API Overview

The Exchange API is for developers looking to create automated betting systems or custom betting interfaces for themselves or for Betfair customers. The Exchange API is available for the Global, Spanish and Italian Betfair Exchange

The API contains a powerful set of features that enable advanced market navigation, search, odds retrieval, bet placement and sports related data retrieval. The Exchange API is made up of the following key components:

- **Betting API** - Contains navigation, odds retrieval and bet placement operations.
- **Accounts API** - Contains account related operations such as the ability to retrieve your available account balance as well as Vendor Services operations that are available to licensed Software Vendors
- **Heartbeat API** - allows you to automatically cancel unmatched bets in the event of your API client/s losing connectivity.
- **Race Status API -** allows you to establish the status of a horse race or greyhound market both prior to and after the start of the race.
- **Exchange Stream API** - allows you to subscribe to market changes (both price and definitions) and orders.

Documentation

There are a number of documentation resources available:

Getting Started Guide - provides all the information required regarding licensing, login and making your first requests via the Betfair API.

Reference Guide - the latest documentation for the Betfair API.

Sample Code - code samples are available in a number of programming languages.

Demo Tools - allow you to quickly test API operations via an easy to use interface.

Developers Forum - discuss the Betfair API, share your knowledge and ask questions of the developer community.

Benefits & Features

The main benefits and features of the Exchange API include:

- Access to the Exchange API is free of charge for development purposes*# to all developers for personal use only.
- No data request charges for requests made via the Exchange API
- Lightweight protocol (JSON/JSON-RPC).
- Configure the depth of the best prices returned to you.
- Rollup available prices - you can configure the rollup amount and type.
- Retrieve data from multiple markets in one request.
- Retrieve matched and unmatched bets and prices available via a single request.
- Search by MarketType (MATCH_ODDS, WIN, PLACE etc.) flags which remain the same, regardless of language.
- Search for in-play markets.
- View ‘result’ by selection after settlement.
- View virtual prices.

*** does not apply to commercial access**. Please see Commercial Opportunities for details of commercial licensing

# You should use your **Delayed Application key** for development purposes. To apply for a **Live Application key** please click here and select **Exchange API > For My Personal Betting** and complete the application form at the bottom of the page. A one-off activation fee of **£299** applies; this is debited directly from your Betfair account once access is approved.

Recently Updated

Interactive Login - API Endpoint
about 5 hours ago • updated by built In User • view change

Non-Interactive (bot) login
about 5 hours ago • updated by built In User • view change

Exchange Stream API

Navigate space

Apr 21, 2017 • updated by built In User • view change

Betting Enums
Apr 13, 2017 • updated by built In User • view change

API Roadmap
Apr 12, 2017 • updated by built In User • view change

Betting Type Definitions
Apr 11, 2017 • updated by built In User • view change

Navigation Data For Applications
Apr 10, 2017 • updated by built In User • view change

listRunnerBook
Apr 06, 2017 • updated by built In User • view change

Betfair Price Increments
Apr 06, 2017 • updated by built In User • view change

Best Practice
Apr 06, 2017 • updated by built In User • view change

Application Keys
Apr 06, 2017 • updated by built In User • view change

Development Life Cycle
Apr 06, 2017 • created by built In User

Heartbeat API
Apr 06, 2017 • updated by built In User • view change

Interface Definition Documents
Apr 04, 2017 • updated by built In User • view change

Release Notes
Mar 28, 2017 • updated by built In User • view change

Getting Started

How Do I Get Started?

To use the Exchange API you require the following:

1. A **Betfair account**. you can open a Betfair account here
2. An **Application Key** - you can create an Application Key by following the instructions here
3. A **sessionToken -** you can create a session token by using either of the API login methods or by following the instructions here

Login

The Betfair API offers three login flows for developers, depending on the use case of your application:.

- **Non-Interactive login** - if you are building an application which will run autonomously, there is a separate login flow to follow to ensure your account remains secure.
- **Interactive login** - if you are building an application which will be used interactively, then this is the flow for you. This flow has two variants:
  - Interactive login - API method - This flow makes use of a JSON API Endpoint and is the simplest way to get started if you are looking to create your own login form.
  - Interactive login - Desktop Application- This login flow makes use of Betfair's login pages and allows your app to gracefully handle all errors and re-directions in the same way as the Betfair website

Making a Request to the API

- All requests must include a HTTP header named **"X-Application"** containing the Application Key assigned to you.
- All requests must include a HTTP header **"X-Authentication"** containing your sessionToken.

Please note: The only exceptions to the above are some of the Account Operations (Vendor API) which require the **X-Authentication** HTTP header only.

You can call the API at one of two endpoints, depending on which style of request you want to use.

API Endpoints

You can make requests and place bets on UK & international markets by accessing the Global Exchange via the following endpoints.

Please find the details for the current Betting API endpoints:

Global Exchange

Interface	Endpoint	JSON-RPC Prefix	<methodname> Example
JSON-RPC	https://api.betfair.com/exchange/betting/json-rpc/v1	<methodname>	SportsAPING/v1.0/listMarketBook
JSON REST	https://api.betfair.com/exchange/betting/rest/v1.0/		listMarketBook/

You can make requests for your UK Exchange wallet information by accessing the Global Exchange via the following endpoints.

Global Exchange

Interface	Endpoint	JSON-RPC Prefix	<methodname> Example
JSON-RPC	https://api.betfair.com/exchange/account/json-rpc/v1	<methodname>	AccountAPING/v1.0/getAccountFunds
JSON REST	https://api.betfair.com/exchange/account/rest/v1.0		getAccountFunds/

Please see separate documentation for the Spanish & Italian Exchange

JSON

You can POST a request to the API at:

https://api.betfair.com/exchange/betting/rest/v1.0/<operation name>. So, to call the listEventTypes method, you would POST to: https://api.betfair.com/exchange/betting/rest/v1.0/listEventTypes/

The POST data contains the request parameters. For listEventTypes, the only required parameter is a filter to select markets. You can pass an empty filter to select all markets, in which case listEventTypes returns the EventTypes associated with all available markets.

JSON POST Data

```
{
 "filter" : { }
}
```

Python Example JSON Request

```
import requests
import json

endpoint = "https://api.betfair.com/exchange/betting/rest/v1.0/"

header = { 'X-Application' : 'APP_KEY_HERE', 'X-Authentication' :
'SESSION_TOKEN_HERE' ,'content-type' : 'application/json' }

json_req='{"filter":{ }}'

url = endpoint + "listEventTypes/"

response = requests.post(url, data=json_req, headers=header)


print json.dumps(json.loads(response.text), indent=3)
```

JSON-RPC

You can POST a request to the API using JSON-RPC at:

https://api.betfair.com/exchange/betting/json-rpc/v1

The POST data should contain a valid JSON-RPC formatted request where the "params" field contains the request parameters and the "method" field contains the API method you are calling, specified like "SportsAPING/v1.0/<operation name>.

For example, if you were calling the listCompetitions operation and passing in a filter to find all markets with a corresponding event type id of 1 (i.e., all Football markets), the POST data for the JSON-RPC endpoint would be:

Example JSON-RPC POST data

```
{
   "params": {
      "filter": {
         "eventTypeIds": [1]
      }
   },
   "jsonrpc": "2.0",
   "method": "SportsAPING/v1.0/listCompetitions",
   "id": 1
}
```

Here's a quick example Python program that uses JSON-RPC and returns the list of EventTypes (Sports) available:

JSON-RPC Python Example

```
import requests
import json

url="https://api.betfair.com/exchange/betting/json-rpc/v1"
header = { 'X-Application' : 'APP_KEY_HERE', 'X-Authentication' :
'SESSION_TOKEN' ,'content-type' : 'application/json' }

jsonrpc_req='{"jsonrpc": "2.0", "method":
"SportsAPING/v1.0/listEventTypes", "params": {"filter":{ }}, "id": 1}'

response = requests.post(url, data=jsonrpc_req, headers=header)

print json.dumps(json.loads(response.text), indent=3)
```

And the response from the above:

EventTypeResult

```
{
   "jsonrpc": "2.0",
   "result": [
      {
         "eventType": {
            "id": "468328",
            "name": "Handball"
         },
         "marketCount": 59
      },
      {
         "eventType": {
            "id": "1",
            "name": "Soccer"
         },
         "marketCount": 14792
      },
      {
         "eventType": {
            "id": "2",
            "name": "Tennis"
         },
         "marketCount": 51
      },
      {
         "eventType": {
            "id": "3",
            "name": "Golf"
         },
         "marketCount": 12
      },
      {
         "eventType": {
            "id": "4",
            "name": "Cricket"
         },
         "marketCount": 139
      },
      {
         "eventType": {
            "id": "5",
            "name": "Rugby Union"
         },
         "marketCount": 100
      },
      {
         "eventType": {
            "id": "6",
            "name": "Boxing"
         },
         "marketCount": 12
      },
      {
         "eventType": {
            "id": "7",
            "name": "Horse Racing"
         },
         "marketCount": 187
      },
      {
         "eventType": {
            "id": "8",
            "name": "Motor Sport"
         },
         "marketCount": 3
      },
      {
         "eventType": {
            "id": "7524",
            "name": "Ice Hockey"
         },
         "marketCount": 8
      },
      {
         "eventType": {
            "id": "10",
            "name": "Special Bets"
         },
         "marketCount": 30
      },
      {
            "eventType": {
               "id": "451485",
               "name": "Winter Sports"
            },
            "marketCount": 7
         },
         {
            "eventType": {
               "id": "7522",
               "name": "Basketball"
            },
            "marketCount": 559
         },
         {
            "eventType": {
               "id": "1477",
               "name": "Rugby League"
            },
            "marketCount": 3
         },
         {
            "eventType": {
               "id": "4339",
               "name": "Greyhound Racing"
            },
            "marketCount": 269
         },
         {
            "eventType": {
               "id": "2378961",
               "name": "Politics"
            },
            "marketCount": 19
         },
         {
            "eventType": {
               "id": "6231",
               "name": "Financial Bets"
            },
            "marketCount": 51
         },
         {
            "eventType": {
               "id": "998917",
               "name": "Volleyball"
            },
            "marketCount": 69
         },
         {
            "eventType": {
               "id": "998919",
               "name": "Bandy"
            },
            "marketCount": 2
         },
         {
            "eventType": {
               "id": "998918",
               "name": "Bowls"
            },
            "marketCount": 10
         },
         {
            "eventType": {
               "id": "3503",
               "name": "Darts"
            },
            "marketCount": 446
         },
         {
            "eventType": {
               "id": "72382",
               "name": "Pool"
            },
            "marketCount": 1
         },
         {
            "eventType": {
               "id": "6422",
               "name": "Snooker"
            },
            "marketCount": 3
         },
         {
            "eventType": {
               "id": "6423",
               "name": "American Football"
            },
            "marketCount": 86
         },
         {
            "eventType": {
               "id": "7511",
               "name": "Baseball"
            },
            "marketCount": 1
         }
```

```
    ],
    "id": 1
}
```

Example Requests

This section shows how you might call the Betting API to retrieve information.

This section includes examples of how to request the following information in jsonrpc format:

- Request a List Of Available Event Types
- Request a List of Events for an Event Type
- Request the Market Information for an Event
- Football Competitions
- Market Prices
- Placing a Bet
- Placing a SP Bet
- Retrieving Details of Bet/s Placed on a Market/s
- Retrieving the Result of a Settled Market

Request A List Of Available Event Types

You can make a request using the listEventTypes service which will return a response containing the eventTypes (e.g. Soccer, Horse Racing etc.) that are currently available on Betfair.

listEventTypes Request

```
[
    {
        "jsonrpc": "2.0",
        "method": "SportsAPING/v1.0/listEventTypes",
        "params": {
            "filter": {}
        },
        "id": 1
    }
]
```

listEventTypes Response

```
[
    {
        "jsonrpc": "2.0",
        "result": [
            {
                "eventType": {
                    "id": "468328",
                    "name": "Handball"
                },
                "marketCount": 11
            },
            {
                "eventType": {
                    "id": "1",
                    "name": "Soccer"
                },
                "marketCount": 25388
            },
            {
                "eventType": {
                    "id": "2",
                    "name": "Tennis"
                },
                "marketCount": 402
            },
            {
                "eventType": {
                    "id": "3",
                    "name": "Golf"
                },
                "marketCount": 79
            },
            {
                "eventType": {
                    "id": "4",
                    "name": "Cricket"
                },
                "marketCount": 192
            },
            {
                "eventType": {
                    "id": "5",
                    "name": "Rugby Union"
                },
                "marketCount": 233
            },
            {
                "eventType": {
                    "id": "6",
                    "name": "Boxing"
                },
                "marketCount": 18
            },
            {
                "eventType": {
                    "id": "7",
                    "name": "Horse Racing"
                },
                "marketCount": 398
            },
            {
                "eventType": {
                    "id": "8",
                    "name": "Motor Sport"
                },
                "marketCount": 50
            },
            {
                "eventType": {
                    "id": "7524",
                    "name": "Ice Hockey"
                },
                "marketCount": 521
            },
            {
                "eventType": {
                    "id": "10",
                    "name": "Special Bets"
                },
                "marketCount": 39
            },
            {
                "eventType": {
                    "id": "451485",
                    "name": "Winter Sports"
                },
                "marketCount": 7
            },
            {
                "eventType": {
                    "id": "11",
                    "name": "Cycling"
                },
                "marketCount": 1
            },
            {
                "eventType": {
                    "id": "136332",
                    "name": "Chess"
                },
                "marketCount": 1
            },
            {
                "eventType": {
                    "id": "7522",
                    "name": "Basketball"
                },
                "marketCount": 617
            },
            {
                "eventType": {
                    "id": "1477",
                    "name": "Rugby League"
                },
                "marketCount": 91
            },
            {
                "eventType": {
```

```
                "id": "4339",
                "name": "Greyhound Racing"
            },
            "marketCount": 298
        },
        {
            "eventType": {
                "id": "6231",
                "name": "Financial Bets"
            },
            "marketCount": 44
        },
        {
            "eventType": {
                "id": "2378961",
                "name": "Politics"
            },
            "marketCount": 23
        },
        {
            "eventType": {
                "id": "998917",
                "name": "Volleyball"
            },
            "marketCount": 66
        },
        {
            "eventType": {
                "id": "998919",
                "name": "Bandy"
            },
            "marketCount": 4
        },
        {
            "eventType": {
                "id": "998918",
                "name": "Bowls"
            },
            "marketCount": 17
        },
        {
            "eventType": {
                "id": "26420387",
                "name": "Mixed Martial Arts"
            },
            "marketCount": 52
        },
        {
            "eventType": {
                "id": "3503",
                "name": "Darts"
            },
            "marketCount": 21
            },
            {
                "eventType": {
                    "id": "2152880",
                    "name": "Gaelic Games"
                },
                "marketCount": 2
            },
            {
                "eventType": {
                    "id": "6422",
                    "name": "Snooker"
                },
                "marketCount": 22
            },
            {
                "eventType": {
                    "id": "6423",
                    "name": "American Football"
                },
                "marketCount": 171
            },
            {
                "eventType": {
                    "id": "315220",
                    "name": "Poker"
                },
                "marketCount": 2
            },
            {
                "eventType": {
                    "id": "7511",
                    "name": "Baseball"
                },
                "marketCount": 7
            }
        ],
        "id": 1
    }
]
```

Request a List of Events for an Event Type

The below example demonstrates how to retrieve a list of events (eventIds) for a specific event type. The request shows how to retrieve all Soccer events that are taking place in a single day.

listEvents Request

```
[
    {
        "jsonrpc": "2.0",
        "method": "SportsAPING/v1.0/listEvents",
        "params": {
            "filter": {
                "eventTypeIds": [
                    "1"
                ],
                "marketStartTime": {
                    "from": "2014-03-13T00:00:00Z",
                    "to": "2014-03-13T23:59:00Z"
                }
            }
        },
        "id": 1
    }
]
```

listEvents Response

```
[
    {
        "jsonrpc": "2.0",
        "result": [
            {
                "event": {
                    "id": "27165668",
                    "name": "Al-Wahda (KSA) v Hajer (KSA)",
                    "countryCode": "SA",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T13:30:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27165665",
                    "name": "Al Hussein v Mansheyat Bani Hasan",
                    "countryCode": "JO",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T15:00:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27165425",
                    "name": "Daily Goals",
                    "countryCode": "GB",
                    "timezone": "Europe/London",
                    "openDate": "2014-03-13T18:00:00.000Z"
                },
                "marketCount": 1
            },
            {
                "event": {
                    "id": "27165667",
                    "name": "Al Jeel v Al Draih",
                    "countryCode": "SA",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T12:45:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27165677",
                    "name": "Daventry Town v Kettering",
                    "countryCode": "GB",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T19:45:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27160160",
                    "name": "Porto v Napoli",
                    "countryCode": "PT",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T18:00:00.000Z"
                },
                "marketCount": 84
            },
            {
                "event": {
                    "id": "27162435",
                    "name": "Bishops Stortford v Hayes And Yeading",
                    "countryCode": "GB",
                    "timezone": "GMT",
        "openDate": "2014-03-13T19:45:00.000Z"
    },
    "marketCount": 2
},
{
    "event": {
        "id": "27166333",
        "name": "Bosnia U19 v Serbia U19",
        "timezone": "GMT",
        "openDate": "2014-03-13T12:30:00.000Z"
    },
    "marketCount": 25
},
{
    "event": {
        "id": "27162436",
        "name": "Maidenhead v Gosport Borough",
        "countryCode": "GB",
        "timezone": "GMT",
        "openDate": "2014-03-13T19:45:00.000Z"
    },
    "marketCount": 20
},
{
    "event": {
        "id": "27165673",
        "name": "ASA Tel Aviv Uni (W) v FC Ramat Hasharon (W)",
        "countryCode": "IL",
        "timezone": "GMT",
        "openDate": "2014-03-13T17:15:00.000Z"
    },
    "marketCount": 20
},
{
    "event": {
        "id": "27164435",
        "name": "Forest Green v Braintree",
        "countryCode": "GB",
        "timezone": "GMT",
        "openDate": "2014-03-13T19:45:00.000Z"
    },
    "marketCount": 15
},
{
    "event": {
        "id": "27165685",
        "name": "FC Samtredia v FC Betlemi Keda",
        "countryCode": "GE",
        "timezone": "GMT",
        "openDate": "2014-03-13T11:00:00.000Z"
    },
    "marketCount": 20
},
            {
                "event": {
                    "id": "27165684",
                    "name": "FC Lokomotivi Tbilisi v FC Saburtalo Tbilisi",
                    "countryCode": "GE",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T11:00:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27165686",
                    "name": "FC Sasco Tbilisi v Matchak Khelvachauri",
                    "countryCode": "GE",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T11:00:00.000Z"
                },
                "marketCount": 18
            },
            {
                "event": {
                    "id": "27165680",
                    "name": "FAR Rabat v Maghreb Fes",
                    "countryCode": "MA",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T15:30:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27165683",
                    "name": "FC Kolkheti Khobi v Samgurali Tskaltubo",
                    "countryCode": "GE",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T11:00:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27165682",
                    "name": "FC Dila Gori II v FC Dinamo Batumi",
                    "countryCode": "GE",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T11:00:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27165693",
                    "name": "Tilbury FC v Redbridge",
                    "countryCode": "GB",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T19:45:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27165688",
                    "name": "HUJK Emmaste v Kohtla-Jarve JK Jarve",
                    "countryCode": "EE",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T17:00:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27165690",
                    "name": "M Kishronot Hadera (W) v Maccabi Holon FC
(W)",
                    "countryCode": "IL",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T17:30:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27166225",
                    "name": "Litex Lovech v Cherno More",
                    "countryCode": "BG",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T15:30:00.000Z"
                },
                "marketCount": 27
            },
            {
                "event": {
                    "id": "27162412",
                    "name": "KR Reykjavik v IA Akranes",
                    "countryCode": "IS",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T19:00:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27162473",
                    "name": "Atletico Huila v Tolima",
                    "countryCode": "CO",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T23:00:00.000Z"
                },
                "marketCount": 27
            },
            {
                "event": {
                    "id": "27162413",
                    "name": "KV v Selfoss",
                    "countryCode": "IS",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T21:00:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27165159",
                    "name": "August Town FC v Boys Town FC",
                    "countryCode": "JM",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T20:30:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27165161",
                    "name": "Bogota v CD Barranquilla",
                    "countryCode": "CO",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T20:00:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27166474",
                    "name": "Brasilia FC v Formosa",
                    "countryCode": "BR",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T19:00:00.000Z"
                },
                "marketCount": 15
            },
            {
                "event": {
                    "id": "27162538",
                    "name": "Arsenal FC v Penarol",
                    "countryCode": "AR",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T22:00:00.000Z"
                },
            "marketCount": 40
        },
        {
            "event": {
                "id": "27166478",
                "name": "Ware FC v AFC Sudbury",
                "countryCode": "GB",
                "timezone": "GMT",
                "openDate": "2014-03-13T19:45:00.000Z"
            },
            "marketCount": 15
        },
        {
            "event": {
                "id": "27165505",
                "name": "Tomsk v Tyumen",
                "countryCode": "RU",
                "timezone": "GMT",
                "openDate": "2014-03-13T11:30:00.000Z"
            },
            "marketCount": 28
        },
        {
            "event": {
                "id": "27166477",
                "name": "Needham Market FC v Thurrock",
                "countryCode": "GB",
                "timezone": "GMT",
                "openDate": "2014-03-13T19:45:00.000Z"
            },
            "marketCount": 15
        },
        {
            "event": {
                "id": "27160154",
                "name": "Lyon v Plzen",
                "countryCode": "FR",
                "timezone": "GMT",
                "openDate": "2014-03-13T20:05:00.000Z"
            },
            "marketCount": 41
        },
        {
            "event": {
                "id": "27160155",
                "name": "Ludogorets v Valencia",
                "countryCode": "BG",
                "timezone": "GMT",
                "openDate": "2014-03-13T18:00:00.000Z"
            },
            "marketCount": 84
        },
        {
```

```
                    "event": {
                        "id": "27160152",
                        "name": "Tottenham v Benfica",
                        "countryCode": "GB",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T20:05:00.000Z"
                    },
                    "marketCount": 84
                },
                {
                    "event": {
                        "id": "27162428",
                        "name": "Wadi Degla v El Shorta",
                        "countryCode": "EG",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T13:00:00.000Z"
                    },
                    "marketCount": 20
                },
                {
                    "event": {
                        "id": "27160158",
                        "name": "FC Basel v Red Bull Salzburg",
                        "countryCode": "CH",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T18:00:00.000Z"
                    },
                    "marketCount": 84
                },
                {
                    "event": {
                        "id": "27162427",
                        "name": "Ismaily v El Qanah",
                        "countryCode": "EG",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T13:00:00.000Z"
                    },
                    "marketCount": 20
                },
                {
                    "event": {
                        "id": "27160159",
                        "name": "AZ Alkmaar v Anzhi Makhachkala",
                        "countryCode": "NL",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T20:05:00.000Z"
                    },
                    "marketCount": 41
                },
                {
                    "event": {
                        "id": "27162426",
                        "name": "Al Ahly v El Entag El Harby",
                        "countryCode": "EG",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T15:30:00.000Z"
                    },
                    "marketCount": 15
                },
                {
                    "event": {
                        "id": "27160156",
                        "name": "Sevilla v Betis",
                        "countryCode": "ES",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T20:05:00.000Z"
                    },
                    "marketCount": 84
                },
                {
                    "event": {
                        "id": "27160157",
                        "name": "Juventus v Fiorentina",
                        "countryCode": "IT",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T20:05:00.000Z"
                    },
                    "marketCount": 84
                },
                {
                    "event": {
                        "id": "27166336",
                        "name": "Becamex Binh Duong U19 v Khanh Hoa U19",
                        "countryCode": "VN",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T11:00:00.000Z"
                    },
                    "marketCount": 15
                },
                {
                    "event": {
                        "id": "27163800",
                        "name": "Lokomotiv Sofia v Chernomorets Burgas",
                        "countryCode": "BG",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T12:00:00.000Z"
                    },
                    "marketCount": 20
                },
                {
                    "event": {
                        "id": "27162481",
                        "name": "Ljungskile v Torslanda",
                        "countryCode": "SE",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T18:00:00.000Z"
                        },
                        "marketCount": 27
                    },
                    {
                        "event": {
                            "id": "27166338",
                            "name": "H Ironi Petah Tikva (W) v Maccabi Beer Sheva
(W)",
                    "countryCode": "IL",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T18:15:00.000Z"
                },
                "marketCount": 15
            },
            {
                "event": {
                    "id": "27163801",
                    "name": "Concord Rangers v Havant and W",
                    "countryCode": "GB",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T19:45:00.000Z"
                },
                "marketCount": 2
            },
            {
                "event": {
                    "id": "27166340",
                    "name": "Maccabi Ironi Bat Yam v Hapoel Mahane Yehuda",
                    "countryCode": "IL",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T17:00:00.000Z"
                },
                "marketCount": 15
            },
            {
                "event": {
                    "id": "27162418",
                    "name": "Courts Young Lions v Woodlands Wellington",
                    "countryCode": "SG",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T11:30:00.000Z"
                },
                "marketCount": 20
            },
            {
                "event": {
                    "id": "27162417",
                    "name": "Balestier Khalsa v Tanjong Pagar Utd",
                    "countryCode": "SG",
                    "timezone": "GMT",
                    "openDate": "2014-03-13T11:30:00.000Z"
                },
                "marketCount": 20
```

```
        }
    ],
        "id": 1
    }
]
```

Request the Market Information for an Event

The below example demonstrates how to retrieve all the market information that belongs to an event (excluding price data). You can include one or more eventId's in the requests provide that you stay within the Market Data Limits

listMarketCatalogue Request

```
[
    {
        "jsonrpc": "2.0",
        "method": "SportsAPING/v1.0/listMarketCatalogue",
        "params": {
            "filter": {
                "eventIds": [
                    "27165685"
                ]
            },
            "maxResults": "200",
            "marketProjection": [
                "COMPETITION",
                "EVENT",
                "EVENT_TYPE",
                "RUNNER_DESCRIPTION",
                "RUNNER_METADATA",
                "MARKET_START_TIME"
            ]
        },
        "id": 1
    }
]
```

