available. Each command may have unique parameter requirements and so are specified per service and command.

### Example Request

For example, one of the services exposed via the API is the Sponsored Access service. One of the commands available in this service is “viewClient”. Since this command does not modify anything, you would use a GET.

Service URL: [https://api.batstrading.co.uk/account/sponsored\\_access/](https://api.batstrading.co.uk/account/sponsored_access/)  
Method: GET  
Required request parameters:  
    command: “viewClient”  
    client: “ABCD”

Since this example is a GET, the API request could be made using this URL:

[https://api.batstrading.co.uk/account/sponsored\\_access/?command=viewClient&client=ABCD](https://api.batstrading.co.uk/account/sponsored_access/?command=viewClient&client=ABCD)

### 3 Response Structure

The response will be in JSON format. There are some key-value pairs that you can always expect while other keys will optionally exist. In the table below, the optional fields are shown in *italic*.

**Additional key-value pairs and data items could be added.** You should develop your application in such a way that it will not break if new items are added in the response. Do not rely on key position when parsing the response.

| Field             | Description   |
|-------------------|---|
| <code>code</code> | Request result code. <ul style="list-style-type: none"><li>• 200 series = Success.</li><li>• Non 200 series = Failure. See 'msg' value for details. The request may have been partially successful.</li></ul> |
| <i>ttl</i>        | "Time to Live" The response you received is cached for the TTL period (expressed in milliseconds). This means it would be useless to resend this same request within the TTL period.                          |
| <i>msg</i>        | A string description of the success or failure. Usually only present when code is not 200.  |
| <i>data</i>       | A list of data items. The exact format is specific to the command requested.  |
| <i>data_ts</i>    | Timestamp when the returned data was created. Format is 'YYYY-MM-DD HH:MM:SS'.  |

#### Example Response Structure

```
{
  "code": "200",
  "data": [{"Color": "green", "Size": 12}, {"Color": "blue", "Size": 9}],
  "data_ts": "2009-10-05 13:27:08"
}
```

## 4 Services

### 4.1 Sponsored Access

**Service URL**

[https://api.batstrading.co.uk/account/sponsored\\_access/](https://api.batstrading.co.uk/account/sponsored_access/)

**Description**

Used by a Sponsoring Participant to monitor and limit Sponsored Participant activity.

**Specification Document**

[Sponsored Access Secure Web API](#)

### 4.2 Market Analytics Service

**Service URL**

[https://api.batstrading.co.uk/account/market\\_analytics/](https://api.batstrading.co.uk/account/market_analytics/)

**Description**

Used by a trading firm or market data recipient to get up to date analysis on their activity on the BATS Europe market as well as pan-European activity

**Specification Document**

[Market Analytics Service API](#)

### 4.3 Liquidity Provider

**Service URL**

[https://api.batstrading.co.uk/account/market\\_maker/](https://api.batstrading.co.uk/account/market_maker/)

**Description**

Used by a liquidity provider to manage the symbols, bank codes and ports for which they intend to provide liquidity under the terms of the BATS Europe Liquidity Provider Program

**Specification Document**

[Liquidity Provider API](#)

## 5 Support

Please e-mail questions or comments regarding this specification to [tradedeskeurope@batstrading.com](mailto:tradedeskeurope@batstrading.com).

## Revision History

| Protocol Version | Date       | Description  |
|------------------|------------|--|
| 1.0              | 12/02/2010 | Release 1.0 distributed.                           |
| 1.1              | 19/03/2010 | Added Firm Analysis                                |
| 1.2              | 24/03/2010 | Moved individual services into their own documents |
| 1.3              | 30/06/2010 | Firm Analysis →Market Analytics                    |
| 1.4              | 29/08/2011 | Added Liquidity Provider API                       |