listMarketCatalogue Response

```
[
    {
        "jsonrpc": "2.0",
        "result": [
            {
                "marketId": "1.113197547",
                "marketName": "FC Betlemi Keda +1",
                "marketStartTime": "2014-03-13T11:00:00.000Z",
                "totalMatched": 12,
                "runners": [
                    {
                        "selectionId": 6843871,
                        "runnerName": "FC Betlemi Keda +1",
            "handicap": 0,
            "sortPriority": 1,
            "metadata": {
                "runnerId": "63123618"
            }
        },
        {
            "selectionId": 6830600,
            "runnerName": "FC Samtredia -1",
            "handicap": 0,
            "sortPriority": 2,
            "metadata": {
                "runnerId": "63123619"
            }
        },
        {
            "selectionId": 151478,
            "runnerName": "Draw",
            "handicap": 0,
            "sortPriority": 3,
            "metadata": {
                "runnerId": "63123620"
            }
        }
    ],
    "eventType": {
        "id": "1",
        "name": "Soccer"
    },
    "competition": {
        "id": "2356065",
        "name": "Pirveli Liga"
    },
    "event": {
        "id": "27165685",
        "name": "FC Samtredia v FC Betlemi Keda",
        "countryCode": "GE",
        "timezone": "GMT",
        "openDate": "2014-03-13T11:00:00.000Z"
    }
},
{
    "marketId": "1.113197546",
    "marketName": "FC Samtredia +1",
    "marketStartTime": "2014-03-13T11:00:00.000Z",
    "totalMatched": 0,
    "runners": [
        {
            "selectionId": 6830597,
            "runnerName": "FC Samtredia +1",
            "handicap": 0,
            "sortPriority": 1,
            "metadata": {
```

```
                                "runnerId": "63123615"
                            }
                        },
                        {
                            "selectionId": 6843874,
                            "runnerName": "FC Betlemi Keda -1",
                            "handicap": 0,
                            "sortPriority": 2,
                            "metadata": {
                                "runnerId": "63123616"
                            }
                        },
                        {
                            "selectionId": 151478,
                            "runnerName": "Draw",
                            "handicap": 0,
                            "sortPriority": 3,
                            "metadata": {
                                "runnerId": "63123617"
                            }
                        }
                    ],
                    "eventType": {
                        "id": "1",
                        "name": "Soccer"
                    },
                    "competition": {
                        "id": "2356065",
                        "name": "Pirveli Liga"
                    },
                    "event": {
                        "id": "27165685",
                        "name": "FC Samtredia v FC Betlemi Keda",
                        "countryCode": "GE",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T11:00:00.000Z"
                    }
                },
                {
                    "marketId": "1.113197492",
                    "marketName": "Total Goals",
                    "marketStartTime": "2014-03-13T11:00:00.000Z",
                    "totalMatched": 246.82,
                    "runners": [
                        {
                            "selectionId": 285469,
                            "runnerName": "1 goals or more",
                            "handicap": -1,
                            "sortPriority": 1,
                            "metadata": {
                                "runnerId": "63123486"
                            }
                        },
                        {
                            "selectionId": 285470,
                            "runnerName": "2 goals or more",
                            "handicap": -2,
                            "sortPriority": 2,
                            "metadata": {
                                "runnerId": "63123487"
                            }
                        },
                        {
                            "selectionId": 285471,
                            "runnerName": "3 goals or more",
                            "handicap": -3,
                            "sortPriority": 3,
                            "metadata": {
                                "runnerId": "63123488"
                            }
                        },
                        {
                            "selectionId": 2795170,
                            "runnerName": "4 goals or more",
                            "handicap": -4,
                            "sortPriority": 4,
                            "metadata": {
                                "runnerId": "63123489"
                            }
                        },
                        {
                            "selectionId": 285473,
                            "runnerName": "5 goals or more",
                            "handicap": -5,
                            "sortPriority": 5,
                            "metadata": {
                                "runnerId": "63123490"
                            }
                        },
                        {
                            "selectionId": 285474,
                            "runnerName": "6 goals or more",
                            "handicap": -6,
                            "sortPriority": 6,
                            "metadata": {
                                "runnerId": "63123491"
                            }
                        },
                        {
                            "selectionId": 8215951,
                            "runnerName": "7 goals or more",
                            "handicap": -7,
                            "sortPriority": 7,
                            "metadata": {
                                "runnerId": "63123492"
                            }
```

```
                }
            ],
            "eventType": {
                "id": "1",
                "name": "Soccer"
            },
            "competition": {
                "id": "2356065",
                "name": "Pirveli Liga"
            },
            "event": {
                "id": "27165685",
                "name": "FC Samtredia v FC Betlemi Keda",
                "countryCode": "GE",
                "timezone": "GMT",
                "openDate": "2014-03-13T11:00:00.000Z"
            }
        },
        {
            "marketId": "1.113197491",
            "marketName": "Match Odds",
            "marketStartTime": "2014-03-13T11:00:00.000Z",
            "totalMatched": 7707.52,
            "runners": [
                {
                    "selectionId": 6830593,
                    "runnerName": "FC Samtredia",
                    "handicap": 0,
                    "sortPriority": 1,
                    "metadata": {
                        "runnerId": "63123483"
                    }
                },
                {
                    "selectionId": 6843866,
                    "runnerName": "FC Betlemi Keda",
                    "handicap": 0,
                    "sortPriority": 2,
                    "metadata": {
                        "runnerId": "63123484"
                    }
                },
                {
                    "selectionId": 58805,
                    "runnerName": "The Draw",
                    "handicap": 0,
                    "sortPriority": 3,
                    "metadata": {
                        "runnerId": "63123485"
                    }
                }
            ],
            "eventType": {
                "id": "1",
                "name": "Soccer"
            },
            "competition": {
                "id": "2356065",
                "name": "Pirveli Liga"
            },
            "event": {
                "id": "27165685",
                "name": "FC Samtredia v FC Betlemi Keda",
                "countryCode": "GE",
                "timezone": "GMT",
                "openDate": "2014-03-13T11:00:00.000Z"
            }
        },
        {
            "marketId": "1.113197550",
            "marketName": "Both teams to Score?",
            "marketStartTime": "2014-03-13T11:00:00.000Z",
            "totalMatched": 14.78,
            "runners": [
                {
                    "selectionId": 30246,
                    "runnerName": "Yes",
                    "handicap": 0,
                    "sortPriority": 1,
                    "metadata": {
                        "runnerId": "63123625"
                    }
                },
                {
                    "selectionId": 30247,
                    "runnerName": "No",
                    "handicap": 0,
                    "sortPriority": 2,
                    "metadata": {
                        "runnerId": "63123626"
                    }
                }
            ],
            "eventType": {
                "id": "1",
                "name": "Soccer"
            },
            "competition": {
                "id": "2356065",
                "name": "Pirveli Liga"
            },
            "event": {
                "id": "27165685",
                "name": "FC Samtredia v FC Betlemi Keda",
                "countryCode": "GE",
                "timezone": "GMT",
                "openDate": "2014-03-13T11:00:00.000Z"
            }
        },
        {
            "marketId": "1.113197501",
            "marketName": "Next Goal",
            "marketStartTime": "2014-03-13T11:00:00.000Z",
            "totalMatched": 3.34,
            "runners": [
                {
                    "selectionId": 6830593,
                    "runnerName": "FC Samtredia",
                    "handicap": 0,
                    "sortPriority": 1,
                    "metadata": {
                        "runnerId": "63123495"
                    }
                },
                {
                    "selectionId": 6843866,
                    "runnerName": "FC Betlemi Keda",
                    "handicap": 0,
                    "sortPriority": 2,
                    "metadata": {
                        "runnerId": "63123496"
                    }
                },
                {
                    "selectionId": 69852,
                    "runnerName": "No Goal",
                    "handicap": 0,
                    "sortPriority": 3,
                    "metadata": {
                        "runnerId": "63123497"
                    }
                }
            ],
            "eventType": {
                "id": "1",
                "name": "Soccer"
            },
            "competition": {
                "id": "2356065",
                "name": "Pirveli Liga"
            },
            "event": {
                "id": "27165685",
                "name": "FC Samtredia v FC Betlemi Keda",
                "countryCode": "GE",
                "timezone": "GMT",
                "openDate": "2014-03-13T11:00:00.000Z"
            }
        },
                {
                    "marketId": "1.113197502",
                    "marketName": "Over/Under 6.5 Goals",
                    "marketStartTime": "2014-03-13T11:00:00.000Z",
                    "totalMatched": 1255.79,
                    "runners": [
                        {
                            "selectionId": 2542448,
                            "runnerName": "Under 6.5 Goals",
                            "handicap": 0,
                            "sortPriority": 1,
                            "metadata": {
                                "runnerId": "63123498"
                            }
                        },
                        {
                            "selectionId": 2542449,
                            "runnerName": "Over 6.5 Goals",
                            "handicap": 0,
                            "sortPriority": 2,
                            "metadata": {
                                "runnerId": "63123499"
                            }
                        }
                    ],
                    "eventType": {
                        "id": "1",
                        "name": "Soccer"
                    },
                    "competition": {
                        "id": "2356065",
                        "name": "Pirveli Liga"
                    },
                    "event": {
                        "id": "27165685",
                        "name": "FC Samtredia v FC Betlemi Keda",
                        "countryCode": "GE",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T11:00:00.000Z"
                    }
                },
                {
                    "marketId": "1.113197505",
                    "marketName": "Correct Score",
                    "marketStartTime": "2014-03-13T11:00:00.000Z",
                    "totalMatched": 2380.92,
                    "runners": [
                        {
                            "selectionId": 1,
                            "runnerName": "0 - 0",
                            "handicap": 0,
                            "sortPriority": 1,
                            "metadata": {
                                "runnerId": "63123505"
                            }
                        },
                        {
                            "selectionId": 4,
                            "runnerName": "0 - 1",
                            "handicap": 0,
                            "sortPriority": 2,
                            "metadata": {
                                "runnerId": "63123506"
                            }
                        },
                        {
                            "selectionId": 9,
                            "runnerName": "0 - 2",
                            "handicap": 0,
                            "sortPriority": 3,
                            "metadata": {
                                "runnerId": "63123507"
                            }
                        },
                        {
                            "selectionId": 16,
                            "runnerName": "0 - 3",
                            "handicap": 0,
                            "sortPriority": 4,
                            "metadata": {
                                "runnerId": "63123508"
                            }
                        },
                        {
                            "selectionId": 2,
                            "runnerName": "1 - 0",
                            "handicap": 0,
                            "sortPriority": 5,
                            "metadata": {
                                "runnerId": "63123509"
                            }
                        },
                        {
                            "selectionId": 3,
                            "runnerName": "1 - 1",
                            "handicap": 0,
                            "sortPriority": 6,
                            "metadata": {
                                "runnerId": "63123510"
                            }
                        },
                        {
                            "selectionId": 8,
                            "runnerName": "1 - 2",
                            "handicap": 0,
                            "sortPriority": 7,
                            "metadata": {
                                "runnerId": "63123511"
                            }
                        },
                        {
                            "selectionId": 15,
                            "runnerName": "1 - 3",
                            "handicap": 0,
                            "sortPriority": 8,
                            "metadata": {
                                "runnerId": "63123512"
                            }
                        },
                        {
                            "selectionId": 5,
                            "runnerName": "2 - 0",
                            "handicap": 0,
                            "sortPriority": 9,
                            "metadata": {
                                "runnerId": "63123513"
                            }
                        },
                        {
                            "selectionId": 6,
                            "runnerName": "2 - 1",
                            "handicap": 0,
                            "sortPriority": 10,
                            "metadata": {
                                "runnerId": "63123514"
                            }
                        },
                        {
                            "selectionId": 7,
                            "runnerName": "2 - 2",
                            "handicap": 0,
                            "sortPriority": 11,
                            "metadata": {
                                "runnerId": "63123515"
                            }
                        },
                        {
                            "selectionId": 14,
                            "runnerName": "2 - 3",
                            "handicap": 0,
                            "sortPriority": 12,
                            "metadata": {
                                "runnerId": "63123516"
                            }
                        },
                        {
                            "selectionId": 10,
                            "runnerName": "3 - 0",
                            "handicap": 0,
                                "sortPriority": 13,
                                "metadata": {
                                    "runnerId": "63123517"
                                }
                            },
                            {
                                "selectionId": 11,
                                "runnerName": "3 - 1",
                                "handicap": 0,
                                "sortPriority": 14,
                                "metadata": {
                                    "runnerId": "63123518"
                                }
                            },
                            {
                                "selectionId": 12,
                                "runnerName": "3 - 2",
                                "handicap": 0,
                                "sortPriority": 15,
                                "metadata": {
                                    "runnerId": "63123519"
                                }
                            },
                            {
                                "selectionId": 13,
                                "runnerName": "3 - 3",
                                "handicap": 0,
                                "sortPriority": 16,
                                "metadata": {
                                    "runnerId": "63123520"
                                }
                            },
                            {
                                "selectionId": 4506345,
                                "runnerName": "Any Unquoted ",
                                "handicap": 0,
                                "sortPriority": 17,
                                "metadata": {
                                    "runnerId": "63123521"
                                }
                            }
                        ],
                        "eventType": {
                            "id": "1",
                            "name": "Soccer"
                        },
                        "competition": {
                            "id": "2356065",
                            "name": "Pirveli Liga"
                        },
                        "event": {
                            "id": "27165685",
                            "name": "FC Samtredia v FC Betlemi Keda",
                                "countryCode": "GE",
                                "timezone": "GMT",
                                "openDate": "2014-03-13T11:00:00.000Z"
                            }
                        },
                        {
                            "marketId": "1.113197506",
                            "marketName": "Over/Under 2.5 Goals",
                            "marketStartTime": "2014-03-13T11:00:00.000Z",
                            "totalMatched": 4149.36,
                            "runners": [
                                {
                                    "selectionId": 47972,
                                    "runnerName": "Under 2.5 Goals",
                                    "handicap": 0,
                                    "sortPriority": 1,
                                    "metadata": {
                                        "runnerId": "63123522"
                                    }
                                },
                                {
                                    "selectionId": 47973,
                                    "runnerName": "Over 2.5 Goals",
                                    "handicap": 0,
                                    "sortPriority": 2,
                                    "metadata": {
                                        "runnerId": "63123523"
                                    }
                                }
                            ],
                            "eventType": {
                                "id": "1",
                                "name": "Soccer"
                            },
                            "competition": {
                                "id": "2356065",
                                "name": "Pirveli Liga"
                            },
                            "event": {
                                "id": "27165685",
                                "name": "FC Samtredia v FC Betlemi Keda",
                                "countryCode": "GE",
                                "timezone": "GMT",
                                "openDate": "2014-03-13T11:00:00.000Z"
                            }
                        },
                        {
                            "marketId": "1.113197504",
                            "marketName": "Over/Under 5.5 Goals",
                            "marketStartTime": "2014-03-13T11:00:00.000Z",
                            "totalMatched": 2216.24,
                            "runners": [
                                {
```

```
                "selectionId": 1485567,
                "runnerName": "Under 5.5 Goals",
                "handicap": 0,
                "sortPriority": 1,
                "metadata": {
                    "runnerId": "63123503"
                }
            },
            {
                "selectionId": 1485568,
                "runnerName": "Over 5.5 Goals",
                "handicap": 0,
                "sortPriority": 2,
                "metadata": {
                    "runnerId": "63123504"
                }
            }
        ],
        "eventType": {
            "id": "1",
            "name": "Soccer"
        },
        "competition": {
            "id": "2356065",
            "name": "Pirveli Liga"
        },
        "event": {
            "id": "27165685",
            "name": "FC Samtredia v FC Betlemi Keda",
            "countryCode": "GE",
            "timezone": "GMT",
            "openDate": "2014-03-13T11:00:00.000Z"
        }
    },
    {
        "marketId": "1.113197509",
        "marketName": "Half Time/Full Time",
        "marketStartTime": "2014-03-13T11:00:00.000Z",
        "totalMatched": 97.3,
        "runners": [
            {
                "selectionId": 6830596,
                "runnerName": "FC Samtredia/FC Samtredia",
                "handicap": 0,
                "sortPriority": 1,
                "metadata": {
                    "runnerId": "63123536"
                }
            },
            {
                "selectionId": 6830599,
                "runnerName": "FC Samtredia/Draw",
                "handicap": 0,
                "sortPriority": 2,
                "metadata": {
                    "runnerId": "63123537"
                }
            },
            {
                "selectionId": 8380726,
                "runnerName": "FC Samtredia/FC Betlemi Ked",
                "handicap": 0,
                "sortPriority": 3,
                "metadata": {
                    "runnerId": "63123538"
                }
            },
            {
                "selectionId": 6830595,
                "runnerName": "Draw/FC Samtredia",
                "handicap": 0,
                "sortPriority": 4,
                "metadata": {
                    "runnerId": "63123539"
                }
            },
            {
                "selectionId": 3710152,
                "runnerName": "Draw/Draw",
                "handicap": 0,
                "sortPriority": 5,
                "metadata": {
                    "runnerId": "63123540"
                }
            },
            {
                "selectionId": 6843869,
                "runnerName": "Draw/FC Betlemi Ked",
                "handicap": 0,
                "sortPriority": 6,
                "metadata": {
                    "runnerId": "63123541"
                }
            },
            {
                "selectionId": 8380727,
                "runnerName": "FC Betlemi Ked/FC Samtredia",
                "handicap": 0,
                "sortPriority": 7,
                "metadata": {
                    "runnerId": "63123542"
                }
            },
            {
                "selectionId": 6843873,
                "runnerName": "FC Betlemi Ked/Draw",
                "handicap": 0,
                "sortPriority": 8,
                "metadata": {
                    "runnerId": "63123543"
                }
            },
            {
                "selectionId": 6843870,
                "runnerName": "FC Betlemi Ked/FC Betlemi Ked",
                "handicap": 0,
                "sortPriority": 9,
                "metadata": {
                    "runnerId": "63123544"
                }
            }
        ],
        "eventType": {
            "id": "1",
            "name": "Soccer"
        },
        "competition": {
            "id": "2356065",
            "name": "Pirveli Liga"
        },
        "event": {
            "id": "27165685",
            "name": "FC Samtredia v FC Betlemi Keda",
            "countryCode": "GE",
            "timezone": "GMT",
            "openDate": "2014-03-13T11:00:00.000Z"
        }
    },
    {
        "marketId": "1.113197510",
        "marketName": "Over/Under 1.5 Goals",
        "marketStartTime": "2014-03-13T11:00:00.000Z",
        "totalMatched": 1879.69,
        "runners": [
            {
                "selectionId": 1221385,
                "runnerName": "Under 1.5 Goals",
                "handicap": 0,
                "sortPriority": 1,
                "metadata": {
                    "runnerId": "63123545"
                }
            },
            {
                "selectionId": 1221386,
                "runnerName": "Over 1.5 Goals",
                "handicap": 0,
                "sortPriority": 2,
                "metadata": {
```

```
                        "runnerId": "63123546"
                    }
                }
            ],
            "eventType": {
                "id": "1",
                "name": "Soccer"
            },
            "competition": {
                "id": "2356065",
                "name": "Pirveli Liga"
            },
            "event": {
                "id": "27165685",
                "name": "FC Samtredia v FC Betlemi Keda",
                "countryCode": "GE",
                "timezone": "GMT",
                "openDate": "2014-03-13T11:00:00.000Z"
            }
        },
        {
            "marketId": "1.113197507",
            "marketName": "Over/Under 4.5 Goals",
            "marketStartTime": "2014-03-13T11:00:00.000Z",
            "totalMatched": 3189.28,
            "runners": [
                {
                    "selectionId": 1222347,
                    "runnerName": "Under 4.5 Goals",
                    "handicap": 0,
                    "sortPriority": 1,
                    "metadata": {
                        "runnerId": "63123524"
                    }
                },
                {
                    "selectionId": 1222346,
                    "runnerName": "Over 4.5 Goals",
                    "handicap": 0,
                    "sortPriority": 2,
                    "metadata": {
                        "runnerId": "63123525"
                    }
                }
            ],
            "eventType": {
                "id": "1",
                "name": "Soccer"
            },
            "competition": {
                "id": "2356065",
                "name": "Pirveli Liga"
            },
            "event": {
                "id": "27165685",
                "name": "FC Samtredia v FC Betlemi Keda",
                "countryCode": "GE",
                "timezone": "GMT",
                "openDate": "2014-03-13T11:00:00.000Z"
            }
        },
        {
            "marketId": "1.113197511",
            "marketName": "Over/Under 3.5 Goals",
            "marketStartTime": "2014-03-13T11:00:00.000Z",
            "totalMatched": 3934.41,
            "runners": [
                {
                    "selectionId": 1222344,
                    "runnerName": "Under 3.5 Goals",
                    "handicap": 0,
                    "sortPriority": 1,
                    "metadata": {
                        "runnerId": "63123547"
                    }
                },
                {
                    "selectionId": 1222345,
                    "runnerName": "Over 3.5 Goals",
                    "handicap": 0,
                    "sortPriority": 2,
                    "metadata": {
                        "runnerId": "63123548"
                    }
                }
            ],
            "eventType": {
                "id": "1",
                "name": "Soccer"
            },
            "competition": {
                "id": "2356065",
                "name": "Pirveli Liga"
            },
            "event": {
                "id": "27165685",
                "name": "FC Samtredia v FC Betlemi Keda",
                "countryCode": "GE",
                "timezone": "GMT",
                "openDate": "2014-03-13T11:00:00.000Z"
            }
        },
        {
            "marketId": "1.113197512",
            "marketName": "Asian Handicap",
            "marketStartTime": "2014-03-13T11:00:00.000Z",
                    "totalMatched": 1599.38,
                    "runners": [
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": -4,
                            "sortPriority": 1
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": 4,
                            "sortPriority": 2
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": -3.75,
                            "sortPriority": 3
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": 3.75,
                            "sortPriority": 4
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": -3.5,
                            "sortPriority": 5
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": 3.5,
                            "sortPriority": 6
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": -3.25,
                            "sortPriority": 7
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": 3.25,
                            "sortPriority": 8
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                                "handicap": -3,
                                "sortPriority": 9
                            },
                            {
                                "selectionId": 6843866,
                                "runnerName": "FC Betlemi Keda",
                                "handicap": 3,
                                "sortPriority": 10
                            },
                            {
                                "selectionId": 6830593,
                                "runnerName": "FC Samtredia",
                                "handicap": -2.75,
                                "sortPriority": 11
                            },
                            {
                                "selectionId": 6843866,
                                "runnerName": "FC Betlemi Keda",
                                "handicap": 2.75,
                                "sortPriority": 12
                            },
                            {
                                "selectionId": 6830593,
                                "runnerName": "FC Samtredia",
                                "handicap": -2.5,
                                "sortPriority": 13
                            },
                            {
                                "selectionId": 6843866,
                                "runnerName": "FC Betlemi Keda",
                                "handicap": 2.5,
                                "sortPriority": 14
                            },
                            {
                                "selectionId": 6830593,
                                "runnerName": "FC Samtredia",
                                "handicap": -2.25,
                                "sortPriority": 15
                            },
                            {
                                "selectionId": 6843866,
                                "runnerName": "FC Betlemi Keda",
                                "handicap": 2.25,
                                "sortPriority": 16
                            },
                            {
                                "selectionId": 6830593,
                                "runnerName": "FC Samtredia",
                                "handicap": -2,
                                "sortPriority": 17
                            },
                            {
                                "selectionId": 6843866,
                        "runnerName": "FC Betlemi Keda",
                        "handicap": 2,
                        "sortPriority": 18
                    },
                    {
                        "selectionId": 6830593,
                        "runnerName": "FC Samtredia",
                        "handicap": -1.75,
                        "sortPriority": 19
                    },
                    {
                        "selectionId": 6843866,
                        "runnerName": "FC Betlemi Keda",
                        "handicap": 1.75,
                        "sortPriority": 20
                    },
                    {
                        "selectionId": 6830593,
                        "runnerName": "FC Samtredia",
                        "handicap": -1.5,
                        "sortPriority": 21
                    },
                    {
                        "selectionId": 6843866,
                        "runnerName": "FC Betlemi Keda",
                        "handicap": 1.5,
                        "sortPriority": 22
                    },
                    {
                        "selectionId": 6830593,
                        "runnerName": "FC Samtredia",
                        "handicap": -1.25,
                        "sortPriority": 23
                    },
                    {
                        "selectionId": 6843866,
                        "runnerName": "FC Betlemi Keda",
                        "handicap": 1.25,
                        "sortPriority": 24
                    },
                    {
                        "selectionId": 6830593,
                        "runnerName": "FC Samtredia",
                        "handicap": -1,
                        "sortPriority": 25
                    },
                    {
                        "selectionId": 6843866,
                        "runnerName": "FC Betlemi Keda",
                        "handicap": 1,
                        "sortPriority": 26
                    },
                    {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": -0.75,
                            "sortPriority": 27
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": 0.75,
                            "sortPriority": 28
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": -0.5,
                            "sortPriority": 29
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": 0.5,
                            "sortPriority": 30
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": -0.25,
                            "sortPriority": 31
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": 0.25,
                            "sortPriority": 32
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": 0,
                            "sortPriority": 33
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": 0,
                            "sortPriority": 34
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": 0.25,
                            "sortPriority": 35
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": -0.25,
                            "sortPriority": 36
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": 0.5,
                            "sortPriority": 37
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": -0.5,
                            "sortPriority": 38
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": 0.75,
                            "sortPriority": 39
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": -0.75,
                            "sortPriority": 40
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": 1,
                            "sortPriority": 41
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": -1,
                            "sortPriority": 42
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": 1.25,
                            "sortPriority": 43
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": -1.25,
                            "sortPriority": 44
```

```
                },
                {
                    "selectionId": 6830593,
                    "runnerName": "FC Samtredia",
                    "handicap": 1.5,
                    "sortPriority": 45
                },
                {
                    "selectionId": 6843866,
                    "runnerName": "FC Betlemi Keda",
                    "handicap": -1.5,
                    "sortPriority": 46
                },
                {
                    "selectionId": 6830593,
                    "runnerName": "FC Samtredia",
                    "handicap": 1.75,
                    "sortPriority": 47
                },
                {
                    "selectionId": 6843866,
                    "runnerName": "FC Betlemi Keda",
                    "handicap": -1.75,
                    "sortPriority": 48
                },
                {
                    "selectionId": 6830593,
                    "runnerName": "FC Samtredia",
                    "handicap": 2,
                    "sortPriority": 49
                },
                {
                    "selectionId": 6843866,
                    "runnerName": "FC Betlemi Keda",
                    "handicap": -2,
                    "sortPriority": 50
                },
                {
                    "selectionId": 6830593,
                    "runnerName": "FC Samtredia",
                    "handicap": 2.25,
                    "sortPriority": 51
                },
                {
                    "selectionId": 6843866,
                    "runnerName": "FC Betlemi Keda",
                    "handicap": -2.25,
                    "sortPriority": 52
                },
                {
                    "selectionId": 6830593,
                    "runnerName": "FC Samtredia",
                    "handicap": 2.5,
                        "sortPriority": 53
                    },
                    {
                        "selectionId": 6843866,
                        "runnerName": "FC Betlemi Keda",
                        "handicap": -2.5,
                        "sortPriority": 54
                    },
                    {
                        "selectionId": 6830593,
                        "runnerName": "FC Samtredia",
                        "handicap": 2.75,
                        "sortPriority": 55
                    },
                    {
                        "selectionId": 6843866,
                        "runnerName": "FC Betlemi Keda",
                        "handicap": -2.75,
                        "sortPriority": 56
                    },
                    {
                        "selectionId": 6830593,
                        "runnerName": "FC Samtredia",
                        "handicap": 3,
                        "sortPriority": 57
                    },
                    {
                        "selectionId": 6843866,
                        "runnerName": "FC Betlemi Keda",
                        "handicap": -3,
                        "sortPriority": 58
                    },
                    {
                        "selectionId": 6830593,
                        "runnerName": "FC Samtredia",
                        "handicap": 3.25,
                        "sortPriority": 59
                    },
                    {
                        "selectionId": 6843866,
                        "runnerName": "FC Betlemi Keda",
                        "handicap": -3.25,
                        "sortPriority": 60
                    },
                    {
                        "selectionId": 6830593,
                        "runnerName": "FC Samtredia",
                        "handicap": 3.5,
                        "sortPriority": 61
                    },
                    {
                        "selectionId": 6843866,
                        "runnerName": "FC Betlemi Keda",
                            "handicap": -3.5,
                            "sortPriority": 62
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": 3.75,
                            "sortPriority": 63
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": -3.75,
                            "sortPriority": 64
                        },
                        {
                            "selectionId": 6830593,
                            "runnerName": "FC Samtredia",
                            "handicap": 4,
                            "sortPriority": 65,
                            "metadata": {
                                "runnerId": "63123613"
                            }
                        },
                        {
                            "selectionId": 6843866,
                            "runnerName": "FC Betlemi Keda",
                            "handicap": -4,
                            "sortPriority": 66,
                            "metadata": {
                                "runnerId": "63123614"
                            }
                        }
                    ],
                    "eventType": {
                        "id": "1",
                        "name": "Soccer"
                    },
                    "competition": {
                        "id": "2356065",
                        "name": "Pirveli Liga"
                    },
                    "event": {
                        "id": "27165685",
                        "name": "FC Samtredia v FC Betlemi Keda",
                        "countryCode": "GE",
                        "timezone": "GMT",
                        "openDate": "2014-03-13T11:00:00.000Z"
                    }
                }
            ],
            "id": 1
        }
    ]
```

Horse Racing - Today's Win & Place Markets

The below request is an example of retrieving the available win/place horse racing markets for a specific day. The marketStartTime (from & to) should be updated accordingly.

listMarketCatalogue Request

```
[
    {
        "jsonrpc": "2.0",
        "method": "SportsAPING/v1.0/listMarketCatalogue",
        "params": {
            "filter": {
                "eventTypeIds": [
                    "7"
                ],
                "marketTypeCodes": [
                    "WIN",
                    "PLACE"
                ],
                "marketStartTime": {
                    "from": "2013-10-16T00:00:00Z",
                    "to": "2013-10-16T23:59:00Z"
                }
            },
            "maxResults": "200",
            "marketProjection": [
                "MARKET_START_TIME",
                "RUNNER_METADATA",
                "RUNNER_DESCRIPTION",
                "EVENT_TYPE",
                "EVENT",
                "COMPETITION"
            ]
        },
        "id": 1
    }
]
```

Football Competitions

To retrieve all football competitions, you call the eventTypeId; operation with the following market filter:

listCompetitions Request

```
{
   "params": {
      "filter": {
         "eventTypeIds": [1]
      }
   },
   "jsonrpc": "2.0",
   "method": "SportsAPING/v1.0/listCompetitions",
   "id": 1
}
```

The "filter" selects all markets that have an eventTypeId of 1 (which is the event type for Football).

Then returns a list of Competitions and their Ids and how many markets are in each competition which are associated with those markets:

listCompetitions Response

```
{
   "jsonrpc": "2.0",
   "result": [
      {
         "marketCount": 16,
         "competition": {
            "id": "833222",
            "name": "Turkish Division 2"
         }
      },
      {
         "marketCount": 127,
         "competition": {
            "id": "5",
            "name": "A-League 2012/13"
         }
      },
      {
         "marketCount": 212,
         "competition": {
            "id": "7",
            "name": "Austrian Bundesliga"
         }
      },
      {
         "marketCount": 243,
         "competition": {
            "id": "11",
            "name": "Dutch Jupiler League"
         }
      },
      {
         "marketCount": 206,
         "competition": {
            "id": "26207",
            "name": "Greek Cup"
         }
      },
      {
         "marketCount": 117,
         "competition": {
            "id": "2129602",
            "name": "Professional Development League"
         }
      },
      {
         "marketCount": 68,
         "competition": {
            "id": "803237",
            "name": "Argentinian Primera B"
         }
      },
      {
         "marketCount": 1,
         "competition": {
            "id": "1842928",
            "name": "OTB Bank Liga"
         }
      }
```

```
    ],
    "id": 1
}
```

Python Example

```
import requests
import json

url="https://api.betfair.com/betting/json-rpc"
header = { 'X-Application' : "APP_KEY_HERE",  'X-Authentication :
'SESSION_TOKEN', 'content-type' : 'application/json' }


jsonrpc_req='{"jsonrpc": "2.0", "method":
"SportsAPING/v1.0/listCompetitions", "params": {"filter":{ "eventTypeIds" :
[1]  }}, "id": 1}'


print json.dumps(json.loads(jsonrpc_req), indent=3)
print " "



response = requests.post(url, data=jsonrpc_req, headers=header)

print json.dumps(json.loads(response.text), indent=3)
```

Market Prices

Once you have identified the market (marketId) that your interested in using the listMarketCatalogue service, you can request prices for that market using the listMarketBook API call.

This is an example showing a request for the best prices (back and lay) and trading volume including virtual bets.

listMarketBook Request

```
[{
 "jsonrpc": "2.0",
 "method": "SportsAPING/v1.0/listMarketBook",
 "params": {
  "marketIds": ["1.127771425"],
  "priceProjection": {
   "priceData": ["EX_BEST_OFFERS", "EX_TRADED"],
   "virtualise": "true"
  }
 },
 "id": 1
}][{
 "jsonrpc": "2.0",
 "result": [{
  "marketId": "1.127771425",
  "isMarketDataDelayed": false,
  "status": "OPEN",
  "betDelay": 0,
  "bspReconciled": false,
  "complete": true,
  "inplay": false,
  "numberOfWinners": 1,
  "numberOfRunners": 3,
  "numberOfActiveRunners": 3,
  "lastMatchTime": "2016-10-28T12:25:30.235Z",
  "totalMatched": 188959.22,
  "totalAvailable": 172932.96,
  "crossMatching": true,
  "runnersVoidable": false,
  "version": 1469071410,
  "runners": [{
   "selectionId": 44790,
   "handicap": 0.0,
   "status": "ACTIVE",
   "lastPriceTraded": 7.0,
   "totalMatched": 12835.46,
   "ex": {
    "availableToBack": [{
     "price": 7.0,
     "size": 75.53
    }, {
     "price": 6.8,
     "size": 538.6
    }, {
     "price": 6.6,
     "size": 612.2
    }],
    "availableToLay": [{
     "price": 7.2,
     "size": 152.12
    }, {
     "price": 7.4,
     "size": 1446.28
    }, {
     "price": 7.6,
     "size": 1250.26
    }],
    "tradedVolume": [{
     "price": 6.4,
     "size": 3.0
    }, {
     "price": 6.6,
   "size": 3.59
  }, {
   "price": 6.8,
   "size": 1.42
  }, {
   "price": 7.0,
   "size": 1736.55
  }, {
   "price": 7.2,
   "size": 1601.31
  }, {
   "price": 7.4,
   "size": 3580.1
  }, {
   "price": 7.6,
   "size": 4236.59
  }, {
   "price": 7.8,
   "size": 1367.18
  }, {
   "price": 8.0,
   "size": 305.29
  }, {
   "price": 8.2,
   "size": 0.39
  }]
 }
}, {
 "selectionId": 489720,
 "handicap": 0.0,
 "status": "ACTIVE",
 "lastPriceTraded": 1.63,
 "totalMatched": 163020.94,
 "ex": {
  "availableToBack": [{
   "price": 1.62,
   "size": 4921.06
  }, {
   "price": 1.61,
   "size": 3230.34
  }, {
   "price": 1.6,
   "size": 2237.71
  }],
  "availableToLay": [{
   "price": 1.63,
   "size": 1001.76
  }, {
   "price": 1.64,
   "size": 6737.59
  }, {
   "price": 1.65,
   "size": 1701.27
```

```
      }],
      "tradedVolume": [{
       "price": 1.53,
       "size": 31.38
      }, {
       "price": 1.54,
       "size": 3.77
      }, {
       "price": 1.57,
       "size": 3582.76
      }, {
       "price": 1.58,
       "size": 12037.21
      }, {
       "price": 1.59,
       "size": 16530.57
      }, {
       "price": 1.6,
       "size": 54607.84
      }, {
       "price": 1.61,
       "size": 36015.53
      }, {
       "price": 1.62,
       "size": 21108.23
      }, {
       "price": 1.63,
       "size": 17575.94
      }, {
       "price": 1.64,
       "size": 1527.67
      }]
     }
    }, {
     "selectionId": 58805,
     "handicap": 0.0,
     "status": "ACTIVE",
     "lastPriceTraded": 4.2,
     "totalMatched": 13102.81,
     "ex": {
      "availableToBack": [{
       "price": 4.1,
       "size": 3243.85
      }, {
       "price": 4.0,
       "size": 1158.17
      }, {
       "price": 3.95,
       "size": 254.04
      }],
      "availableToLay": [{
       "price": 4.2,
       "size": 1701.15
```

```
  }, {
   "price": 4.3,
   "size": 3072.55
  }, {
   "price": 4.4,
   "size": 2365.76
  }],
  "tradedVolume": [{
   "price": 4.0,
   "size": 457.79
  }, {
   "price": 4.1,
   "size": 4434.67
  }, {
   "price": 4.2,
   "size": 7845.01
  }, {
   "price": 4.3,
   "size": 354.7
  }, {
   "price": 4.4,
   "size": 6.6
  }, {
   "price": 4.9,
   "size": 4.0
  }]
 }
}]
```

```
 }],
 "id": 1
}]
```

Placing a Bet

To place a bet you require the marketId and selectionId parameters from the listMarketCatalogue API call. The below parameters will place a normal Exchange bet at odds of 3.0 for a stake of £2.0.

If the bet is placed successfully, a betId is returned in the placeOrders response

placeOrders Request

```
[
    {
        "jsonrpc": "2.0",
        "method": "SportsAPING/v1.0/placeOrders",
        "params": {
            "marketId": "1.109850906",
            "instructions": [
                {
                    "selectionId": "237486",
                    "handicap": "0",
                    "side": "LAY",
                    "orderType": "LIMIT",
                    "limitOrder": {
                        "size": "2",
                        "price": "3",
                        "persistenceType": "LAPSE"
                    }
                }
            ]
        },
        "id": 1
    }
]
```

placeOrder Response

```
[
    {
        "jsonrpc": "2.0",
        "result": {
            "marketId": "1.109850906",
            "instructionReports": [
                {
                    "instruction": {
                        "selectionId": 237486,
                        "handicap": 0,
                        "limitOrder": {
                            "size": 2,
                            "price": 3,
                            "persistenceType": "LAPSE"
                        },
                        "orderType": "LIMIT",
                        "side": "LAY"
                    },
                    "betId": "31242604945",
                    "placedDate": "2013-10-30T14:22:47.000Z",
                    "averagePriceMatched": 0,
                    "sizeMatched": 0,
                    "status": "SUCCESS"
                }
            ],
            "status": "SUCCESS"
        },
        "id": 1
    }
]
```

Placing a Betfair SP Bet

To place a bet on a selection at Betfair SP, you need to specify the parameters below in the placeOrders request. The below example would place a Betfair SP back bet on the required selection for a stake of £2.00.

Request

placeOrders Request

```
[
    {
        "jsonrpc": "2.0",
        "method": "SportsAPING/v1.0/placeOrders",
        "params": {
            "marketId": "1.111836557",
            "instructions": [
                {
                    "selectionId": "5404312",
                    "handicap": "0",
                    "side": "BACK",
                    "orderType": "MARKET_ON_CLOSE",
                    "marketOnCloseOrder": {
                        "liability": "2"
                    }
                }
            ]
        },
        "id": 1
    }
]
```

placeOrders Response

```
[
    {
        "jsonrpc": "2.0",
        "result": {
            "marketId": "1.111836557",
            "instructionReports": [
                {
                    "instruction": {
                        "selectionId": 5404312,
                        "handicap": 0,
                        "marketOnCloseOrder": {
                            "liability": 2
                        },
                        "orderType": "MARKET_ON_CLOSE",
                        "side": "BACK"
                    },
                    "betId": "31645233727",
                    "placedDate": "2013-11-12T12:07:29.000Z",
                    "status": "SUCCESS"
                }
            ],
            "status": "SUCCESS"
        },
        "id": 1
    }
]
```

Retrieving Details of Bet/s Placed on a Market/s

You can use the listCurrentOrders request to retrieve of a bet/s placed on a specific market/s.

listCurrentOrders request

```
[{"jsonrpc": "2.0", "method": "SportsAPING/v1.0/listCurrentOrders",
"params":
{"marketIds":["1.117020524"],"orderProjection":"ALL","dateRange":{}}, "id":
1}]
```

```
[{"jsonrpc":"2.0","result":{"currentOrders":[{"betId":"45496907354","marke
tId":"1.117020524","selectionId":9170340,"handicap":0.0,"priceSize":{"pric
e":10.0,"size":5.0},"bspLiability":0.0,"side":"BACK","status":"EXECUTABLE"
,"persistenceType":"LAPSE","orderType":"LIMIT","placedDate":"2015-01-22T13
:01:53.000Z","averagePriceMatched":0.0,"sizeMatched":0.0,"sizeRemaining":5
.0,"sizeLapsed":0.0,"sizeCancelled":0.0,"sizeVoided":0.0,"regulatorCode":"
GIBRALTAR REGULATOR"}],"moreAvailable":false},"id":1}]
```

Retrieving the Result of a Settled Market

To retrieve the result of a settled market simply make a request to listMarketBook after the market has been settled. The response will indicate whether the selection was settled as a 'WINNER' or 'LOSER' in the runners 'status' field. Settled market information is available for 90 days after settlement.

listMarketBook Request to a Settled market

```
[{"jsonrpc": "2.0", "method": "SportsAPING/v1.0/listMarketBook", "params":
{"marketIds":["1.114363660"],"priceProjection":{"priceData":["EX_BEST_OFFE
RS"]}}, "id": 1}]
```

listMarketBook Response

```
[
    {
        "jsonrpc": "2.0",
        "result": [
            {
                "marketId": "1.114363660",
                "isMarketDataDelayed": false,
                "status": "CLOSED",
                "betDelay": 0,
                "bspReconciled": false,
                "complete": true,
                "inplay": false,
                "numberOfWinners": 1,
                "numberOfRunners": 14,
                "numberOfActiveRunners": 0,
                "totalMatched": 0,
                "totalAvailable": 0,
                "crossMatching": false,
                "runnersVoidable": false,
                "version": 767398001,
                "runners": [
                    {
                        "selectionId": 8611526,
                        "handicap": 0,
                        "status": "REMOVED",
                        "adjustmentFactor": 9.1,
                        "removalDate": "2014-06-13T08:44:17.000Z",
                        "ex": {
                            "availableToBack": [],
                            "availableToLay": [],
                            "tradedVolume": []
                        }
                    },
                    {
                        "selectionId": 8611527,
                        "handicap": 0,
                        "status": "REMOVED",
                        "adjustmentFactor": 3.5,
                        "removalDate": "2014-06-13T08:44:29.000Z",
                        "ex": {
                            "availableToBack": [],
                            "availableToLay": [],
                            "tradedVolume": []
                        }
                    },
                    {
                        "selectionId": 7920154,
                        "handicap": 0,
                        "status": "WINNER",
                        "adjustmentFactor": 17.5,
                        "ex": {
                            "availableToBack": [],
                            "availableToLay": [],
                            "tradedVolume": []
                        }
                    },
                    {
                        "selectionId": 1231425,
                        "handicap": 0,
                        "status": "LOSER",
                        "adjustmentFactor": 4.3,
                        "ex": {
                            "availableToBack": [],
                            "availableToLay": [],
                            "tradedVolume": []
                        }
                    },
                    {
                        "selectionId": 7533866,
                        "handicap": 0,
                        "status": "LOSER",
                        "adjustmentFactor": 11.4,
                        "ex": {
                            "availableToBack": [],
                            "availableToLay": [],
                            "tradedVolume": []
                        }
```

```
},
{
    "selectionId": 8220841,
    "handicap": 0,
    "status": "LOSER",
    "adjustmentFactor": 5.4,
    "ex": {
        "availableToBack": [],
        "availableToLay": [],
        "tradedVolume": []
    }
},
{
    "selectionId": 7533883,
    "handicap": 0,
    "status": "LOSER",
    "adjustmentFactor": 8.7,
    "ex": {
        "availableToBack": [],
        "availableToLay": [],
        "tradedVolume": []
    }
},
{
    "selectionId": 8476712,
    "handicap": 0,
    "status": "LOSER",
    "adjustmentFactor": 8.7,
    "ex": {
        "availableToBack": [],
        "availableToLay": [],
        "tradedVolume": []
    }
},
{
    "selectionId": 7012263,
    "handicap": 0,
    "status": "LOSER",
    "adjustmentFactor": 5.4,
    "ex": {
        "availableToBack": [],
        "availableToLay": [],
        "tradedVolume": []
    }
},
{
    "selectionId": 7374225,
    "handicap": 0,
    "status": "LOSER",
    "adjustmentFactor": 8.7,
    "ex": {
        "availableToBack": [],
        "availableToLay": [],
                        "tradedVolume": []
                    }
                },
                {
                    "selectionId": 8611525,
                    "handicap": 0,
                    "status": "LOSER",
                    "adjustmentFactor": 0.7,
                    "ex": {
                        "availableToBack": [],
                        "availableToLay": [],
                        "tradedVolume": []
                    }
                },
                {
                    "selectionId": 7659121,
                    "handicap": 0,
                    "status": "LOSER",
                    "adjustmentFactor": 26.8,
                    "ex": {
                        "availableToBack": [],
                        "availableToLay": [],
                        "tradedVolume": []
                    }
                },
                {
                    "selectionId": 6996332,
                    "handicap": 0,
                    "status": "LOSER",
                    "adjustmentFactor": 0.7,
                    "ex": {
                        "availableToBack": [],
                        "availableToLay": [],
                        "tradedVolume": []
                    }
                },
                {
                    "selectionId": 7137541,
                    "handicap": 0,
                    "status": "LOSER",
                    "adjustmentFactor": 1.7,
                    "ex": {
                        "availableToBack": [],
                        "availableToLay": [],
                        "tradedVolume": []
                    }
                }
            ]
        }
    ],
            "id": 1
        }
    ]
```

Application Keys

What is an Application Key?

In order to use the Betting & Accounts API, you need to have an Application Key. The Application Key identifies your API client. Two App Keys are assigned to a single Betfair account, one **live** App Key and one **delayed** App Key for testing.

You must pass the Application Key with every HTTP request. You do this by setting the HTTP header with the value of the key assigned by Betfair.

```
X-Application: APP_KEY_ASSIGNED
```

Commercial Usage
Please note: App Key generation is for **personal betting** purposes only. All data/API usage in any commercial context must be approved by Betfair. Please see the following licensing information for further details (https://developer.betfair.com/get-started/)

Unauthorised commercial usage will be identified & blocked.

How to Create An Application Key

You can create an Application Key for your Betfair account using the Accounts API Visualiser and **createDeveloperAppKeys** operation

1. Click on the Accounts API Visualiser link & ensure the the **Endpoint "PROD"/"UK"** is selected.
2. Select the **createDeveloperAppKeys** operation from the list of Operations on the top left hand side of the visualiser.
3. Enter a **sessionToken** in the 'Session Token (ssoid)' text box. You can find instructions on how to find your sessionToken via your browser here.
4. Enter your **Application Name** (this must be unique) in the 'Request' column. The **Application Name** can be any name of your choice, but like your Betfair username, must be unique.
5. Press **Execute** at the bottom of the 'Request' column.

Two Application Keys will then be created and displayed in the **Developer Apps** column of the demo tool

Please note:

- The X-Application header is not required when using the **createDeveloperAppKeys** or the **getDeveloperAppKeys** service.
- The **Application Name** must be unique.

We are aware that when using some browser versions to create App Keys the Visualiser throws an **UNEXPECTED_ERROR** when requesting **createDeveloperAppKeys.** Using an alternative browser/s should resolve this problem. You should also ensure that your Application Name is unique as attempts to create a duplicate Application Name will return an **UNEXPECTED_ERROR** response.

Live & Delayed Application Keys

The **createDeveloperAppKeys** service will assign two Application Keys (App Keys) to your Betfair account.

One 'live' App Key and one 'delayed' App Key. A delayed App Key is displayed as 'Version 1.0-DELAY' via createDeveloperAppKeys/getDeveloperAppKeys

- Upon creation, the **Live Application Key will be inactive.**
- To apply for a **Live Application key** please click here and select **Exchange API > For My Personal Betting** and complete the application form at the bottom of the page. A one-off activation fee of **£299** applies; this is debited directly from your Betfair account once access is approved
- The **Delayed App Key** operates on the live Betfair Exchange and not a testbed/sandbox environment.
- The **Delayed App Key** should be use for **development purposes** and **any functional testing** and provides **delayed Betfair price data (EX_BEST_OFFERS only).** The delay is variable between 1-60 second snapshots.
- The **Delayed App Key must also be used in simulation/practice applications** where the facility to bet into live Betfair markets is not available.
- The delayed App Key does not return traded volume data '**totalMatched**' or **EX_ALL_OFFERS** via listMarketBook.
- The Stream API is only available via the Live Application Key subject to approval.

* **Historical Data** is additionally made available for testing and analysis purposes. Please see the App Directory for further information on these services.

Personal Betting Access - Application Key Overview

Please see below table for a summary of the data/services available to Delayed & Live Application Keys.

	Delayed Application Key	**Live Application Key**
Use For	**Development**	**Live betting applications**
Activation Fee	**None**	**£299**
Live Price Data	**Delayed***	**Yes**
Bet Placement (Live Exchange)	**Yes**	**Yes**
Stream API	**Yes (contact Developer Support)**	**Yes (on application)**
Price Levels	**3**	**All**
Total Matched by Selection	**Not Available**	**Yes**
Total Matched by Market	**Yes**	**No**
`*Delay is variable between 1-60 seconds`		

Login & Session Management

Login

The Betfair API offers three login flows for developers, depending on the use case of your application:.

Non-Interactive login

if you are building an application which will run autonomously, there is a separate login flow to follow to ensure your account remains secure.

Interactive login

if you are building an application which will be used interactively, then this is the flow for you. This flow has two variants:

Interactive login - Desktop Application

This login flow makes use of Betfair's login pages and allows your app to gracefully handle all errors and re-directions in the same way as the Betfair website

Interactive login - API method

This flow makes use of a JSON API Endpoint and is the simplest way to get started if you are looking to create your own login form.

> If you're looking for the quickest way to get started, try the curl example in the Interactive login - API Method.

Login Request Limits

> Successful login requests are restricted to **100 request per minute**.. In the event of a breach of the login limit the account will be prevented from creating new login session for a 20 minute period. The error **TEMPORARY_BAN_TOO_MANY_REQUESTS** will be returned in these circumstances. All existing sessions will continue to be valid.

Keep Alive

You can use Keep Alive to extend the session timeout period. The minimum session time is currently **20 minutes (Italian Exchange)**. On the **international (.com) Exchange** the current session time is **4 hours**. Therefore, you should request Keep Alive within this time to prevent session expiry. If you don't call Keep Alive within the specified timeout period, the session will expire. **Please note:** Session times aren't determined or extended based on API activity.

Headers

Name	Description	Sample
Accept (mandatory)	Header that signals that the response should be returned as JSON	application/json
X-Authentication (mandatory)	Header that represents the session token that needs to be keep alive	Session Token
X-Application (optional)	Header the Application Key used by the customer to identify the product.	App Key

> The presence of the "Accept: application/json" header will signal that the service should respond with JSON and not an HTML page

URL Definition

International jurisdictions:

```
https://identitysso.betfair.com/api/keepAlive
```

Spanish jurisdiction

```
https://identitysso.betfair.es/api/keepAlive
```

Romania jurisdiction:

```
https://identitysso.betfair.ro/api/keepAlive
```

Parameters - NONE

Response structure

```
{
  "token":"<token_passed_as_header>",
  "product":"product_passed_as_header",
  "status":"<status>",
  "error":"<error>"
}
```

Status values

```
SUCCESS
FAIL
```

Error values

```
INPUT_VALIDATION_ERROR
INTERNAL_ERROR
NO_SESSION
```

Call sample

```
# full request
curl -k -i -H "Accept: application/json" -H "X-Application: AppKey" -H "X-Authentication: <token>" https://identitysso.betfair.com/api/keepAlive
```

> You can use Keep Alive to extend the session timeout period. The minimum session time is currently 20 minutes (Italian Exchange). On the international (.com) Exchange the current session time is 4 hours. Therefore, you should request Keep Alive within this time to prevent session expiry. If you don't call Keep Alive within the specified timeout period, the session will expire. Session times aren't determined or extended based on API activity.

Keep Alive success

```
curl -k -i -H "Accept: application/json" -H "X-Application: AppKey" -H "X-Authentication: SESSIONTOKEN" https://identitysso.betfair.com/api/keepAlive

{
  "token":"SESSIONTOKEN",
  "product":"AppKey",
  "status":"SUCCESS",
  "error":""
}
```

Logout

You can use Logout to terminate your existing session.

URL Definition

`https://identitysso.betfair.com/api/logout`

The presence of the "Accept: application/json" header will signal that the service should respond with JSON and not an HTML page

Headers

Name	Description	Sample
Accept (mandatory)	Header that signals that the response should be returned as JSON	application/json
X-Authentication (mandatory)	Header that represents the session token created at login.	Session Token
X-Application (optional)	Header the Application Key used by the customer to identify the product.	App Key

Response structure

```
{
  "token":"<token_passed_as_header>",
  "product":"product_passed_as_header",
  "status":"<status>",
  "error":"<error>"
}
```

Status values

```
SUCCESS
FAIL
```

Error values

```
INPUT_VALIDATION_ERROR
INTERNAL_ERROR
NO_SESSION
```

Call sample

```
# full request
curl -k -i -H "Accept: application/json" -H "X-Application: AppKey" -H "X-Authentication: <token>" https://
identitysso.betfair.com/api/logout
```

Login FAQ's

When should I use the non-interactive login?

You should use the non-interactive login when the user will not be present to log into the application themselves. An example of this is an automated bot that might need to login without the user triggering a login. 3rd Party interfaces to Betfair, used by multiple users, and which act as a direct proxy of a user request should use the interactive login instead.

Why is the redirect URL required for the interactive login?

The redirect URL is required in order to post the session token to the application at the end of the login process. For further details of how to handle the session token please see Interactive Login from a Desktop

Why isn’t there a non-interactive endpoint that accepts only a username and a password?

Betfair take user security very seriously and have made many enhancements to the login process alongside additional changes which have been made at the request of some of our regulators. This means that you cannot rely upon a username and password being the only pieces of information that may be required at login. Some examples of workflows currently in use are 2 factor authorisation codes, additional National Identifiers for a region or requests for additional account information or account migration.

Why does my session time out, even though I’ve been retrieving prices?

For security reasons, we require that the application using the API explicitly calls the Keep Alive operation no more than once within every 4 hours in a response to user activity. In the case of non-interactive applications, these should call the keep-alive operation within every 4 hours whilst they are active.

Why is my interactive login/logout request failing with errorCode=FORBIDDEN?

Your Application Key App Key is not using the correct redirect URL. By default only https://www.betfair.com will be allowed as the redirect URL.

Non-Interactive (bot) login

The non-interactive login method for API-NG requires that you create and upload a self-signed certificate which will be used, alongside your username and password to authenticate your credentials and generate a session token.

For the purposes of this guide, we have used openssl to generate this client, details of which can be found at **http://www.openssl.org/**

Getting Started

There are a couple of steps required before we can actually log in:

1. Create a self-signed certificate
2. Link the certificate to your Betfair account

Creating a Self Signed Certificate

API-NG requires that a 1024-bit or 2048-bit RSA certificate be used. There are various tutorials available on the Internet but be aware that the certificate needs to be for client authentication (most tutorials only cover server authentication).

Create a public/private RSA key pair using openssl

```
openssl genrsa -out client-2048.key 2048
```

Update or Create the openssl configuration file (openssl.cnf) for OpenSSL to override some of the default settings:

```
[ ssl_client ]
basicConstraints = CA:FALSE
nsCertType = client
keyUsage = digitalSignature, keyEncipherment
extendedKeyUsage = clientAuth
```

In Windows, the config file is located in the installation directory of OpenSSL

In Linux distributions, the config file is located at /usr/lib/ssl/openssl.cnf or /etc/ssl/openssl.cnf

Create a certificate signing request (CSR).

```
openssl req -new -config openssl.cnf -key client-2048.key -out
client-2048.csr


Country Name (2 letter code) [AU]:GB
State or Province Name (full name) [Some-State]:London
Locality Name (eg, city) []:London
Organization Name (eg, company) [Internet Widgits Pty Ltd]:yourcompany.com
Organizational Unit Name (eg, section) []:Security Team
Common Name (e.g. server FQDN or YOUR name) []:Test API-NG Certificate
Email Address []:my.name@mydomain.com

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
```

Self-sign the certificate request to create a certificate

```
openssl x509 -req -days 365 -in client-2048.csr -signkey client-2048.key
-out client-2048.crt -extfile openssl.cnf -extensions ssl_client
```

> In Windows, using any text editor, copy the contents of the .crt file and the .key file into a new file. Save this new file as client-2048.pem.

Linking the Certificate to Your Betfair Account

The previous steps should have created the following files:

File name	Description
client-2048.key	The private key. This file is needed in order to use the certificate and should be protected and shouldn't be shared with anyone.
client-2048.csr	A certificate signing request. This file is no longer needed and can be deleted.
client-2048.crt	The certificate. This file is not sensitive in security terms and can be shared with anyone.

Before you login using the certificate, it must be attached to your Betfair account, as follows:

1. Log in to your Betfair account through betfair.comPaste the following URL into the address bar of your browser
2. Navigate to https://myaccount.betfair.com/accountdetails/mysecurity?showAPI=1 **-** **Note:** Please use https://myaccount.betfair.it/accountdetails/mysecurity?showAPI=1 for the **Italian Exchange or the endpoint relevant to your own jusristiction. See the URL Definition section for more details**
3. Scroll to the section titled "**Automated Betting Program Access**" and click **'Edit'**
4. Click on "Browse" and then locate and select the file client-2048.crt (client-2048.pem if applicable) created above.
5. Click on the "**Upload Certificate**" button.

Scroll down to the "**Automated Betting Program Access**" section if required and the certificate details should be shown. You should now be able to log in to your Betfair account using the API-NG endpoint.

Note on File Formats

Some systems require that client certificates are in a different format to the ones we've created. The two most common formats are (a) PEM format key and certificate in a single file and (b) PKCS#12 format file. .NET applications require a PKCS#12 format file.

To create a PEM format file that contains both the private key and the certificate you can use the following command:

Linux

```
cat client-2048.crt client-2048.key > client-2048.pem
```

Create the PKCS#12 format using crt and key

```
openssl pkcs12 -export -in client-2048.crt -inkey client-2048.key -out
client-2048.p12
```

> Don't circulate the key, PEM file or PCKS#12 format files as these files are security sensitive

Details of a Login Request

A login request can now be made as follows:

1. Submit a HTTP "POST" request to: https://identitysso.betfair.com/api/certlogin
2. As part of the SSL connection, the certificate created previously must be supplied.
3. Include a custom Header called "X-Application" with a value that identifies your application. The value is not validated and is only used to help with troubleshooting and diagnosing any problems.
4. Ensure the POST's Content-Type is "application/x-www-form-urlencoded" rather than MIME attachment encoded.
5. As part of the POST body include two parameters "username" and "password" which should have the relevant username/password for your account.

Certificate Login Interface Details

URL Definition

Certificate Endpoint

```
https://identitysso.betfair.com/api/certlogin
```

This endpoint is also available under the following:

- identitysso.betfair.com
- identitysso.betfair.es
- identitysso.betfair.it
- idenititysso.betfair.ro
- identitysso.w-con.betfair.com
- identitysso.betfaironline.eu

> **Please note: Danish residents** cannot use the Non-Interactive (bot) login method due to the NEMID requirement which is only supported by the Interactive Login - Desktop Application method

Request headers

- **X-Application** - You must set the X-Application header to your application key.

Request Parameters

- **username** (mandatory) - The username of the user logging in.

- **password** (mandatory) - The password of the user logging in.

Please note: The username and password values should be encoded when making the login request. All method names are case sensitive, this includes login, keepAlive and logout.

Response

The response returned is a json string. If the response is successful then the loginStatus key will contain SUCCESS, for example:

```
{
  sessionToken: xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx;
  loginStatus: SUCCESS;
}
```

Should a failure or exception be returned, the response will be structured as below and loginStatus will contain a failure reason:

```
{
  loginStatus: INVALID_USERNAME_OR_PASSWORD;
}
```

The possible failure and exceptional return codes are:

loginStatus	Description
INVALID_USERNAME_OR_PASSWORD	the username or password are invalid
ACCOUNT_NOW_LOCKED	the account was just locked
ACCOUNT_ALREADY_LOCKED	the account is already locked
PENDING_AUTH	pending authentication
TELBET_TERMS_CONDITIONS_NA	Telbet terms and conditions rejected
DUPLICATE_CARDS	duplicate cards
SECURITY_QUESTION_WRONG_3X	the user has entered wrong the security answer 3 times
KYC_SUSPEND	KYC suspended
SUSPENDED	the account is suspended
CLOSED	the account is closed
SELF_EXCLUDED	the account has been self-excluded
INVALID_CONNECTIVITY_TO_REGULATOR_DK	the DK regulator cannot be accessed due to some internal problems in the system behind or in at regulator; timeout cases included.
NOT_AUTHORIZED_BY_REGULATOR_DK	the user identified by the given credentials is not authorized in the DK's jurisdictions due to the regulators' policies. Ex: the user for which this session should be created is not allowed to act(play, bet) in the DK's jurisdiction.
INVALID_CONNECTIVITY_TO_REGULATOR_IT	the IT regulator cannot be accessed due to some internal problems in the system behind or in at regulator; timeout cases included.
NOT_AUTHORIZED_BY_REGULATOR_IT	the user identified by the given credentials is not authorized in the IT's jurisdictions due to the regulators' policies. Ex: the user for which this session should be created is not allowed to act(play, bet) in the IT's jurisdiction.
SECURITY_RESTRICTED_LOCATION	the account is restricted due to security concerns
BETTING_RESTRICTED_LOCATION	the account is accessed from a location where betting is restricted

TRADING_MASTER	Trading Master Account
TRADING_MASTER_SUSPENDED	Suspended Trading Master Account
AGENT_CLIENT_MASTER	Agent Client Master
AGENT_CLIENT_MASTER_SUSPENDED	Suspended Agent Client Master
DANISH_AUTHORIZATION_REQUIRED	Danish authorization required
SPAIN_MIGRATION_REQUIRED	Spain migration required
DENMARK_MIGRATION_REQUIRED	Denmark migration required
SPANISH_TERMS_ACCEPTANCE_REQUIRED	The latest Spanish terms and conditions version must be accepted. You must login to the website to accept the new conditions.
ITALIAN_CONTRACT_ACCEPTANCE_REQUIRED	The latest Italian contract version must be accepted. You must login to the website to accept the new conditions.
CERT_AUTH_REQUIRED	Certificate required or certificate present but could not authenticate with it
CHANGE_PASSWORD_REQUIRED	Change password required
PERSONAL_MESSAGE_REQUIRED	Personal message required for the user
INTERNATIONAL_TERMS_ACCEPTANCE_REQUIRED	The latest international terms and conditions must be accepted prior to logging in.
EMAIL_LOGIN_NOT_ALLOWED	This account has not opted in to log in with the email
MULTIPLE_USERS_WITH_SAME_CREDENTIAL	There is more than one account with the same credential
ACCOUNT_PENDING_PASSWORD_CHANGE	The account must undergo password recovery to reactivate via https://identitysso.betfair.com/view/recoverpassword
TEMPORARY_BAN_TOO_MANY_REQUESTS	The limit for successful login requests per minute has been exceeded. New login attempts will be banned for 20 minutes
ITALIAN_PROFILING_ACCEPTANCE_REQUIRED	You must login to the website to accept the new conditions

Sample curl command to quickly check the certificate based login

```
curl -q -k --cert client-2048.crt --key client-2048.key
https://identitysso.betfair.com/api/certlogin -d
"username=testuser&password=testpassword" -H "X-Application:
curlCommandLineTest"

Response should be

{"sessionToken":"Zx8i4oigut5nc+l4L8qFb0DSxG+mwLn2t0AMGFxjrMJI=","loginStat
us":"SUCCESS"}
```

Sample Code for Non-Interactive Login

Sample C# code using PKCS#12 key store

Please see code sample via https://github.com/betfair/API-NG-sample-code/tree/master/loginCode/Non-interactive-cSharp

Sample Java code using Apache http client library and PKCS#12 key store

Java API-NG login

```
package com.test.aping.client;



import org.apache.http.HttpEntity;
import org.apache.http.HttpResponse;
import org.apache.http.NameValuePair;
import org.apache.http.client.entity.UrlEncodedFormEntity;
import org.apache.http.client.methods.HttpPost;
import org.apache.http.conn.ClientConnectionManager;
import org.apache.http.conn.scheme.Scheme;
import org.apache.http.conn.ssl.SSLSocketFactory;
import org.apache.http.conn.ssl.StrictHostnameVerifier;
import org.apache.http.impl.client.DefaultHttpClient;
import org.apache.http.message.BasicNameValuePair;
import org.apache.http.util.EntityUtils;

import javax.net.ssl.KeyManager;
import javax.net.ssl.KeyManagerFactory;
import javax.net.ssl.SSLContext;
import java.io.File;
import java.io.FileInputStream;
import java.io.InputStream;
import java.security.KeyStore;
import java.security.SecureRandom;
import java.util.ArrayList;
import java.util.List;


public class HttpClientSSO {

    private static int port = 443;

    public static void main(String[] args) throws Exception {

        DefaultHttpClient httpClient = new DefaultHttpClient();

        try {
            SSLContext ctx = SSLContext.getInstance("TLS");
            KeyManager[] keyManagers = getKeyManagers("pkcs12", new
FileInputStream(new File("C:\\sslcerts\\client-2048.p12")), "password");
            ctx.init(keyManagers, null, new SecureRandom());
            SSLSocketFactory factory = new SSLSocketFactory(ctx, new
StrictHostnameVerifier());

            ClientConnectionManager manager =
httpClient.getConnectionManager();
            manager.getSchemeRegistry().register(new Scheme("https", port,
factory));
            HttpPost httpPost = new
HttpPost("https://identitysso.betfair.com/api/certlogin");
```

```
            List<NameValuePair> nvps = new ArrayList<NameValuePair>();
            nvps.add(new BasicNameValuePair("username", "testuser"));
            nvps.add(new BasicNameValuePair("password", "testpassword"));

            httpPost.setEntity(new UrlEncodedFormEntity(nvps));



            httpPost.setHeader("X-Application","appkey");


            System.out.println("executing request" +
httpPost.getRequestLine());

            HttpResponse response = httpClient.execute(httpPost);
            HttpEntity entity = response.getEntity();

            System.out.println("----------------------------------------");
            System.out.println(response.getStatusLine());
            if (entity != null) {
                String responseString = EntityUtils.toString(entity);
    //extract the session token from responsestring
    System.out.println("responseString" + responseString);
            }

        } finally {
            httpClient.getConnectionManager().shutdown();
        }
    }



    private static KeyManager[] getKeyManagers(String keyStoreType,
InputStream keyStoreFile, String keyStorePassword) throws Exception {
        KeyStore keyStore = KeyStore.getInstance(keyStoreType);
        keyStore.load(keyStoreFile, keyStorePassword.toCharArray());
        KeyManagerFactory kmf =
KeyManagerFactory.getInstance(KeyManagerFactory.getDefaultAlgorithm());
        kmf.init(keyStore, keyStorePassword.toCharArray());
        return kmf.getKeyManagers();
    }
}
```

Sample Python code

```
#!/usr/bin/env python

import requests

#openssl x509 -x509toreq -in certificate.crt -out CSR.csr -signkey
privateKey.key


payload = 'username=myusername&password=password'
headers = {'X-Application': 'SomeKey', 'Content-Type':
'application/x-www-form-urlencoded'}

resp = requests.post('https://identitysso.betfair.com/api/certlogin',
data=payload, cert=('client-2048.crt', 'client-2048.key'), headers=headers)

if resp.status_code == 200:
  resp_json = resp.json()
  print resp_json['loginStatus']
  print resp_json['sessionToken']
else:
  print "Request failed."
```

ITALIAN_PROFILING_ACCEPTANCE_REQUIRED

Certificate Generation With XCA

If you aren't able (or willing) to setup openssl on your windows machine, there are various GUI wrappers around the toolset which you might be able to use instead. XCA is an open source wrapper around the OpenSSL toolset which allows you to create keys, csrs and certificates via a GUI and stores all of the generated items in a database file.

First you need to download XCA, which is an open source wrapper around the openssl toolset from:
http://sourceforge.net/projects/xca/

Step by Step Guide

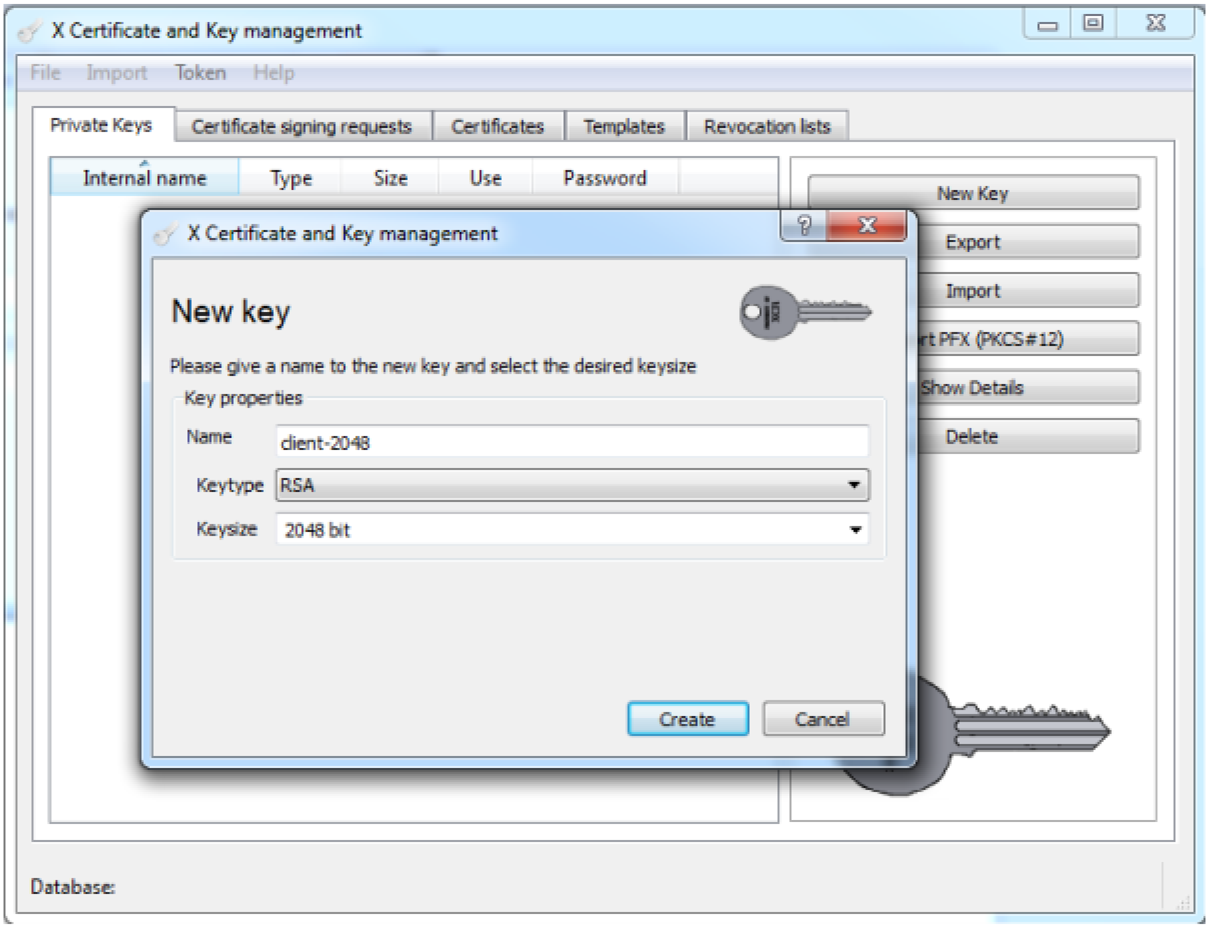
1. Install XCA and run it.
2. Create a new Database,
3. Name it something sensible
4. Save it somewhere appropriate.

This proprietary database is useful for the xca tool only and helps you store your keys, csrs, and certificates, the database file is not used in any part of the process with Betfair.

Equivalent Open SSL Command - Create a public/private RSA key pair using openssl

```
openssl genrsa -out client-2048.key 2048
```

- Create a public/private RSA key pair using xca:



New key

Please give a name to the new key and select the desired keysize

Key properties

Name: client-2048

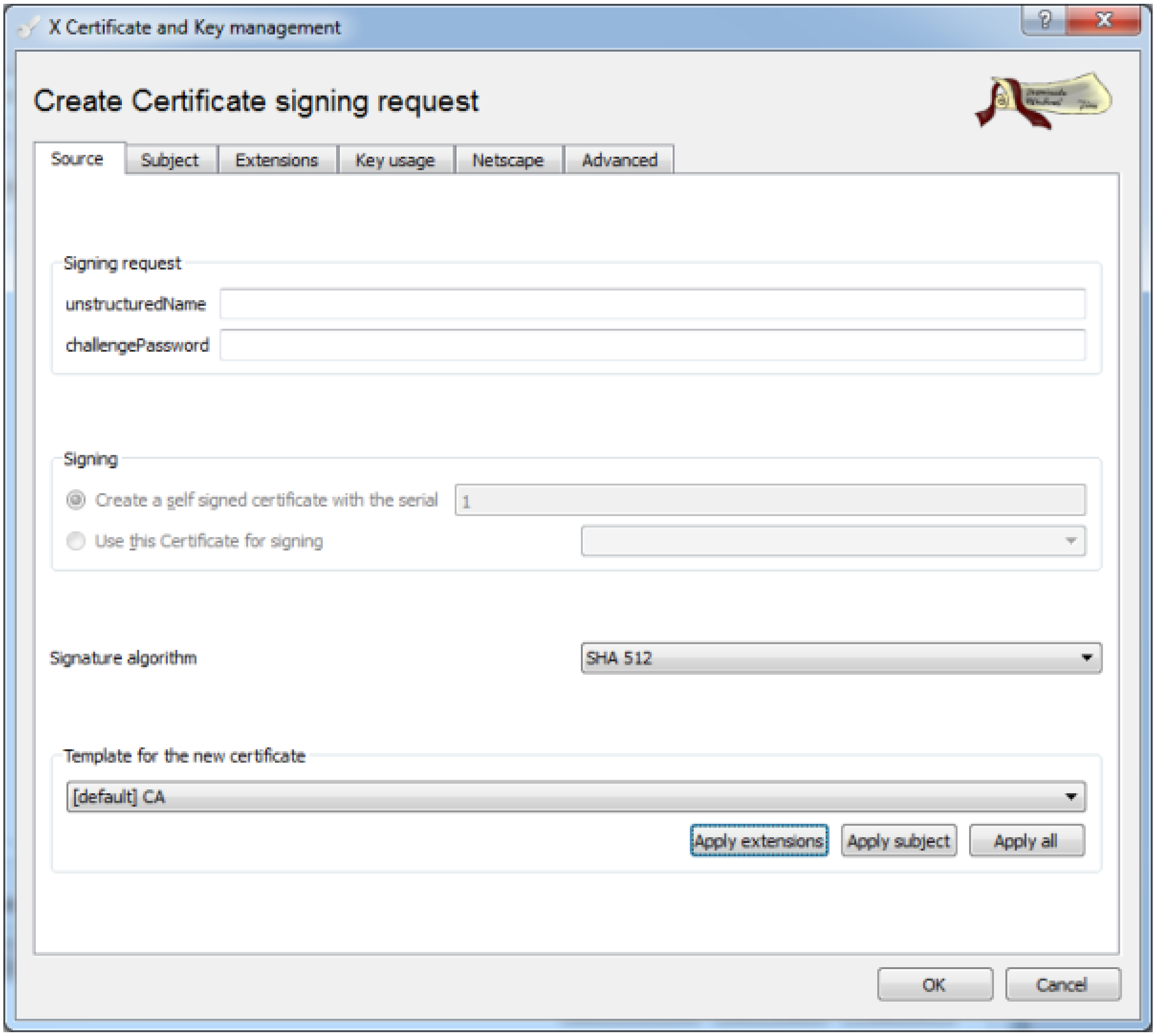
Keytype: RSA

Keysize: 2048 bit

Equivalent Open SSL Command - Create a certificate signing request (CSR).

```
openssl req -new -config openssl.cnf -key client-2048.key -out client-2048.csr
```

- Select the Certificate signing requests tab and click **New Request**
- Select the CA template and click the **Apply Extensions** button.



Create Certificate signing request

Signing request

unstructuredName

challengePassword

Signing

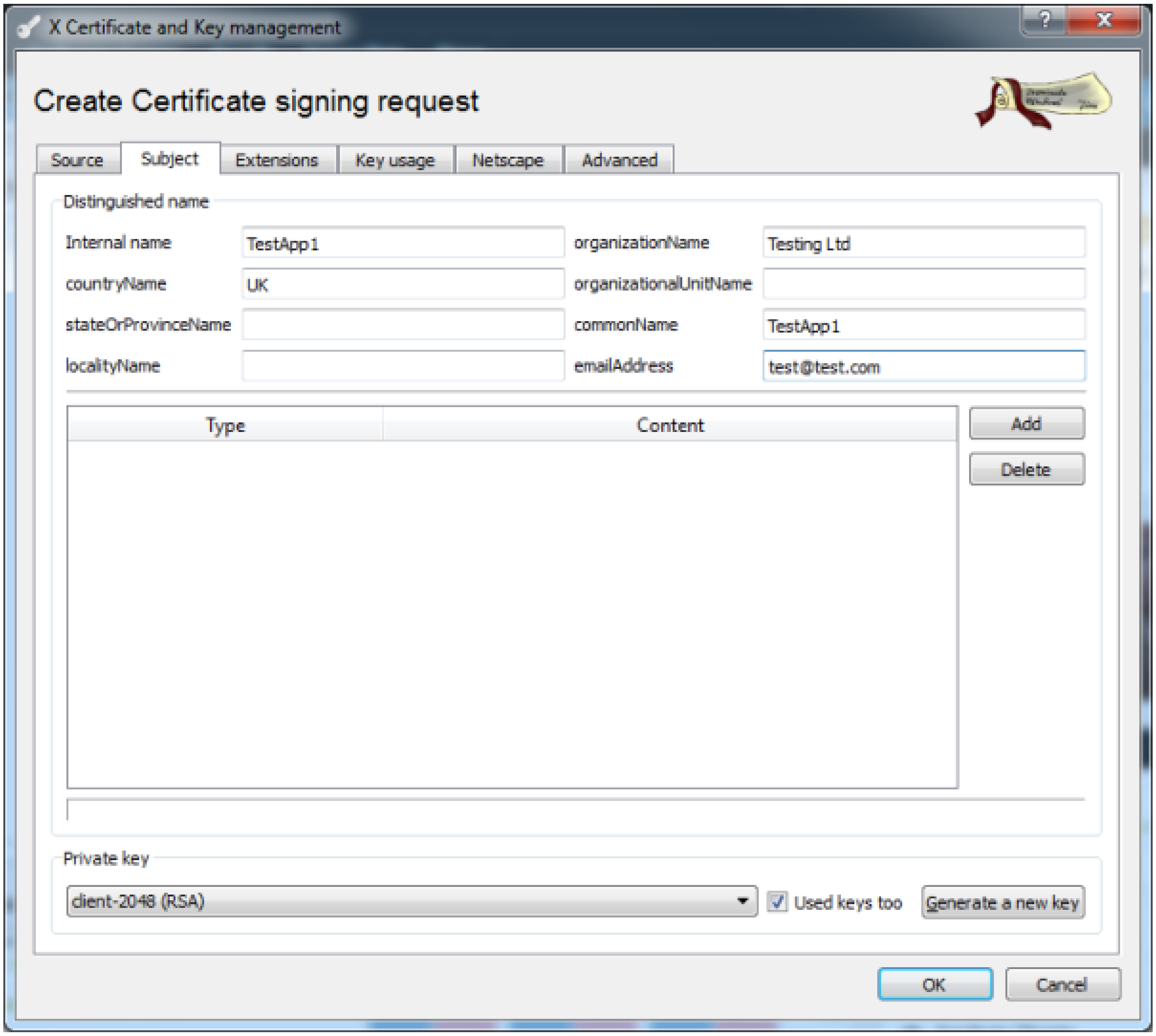
● Create a self signed certificate with the serial: 1

○ Use this Certificate for signing

Signature algorithm: SHA 512

Template for the new certificate: [default] CA

- Click on the **Subject** tab and enter the name etc.
- The key that you generated in the first step must be selected (if the key doesn't appear, check the "Used keys tool" box)



Create Certificate signing request

Distinguished name

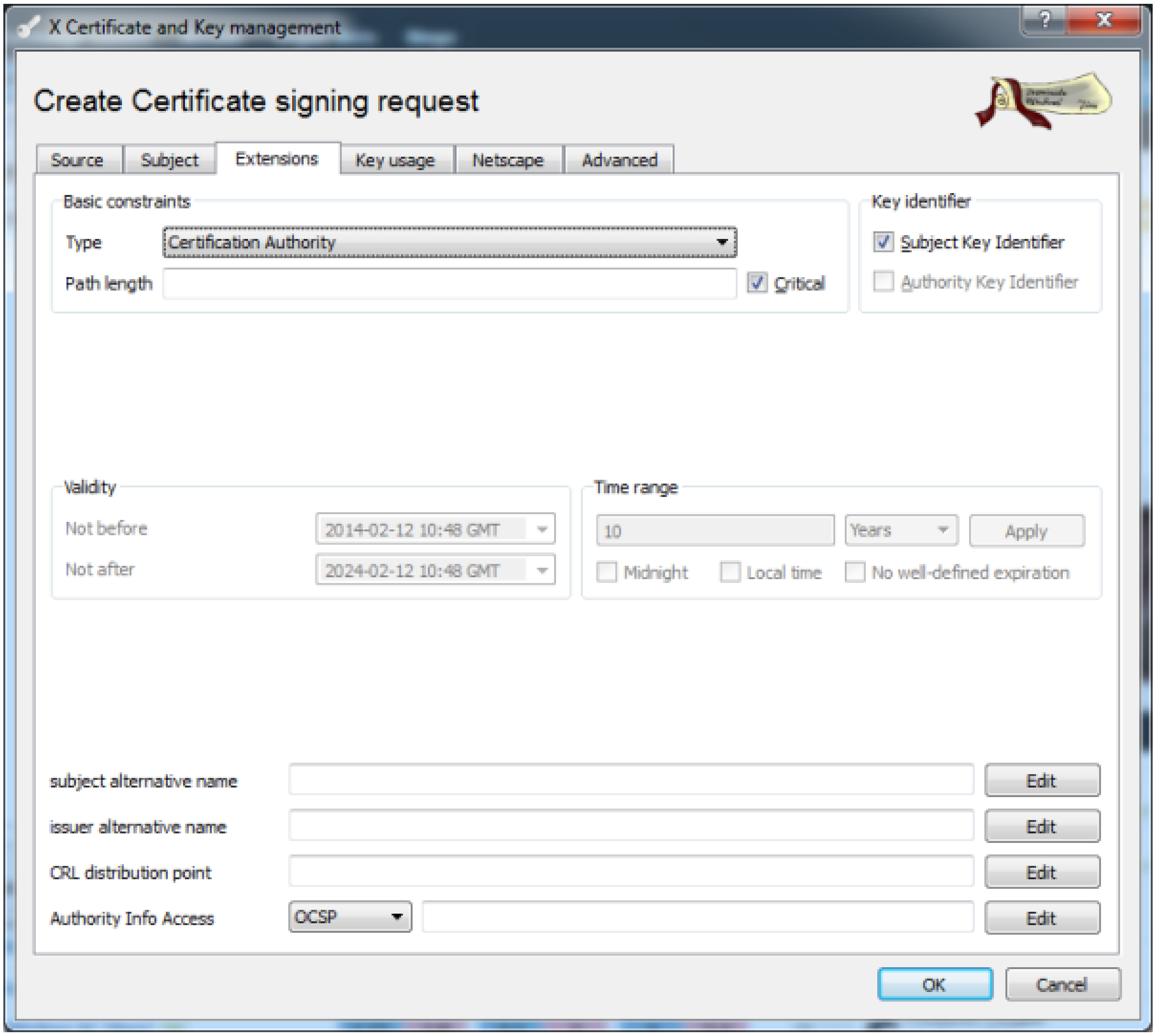
Internal name	TestApp1	organizationName	Testing Ltd
countryName	UK	organizationalUnitName	
stateOrProvinceName		commonName	TestApp1
localityName		emailAddress	test@test.com

Type	Content

Private key

client-2048 (RSA) ☑ Used keys too

- Click on the **Extensions** tab
- Ensure that **Certification Authority** is selected as the type.
- Click OK.



Create Certificate signing request

Type: Certification Authority

Path length:

☑ Critical

☑ Subject Key Identifier

□ Authority Key Identifier

Not before: 2014-02-12 10:48 GMT

Not after: 2024-02-12 10:48 GMT

10 Years

□ Midnight □ Local time □ No well-defined expiration

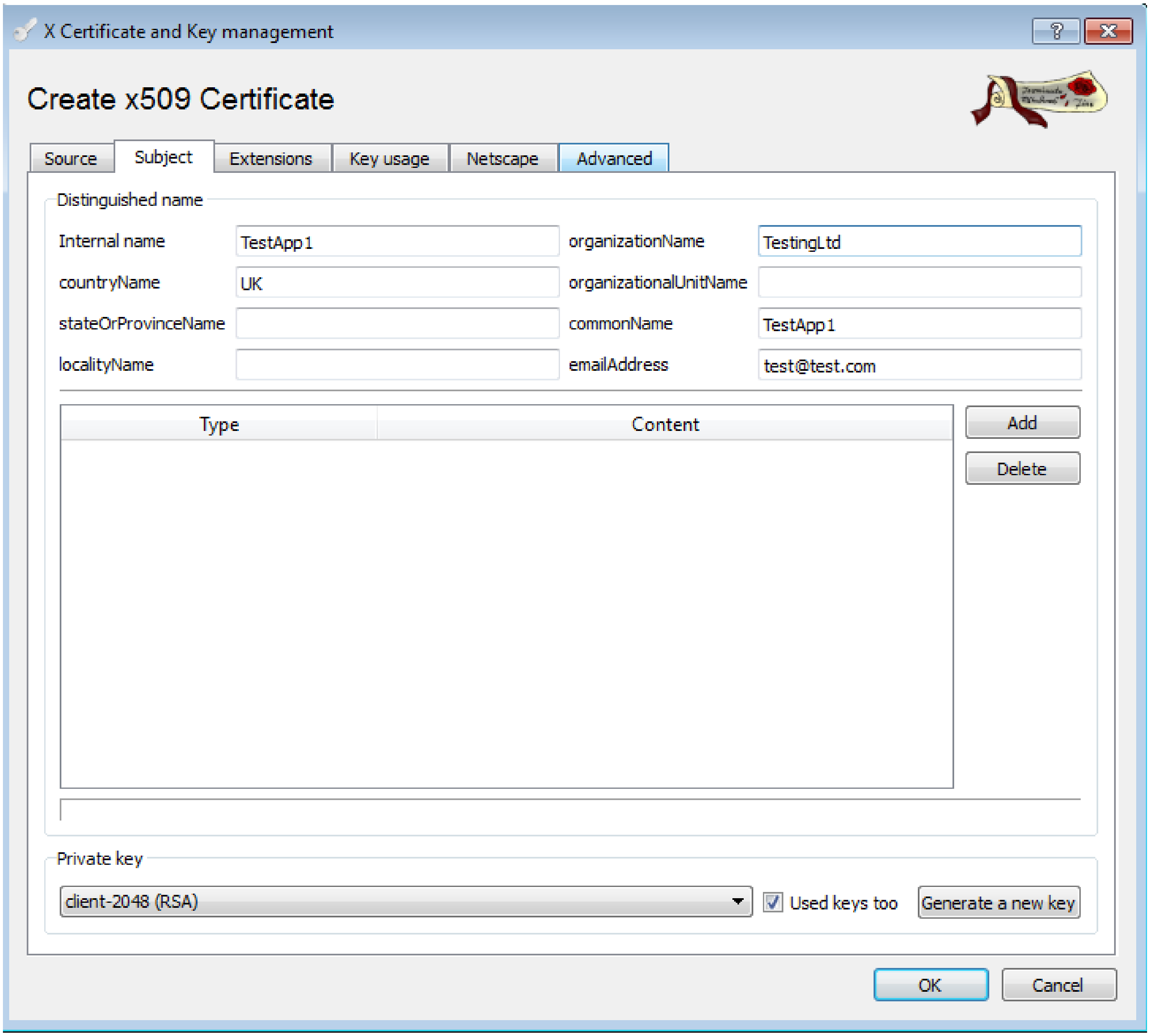
Authority Info Access: OCSP

Equivalent Open SSL Command - Self-sign the certificate request to create a certificate

```
openssl x509 -req -days 365 -in client-2048.csr -signkey client-2048.key -out client-2048.crt -extfile openssl.cnf -extensions ssl_client
```

We will now create and sign a certificate from the first two steps:

- Click the Certificates Tab and click New Certificate
- Click on the **Subject** tab and enter the name etc.
- The key that you generated in the first step must be selected (if the key doesn't appear, check the "Used keys tool" box)



Create x509 Certificate

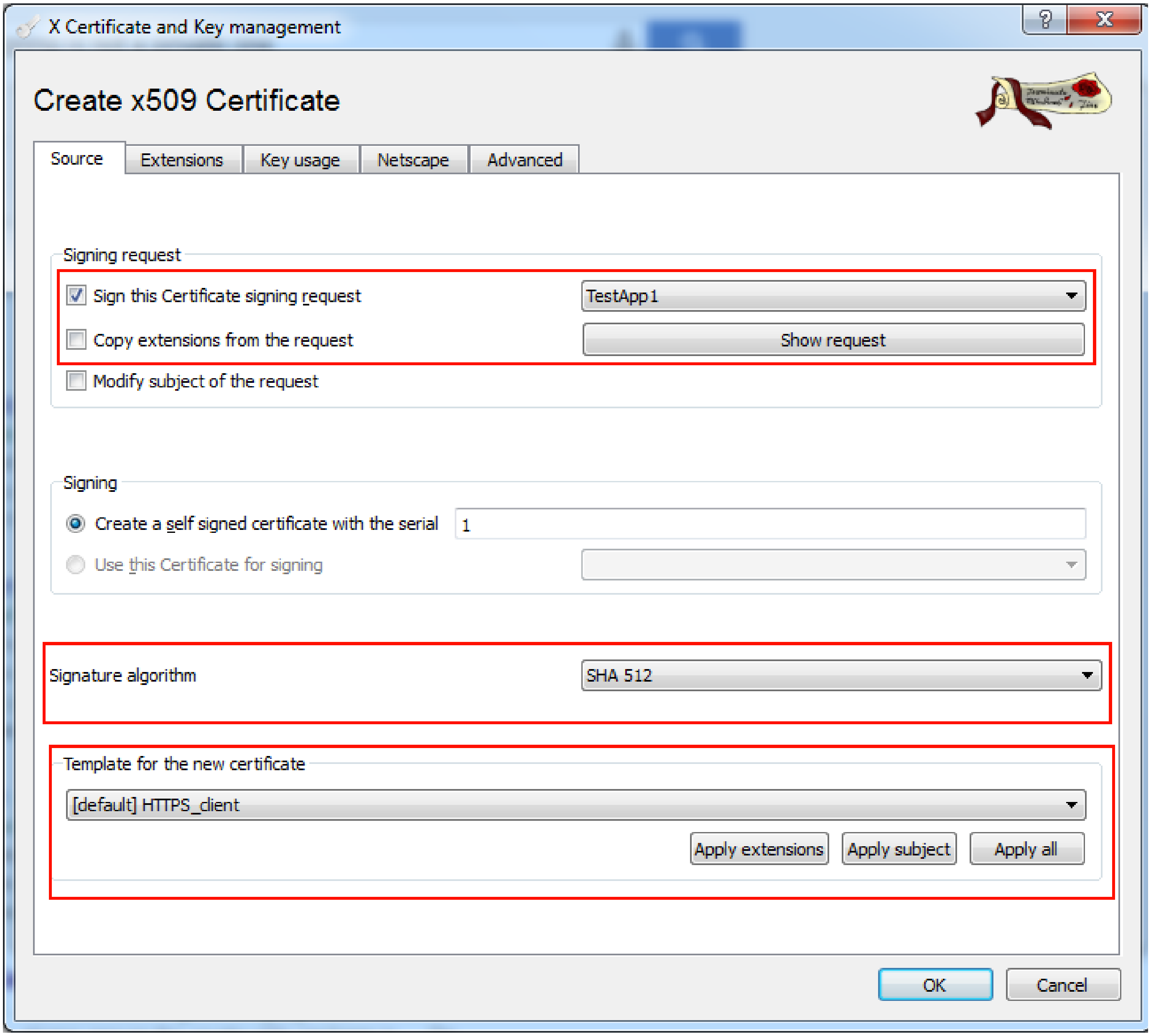
Distinguished name

Internal name	TestApp1	organizationName	TestingLtd
countryName	UK	organizationalUnitName	
stateOrProvinceName		commonName	TestApp1
localityName		emailAddress	test@test.com

Private key

client-2048 (RSA) ☑ Used keys too Generate a new key

- Click on the **Source** tab and select the parameters shown below:



X Certificate and Key management

Create x509 Certificate

Signing request

☑ Sign this Certificate signing request — TestApp1

□ Copy extensions from the request — Show request

□ Modify subject of the request

Signing

● Create a self signed certificate with the serial 1

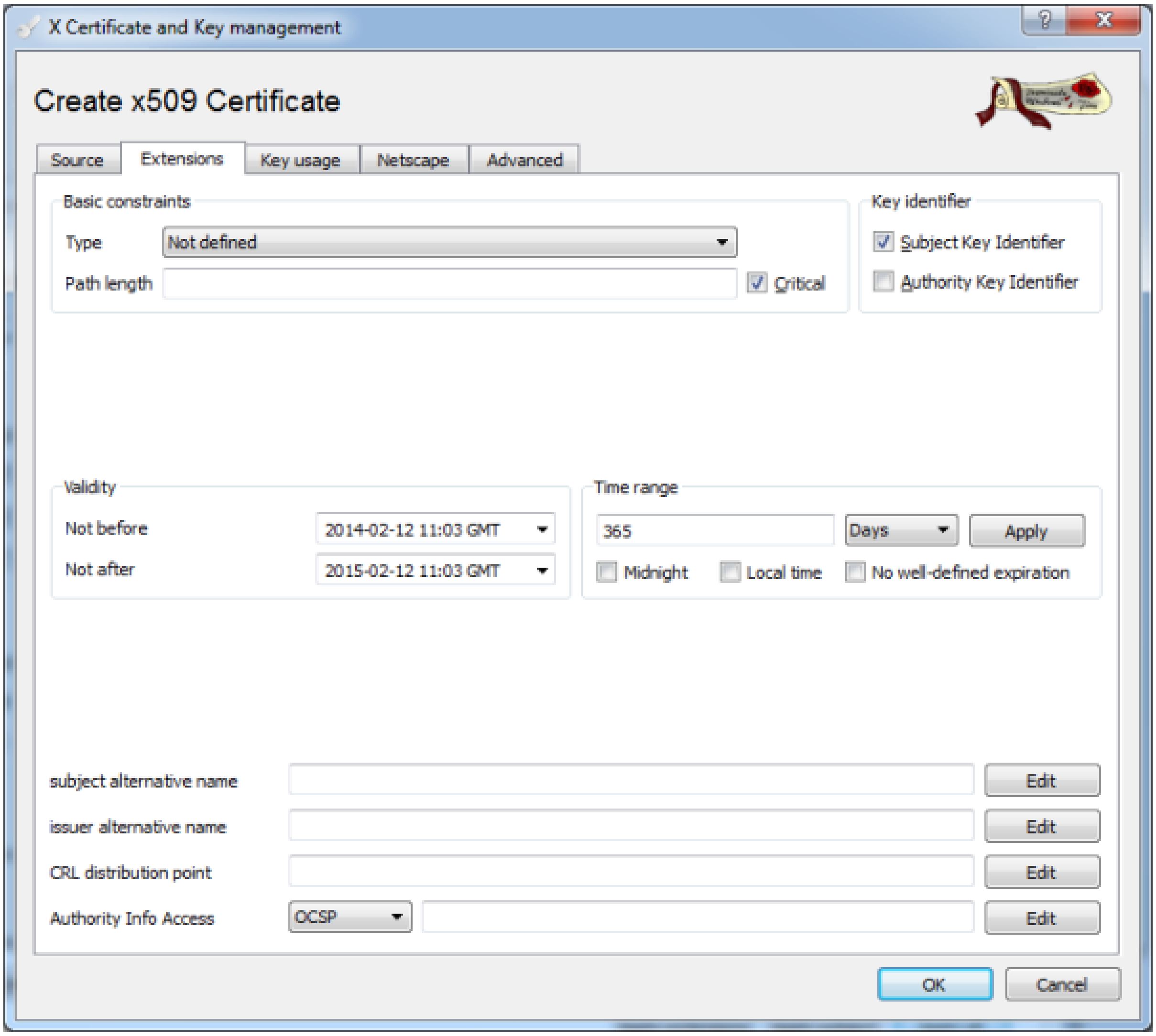
○ Use this Certificate for signing

Signature algorithm: SHA 512

Template for the new certificate

[default] HTTPS_client

- You should make sure that your CSR is selected but that “Copy extensions from the request” is unticked and that you have selected the [default] HTTPS_client and pressed the “Apply extensions” button.



Type: Not defined

Path length: ☑ Critical

☑ Subject Key Identifier

□ Authority Key Identifier

Not before: 2014-02-12 11:03 GMT

Not after: 2015-02-12 11:03 GMT

365 Days

□ Midnight □ Local time □ No well-defined expiration

subject alternative name

issuer alternative name

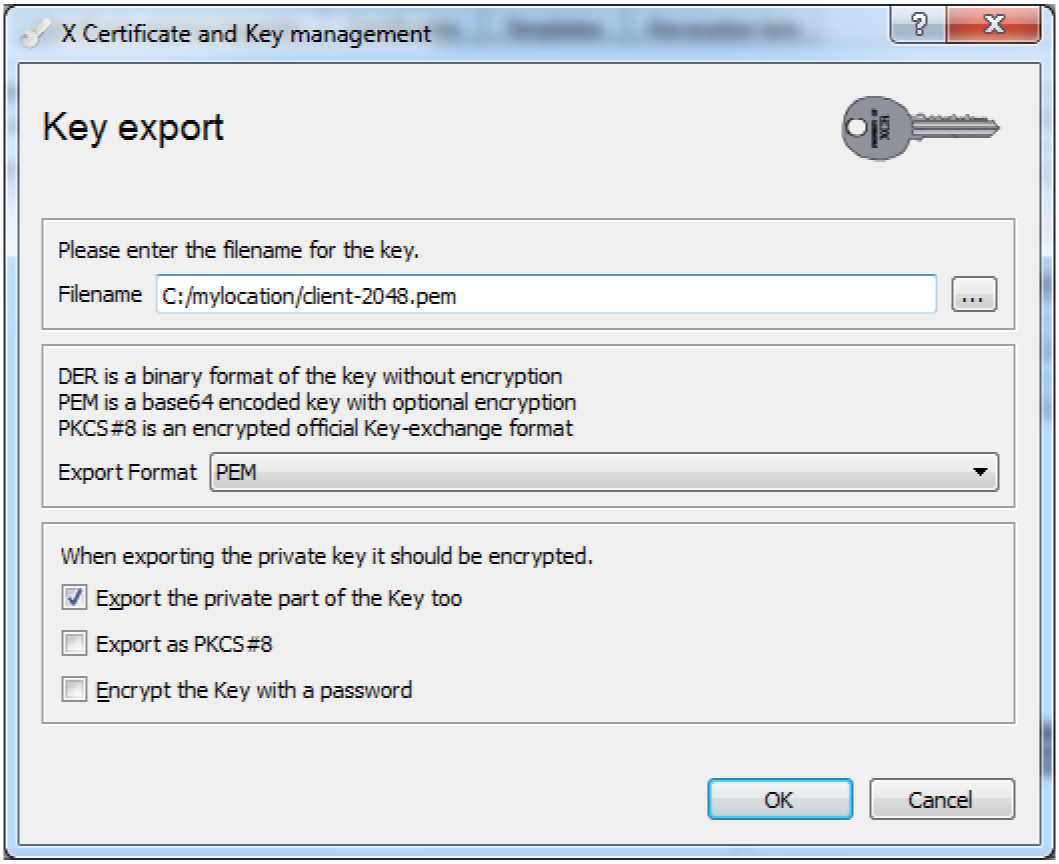
CRL distribution point

Authority Info Access: OCSP

You can then press OK to create the self-signed certificate.

Next we need to export the key and the certificate for use in our application.

- Export the private key and Certificate. These files should be used in the authentication process and should be referred to in your code where the private key (the .pem file) and certificate file (the .crt file) are mentioned. You can also export to other formats such as PKCS 12 depending upon the format expected by your language/library of choice.



Key export

Please enter the filename for the key.

Filename C:/mylocation/client-2048.pem

DER is a binary format of the key without encryption
PEM is a base64 encoded key with optional encryption
PKCS#8 is an encrypted official Key-exchange format

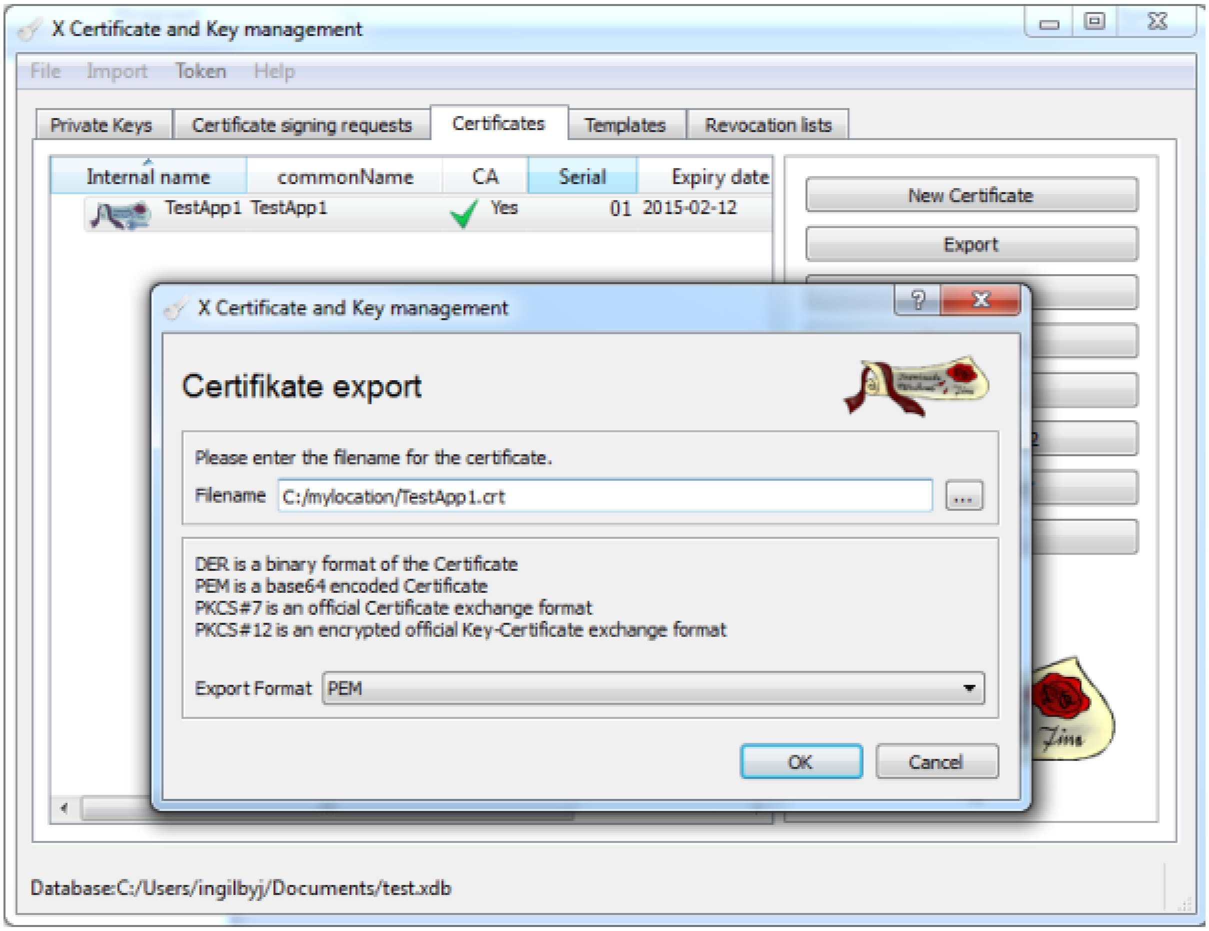
Export Format PEM

When exporting the private key it should be encrypted.

☑ Export the private part of the Key too

□ Export as PKCS#8

□ Encrypt the Key with a password



Internal name	commonName	CA	Serial	Expiry date
TestApp1	TestApp1	✓ Yes	01	2015-02-12

Certifikate export

Please enter the filename for the certificate.

Filename C:/mylocation/TestApp1.crt

DER is a binary format of the Certificate
PEM is a base64 encoded Certificate
PKCS#7 is an official Certificate exchange format
PKCS#12 is an encrypted official Key-Certificate exchange format

Export Format PEM

Database:C:/Users/ingilbyj/Documents/test.xdb

You should upload the .crt file exported to the My Security page on Betfair.com to allow this certificate access to your account.

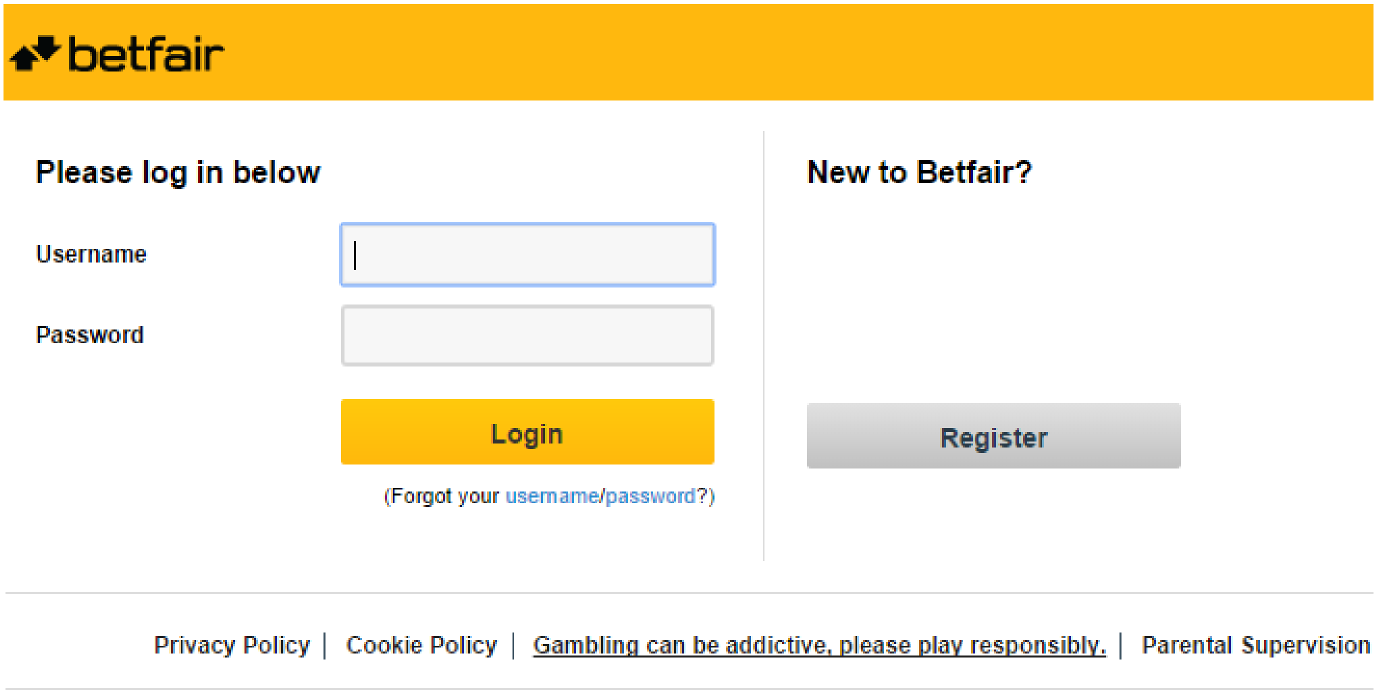
Interactive Login - Desktop Application

- Overview
- Obtaining the sessionToken from the POST data
- URL Definition (Global)
- URL Definition - Other Jurisdictions
- Parameters

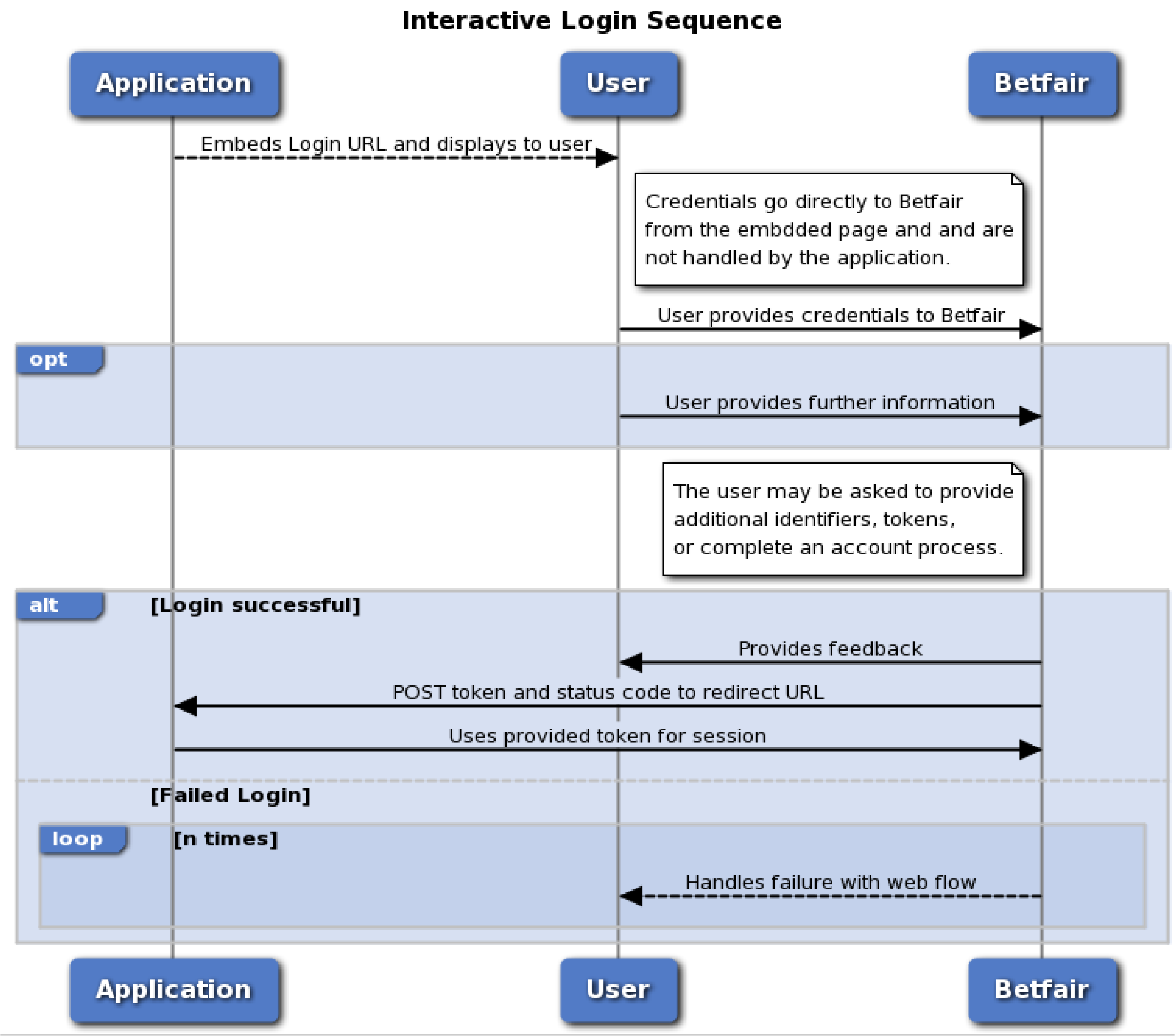
Overview

Interactive login is to be used when the user is present to login (for example, 3rd Party Desktop Applications) and will manage any additional information required at login depending upon a customer's account (such as 2 Factor Authentication codes or National Identifiers).

This is achieved by embedding the Betfair IdentitySSO login page in your application and then obtaining a successful session token upon login. The Keep Alive operation should be called within session expiry time if the user is still actively using your application. The embedded login page initially looks like this:


betfair
Please log in below
Username
Password
Login
(Forgot your username/password?)
New to Betfair?
Register
Privacy Policy | Cookie Policy | Gambling can be addictive, please play responsibly. | Parental Supervision

The interactive login sequence looks like this:


Interactive Login Sequence
Application
User
Betfair
Embeds Login URL and displays to user
Credentials go directly to Betfair from the embdded page and and are not handled by the application.
User provides credentials to Betfair
opt
User provides further information
The user may be asked to provide additional identifiers, tokens, or complete an account process.
alt
[Login successful]
Provides feedback
POST token and status code to redirect URL
Uses provided token for session
[Failed Login]
loop
[n times]
Handles failure with web flow
Application
User
Betfair

Obtaining the sessionToken from the POST data

Once a login has been successfully made, the Javascript in the page will POST the session token (**ssoid**) to the URL provided as a redirect URL. For a desktop application, this is not required to be a real page as the desktop application can intercept the POST request as it happens via the embedded browser container. A Windows based application can embed a web browser into the application and use the BeforeNavigate2 event to catch the post data sent to the redirect URL and there are platform specific alternatives. The POST request body will contain two URL encoded parameters (which you will need to URL Decode):

- **ssoid** - This is your session token and should be attached to requests made to API-NG in the X-Authentication header.
- **errorCode** - This is returned in a URL by Betfair and provides the reason for the login failure.

> This flow protects the implementing application from user login complexities, such as 2 factor auth, requiring national identifiers or jurisdictional migrations.

The Interactive Login is the same login flow used by the Betfair website and therefore, any message's will be returned directly by Betfair & handled in the same way.

errorCode	
ACCOUNT_ALREADY_LOCKED	the account is already locked
ACCOUNT_NOW_LOCKED	the account was just locked
ACCOUNT_PENDING_PASSWORD_CHANGE	the account must undergo password recovery to reactivate via https://identitysso.betfair.com/view/recoverpassword
AGENT_CLIENT_MASTER	Agent Client Master
AGENT_CLIENT_MASTER_SUSPENDED	Suspended Agent Client Master
BETTING_RESTRICTED_LOCATION	the account is accessed from a location where betting is restricted
CERT_AUTH_REQUIRED	Certificate required or certificate present but could not authenticate with it
CHANGE_PASSWORD_REQUIRED	change password required
CLOSED	the account is closed
DANISH_AUTHORIZATION_REQUIRED	danish authorization required
DENMARK_MIGRATION_REQUIRED	denmark migration required
DUPLICATE_CARDS	duplicate cards
EMAIL_LOGIN_NOT_ALLOWED	This account has not opted in to log in with the email
INVALID_CONNECTIVITY_TO_REGULATOR_DK	the DK regulator cannot be accessed due to some internal problems in the system behind or in at regulator; timeout cases included.
INVALID_CONNECTIVITY_TO_REGULATOR_IT	the IT regulator cannot be accessed due to some internal problems in the system behind or in at regulator; timeout cases included.
INVALID_USERNAME_OR_PASSWORD	the username or password are invalid
ITALIAN_CONTRACT_ACCEPTANCE_REQUIRED	The latest italian contract version must be accepted
KYC_SUSPEND	KYC suspended
NOT_AUTHORIZED_BY_REGULATOR_DK	the user identified by the given credentials is not authorized in the DK's jurisdictions due to the regulators' policies. Ex: the user for which this session should be created is not allowed to act(play, bet) in the DK's jurisdiction.
NOT_AUTHORIZED_BY_REGULATOR_IT	the user identified by the given credentials is not authorized in the IT's jurisdictions due to the regulators' policies. Ex: the user for which this session should be created is not allowed to act(play, bet) in the IT's jurisdiction.
MULTIPLE_USERS_WITH_SAME_CREDENTIAL	There is more than one account with the same credential
PENDING_AUTH	pending authentication
PERSONAL_MESSAGE_REQUIRED	personal message required for the user
SECURITY_QUESTION_WRONG_3X	the user has entered wrong the security question 3 times
SECURITY_RESTRICTED_LOCATION	the account is restricted due to security concerns
SELF_EXCLUDED	the account has been self excluded
SPAIN_MIGRATION_REQUIRED	spain migration required
SPANISH_TERMS_ACCEPTANCE_REQUIRED	The latest spanish terms and conditions version must be accepted
SUSPENDED	the account is suspended
TELBET_TERMS_CONDITIONS_NA	Telbet terms and conditions rejected
TRADING_MASTER	Trading Master Account
TRADING_MASTER_SUSPENDED	Suspended Trading Master Account

TEMPORARY_BAN_TOO_MANY_REQUESTS	The limit for successful login requests per minute has been exceeded. New login attempts will be banned for 20 minutes

URL Definition (Global)

```
https://identitysso.betfair.com/view/login?product=<theProductDescriptor>&url=<theRedirectUrl>
```

URL Definition - Other Jurisdictions

Australian jurisdiction users:

```
https://identitysso.betfair.com.au/view/login?product=<theProductDescriptor>&url=<theRedirectUr
```

Italian jurisdiction users:

```
https://identitysso.betfair.it/view/login?product=<theProductDescriptor>&url=<theRedirectUrl>
```

Spanish jurisdiction users:

```
https://identitysso.betfair.es/view/login?product=<theProductDescriptor>&url=<theRedirectUrl>
```

Romania jurisdiction users:

```
https://identitysso.betfair.ro/view/login?product=<theProductDescriptor>&url=<theRedirectUrl>
```

Parameters

Name	Description	Sample
product(mandatory)	The product for which the login page is used and on which the user will do the login; This should be your application key.	"IhDSui3ODdsdwo"
url (mandatory)	The url to which the the browser should be redirected in case of a successful login. By default only https://www.betfair.com will be allowed	https://www.betfair.com

Please note that all method names are case sensitive, this includes login, keepAlive and logout.

Interactive Login - API Endpoint

- Status Codes & Error values
- Possible failure and exceptional return codes

Overview and limitations

The API login endpoint is the simplest method of integration for most applications in terms of expected development time but comes at the cost of being less flexible to edge-cases than the embedded Betfair embedded login page. It will allow a user to provide a username and password or a username and (password + 2 factor auth code) if they have strong authentication enabled.

- Customers who writing bots are for their own use are **strongly recommended** to use the non-interactive endpoint with an SSL certificate.
- We **recommend** that **3rd party applications** which will be exposed to a wide range of users use the Interactive Login method of embedding the Betfair embedded login page as this will allow your application to handle additional workflows, such as terms and conditions updates as well as additional jurisdictional specific identifiers.

The Keep alive and logout methods remain the same with this method of login.

URL Definition (Global)

API Login Endpoint

```
https://identitysso.betfair.com/api/login
```

Other Jurisdictions

Italian jurisdiction users:

```
https://identitysso.betfair.it/api/login
```

Spanish jurisdiction users:

```
https://identitysso.betfair.es/api/login
```

Romania jurisdiction users:

```
https://identitysso.betfair.ro/api/login
```

Parameters (POST)

Name	Description	Sample
username (mandatory)	The username to be used for the login	
password (mandatory)	The password to be used for the login. For strong auth customers, this should be their password with a 2 factor auth code appended to the password string.	

Headers

Name	Description	Sample
Accept (mandatory)	Signals that the response should be returned as JSON	application/json
X-Application (mandatory)	AppKey used by the customer to identify the product.	

The presence of the "Accept: application/json" will signal SSO that it should respond with JSON and not with a HTML page.

POST Example

Accept: application/json

X-Application: <AppKey>

Content-Type: application/x-www-form-urlencoded

URL endpoint: https://identitysso.betfair.com/api/login

Payload

username=username&password=password

Curl call sample

```
curl -k -i -H "Accept: application/json" -H "X-Application: <AppKey>" -X
POST -d 'username=<username>&password=<password>'
https://identitysso.betfair.com/api/login
```

Example of a successful login:

```
curl -k -i -H "Accept: application/json" -H "X-Application: <AppKey>" -X
POST -d 'username=<username>&password=<password>'
https://identitysso.betfair.com/api/login

{
  "token":"SESSION_TOKEN",
  "product":"APP_KEY",
  "status":"SUCCESS",
  "error":""
}
```

Response Structure

```
{
  "token":"<token_passed_as_header>",
  "product":"product_passed_as_header",
  "status":"<status>",
  "error":"<error>"
}
```

Status Values

```
SUCCESS
LIMITED_ACCESS
LOGIN_RESTRICTED
FAIL
```

Status Codes & Error values

The below describes the status codes that can be returned and the associated error values:

LIMITED_ACCESS - Access is limited (e.g. accounts can login but can't bet due to account suspension), product session will be provided.

```
{
  "token": product_token,
  "product": product,
  "status": LIMITED_ACCESS,
  "error": error
}

error = {PENDING_AUTH | SECURITY_QUESTION_WRONG_3X | KYC_SUSPEND |
SUSPENDED}
```

LOGIN_RESTRICTED - login is restricted (in case of indirection point this is what will be returned), product session will not be provided:

```
{
  "token": "",
  "product": product,
  "status": LOGIN_RESTRICTED,
  "error": error
}

error = {STRONG_AUTH_CODE_REQUIRED | DENMARK_MIGRATION_REQUIRED |
DANISH_AUTHORIZATION_REQUIRED | SPAIN_MIGRATION_REQUIRED |
SPANISH_TERMS_ACCEPTANCE_REQUIRED | ITALY_MIGRATION_REQUIRED |
ITALIAN_CONTRACT_ACCEPTANCE_REQUIRED | CHANGE_PASSWORD_REQUIRED |
PERSONAL_MESSAGE_REQUIRED}
```

FAIL - All other cases are treated as errors, product session will not be provided:

```
{
  "token": "",
  "product": product,
  "status": FAIL,
  "error": error
}

error = {TRADING_MASTER | TRADING_MASTER_SUSPENDED | AGENT_CLIENT_MASTER |
AGENT_CLIENT_MASTER_SUSPENDED | DENMARK_MIGRATION_REQUIRED | INVALID_PIN |
INVALID_USERNAME_OR_PASSWORD | PIN_DELETED_ON_FAILED_COUNT_EXCEEDED |
UNRECOGNIZED_DEVICE | DUPLICATE_CARDS | ACCOUNT_NOW_LOCKED |
ACCOUNT_ALREADY_LOCKED | SECURITY_RESTRICTED_LOCATION |
BETTING_RESTRICTED_LOCATION | INVALID_CONNECTIVITY_TO_REGULATOR |
INVALID_CONNECTIVITY_TO_REGULATOR | INVALID_CONNECTIVITY_TO_REGULATOR_IT |
INVALID_CONNECTIVITY_TO_REGULATOR_DK| NOT_AUTHORIZED_BY_REGULATOR |
NOT_AUTHORIZED_BY_REGULATOR | NOT_AUTHORIZED_BY_REGULATOR_DK |
NOT_AUTHORIZED_BY_REGULATOR_IT | TELBET_TERMS_CONDITIONS_NA | CLOSED |
SELF_EXCLUDED | NOT_AUTHORIZED_FOR_DOMAIN_ES | NOT_AUTHORIZED_FOR_DOMAIN_IT
| NOT_AUTHORIZED_FOR_DOMAIN_COM | AUTHORIZED_ONLY_FOR_DOMAIN_ES}
```

Please note that master account access is restricted for API/JSON requests.

```
{
  "token": "",
  "product": "APP_KEY",
  "status": FAIL,
  "error": error
}

error = {INPUT_VALIDATION_ERROR | FORBIDDEN | INVALID_USERNAME_OR_PASSWORD
| NO_SESSION | INVALID_PIN | INVALID_PIN_LOGIN_REQUEST |
INVALID_PIN_LOGIN_REQUEST}
```

Possible failure and exceptional return codes

loginStatus	Description
TRADING_MASTER_SUSPENDED	Suspended Trading Master Account
TRADING_MASTER	Trading Master Account
TELBET_TERMS_CONDITIONS_NA	Telbet terms and conditions rejected
SUSPENDED	the account is suspended

SPANISH_TERMS_ACCEPTANCE_REQUIRED	The latest Spanish terms and conditions version must be accepted
SPAIN_MIGRATION_REQUIRED	Spain migration required
SELF_EXCLUDED	the account has been self excluded
SECURITY_RESTRICTED_LOCATION	the account is restricted due to security concerns
SECURITY_QUESTION_WRONG_3X	the user has entered wrong the security question 3 times
PERSONAL_MESSAGE_REQUIRED	personal message required for the user
PENDING_AUTH	pending authentication
NOT_AUTHORIZED_BY_REGULATOR_IT	the user identified by the given credentials is not authorized in the IT's jurisdictions due to the regulators' policies. Ex: the user for which this session should be created is not allowed to act(play, bet) in the IT's jurisdiction.
NOT_AUTHORIZED_BY_REGULATOR_DK	the user identified by the given credentials is not authorized in the DK's jurisdictions due to the regulators' policies. Ex: the user for which this session should be created is not allowed to act(play, bet) in the DK's jurisdiction.
KYC_SUSPEND	KYC suspended
ITALIAN_CONTRACT_ACCEPTANCE_REQUIRED	The latest Italian contract version must be accepted
INVALID_USERNAME_OR_PASSWORD	the username or password are invalid
INVALID_CONNECTIVITY_TO_REGULATOR_IT	the IT regulator cannot be accessed due to some internal problems in the system behind or in at regulator; timeout cases included.
INVALID_CONNECTIVITY_TO_REGULATOR_DK	the DK regulator cannot be accessed due to some internal problems in the system behind or in at regulator; timeout cases included.
DUPLICATE_CARDS	duplicate cards
DENMARK_MIGRATION_REQUIRED	Denmark migration required
DANISH_AUTHORIZATION_REQUIRED	Danish authorization required
CLOSED	the account is closed
CHANGE_PASSWORD_REQUIRED	change password required
CERT_AUTH_REQUIRED	Certificate required or certificate present but could not authenticate with it
BETTING_RESTRICTED_LOCATION	the account is accessed from a location where betting is restricted
AGENT_CLIENT_MASTER_SUSPENDED	Suspended Agent Client Master
AGENT_CLIENT_MASTER	Agent Client Master
ACCOUNT_NOW_LOCKED	the account was just locked
ACCOUNT_ALREADY_LOCKED	the account is already locked
TEMPORARY_BAN_TOO_MANY_REQUESTS	The limit for successful login requests per minute has been exceeded. New login attempts will be banned for 20 minutes
ACCOUNT_PENDING_PASSWORD_CHANGE	the account must undergo password recovery to reactivate via https://identitysso.betfair.com/view/recoverpassword
ITALIAN_PROFILING_ACCEPTANCE_REQUIRED	You must login to the website to accept the new conditions

Best Practice

- Development & Testing
- Session Management
- Expect: 100 - Continue Header

To optimize performance and ensure that your application is interacting with the Betfair API correctly & as efficiently as possible, we strongly recommend the following as best practice guidelines.

Development & Testing

You should use the Delayed Application Key for any initial development and functional testing. Only apply for Live Application Key access once you are ready to start transacting on the Exchange using your Live Application Key.

Please see the personal betting access overview for more details regarding the difference between Delayed an Live Application Keys

Session Management

Use Login to create a new session and Keep Alive to extend the session beyond the stated session expiry time. A single session can be used across multiple API calls/threads simultaneously.

You should ensure that you handle the INVALID_SESSION_TOKEN error within your code by creating a new session token via the API login met hod.

Expect: 100 - Continue Header

> Sending this header will result in the error: **"The remote server returned an error: (417) Expectation Failed."**

You should be aware that if using the .Net Framework you will need to set the relevant property in the ServicePointManager which then prevents the "Expect" header from being added:

System.Net.ServicePointManager.Expect100Continue = false;

Enabling HTTP compression

HTTP compression is a capability built into both web servers and web clients to reduce the number of bytes transmitted in an HTTP response. This makes better use of available bandwidth and increases performance while reducing download time. When enabled, HTTP protocol data is compressed before it is sent from the server. Clients capable of receiving compressed HTTP data announce that they support compression in the HTTP header. Almost all modern web browsers support HTTP Compression by default.

The Betfair API uses HTTP to handle communication between API clients and servers. Therefore, the JSON messages can be compressed using the same HTTP compression used by web browsers. Custom API applications may need some modification before they can take advantage of this feature. Specifically, they need to send an additional HTTP header to indicate they support receipt of compressed responses from the API. In addition, some environments require you to explicitly decompress the response.

We would therefore recommend that all Betfair API request are sent with the '**Accept-Encoding: gzip, deflate'** request header.

HTTP Persistent connection

We recommend that **Connection: keep-alive** header is set for all requests to guarantee a persistent connection and therefore reducing latency.

Other performance tips

Additional advice regarding optimizing HTTPClient performance can be found via http://hc.apache.org/httpclient-3.x/performance.html

API Demo Tools

Demo tools are available for API-NG for both the **Betting** and **Accounts** API. The Demo Tools can be used by developers to allow quick experimentation with the API.

Why Use Demo Tools?

- All existing API operations are available
- You can easily experiment with the parameters for your API queries.
- You can build sample requests and interact with the API directly requests and responses are output to the JavaScript debug console. For example, using Google Chrome, you can open the Javascript console using the shortcut Ctrl+Shift+J. The same shortcut can also be used with Mozilla Firefox.

Please note that the Demo Tools are provided as-is, and is intended solely as a testing resource.

Obtaining a Session Token for the Demo Tools

There are several ways that you can obtain a session token for the visualiser:

1. Using the pre-populated session token from the website

The Demo Tools will populate the session token field with the value of a session token found in your browsers cookie store for the Betfair.com website.   Login to www.betfair.com (via a seperate browser tab) and refresh the demo tools page to automatically input the Session Token field

2. Manually adding the ssoid (session token) from the website cookie.

Using Google Chrome you can inspect and copy the session directly from the browser by doing the following:

1, Press Ctrl+Shif+J
2. Selection Resource > Cookies > www.betfair.com
3. Copy the value for the cookie with name **ssoid** and paste this into the Demo Tool

3. Using the API-NG Interactive login sample

Download SampleAPI.exe which is a compiled version of the sample interactive login code which is described in the documentation here, with the source from the API-NG Sample code github repo.

Enter your application key and follow the login instructions; your session token with be extracted from your session and presented to you via the application.

Market Data Request Limits

Although you can request multiple markets from listMarketBook, listMarketCatalogue and listMarketProfitandLoss, there are limits on the amount of data requested in **one request.**

sum(Weight) * **number market ids** must not exceed 200 points per request

The following tables explain the "weighting" of data for each **MarketProjection** or **PriceProjection**. If you exceed the maximum weighting of 200 points, the API will return a **TOO_MUCH_DATA** error.

listMarketCatalogue

MarketProjection	Weight
MARKET_DESCRIPTION	1
RUNNER_DESCRIPTION	0
EVENT	0

EVENT_TYPE	0
COMPETITION	0
RUNNER_METADATA	1
MARKET_START_TIME	0

listMarketBook

PriceProjection	Weight
Null (No PriceProjection set)	2
SP_AVAILABLE	3
SP_TRADED	7
EX_BEST_OFFERS	5
EX_ALL_OFFERS	17
EX_TRADED	17

Please note: specific combinations of priceProjections will carry different weights that aren't the sum of their individual weights. Please see a summary of these below:

PriceProjection	Weight
EX_BEST_OFFERS + EX_TRADED	20
EX_ALL_OFFERS + EX_TRADED	32

If **exBestOffersOverrides** is used the weight is is calculated by **weight * (requestedDepth/3)**.

listMarketProfitandLoss

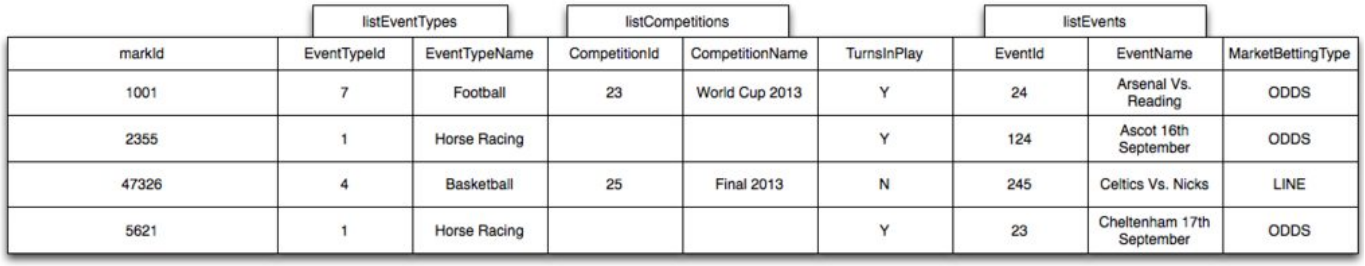
PriceProjection	Weight
Not applicable	4

Understanding Market Navigation

> **Recreating Website Navigation?**
> **Please note:** if you want to recreate the exact navigation hierarchy used by the Betfair website, please use the Navigation Data for Applications service.

In API-NG, navigating markets uses the concepts of faceted search. You've probably seen faceted search used on sites like eBay or Amazon where you pick a category, for example "Shoes" and then you see a list of shoes. Then, you can narrow your search by facets. For example, by colour or price. In a similar way, the new Sports API lets you find markets you're interested in and then get data back about a facet of those markets.

The way to think of the API Navigation operations is to imagine a single table that lists every market, along with various metadata about the market. The columns of the table are the Facets (and the Nav Operations in API return some of them in combination):



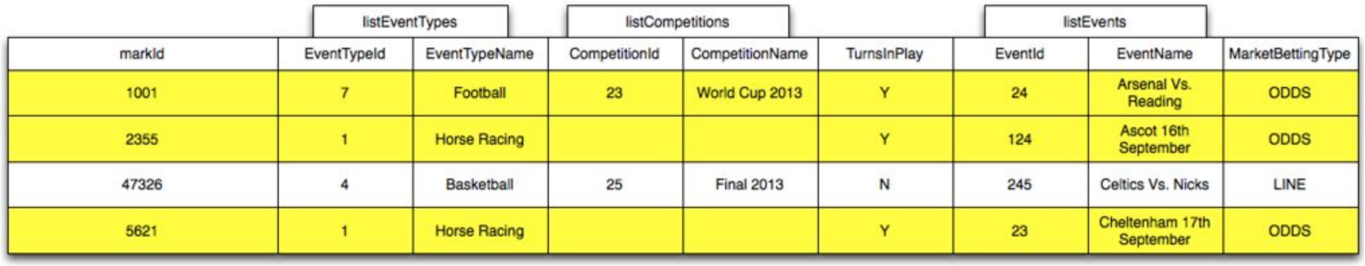
	listEventTypes		listCompetitions			listEvents		
markId	EventTypeId	EventTypeName	CompetitionId	CompetitionName	TurnsInPlay	EventId	EventName	MarketBettingType
1001	7	Football	23	World Cup 2013	Y	24	Arsenal Vs. Reading	ODDS
2355	1	Horse Racing			Y	124	Ascot 16th September	ODDS
47326	4	Basketball	25	Final 2013	N	245	Celtics Vs. Nicks	LINE
5621	1	Horse Racing			Y	23	Cheltenham 17th September	ODDS

As you can see, the table has four markets along with some of the metadata associated with that market. Of course, the actual table would have tens of thousands of markets and more columns.

Along the top, you can see three of the navigation operations. Each of those operations takes a "MarketFilter". The MarketFilter filters the table of markets and selects the ones that match. The MarketFilter can contain things like "going in play" or "eventId 7", etc. So, if you called "listCompetitions" and passed in a filter that looked like:
Code:

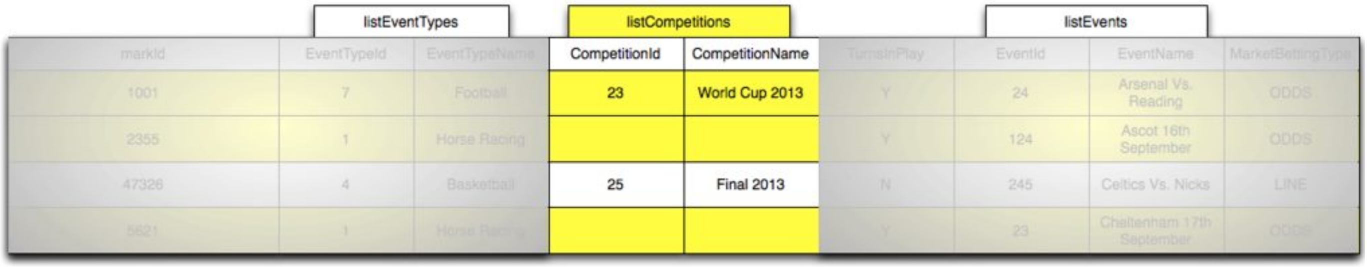
```
{ "filter": {"turnsInPlay" : true }}
```

The selected markets would look like:



	listEventTypes		listCompetitions			listEvents		
markId	EventTypeId	EventTypeName	CompetitionId	CompetitionName	TurnsInPlay	EventId	EventName	MarketBettingType
1001	7	Football	23	World Cup 2013	Y	24	Arsenal Vs. Reading	ODDS
2355	1	Horse Racing			Y	124	Ascot 16th September	ODDS
47326	4	Basketball	25	Final 2013	N	245	Celtics Vs. Nicks	LINE
5621	1	Horse Racing			Y	23	Cheltenham 17th September	ODDS

In the example, the 1001, 2355, and the 5621 market are the ones that match the filter. As you called "listCompetitions" with that filter, then the data returned is the columns containing competition information:



	listEventTypes		listCompetitions			listEvents		
markId	EventTypeId	EventTypeName	CompetitionId	CompetitionName	TurnsInPlay	EventId	EventName	MarketBettingType
1001	7	Football	23	World Cup 2013	Y	24	Arsenal Vs. Reading	ODDS
2355	1	Horse Racing			Y	124	Ascot 16th September	ODDS
47326	4	Basketball	25	Final 2013	N	245	Celtics Vs. Nicks	LINE
5621	1	Horse Racing			Y	23	Cheltenham 17th September	ODDS

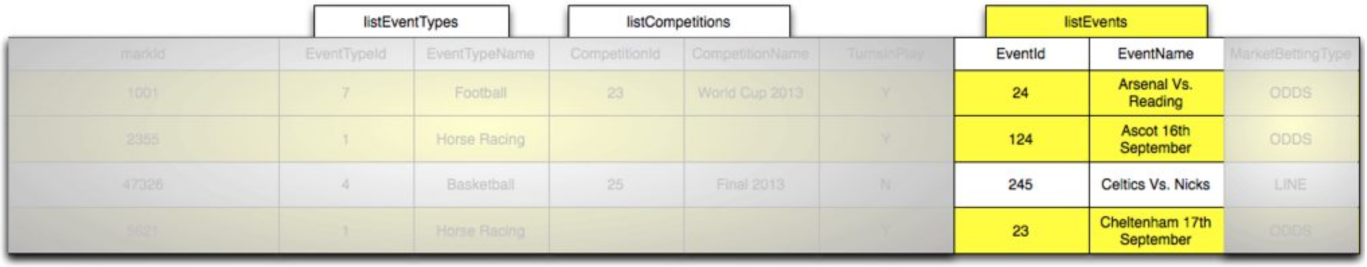
In this example, you'd get a list of competitions like so:

Code:

```
{"competitionId" : "23", "competitionName" : "World Cup 2013"}
```

Notice that the competition "Final 2013" was not returned. This is because the market was not selected by the MarketFilter. Also, notice that although the market 2355 was selected by the MarketFilter, it is not associated with a competition, so no competition id and name were returned for that market.

As a further example, suppose that you used the same MarketFilter as before while calling listEvents. In that case, the data returned is the columns containing eventId and eventName for the markets that match the filter (i.e., turnsInPlay = True):



	listEventTypes		listCompetitions			listEvents		
markId	EventTypeId	EventTypeName	CompetitionId	CompetitionName	TurnsInPlay	EventId	EventName	MarketBettingType
1001	7	Football	23	World Cup 2013	Y	24	Arsenal Vs. Reading	ODDS
2355	1	Horse Racing			Y	124	Ascot 16th September	ODDS
47326	4	Basketball	25	Final 2013	N	245	Celtics Vs. Nicks	LINE
5621	1	Horse Racing			Y	23	Cheltenham 17th September	ODDS

In this example, you'd get a list of events like so:

Code:

```
[{"eventId" : "24", "eventName" : "Arsenal Vs. Reading"}, {"eventId" : "124", "eventName" : "Ascot 16th
September"}, {"eventId" : "23", "eventName" : "Cheltenham 17th September"}]
```

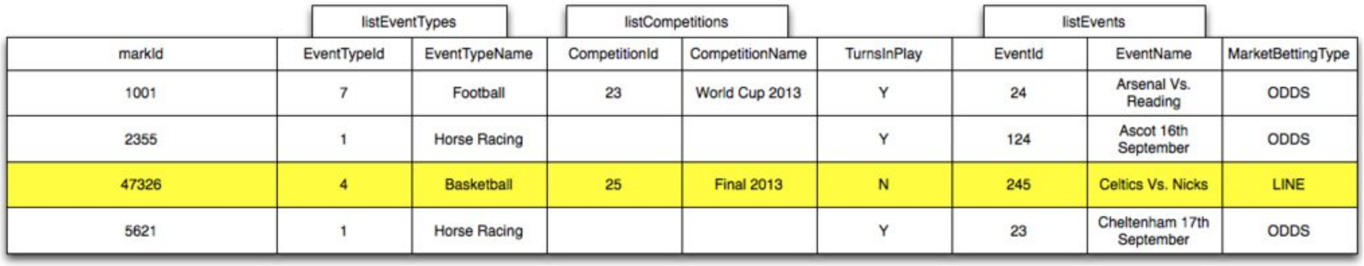
Again, notice that the event "Celtics vs Nicks" was not returned because it was not selected by the MarketFilter.

One further example. Suppose you were only interested in Line markets and you want to find all of the Sports (EventTypes) that contain Line markets. You would create a MarketFilter like this:

Code:

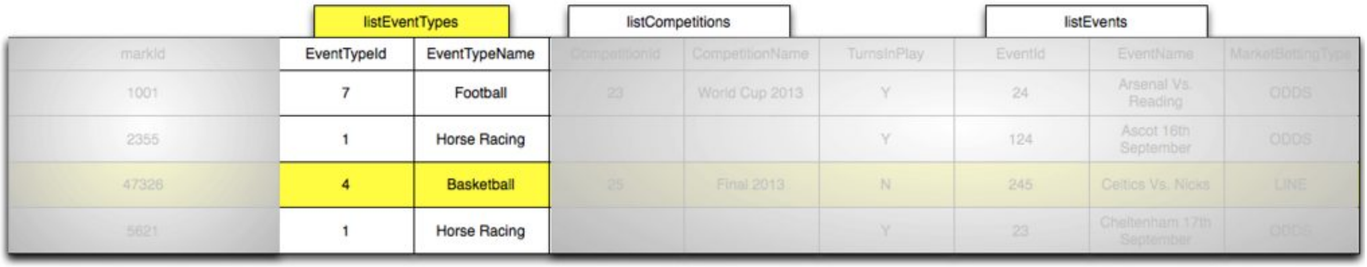
```
{ "filter": {"MarketBettingType" : "LINE" }}
```

And call listEventTypes with that filter. The markets selected by that MarketFilter would be:



	listEventTypes		listCompetitions			listEvents		
markId	EventTypeId	EventTypeName	CompetitionId	CompetitionName	TurnsInPlay	EventId	EventName	MarketBettingType
1001	7	Football	23	World Cup 2013	Y	24	Arsenal Vs. Reading	ODDS
2355	1	Horse Racing			Y	124	Ascot 16th September	ODDS
47326	4	Basketball	25	Final 2013	N	245	Celtics Vs. Nicks	LINE
5621	1	Horse Racing			Y	23	Cheltenham 17th September	ODDS

and the data returned would be:



	listEventTypes		listCompetitions			listEvents		
markId	EventTypeId	EventTypeName	CompetitionId	CompetitionName	TurnsInPlay	EventId	EventName	MarketBettingType
1001	7	Football	23	World Cup 2013	Y	24	Arsenal Vs. Reading	ODDS
2355	1	Horse Racing			Y	124	Ascot 16th September	ODDS
47326	4	Basketball	25	Final 2013	N	245	Celtics Vs. Nicks	LINE
5621	1	Horse Racing			Y	23	Cheltenham 17th September	ODDS

Code:

```
{"eventTypeId" : "4", "EventTypeName" : "Basketball"}
```

Using the set of navigation operations, you can easily create a menu tree of markets in whatever hierarchy you want. You can can listEventTypes to get the Sports as the top of your menu. Or, you could call "listCountries" and use markets in a country as the root of the tree.

There is also listMarketCatalogue so if you didn't care about creating a visual navigation, you could simply call listMarketCatalogue with a MarketFilter defining the markets you're interested in and get back information about those markets.

Reference Guide

Documentation Versions

The below log details any major updates to the API-NG Reference Guide.

Version	Description	Date
V 2.3	Update to include Race Status API with listRaceDetails operations	26-10-2015
V 2.2	Updated to include eachWayDivisor field in listMarketCatalogue response	21-04-2015

V 2.1	Updated to include additional country field in getAccountDetails response	09-01-2015
V 2.0	Updated to include transferFunds, updateApplicationSubscription operation and updates to getAccountfunds	29-09-2014
V 1.9	Updated to include Navigation Data For Applications	26-08-2014
V.1.8	Updated to include changes to error handing in listCurrentOrders	14-07-2014
V.1.7	Updated to include changes to getAccountFunds & Vendor Services API	16-06-2014
V.1.6	Updated to include getAccountStatement and listCurrencyRates	29-04-2014
V.1.5	Updated to include the Heartbeat API	09-04-2014
V.1.4	Updated to include changes to listClearedOrders and listMarketCatalogue.	24-02-2014
V.1.3	Introduced the listMarketProfitAndLoss functionality to the Betting API.	03-02-2014
V.1.2	Introduced the listClearedOrders functionality to the Betting API.	17-12-2013
V.1.1	Updates to the Vendor API, listCurrentOrders and addition of endpoint for Australian Exchange	07-10-2013
V1.0	Introduced login and vendor services functionality, moved to final endpoints.	29-07-2013
V0.4	Introduced additional Account functionality (getAccountFunds and getAccountDetails)	01-07-2013
V0.3	Introduced Account API-NG	03-06-2013
V0.2	Introduced betting functionality and listCurrentOrders	20-05-2013
V0.1	Initial release	09-12-2012

Navigation Data For Applications

Endpoint & Required Headers

This Navigation Data for Applications service allows the retrieval of the full Betfair market navigation menu from a compressed file.

Exchange	HTTP Method	Endpoint
UK	GET	https://api.betfair.com/exchange/betting/rest/v1/en/navigation/menu.json
ITALY	GET	https://api.betfair.it/exchange/betting/rest/v1/en/navigation/menu.json
SPAIN	GET	https://api.betfair.es/exchange/betting/rest/v1/en/navigation/menu.json

> **Best Practice**
> **The file data is cached and new request for the file once an hour should be suitable for those looking to accurately recreate the Betfair navigation menu.**

The following request headers are required:

- **X-Application** - Your Application Key
- **X-Authentication -** Your session token, obtained from the API login response.

Example Request

```
GET
https://api.betfair.com/exchange/betting/rest/v1/en/navigation/menu.json
Connection: keep-alive
X-Application: <AppKey>
X-Authentication: <SessionToken>
Accept: application/json
Accept-Encoding: gzip,deflate
```

Supported Locales

The following languages are supported by the navigation file:

en | en_GB | bg | da | de | el | es | it | pt | ru | sv

English - en

Spanish - es

Italian - it

German - de

Swedish - sv

Portuguese -pt

Russian - ru

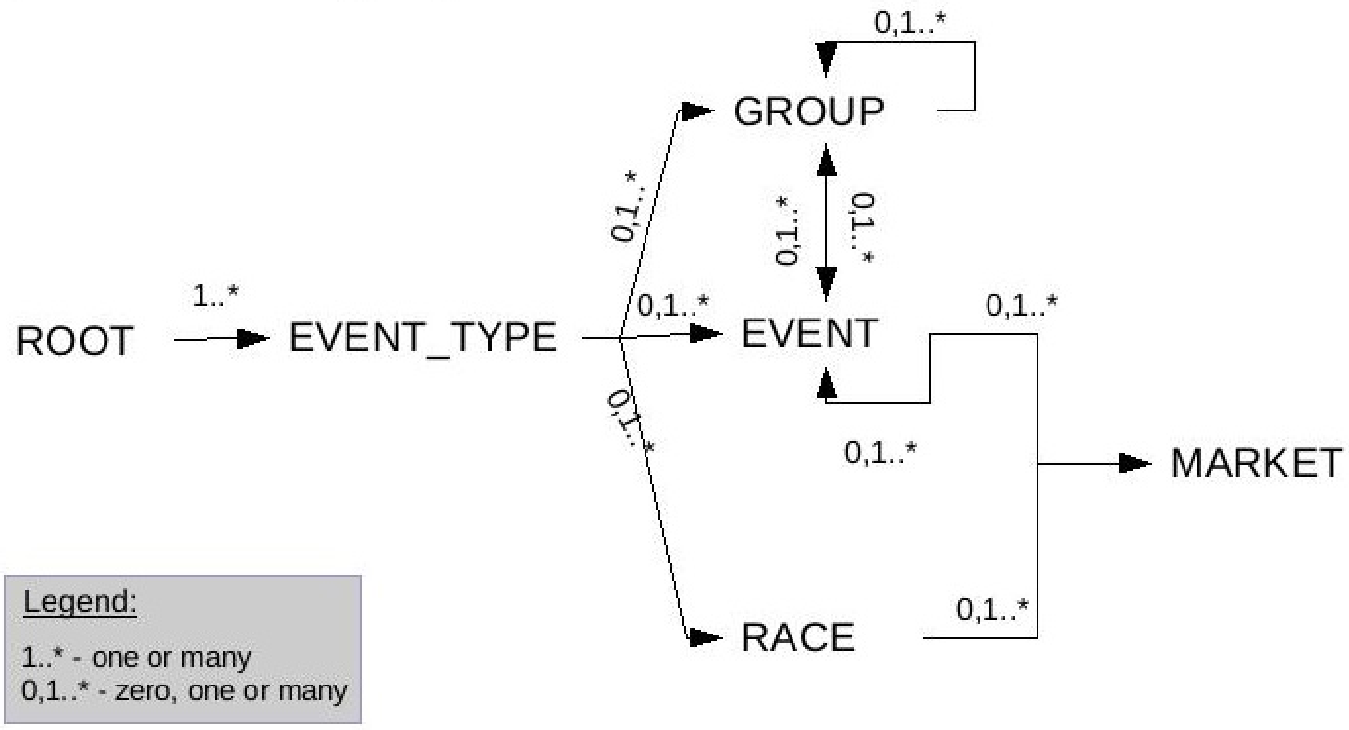
Greek - el

Bulgarian – bg

Danish - da

Navigation Data File Structure

This is a diagram showing how the Navigation Data File is structured.



In plain English:

A **ROOT** group node has one or many **EVENT_TYPE** nodes

An **EVENT_TYPE** node has zero, one or many **GROUP** nodes

An **EVENT_TYPE** node has zero, one or many **EVENT** nodes

A Horse Racing **EVENT_TYPE** node has zero, one or many **RACE** nodes

A **RACE** node has one or many **MARKET** nodes

A **GROUP** node has zero, one or many **EVENT** nodes

A **GROUP** node has zero, one or many **GROUP** nodes

An **EVENT** node has zero, one or many **MARKET** nodes

An **EVENT** node has zero, one or many **GROUP** nodes

An **EVENT** node has zero, one or many **EVENT** nodes

JSON Model Structure

ROOT

```
{
    "children": [
        {
            EVENT_TYPE1
        },
        {
            EVENT_TYPE2
        },
        ...
    ],
    "id": 0, // always 0
    "name": "ROOT", // always ROOT
    "type": "GROUP" // always GROUP
}
```

EVENT_TYPE

```
{
    "children": [
        {
            GROUP or EVENT or RACE (RACE only if Greyhounds/Horse Racing)
        },
        ...
    ],
    "id": "1", // Betfair specific eventTypeId
    "name": "Soccer",
    "type": "EVENT_TYPE"
}
```

GROUP

```
{
    "children": [
        {
            GROUP or EVENT
        },
        ...
    ],
    "id": "74568202414", // Not a Betfair specific id, different for every GROUP
    "name": "Womens Soccer",
    "type": "GROUP"
}
```

EVENT

```
{
    "children": [
        {
            GROUP, MARKET or EVENT
        },
        ...
    ],
    "id": "27244118", // Betfair specific eventId
    "name": "South Korea U20 (W) v Mexico U20 (W)",
    "countryCode": "GB",
    "type": "EVENT"
}
```

RACE

```
{
    "children": [
        {
            MARKET
        },
        ...
    ],
    "id": "27247020.1115", // Betfair specific raceId
    "name": "1300m 3yo",
    "startTime": "2014-08-12T11:15:00.000Z",
    "type": "RACE",
    "venue": "Deauville",
"raceNumber" : "R1", // US specific information about race numbers
    "countryCode": "GB"
}
```

MARKET

```
{
    "exchangeId": "1", // Betfair specific exchangeId
    "id": "1.114881860", // Betfair specific marketId
    "marketStartTime": "2014-08-14T00:00:00.000Z", // Betfair specific marketStartTime
    "marketType": "WIN", // Betfair specific marketType (e.g. PLACE, WIN, FORECAST etc.)
    "numberOfWinners": "2", // Betfair specific number of winners
    "name": "Over/Under 6.5 Goals",
    "type": "MARKET"
}
```

Betting API

Endpoints

Please find the details for the current Betting API endpoints. If you have an **Italian** or **Spanish Exchange** account please see further details via the links below:

Global Exchange

Interface	Endpoint	JSON-RPC Prefix	<methodname> Example

JSON-RPC	https://api.betfair.com/exchange/betting/json-rpc/v1	<methodname>	SportsAPING/v1.0/listMarketBook
JSON REST	https://api.betfair.com/exchange/betting/rest/v1.0/		listMarketBook/

- Betting on Spanish Exchange
- Betting On Italian Exchange

Betting Operations

Operation summary

Type	Operation	Description
List< EventTypeResult >	**listEventTypes (MarketFilter filter** ,Stringlocale **)**	Returns a list of Event Types (i.e. Sports) associated with the markets selected by the MarketFilter.
List< CompetitionResult >	**listCompetitions (MarketFilter filter** ,Stringlocale **)**	Returns a list of Competitions (i.e., World Cup 2013) associated with the markets selected by the MarketFilter. Currently only Football markets have an associated competition.
List< TimeRangeResult >	**listTimeRanges (MarketFilter filter , TimeGranularity granularity)**	Returns a list of time ranges in the granularity specified in the request (i.e. 3PM to 4PM, Aug 14th to Aug 15th) associated with the markets selected by the MarketFilter.
List< EventResult >	**listEvents (MarketFilter filter** ,Stringlocale **)**	Returns a list of Events (i.e, Reading vs. Man United) associated with the markets selected by the MarketFilter.
List< MarketTypeResult >	**listMarketTypes (MarketFilter filter** ,Stringlocale **)**	Returns a list of market types (i.e. MATCH_ODDS, NEXT_GOAL) associated with the markets selected by the MarketFilter. The market types are always the same, regardless of locale.
List< CountryCodeResult >	**listCountries (MarketFilter filter** ,Stringlocale **)**	Returns a list of Countries associated with the markets selected by the MarketFilter.
List< VenueResult >	**listVenues (MarketFilter filter** ,Stringlocale **)**	Returns a list of Venues (i.e. Cheltenham, Ascot) associated with the markets selected by the MarketFilter. Currently, only Horse Racing markets are associated with a Venue.
List< MarketCatalogue >	**listMarketCatalogue (MarketFilter filter** ,Set< MarketProjection >marketProjection, MarketSort sort, int maxResults, Stringlocale **)**	Returns a list of information about published (ACTIVE/SUSPENDED) markets that does not change (or changes very rarely).
List< MarketBook >	**listMarketBook (List<String>marketIds** , PriceProjection priceProjection, OrderProjection orderProjection, MatchProjection matchProjection,StringcurrencyCode,Stringlocale, Date matchedSince, Set<BetId > betIds **)**	Returns a list of dynamic data about markets. Dynamic data includes prices, the status of the market, the status of selections, the traded volume, and the status of any orders you have placed in the market

List<MarketBook>	**List<MarketBook> listRunnerBook (MarketId marketId, SelectionId selectionId**, double handicap, PriceProjection priceProjection, OrderProjection orderProjection, MatchProjection matchProjection, boolean includeOverallPosition, boolean partitionMatchedByStrategyRef, Set<String> customerStrategyRefs, StringcurrencyCode,Stringlocale, Date matchedSince, Set<BetId> betIds**) throws APINGException**	Returns a list of dynamic data about **a market** and a **specified runner**. Dynamic data includes prices, the status of the market, the status of selections, the traded volume, and the status of any orders you have placed in the market..
List<MarketProfitAndLoss>	**listMarketProfitAndLoss (Set<MarketId> marketIds**, boolean includeSettledBets, boolean includeBspBets, boolean netOfCommission **)**	Retrieve profit and loss for a given list of OPEN markets. The values are calculated using matched bets and optionally settled bets
CurrentOrderSummaryReport	**listCurrentOrders (Set<String>betIds**,Set<String>marketIds, OrderProjection orderProjection, TimeRange placedDateRange, OrderBy orderBy, SortDir sortDir,intfromRecord,intrecordCount **)**	Returns a list of your current orders.
ClearedOrderSummaryReport	**listClearedOrders (BetStatus betStatus**, Set<EventTypeId> eventTypeIds, Set<EventId> eventIds, Set<MarketId> marketIds, Set<RunnerId> runnerIds, Set<BetId> betIds, Side side, TimeRange settledDateRange, GroupBy groupBy, boolean includeItemDescription, String locale, int fromRecord, int recordCount **)**	Returns a list of settled bets based on the bet status, ordered by settled date
PlaceExecutionReport	**placeOrders (StringmarketId**, **List<PlaceInstruction> instructions**, StringcustomerRef, MarketVersion marketVersion, String customerStrategyRef, boolean async**)**	Place new orders into market.
CancelExecutionReport	**cancelOrders (StringmarketId**, **List<CancelInstruction>instructions**, StringcustomerRef **)**	Cancel all bets OR cancel all bets on a market OR fully or partially cancel particular orders on a market
ReplaceExecutionReport	**replaceOrders (StringmarketId**, **List< ReplaceInstruction> instructions**, StringcustomerRef, MarketVersion marketVersion, boolean async)	This operation is logically a bulk cancel followed by a bulk place. The cancel is completed first then the new orders are placed.
UpdateExecutionReport	**updateOrders (StringmarketId**, **List< UpdateInstruction> instructions**, StringcustomerRef **)**	Update non-exposure changing fields

listCompetitions

Operation

listCompetitions

List< CompetitionResult > listCompetitions (MarketFilter filter ,Stringlocale **) throws APINGException**

Returns a list of Competitions (i.e., World Cup 2013) associated with the markets selected by the MarketFilter. Currently only Football markets have an associated competition.

Parameter name	Type	Required	Description
filter	MarketFilter	✓	The filter to select desired markets. All markets that match the criteria in the filter are selected.
locale	String		The language used for the response. If not specified, the default is returned.

Return type	Description
List< CompetitionResult >	output data

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

listCountries

Operation

listCountries

List< CountryCodeResult > listCountries (MarketFilter filter ,Stringlocale **) throws APINGException**

Returns a list of Countries associated with the markets selected by the MarketFilter.

Parameter name	Type	Required	Description
filter	MarketFilter	✓	The filter to select desired markets. All markets that match the criteria in the filter are selected.
locale	String		The language used for the response. If not specified, the default is returned.

Return type	Description
List< CountryCodeResult >	output data

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

listCurrentOrders

Operation

listCurrentOrders

CurrentOrderSummaryReport listCurrentOrders (Set<String>betIds,Set<String>marketIds, OrderProjection orderProjection, TimeRange placedDateRange, TimeRange dateRange, OrderBy orderBy, SortDir sortDir,intfromRecord,intrecordCount **) throws APINGException**

Returns a list of your current orders. Optionally you can filter and sort your current orders using the various parameters, setting none of the parameters will return all of your current orders up to a maximum of 1000 bets, ordered BY_BET and sorted EARLIEST_TO_LATEST. To retrieve more than 1000 orders, you need to make use of the fromRecord and recordCount parameters.

> **Best Practice**
> To efficiently track new bet matches from a specific time, customers should use a combination of the **dateRange**, **orderBy "BY_MATCH_TIME"** and **orderProjection "ALL"** to filter fully/partially matched orders from the list of returned bets. The response will then filter out any bet records that have no matched date and provide a list of betIds in the order which they are fully/partially matched from the date and time specified in the dateRange field.

Parameter name	Type	Required	Description
betIds	Set<String>		Optionally restricts the results to the specified bet IDs. A maximum of 250 betId's, or a combination of 250 betId's & marketId's are permitted.
marketIds	Set<String>		Optionally restricts the results to the specified market IDs. A maximum of 250 marketId's, or a combination of 250 marketId's & betId's are permitted.
orderProjection	OrderProjection		Optionally restricts the results to the specified order status.
customerOrderRefs	Set<CustomerOrderRef>		Optionally restricts the results to the specified customer order references.
customerStrategyRefs	Set<CustomerStrategyRef>		Optionally restricts the results to the specified customer strategy references.
dateRange	TimeRange		Optionally restricts the results to be from/to the specified date, these dates are contextual to the orders being returned and therefore the dates used to filter on will change to placed, matched, voided or settled dates depending on the orderBy. This date is inclusive, i.e. if an order was placed on exactly this date (to the millisecond) then it will be included in the results. If the from is later than the to, no results will be returned.
orderBy	OrderBy		Specifies how the results will be ordered. If no value is passed in, it defaults to BY_BET. Also acts as a filter such that only orders with a valid value in the field being ordered by will be returned (i.e. BY_VOID_TIME returns only voided orders, BY_SETTLED_TIME (applies to partially settled markets) returns only settled orders and BY_MATCH_TIME returns only orders with a matched date (voided, settled, matched orders)). Note that specifying an orderBy parameter defines the context of the date filter applied by the dateRange parameter (placed, matched, voided or settled date) - see the dateRange parameter description (above) for more information. See also the OrderBy type definition.
sortDir	SortDir		Specifies the direction the results will be sorted in. If no value is passed in, it defaults to EARLIEST_TO_LATEST.
fromRecord	int		Specifies the first record that will be returned. Records start at index zero, not at index one.
recordCount	int		Specifies how many records will be returned from the index position 'fromRecord'. Note that there is a page size limit of 1000. A value of zero indicates that you would like all records (including and from 'fromRecord') up to the limit.

Return type	Description
CurrentOrderSummaryReport	

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

listClearedOrders

Operation

listClearedOrders

ClearedOrderSummaryReport listClearedOrders (BetStatus betStatus, Set<EventTypeId> eventTypeIds, Set<EventId> eventIds, Set<MarketId> marketIds, Set<RunnerId> runnerIds, Set<BetId> betIds, Side side, TimeRange settledDateRange, Group By groupBy, boolean includeItemDescription, String locale, int fromRecord, int recordCount **) throws APINGException**

Returns a list of settled bets based on the bet status, ordered by settled date. To retrieve more than 1000 records, you need to make use of the fromRecord and recordCount parameters. By default the service will return all available data for the last 90 days (see Best Practice note below). The fields available at each roll-up are available here

Best Practice
You should specify a **settledDateRange "from"** date when making requests for data. This reduces the amount of data that requires downloading & improves the speed of the response. Specifying a "from" date of the last call will ensure that only new data is returned.

Parameter name	Type	Required	Description
betStatus	BetStatus	✓	Restricts the results to the specified status.
eventTypeIds	Set<EventTypeId>		Optionally restricts the results to the specified Event Type IDs.
eventIds	Set<EventId>		Optionally restricts the results to the specified Event IDs.
marketIds	Set<Market Id>		Optionally restricts the results to the specified market IDs.
runnerIds	Set<RunnerId>		Optionally restricts the results to the specified Runners.
betIds	Set<BetId>		Optionally restricts the results to the specified bet IDs.
customerOrderRefs	Set<CustomerOrderRef>		Optionally restricts the results to the specified customer order references.
customerStrategyRefs	Set<CustomerStrategyRef>		Optionally restricts the results to the specified customer strategy references.
side	Side		Optionally restricts the results to the specified side.
settledDateRange	TimeRange		Optionally restricts the results to be from/to the specified settled date. This date is inclusive, i.e. if an order was cleared on exactly this date (to the millisecond) then it will be included in the results. If the from is later than the to, no results will be returned.
groupBy	GroupBy		How to aggregate the lines, if not supplied then the lowest level is returned, i.e. bet by bet This is only applicable to SETTLED BetStatus.
includeItemDescription	boolean		If true then an ItemDescription object is included in the response.
locale	String		The language used for the itemDescription. If not specified, the customer account default is returned.
fromRecord	int		Specifies the first record that will be returned. Records start at index zero.
recordCount	int		Specifies how many records will be returned, from the index position 'fromRecord'. Note that there is a page size limit of 1000. A value of zero indicates that you would like all records (including and from 'fromRecord') up to the limit.

Return type	Description
ClearedOrderSummaryReport	

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

listClearedOrders - Roll-up Fields Available

The below table indicates fields will be available at each roll-up when making requests to **listClearedOrders** using the **groupBy** parameter.

Rollup level:	BET	SIDE	MARKET	EVENT	EVENT_TYPE	EXCHANGE
Settled As	Y	Y	Y	Y	Y	Y
Settled Date	Y	MAX	MAX	MAX	MAX	MAX
Bet Count	Y	Y	Y	Y	Y	Y
Profit	Y	SUM	SUM	SUM	SUM	SUM
Exchange Id	Y	Y	Y	Y	Y	Y
Event Type Id	Y	Y	Y	Y	Y	N
Event Id	Y	Y	Y	Y	N	N
Market Id	Y	Y	Y	N	N	N
Selection Id	Y	Y	N	N	N	N
Handicap	Y	Y	N	N	N	N
Side	Y	Y	N	N	N	N
Price Requested	Y	AVG	N$^{(AVG)}$	N	N	N
Price Matched	Y	AVG	N$^{(AVG)}$	N	N	N
Size Settled	Y	SUM	N$^{(SUM)}$	N	N	N
Price Reduced	Y	Y	Y	N	N	N
Commission	N	N	Y	SUM	SUM	SUM
Bet Id	Y	N	N	N	N	N
Placed Date	Y	MAX	MAX	N	N	N
Persistence Type	Y	Y	Y	N	N	N
Order Type	Y	Y	Y	N	N	N
Regulator Code	Y	Y	Y	N	N	N
Regulator Auth Code	Y	Y	Y	N	N	N
Voided Date(where applicable)	Y	MAX	MAX	N	N	N
BetOutcome	Y	N	N	N	N	N

listEvents

Operation

listEvents

List< EventResult > listEvents (MarketFilter filter ,Stringlocale **) throws APINGException**

Returns a list of Events (i.e, Reading vs. Man United) associated with the markets selected by the MarketFilter.

Parameter name	Type	Required	Description
filter	MarketFilter	✓	The filter to select desired markets. All markets that match the criteria in the filter are selected.
locale	String		The language used for the response. If not specified, the default is returned.

Return type	Description
List< EventResult >	output data

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

listEventTypes

Operation

listEventTypes

List< EventTypeResult > listEventTypes (MarketFilter filter ,Stringlocale **) throws APINGException**

Returns a list of Event Types (i.e. Sports) associated with the markets selected by the MarketFilter.

Parameter name	Type	Required	Description
filter	MarketFilter	✓	The filter to select desired markets. All markets that match the criteria in the filter are selected.
locale	String		The language used for the response. If not specified, the default is returned.

Return type	Description
List< EventTypeResult >	output data

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

listMarketBook

Operation

listMarketBook

List< MarketBook > listMarketBook (List<String>marketIds , PriceProjection priceProjection, OrderProjection orderProjection, MatchProjection matchProjection, boolean includeOverallPosition, boolean partitionMatchedByStrategyRef, Set<String> customerStrategyRefs, StringcurrencyCode,Stringlocale, Date matchedSince, Set<BetId> betIds**) throws APINGException**

Returns a list of dynamic data about markets. Dynamic data includes prices, the status of the market, the status of selections, the traded volume, and the status of any orders you have placed in the market.

Please note: Separate requests should be made for **OPEN & CLOSED** markets. Request that include both **OPEN & CLOSED** markets will only return those markets that are **OPEN.**

> Market Data Request Limits apply to requests made to **listMarketBook** that include price or order projections.
>
> Calls to **listMarketBook** should be made up to a maximum of 5 times per second to a single marketId.

> **Best Practice**
> Customers seeking to use listMarketBook to obtain price, volume, unmatched **(EXECUTABLE)** orders and matched position in a single operation should provide an **OrderProjection**of **“EXECUTABLE”** in their listMarketBook request and receive all unmatched **(EXECUTABLE)** orders and the aggregated matched volume from all orders irrespective of whether they are partially or fully matched. The level of matched volume aggregation (**MatchProjection**) requested should be **ROLLED_UP_BY_AVG_PRICE** or **ROLLED_UP_BY_PRICE**, the former being preferred. This provides a single call in which you can track prices, traded volume, unmatched orders and your evolving matched position with a reasonably fixed, minimally sized response.

Parameter name	Type	Required	Description
marketIds	List<String>	✓	One or more market ids. The number of markets returned depends on the amount of data you request via the price projection.
priceProjection	PriceProjection		The projection of price data you want to receive in the response.
orderProjection	OrderProjection		The orders you want to receive in the response.
matchProjection	MatchProjection		If you ask for orders, specifies the representation of matches.
includeOverallPosition	boolean		If you ask for orders, returns matches for each selection. Defaults to true if unspecified.
partitionMatchedByStrategyRef	boolean		If you ask for orders, returns the breakdown of matches by strategy for each selection. Defaults to false if unspecified.
customerStrategyRefs	Set<String>		If you ask for orders, restricts the results to orders matching any of the specified set of customer defined strategies. Also filters which matches by strategy for selections are returned, if partitionMatchedByStrategyRef is true. An empty set will be treated as if the parameter has been omitted (or null passed).
currencyCode	String		A Betfair standard currency code. If not specified, the default currency code is used.
locale	String		The language used for the response. If not specified, the default is returned.
matchedSince	Date		If you ask for orders, restricts the results to orders that have at least one fragment matched since the specified date (all matched fragments of such an order will be returned even if some were matched before the specified date). All EXECUTABLE orders will be returned regardless of matched date.
betIds	Set<BetId>		If you ask for orders, restricts the results to orders with the specified bet IDs.

Return type	Description
List< MarketBook >	output data

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

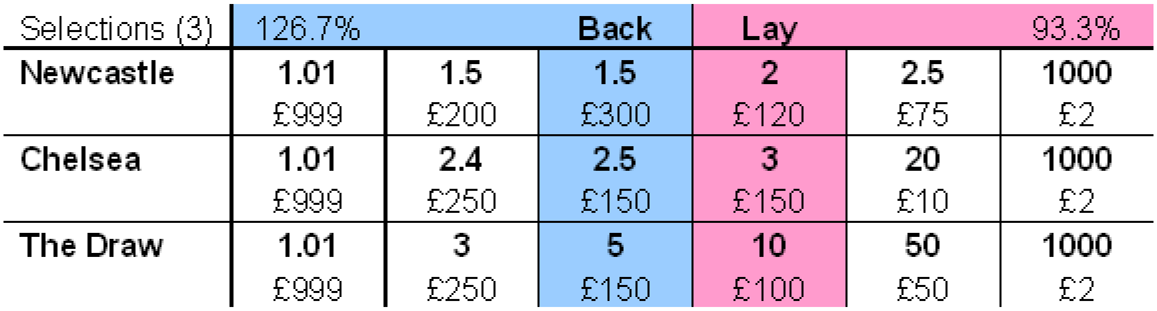
Virtual Bets

The Betfair Exchange uses a 'cross matching' algorithm to display the best possible prices (bets) available by taking into account the back and lay offers (unmatched bets) across all selections.

You can return virtual bets in the response when using API-NG by including the virtualise":"true" in the listMarketBook request e.g. [{"jsonrpc": "2.0", "method": "SportsAPING/v1.0/listMarketBook", "params": {"marketIds":["1.114101556"],"priceProjection":{"priceData":["EX_BEST_OFFERS"],"virtualise":"true"}}, "id": 1}]

One of the easiest ways to understand how we generate virtual bets for cross matching is to work through a couple of examples.

Consider the following market and what would happen if we placed a very large back bet at 1.01 on The Draw:



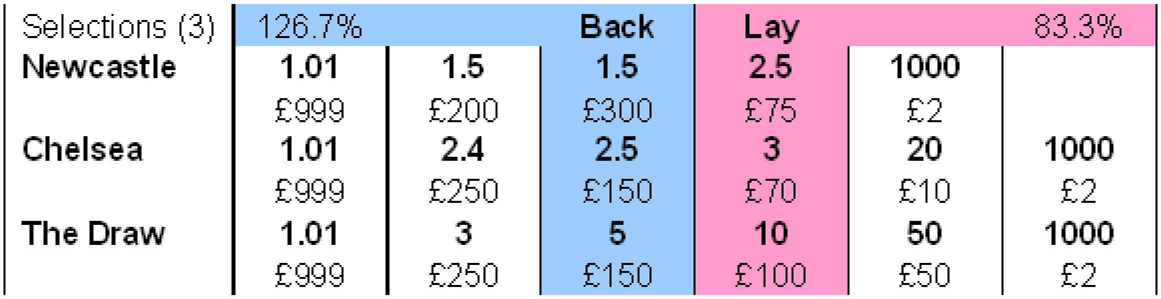
Selections (3)	126.7%		Back	Lay		93.3%
Newcastle	1.01 £999	1.5 £200	1.5 £300	2 £120	2.5 £75	1000 £2
Chelsea	1.01 £999	2.4 £250	2.5 £150	3 £150	20 £10	1000 £2
The Draw	1.01 £999	3 £250	5 £150	10 £100	50 £50	1000 £2

Without cross matching, this bet would be matched in three portions;

1) £150 at 5.0,
2) £250 at 3.0,
and £999 at 1.01 with anything remaining being left unmatched.

With cross matching we can do better.

We have a back bet on Newcastle for £120 at 2.0 and a back bet on Chelsea for £150 at 3.0 (shown in pink on the available to lay side of the market).



Selections (3)	126.7%		Back	Lay		83.3%
Newcastle	1.01 £999	1.5 £200	1.5 £300	2.5 £75	1000 £2	
Chelsea	1.01 £999	2.4 £250	2.5 £150	3 £70	20 £10	1000 £2
The Draw	1.01 £999	3 £250	5 £150	10 £100	50 £50	1000 £2

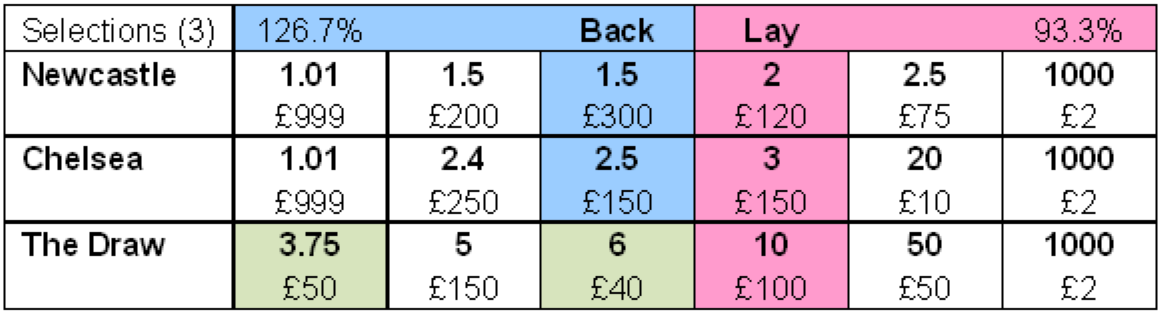
These two bets can be matched against a back bet on The Draw at a price of 6.0, since 2.0, 3.0, and 6.0 form a 100% book. To ensure that the book is balanced, we choose the stakes to be inversely proportional to the prices.

This means that we take the full £120 at 2.0 on Newcastle, only £80 at 3.0 on Chelsea, and £40 at 6.0 on The Draw, which is the first virtual bet

We now have a back bet on Newcastle for £75 at 2.5 and a back bet on Chelsea for £70 at 3.0 (again shown in pink on the available to lay side of the market).

These two bets can be matched against a back bet on The Draw at a price of 3.75, since 2.5, 3.0, and 3.75 also form a 100% book.

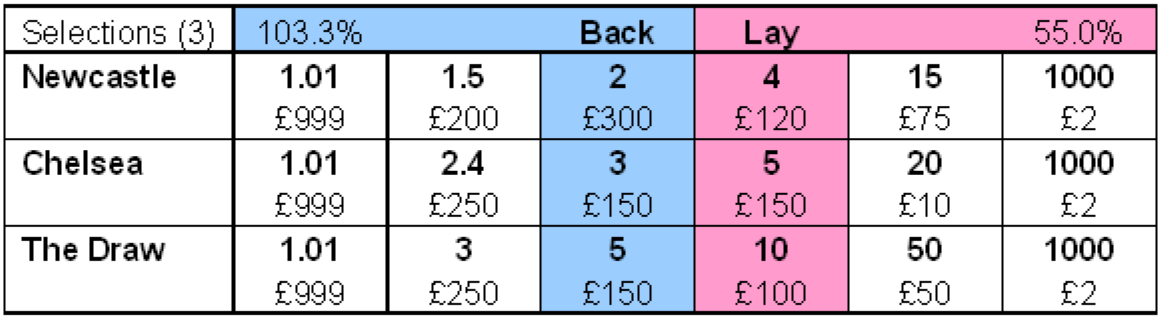
Balancing the stakes means that we need to take the full £75 at 2.5 on Newcastle, only £62.50 at 3.0 on Chelsea, and £50 at 3.75 on The Draw, which is the second virtual bet. Since 3.75 is less than 5.0, the £150 at 5.0 would be matched first followed by £50 at 3.75. If we continued this process we would get further matching at 1.50 and 1.05, but for the purposes of displaying the market view we have the best 3 prices for the available to back bets on The Draw, and so we can stop calculating the virtual bets. The virtual bets are just the bets that would have been matched had we received a sufficiently large back bet at 1.01; in this example, £40 at 6.0 and £50 at 3.75. **We take these virtual bets and merge them with the existing bets on the market to generate the following market view (with the virtual bets shown in green)**



Selections (3)	126.7%		Back	Lay		93.3%
Newcastle	1.01 £999	1.5 £200	1.5 £300	2 £120	2.5 £75	1000 £2
Chelsea	1.01 £999	2.4 £250	2.5 £150	3 £150	20 £10	1000 £2
The Draw	3.75 £50	5 £150	6 £40	10 £100	50 £50	1000 £2

The process is repeated to obtain the virtual lay bets (available to back bets) for Newcastle and Chelsea.

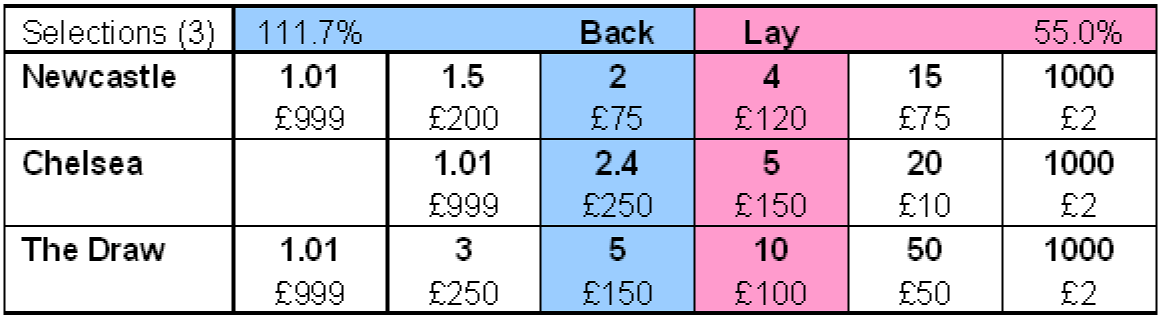
Here we have a slightly different market (as before, chosen to make the numbers nice) and consider what would happen if we placed a very large lay bet at 1000 on The Draw.



Selections (3)	103.3%		Back	Lay		55.0%
Newcastle	1.01 £999	1.5 £200	2 £300	4 £120	15 £75	1000 £2
Chelsea	1.01 £999	2.4 £250	3 £150	5 £150	20 £10	1000 £2
The Draw	1.01 £999	3 £250	5 £150	10 £100	50 £50	1000 £2

Without cross matching, this bet would be matched in three portions; 1) £100 at 10.0, 2) £50 at 50.0, and £2 at 1000 with anything remaining being left unmatched. With cross matching we can do better. We have a lay bet on Newcastle for £300 at 2.0 and a lay bet on Chelsea for £150 at 3.0 (shown in blue on the available to back side of the market). These two bets can be matched against a lay bet on The Draw at a price of 6.0, since 2.0, 3.0, and 6.0 form a 100% book. To ensure that the book is balanced, we choose the stakes to be inversely proportional to the prices. This means that we take only £225 at 2.0 on Newcastle, the full £150 at 3.0 on Chelsea, and £75 at 6.0 on The Draw, which is the first virtual bet.

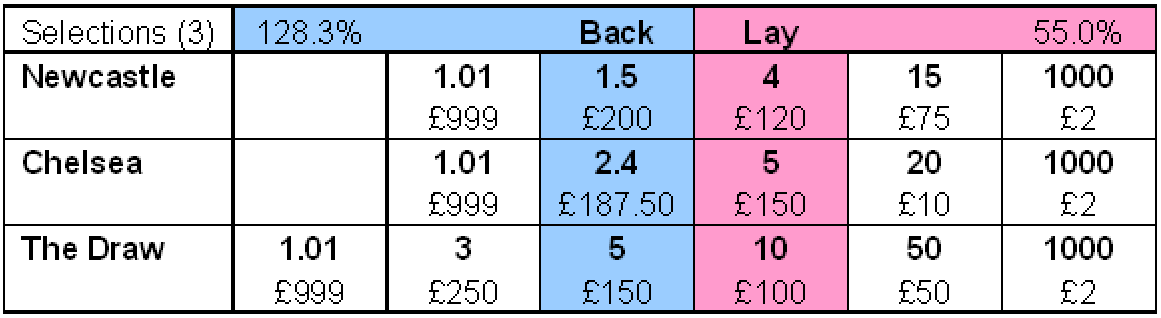
Assuming these bets got matched, the market would look like this:



Selections (3)	111.7%		Back	Lay		55.0%
Newcastle	1.01 £999	1.5 £200	2 £75	4 £120	15 £75	1000 £2
Chelsea		1.01 £999	2.4 £250	5 £150	20 £10	1000 £2
The Draw	1.01 £999	3 £250	5 £150	10 £100	50 £50	1000 £2

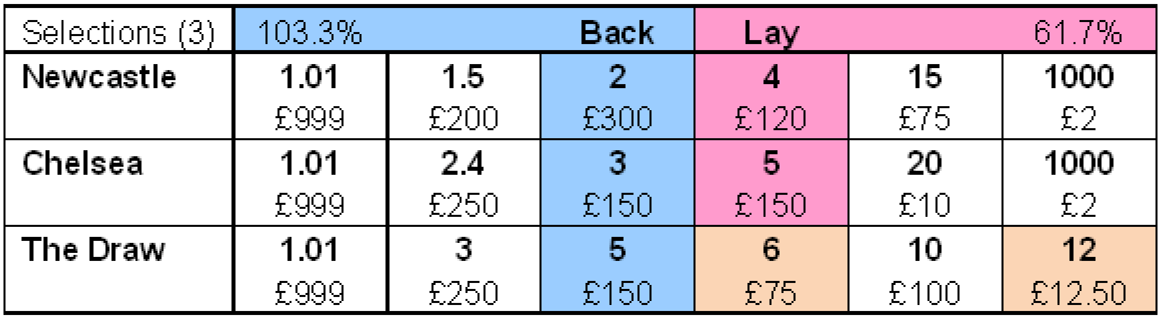
We now have a lay bet on Newcastle for £75 at 2.0 and a lay bet on Chelsea for £250 at 2.4 (again shown in blue on the available to back side of the market). These two bets can be matched against a lay bet on The Draw at a price of 12.0, since 2.0, 2.4, and 12.0 also form a 100% book. Balancing the stakes means that we need to take the full £75 at 2.0 on Newcastle, only £62.50 at 2.4 on Chelsea, and £12.50 at 12.0 on The Draw, which is the second virtual bet.

This leaves the following market:



Selections (3)	128.3%		Back	Lay		55.0%
Newcastle		1.01 £999	1.5 £200	4 £120	15 £75	1000 £2
Chelsea		1.01 £999	2.4 £187.50	5 £150	20 £10	1000 £2
The Draw	1.01 £999	3 £250	5 £150	10 £100	50 £50	1000 £2

This time we can't continue the process since there is no valid price for a virtual bet on The Draw that would result in a 100% book, and so we can stop calculating the virtual bets. Again, the virtual bets are just the bets that would have been matched had we received a sufficiently large lay bet at 1000; in this example, £75 at 6.0 and £12.50 at 12.0. **We take these virtual bets and merge them with the existing bets on the market to generate the following market view (with the virtual bets shown in orange):**



Selections (3)	103.3%		Back	Lay		61.7%
Newcastle	1.01 £999	1.5 £200	2 £300	4 £120	15 £75	1000 £2
Chelsea	1.01 £999	2.4 £250	3 £150	5 £150	20 £10	1000 £2
The Draw	1.01 £999	3 £250	5 £150	6 £75	10 £100	12 £12.50

listRunnerBook

List<MarketBook> listRunnerBook (MarketId marketId, **SelectionId selectionId**, double handicap, PriceProjection priceProjection, OrderProjection orderProjection, MatchProjection matchProjection, boolean includeOverallPosition, boolean partitionMatchedByStrategyRef, Set<String> customerStrategyRefs, StringcurrencyCode,Stringlocale, Date matchedSince, Set<BetId> betIds**) throws APINGException**

Returns a list of dynamic data about **a market** and a **specified runner**. Dynamic data includes prices, the status of the market, the status of selections, the traded volume, and the status of any orders you have placed in the market..

listRunnerBook behaviour
You can only pass in one marketId and one selectionId in that market per request. If the selectionId being passed in is not a valid one / doesn't belong in that market then the call will still work but only the market data is returned

Parameter name	Type	Required	Description
marketId	MarketId	✓	The unique id for the market..
selectionId	SelectionId		The unique id for the selection in the market.
handicap	double		The projection of price data you want to receive in the response.
priceProjection	PriceProjection		The projection of price data you want to receive in the response.
orderProjection	OrderProjection		The orders you want to receive in the response.
matchProjection	MatchProjection		If you ask for orders, specifies the representation of matches.
includeOverallPosition	boolean		If you ask for orders, returns matches for each selection. Defaults to true if unspecified.
partitionMatchedByStrategyRef	boolean		If you ask for orders, returns the breakdown of matches by strategy for each selection. Defaults to false if unspecified.
customerStrategyRefs	Set<String>		If you ask for orders, restricts the results to orders matching any of the specified set of customer defined strategies. Also filters which matches by strategy for selections are returned, if partitionMatchedByStrategyRef is true. An empty set will be treated as if the parameter has been omitted (or null passed).
currencyCode	String		A Betfair standard currency code. If not specified, the default currency code is used.
locale	String		The language used for the response. If not specified, the default is returned.

matchedSince	Date		If you ask for orders, restricts the results to orders that have at least one fragment matched since the specified date (all matched fragments of such an order will be returned even if some were matched before the specified date). All EXECUTABLE orders will be returned regardless of matched date.
betIds	Set<BetId>		If you ask for orders, restricts the results to orders with the specified bet IDs.

Return type	Description
List< MarketBook >	output data

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

listMarketCatalogue

Operation

listMarketCatalogue

List< MarketCatalogue > listMarketCatalogue (MarketFilter filter ,Set< MarketProjection >marketProjection, MarketSort sort, **intmaxResults** ,Stringlocale **) throws APINGException**

Returns a list of information about published (ACTIVE/SUSPENDED) markets that does not change (or changes very rarely). You use listMarketCatalogue to retrieve the name of the market, the names of selections and other information about markets. Market Data Request Limits apply to requests made to listMarketCatalogue.

Please note: listMarketCatalogue does not return markets that are CLOSED.

Parameter name	Type	Required	Description
filter	MarketFilter	✓	The filter to select desired markets. All markets that match the criteria in the filter are selected.
marketProjection	Set< MarketProjection >		The type and amount of data returned about the market.
sort	MarketSort		The order of the results. Will default to **RANK** if not passed. **RANK** is an assigned priority that is determined by our Market Operations team in our back-end system. A result's overall rank is derived from the ranking given to the flowing attributes for the result. EventType, Competition, StartTime, MarketType, MarketId. For example, EventType is ranked by the most popular sports types and marketTypes are ranked in the following order: ODDS ASIAN LINE RANGE If all other dimensions of the result are equal, then the results are ranked in MarketId order.
maxResults	int	✓	limit on the total number of results returned, must be greater than 0 and less than or equal to 1000
locale	String		The language used for the response. If not specified, the default is returned.

Return type	Description
List< MarketCatalogue >	output data

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

RUNNER_METADATA Description

The RUNNER_METADATA returned by listMarketCatalogue for **Horse Racing** (when available) is described in the table below.

Parameter	Description	Example
WEIGHT_UNITS	The unit of weight used	pounds
ADJUSTED_RATING	Adjusted ratings are race-specific ratings which reflect weights allocated in the race and, in some circumstances, the age of the horse. Collectively they represent the chance each runner has on form. https://www.timeform.com/Racing/Articles/How_the_ratings_for_a_race_are_calculated	79
DAM_YEAR_BORN	The year the horse’s mother's birth	1997
DAYS_SINCE_LAST_RUN	The number of days since the horse last ran	66
WEARING	Any extra equipment the horse is wearing	tongue strap

DAMSIRE_YEAR_BORN	The year in which the horse's grandfather was born on its mothers side	1988
SIRE_BRED	The country were the horse's father was bred	IRE
TRAINER_NAME	The name of the horse's trainer	Fergal O'Brien
STALL_DRAW	The stall number the horse is starting from	10
SEX_TYPE	The sex of the horse	f
OWNER_NAME	The owner of the horse	Mr M. C. Fahy
SIRE_NAME	The name of the horse's father	Revoque
FORECASTPRICE_NUMERATOR	The forecast price numerator	13
FORECASTPRICE_DENOMINATOR	The forecast price denominator	8
JOCKEY_CLAIM	The reduction in the weight that the horse carries for a particular jockey	5
WEIGHT_VALUE	The weight of the horse	163
DAM_NAME	The name of the horse's mother	Rare Gesture
AGE	The age of the horse	7
COLOUR_TYPE	The colour of the horse	b
DAMSIRE_BRED	The country were the horse's grandfather was born	IRE
DAMSIRE_NAME	The name of the horse's grandfather	Shalford
SIRE_YEAR_BORN	The year the horse's father was born	1994
OFFICIAL_RATING	The horses official rating	97
FORM	The horses recent form	212246
BRED	The country in which the horse was born	IRE
runnerId	The runnerId for the horse	62434983
JOCKEY_NAME	The name of the jockey. **Please note**: This field will contain '**Reserve**' in the event that the horse has been entered into the market as a reserve runner. Any reserve runners will be withdrawn from the market once it has been confirmed that they will not run.	Paddy Brennar
DAM_BRED	The country where the horse's mother was born	IRE
COLOURS_DESCRIPTION	The textual description of the jockey silk	Royal blue and white sleeves a
COLOURS_FILENAME	A relative URL to an image file corresponding to the jockey silk. You must add the value of this field to the base URL: http://content-cache.betfair.com/feeds_images/Horses/SilkColours/	c20140225lei/0
CLOTH_NUMBER	The number on the saddle-cloth	5
CLOTH_NUMBER ALPHA	The number on the saddle-cloth. For US Racing were the runner is paired, this field will display the cloth number of the paired runner e.g. "1A"	

listMarketProfitAndLoss

listMarketProfitAndLoss

List<MarketProfitAndLoss> listMarketProfitAndLoss (Set<MarketId> marketIds, boolean includeSettledBets, boolean includeBspBets, boolean netOfCommission **) throws APINGException**

Retrieve profit and loss for a given list of OPEN markets. The values are calculated using matched bets and optionally settled bets. **Only odds (MarketBettingType = ODDS) markets are implemented, markets of other types are silently ignored.**

To retrieve your profit and loss for CLOSED markets, please use the listClearedOrders request.

Please note: Market Data Request Limits apply to requests made to listMarketProfitAndLoss

Parameter name	Type	Required	Description
marketIds	Set<MarketId>	✓	List of markets to calculate profit and loss
includeSettledBets	boolean		Option to include settled bets (partially settled markets only). Defaults to false if not specified.
includeBspBets	boolean		Option to include BSP bets. Defaults to false if not specified.
netOfCommission	boolean		Option to return profit and loss net of users current commission rate for this market including any special tariffs. Defaults to false if not specified.

Return type	Description
List<MarketProfitAndLoss>	

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

listMarketTypes

Operation

listMarketTypes

List< MarketTypeResult > listMarketTypes (MarketFilter filter ,Stringlocale) throws APINGException

Returns a list of market types (i.e. MATCH_ODDS, NEXT_GOAL) associated with the markets selected by the MarketFilter. The market types are always the same, regardless of locale.

Parameter name	Type	Required	Description
filter	MarketFilter	✓	The filter to select desired markets. All markets that match the criteria in the filter are selected.
locale	String		The language used for the response. If not specified, the default is returned.

Return type	Description
List< MarketTypeResult >	output data

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

listTimeRanges

Operation

listTimeRanges

List< TimeRangeResult > listTimeRanges (MarketFilter filter , TimeGranularity granularity) throws APINGException

Returns a list of time ranges in the granularity specified in the request (i.e. 3PM to 4PM, Aug 14th to Aug 15th) associated with the markets selected by the MarketFilter.

Parameter name	Type	Required	Description
filter	MarketFilter	✓	The filter to select desired markets. All markets that match the criteria in the filter are selected.
granularity	TimeGranularity	✓	The granularity of time periods that correspond to markets selected by the market filter.

Return type	Description
List< TimeRangeResult >	output data

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

listVenues

Operation

listVenues

List< VenueResult > listVenues (MarketFilter filter ,Stringlocale **) throws APINGException**

Returns a list of Venues (i.e. Cheltenham, Ascot) associated with the markets selected by the MarketFilter. Currently, only Horse Racing markets are associated with a Venue.

Parameter name	Type	Required	Description
filter	MarketFilter	✓	The filter to select desired markets. All markets that match the criteria in the filter are selected.
locale	String		The language used for the response. If not specified, the default is returned.

Return type	Description
List< VenueResult >	output data

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

placeOrders

Operation

placeOrders

PlaceExecutionReport placeOrders (StringmarketId , List< PlaceInstruction >instructions, String customerRef, MarketVersion marketVersion, String customerStrategyRef, boolean async) throws APINGException

Place new orders into market. This operation is atomic in that all orders will be placed or none will be placed. Please note that additional bet sizing rules apply to bets placed into the Italian Exchange.

Parameter name	Type	Required	Description
marketId	String	✓	The market id these orders are to be placed on
instructions	List< PlaceInstruction >	✓	The number of place instructions. The limit of place instructions per request is 200 for the UK/AUS Exchange and 50 for the Italian Exchange.
customerRef	String		Optional parameter allowing the client to pass a unique string (up to 32 chars) that is used to de-dupe mistaken re-submissions. CustomerRef can contain: upper/lower chars, digits, chars : - . _ + * : ; ~ only. **Please note:** There is a time window associated with the de-duplication of duplicate submissions which is 60 seconds.
marketVersion	MarketVersion		Optional parameter allowing the client to specify which version of the market the orders should be placed on. If the current market version is higher than that sent on an order, the bet will be lapsed.
customerStrategyRef	String		An optional reference customers can use to specify which strategy has sent the order. The reference will be returned on order change messages through the stream API. The string is limited to 15 characters. If an empty string is provided it will be treated as null.
async	boolean		An optional flag (not setting equates to false) which specifies if the orders should be placed asynchronously. Orders can be tracked via the Exchange Stream API or or the API-NG by providing a customerOrderRef for each place order. An order's status will be PENDING and no bet ID will be returned. This functionality is available for all bet types - including Market on Close and Limit on Close

Return type	Description
PlaceExecutionReport	

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

Placing a Bet

To place a bet you require the marketId and selectionId parameters from the listMarketCatalogue API call. The below parameters will place a normal Exchange bet at odds of 3.0 for a stake of £2.0.

If the bet is placed successfully, a betId is returned in the placeOrders response

placeOrders Request

```
[
    {
        "jsonrpc": "2.0",
        "method": "SportsAPING/v1.0/placeOrders",
        "params": {
            "marketId": "1.109850906",
            "instructions": [
                {
                    "selectionId": "237486",
                    "handicap": "0",
                    "side": "LAY",
                    "orderType": "LIMIT",
                    "limitOrder": {
                        "size": "2",
                        "price": "3",
                        "persistenceType": "LAPSE"
                    }
                }
            ]
        },
        "id": 1
    }
]
```

placeOrder Response

```
[
    {
        "jsonrpc": "2.0",
        "result": {
            "marketId": "1.109850906",
            "instructionReports": [
                {
                    "instruction": {
                        "selectionId": 237486,
                        "handicap": 0,
                        "limitOrder": {
                            "size": 2,
                            "price": 3,
                            "persistenceType": "LAPSE"
                        },
                        "orderType": "LIMIT",
                        "side": "LAY"
                    },
                    "betId": "31242604945",
                    "placedDate": "2013-10-30T14:22:47.000Z",
                    "averagePriceMatched": 0,
                    "sizeMatched": 0,
                    "status": "SUCCESS"
                }
            ],
            "status": "SUCCESS"
        },
        "id": 1
    }
]
```

Placing a Betfair SP Bet

To place a bet on a selection at Betfair SP, you need to specify the parameters below in the placeOrders request. The below example would place a Betfair SP back bet on the required selection for a stake of £2.00.

placeOrders Request

```
[
    {
        "jsonrpc": "2.0",
        "method": "SportsAPING/v1.0/placeOrders",
        "params": {
            "marketId": "1.111836557",
            "instructions": [
                {
                    "selectionId": "5404312",
                    "handicap": "0",
                    "side": "BACK",
                    "orderType": "MARKET_ON_CLOSE",
                    "marketOnCloseOrder": {
                        "liability": "2"
                    }
                }
            ]
        },
        "id": 1
    }
]
```

placeOrders Response

```
[
    {
        "jsonrpc": "2.0",
        "result": {
            "marketId": "1.111836557",
            "instructionReports": [
                {
                    "instruction": {
                        "selectionId": 5404312,
                        "handicap": 0,
                        "marketOnCloseOrder": {
                            "liability": 2
                        },
                        "orderType": "MARKET_ON_CLOSE",
                        "side": "BACK"
                    },
                    "betId": "31645233727",
                    "placedDate": "2013-11-12T12:07:29.000Z",
                    "status": "SUCCESS"
                }
            ],
            "status": "SUCCESS"
        },
        "id": 1
    }
]
```

Fill or Kill bets

By setting the optional parameter '**TimeInForce**' on a **limitOrder** submission to the value 'FILL_OR_KILL' and optionally passing a **minFillSize** value, the Exchange will only match the order if at least the specified **minFillSize** can be matched (if passed) or the whole order matched (if not). Any order which cannot be so matched, and any remaining unmatched part of the order (if minFillSize is specified) will be immediately cancelled.

Please note: the matching algorithm for Fill or Kill orders behaves slightly differently to that for standard limit orders. Whereas the price on a limit order represents the lowest price at which any fragment should be matched, the price on a Fill or Kill order represents the lower limit of the Volume Weighted Average Price ("VWAP") for the entire volume matched. So, for instance, a Fill or Kill order with price = 5.4 and size = 10 might be matched as £2 @ 5.5, £6 @ 5.4 and £2 @ 5.3.

Examples

FILL_OR_KILL order which was lapsed:

Request
[{"jsonrpc": "2.0", "method": "SportsAPING/v1.0/placeOrders", "params": {"marketId":"1.126124422","instructions":[{"selectionId":"10590221","handicap":"0","side":"BACK","orderType":"LIMIT","limitOrder":{"size":"5","price":"21","persistenceType":"LAPSE","timeInForce":"FILL_OR_KILL","minFillSize":"5"}}]}, "id": 1}]

Response
[{"jsonrpc":"2.0","result":{"status":"SUCCESS","marketId":"1.126124422","instructionReports":[{"status":"SUCCESS","instruction":{"selectionId":10590221,"handicap":0.0,"limitOrder":{"size":5.0,"price":21.0,"minFillSize":5.0,"timeInForce":"FILL_OR_KILL"},"orderType":"LIMIT","side":"BACK"},"betId":"72666364933", "placedDate":"2016-08-12T10:45:15.000Z","averagePriceMatched":0.0,"sizeMatched":0.0}]},"id":1}]

FILL_AND_KILL request a minimum fill size of 3.00:

Request

```
[{"jsonrpc": "2.0", "method": "SportsAPING/v1.0/placeOrders", "params": {"marketId":"1.126124422","instructions":[{"selectionId":"10590221","handicap":"0","side":"BACK","orderType":"LIMIT","limitOrder":{"size":"5","price":"21","persistenceType":"LAPSE","timeInForce":"FILL_OR_KILL","minFillSize":"3"}}]}, "id": 1}]
```

Response

```
[{"jsonrpc":"2.0","result":{"status":"SUCCESS","marketId":"1.126124422","instructionReports":[{"status":"SUCCESS","instruction":{"selectionId":10590221,"handicap":0.0,"limitOrder":{"size":5.0,"price":21.0,"minFillSize":3.0,"timeInForce":"FILL_OR_KILL"},"orderType":"LIMIT","side":"BACK"},"betId":"72666433304","placedDate":"2016-08-12T10:47:42.000Z","averagePriceMatched":21.32267441860465,"sizeMatched":3.4400000000000004}]},"id":1}]
```

Market version parameter

We have added an additional optional parameter '**marketVersion**' to the '**placeOrders**' and '**replaceOrders**' operations. The MarketBook data item, which contains the dynamic data on a market, including its prices, has always returned an integer market 'version'. This 'version' is incremented when significant events – runner removal, turn in-play etc. – occur. Now, by passing that version as 'marketVersion' with your orders, you can specify that if the market version has been incremented beyond that value, your orders should lapse and not be submitted for matching.

This functionality should be of use to those who want to bet right up to the actual 'off' of a horse race or sporting event but be confident that you're not inadvertently bet into the first seconds of in-play after the off. Similarly, in managed football markets, you can avoid your bets reaching the Exchange after the market has reformed following a goal being scored etc.

Notes on 'version' behavior

The 'market version' value (on listMarketBook and on ESA) is incremented for any and all changes to the market.

However, to prevent falsely blocking bets we keep track of the last material change (which we define as one performed under suspension**) and will only accept bets placed with that version or later.

	Market Version	Minimum version to not be rejected	Expected behaviour
Market Activated	1234	1234	
Start time updated (market not suspended)	1235	1234	Non-material change, bets placed before change but received after will be processed normally
Runner removed (under suspension)	1236	1236	Material change, bets placed before change but received after will be rejected.
Market turned in-play	1237	1237	As above

** this includes:

- **Runner removal and addition**
- **Turn in-play**
- **Lapsing or voiding bets (eg on goals being scored in managed Football** market)

But not (e.g.) updating Tennis court times or Golf tee times as they become more accurately known on the day.

Example of a request including market version:

```
[{"jsonrpc": "2.0", "method": "SportsAPING/v1.0/placeOrders", "params": {"marketId":"1.126086207","instructions":[{"selectionId":"63908","handicap":"0","side":"BACK","orderType":"LIMIT","limitOrder":{"size":"2","price":"30"}}],"marketVersion":{"version":"123456789"}}, "id": 1}]
```

Bet to Payout or Profit/Liability

Place a bet specifying your target payout, profit or liability, instead of the backers stake ('size').

Currently, best execution, which guarantees that you'll receive the best possible price, means that you receive a greater potential payout to the same stakes (or risk a smaller potential payout to the same backer's stakes, for layers).

If you wish to benefit by receiving the same potential payout as you originally requested, but to smaller stakes, you can now specify on a LimitOrder (placeOrders) an optional ‘betTargetType’ of ‘PAYOUT’ or ‘BACKERS_PROFIT’ (the latter being identical to layers’ liability) and a ‘betTargetSize’ representing the value of that payout or profit, together with the usual ‘price’ parameter to represent their limit price. Your bet will then be matched to achieve that payout or profit at the specified price or better.

Should all or any of the order be unmatched after first reaching the Exchange, the unmatched portion will be expressed in standard price and backers’ stake terms (by dividing the remaining unmatched payout by the price, or unmatched profit by the price – 1, and placed on the unmatched queue), after this point the bet behaves like any other.

Examples

Placing a back bet targeting a £2 profit

Request

```
[{"jsonrpc": "2.0", "method": "SportsAPING/v1.0/placeOrders", "params": {"marketId":"1.126124417","instructions":[{"selectionId":"11166583","handicap":"0","side":"BACK","orderType":"LIMIT","limitOrder":{"price":"2","betTargetType":"BACKERS_PROFIT","betTargetSize":"2"}}]}, "id": 1}]
```

Response

```
 [{"jsonrpc":"2.0","result":{"status":"SUCCESS","marketId":"1.126124417","instructionReports":[{"status":"SUCCESS","instruction":{"selectionId":11166583,"handicap":0.0,"limitOrder":{"price":2.0,"betTargetSize":2.0,"persistenceType":"LAPSE","betTargetType":"BACKERS_PROFIT"},"orderType":"LIMIT","side":"BACK"},"betId":"72671225671","placedDate":"2016-08-12T13:00:23.000Z","averagePriceMatched":6.3999999999999995,"sizeMatched":0.37}]},"id":1}]
```

Placing a lay bet targeting a £10 Payout

Request

```
[{"jsonrpc": "2.0", "method": "SportsAPING/v1.0/placeOrders", "params": {"marketId":"1.126122473","instructions":[{"selectionId":"11576316","handicap":"0","side":"LAY","orderType":"LIMIT","limitOrder":{"price":"10","betTargetType":"PAYOUT","betTargetSize":"10"}}]}, "id": 1}]
```

Response

```
[{"jsonrpc":"2.0","result":{"status":"SUCCESS","marketId":"1.126122473","instructionReports":[{"status":"SUCCESS","instruction":{"selectionId":11576316,"handicap":0.0,"limitOrder":{"price":10.0,"betTargetSize":10.0,"persistenceType":"LAPSE","betTargetType":"PAYOUT"},"orderType":"LIMIT","side":"LAY"},"betId":"72671531256","placedDate":"2016-08-12T13:05:45.000Z","averagePriceMatched":4.2,"sizeMatched":2.38}]},"id":1}]
```

Ability to place lower minimum stakes at larger prices

In order to allow customers to bet to smaller stakes on longer-priced selections, an extra property has been added to our Currency Parameters – “Min Bet Payout”.

As currently bets where the backer’s stake is at and above the ‘Min Bet Size’ for the currency concerned (£2 for GBP) are valid.   In addition, bets below this value are valid if the payout of the bet would be equal to or greater than the value of ‘**Min Bet Payout’** - £10 for GBP. For example, a bet of £1 @ 10, or 10p @ 100 or 1p @ 1000 are all valid as they all target a payout of £10 or more.

Please note: This function is only enabled for UK & International customers and not .it, .es, .dk jurisdictions.

Each Way Betting

Each Way betting is available via the API. Each Way markets can be identified as marketType EACH_WAY using listMarketCatalogue. The divisor that applies to the EACH_WAY market is returned by listMarketCatalogue via the MARKET_DESCRIPTION MarketProjection.

Please see table a table that indicates how the “Each-Way divisor" is determined for specific race types:

Race Type	Number of Runners(1)	Number of Places	Fraction of Win Odds “Each-Way divisor"
Handicap	16 or more	4	1/4
Handicap	12 to 15	3	1/4
Handicap	8 to 11	3	1/5

Handicap	5 to 7	2	1/4
Non-Handicap	8 or more	3	1/5
Non-Handicap	5 to 7	2	1/4

(1) Number of Runners at Market creation time; place terms are fixed for the life of the market (like Betfair Place market) not dependent on number of runners at the off (like Fixed Odds EW markets)

We will not offer EW markets if the number of runners at market creation time is 4 or fewer

Betfair Price Increments

Below is a list of price increments per price 'group'. Placing a bet outside of these increments will result in an **INVALID_ODDS** error

Odds Markets

Price	Increment
1.01 2	0.01
2 3	0.02
3 4	0.05
4 6	0.1
6 10	0.2
10 20	0.5
20 30	1
30 50	2
50 100	5
100 1000	10

Asian Handicap & Total Goal Markets

Price	Increment
1.01 1000	0.01

Currency Parameters

Guide to available currencies and minimum bet sizes.

Currency name	Symbol	Min Bet Size	Min Deposit Size	Minimum BSP Liability
UK Sterling	£	2	10	10
Euro	€	2	15	20
US Dollar	US$	4	15	20
Hong Kong Dollars	HK$	25	150	125
Australian Dollar	AUD	5	30	30
Canadian Dollar	CAD	6	25	30
Danish Kroner	DKK	30	150	150
Norwegian Kronor	NOK	30	150	150
Swedish Krona	SEK	30	150	150
Singapore Dollar	SGD	6	30	30

cancelOrders

Operation

cancelOrders

CancelExecutionReport cancelOrders (StringmarketId,List< CancelInstruction >instructions,StringcustomerRef **) throws APINGException**

Cancel all bets OR cancel all bets on a market OR fully or partially cancel particular orders on a market. Only LIMIT orders can be cancelled or partially cancelled once placed.

Parameter name	Type	Required	Description
marketId	String		If marketId **and** betId aren't supplied **all bets** are cancelled
instructions	List< CancelInstruction >		All instructions need to be on the same market. If not supplied all bets on the market (if market id is passed) are fully cancelled. The limit of cancel instructions per request is 60
customerRef	String		Optional parameter allowing the client to pass a unique string (up to 32 chars) that is used to de-dupe mistaken re-submissions.

Return type	Description
CancelExecutionReport	

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

replaceOrders

Operation

replaceOrders

ReplaceExecutionReport replaceOrders (StringmarketId , List< ReplaceInstruction >instructions , StringcustomerRef, MarketVersion marketVersion, boolean async **) throws APINGException**

This operation is logically a bulk cancel followed by a bulk place. The cancel is completed first then the new orders are placed. The new orders will be placed atomically in that they will all be placed or none will be placed. In the case where the new orders cannot be placed the cancellations will not be rolled back. See ReplaceInstruction.

Parameter name	Type	Required	Description
marketId	String	✓	The market id these orders are to be placed on
instructions	List< ReplaceInstruction >	✓	The number of replace instructions. The limit of replace instructions per request is 60.
customerRef	String		Optional parameter allowing the client to pass a unique string (up to 32 chars) that is used to de-dupe mistaken re-submissions.
marketVersion	MarketVersion		Optional parameter allowing the client to specify which version of the market the orders should be placed on. If the current market version is higher than that sent on an order, the bet will be lapsed.
async	boolean		An optional flag (not setting equates to false) which specifies if the orders should be replaced asynchronously. Orders can be tracked via the Exchange Stream API or the API-NG by providing a customerOrderRef for each replace order. Not available for MOC or LOC bets.

Return type	Description
ReplaceExecutionReport	

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

updateOrders

Operation

updateOrders

UpdateExecutionReport updateOrders (StringmarketId , List< UpdateInstruction >instructions ,StringcustomerRef) throws APINGException

Update non-exposure changing fields

Parameter name	Type	Required	Description
marketId	String	✓	The market id these orders are to be placed on
instructions	List< UpdateInstruction >	✓	The number of update instructions. The limit of update instructions per request is 60
customerRef	String		Optional parameter allowing the client to pass a unique string (up to 32 chars) that is used to de-dupe mistaken re-submissions.

Return type	Description
UpdateExecutionReport	

Throws	Description
APINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

Betfair Starting Price Betting (BSP)

The Betfair Starting Price will be determined by balancing bets from customers who want to back and lay at Starting Price and matching into the Betfair exchange markets to balance out any residual demand.

The Betfair Starting Price will be calculated exactly to ensure the fairest and most transparent odds possible for both backers and layers. The BSP does not need to account for a profit margin but instead is calculated at the start of an event by looking at the relationship between the amounts of money requested at SP by opposing betting parties. To give an even more accurate price, we will use money where possible that is trading on the exchange at the start of the event. This gives a true reflection of public opinion on a selection.

How is the BSP calculated?

The Near Price is based on money currently on the site at SP as well as unmatched money on the same selection in the exchange. To understand this properly, you first need to understand the calculation of the Far Price, which only takes into account the SP bets that have been made. The Far Price is not as complicated but not as accurate and only accounts for money on the site at SP.

Excluding money requested at a fixed price on the exchange, if there are £1000 worth of backers stakes on a selection at SP and £6000 worth of layers liability, we can return an SP at the start of the event of 6/1 (7.0).

If however there were £6000 worth of backers stakes on the selection and £1000 worth of layers liability, we would return an SP of 1/6 (1.17). These are calculations of the Far Price.

The calculation of the final starting price occurs when the market is turned in-play. This is when the market is reconciled.

Further information and detailed working demonstrating how the Betfair SP price is calculated can be found via https://promo.betfair.com/betfairsp/FAQs_theBasics.html

Betting Exceptions

Exceptions

APINGException

This exception is thrown when an operation fails

Error code	Description
TOO_MUCH_DATA	The operation requested too much data, exceeding the Market Data Request Limits.
INVALID_INPUT_DATA	The data input is invalid. A specific description is returned via errorDetails as shown below.
INVALID_SESSION_INFORMATION	The session token hasn't been provided, is invalid or has expired.
NO_APP_KEY	An application key header ('X-Application') has not been provided in the request
NO_SESSION	A session token header ('X-Authentication') has not been provided in the request
UNEXPECTED_ERROR	An unexpected internal error occurred that prevented successful request processing.
INVALID_APP_KEY	The application key passed is invalid or is not present
TOO_MANY_REQUESTS	There are too many pending requests e.g. a listMarketBook with Order/Match projections is limited to 3 concurrent requests. The error also applies to listCurrentOrders, listMarketProfitAndLoss and listClearedOrders if you have 3 or more requests currently in execution
SERVICE_BUSY	The service is currently too busy to service this request.
TIMEOUT_ERROR	The Internal call to downstream service timed out. **Please note:** If a TIMEOUT_ERROR error occurs on a **placeOrders/replaceOrders** request, you should check **listCurrentOrders** to verify the status of your bets before placing further orders. Please allow up to 2 minutes for timed out order to appear.
REQUEST_SIZE_EXCEEDS_LIMIT	The request exceeds the request size limit. Requests are limited to a total of 250 betId's/marketId's (or a combination of both).
ACCESS_DENIED	The calling client is not permitted to perform the specific action e.g. the using a Delayed App Key when placing bets or attempting to place a bet from a restricted jurisdiction.

Other parameters	Type	Required	Description	Values
errorDetails	String		the stack trace of the error	"market id passed is invalid" "locale must use valid iso-639 locale names" "currency must use valid iso2 currency code name" "country code must use valid iso2 country code name" "text query has invalid content" "language must use valid iso language name"
requestUUID	String			

Generic JSON-RPC Exceptions

Error Code	Description
-32700	Invalid JSON was received by the server. An error occurred on the server while parsing the JSON text.
-32601	Method not found
-32602	Problem parsing the parameters, or a mandatory parameter was not found
-32603	Internal JSON-RPC error

Betting Enums

Enums

MarketProjection

Value	Description
COMPETITION	If not selected then the competition will not be returned with marketCatalogue
EVENT	If not selected then the event will not be returned with marketCatalogue
EVENT_TYPE	If not selected then the eventType will not be returned with marketCatalogue
MARKET_START_TIME	If not selected then the start time will not be returned with marketCatalogue
MARKET_DESCRIPTION	If not selected then the description will not be returned with marketCatalogue
RUNNER_DESCRIPTION	If not selected then the runners will not be returned with marketCatalogue
RUNNER_METADATA	If not selected then the runner metadata will not be returned with marketCatalogue. If selected then RUNNER_DESCRIPTION will also be returned regardless of whether it is included as a market projection.

PriceData

Value	Description
SP_AVAILABLE	Amount available for the BSP auction.
SP_TRADED	Amount traded in the BSP auction.
EX_BEST_OFFERS	Only the best prices available for each runner, to requested price depth.
EX_ALL_OFFERS	EX_ALL_OFFERS trumps EX_BEST_OFFERS if both settings are present
EX_TRADED	Amount traded on the exchange.

MatchProjection

Value	Description
NO_ROLLUP	No rollup, return raw fragments
ROLLED_UP_BY_PRICE	Rollup matched amounts by distinct matched prices per side.
ROLLED_UP_BY_AVG_PRICE	Rollup matched amounts by average matched price per side

OrderProjection

Value	Description
ALL	EXECUTABLE and EXECUTION_COMPLETE orders
EXECUTABLE	An order that has a remaining unmatched portion
EXECUTION_COMPLETE	An order that does not have any remaining unmatched portion

MarketStatus

Value	Description
INACTIVE	The market has been created but isn't yet available.
OPEN	The market is open for betting.
SUSPENDED	The market is suspended and not available for betting.
CLOSED	The market has been settled and is no longer available for betting.

RunnerStatus

Value	Description
ACTIVE	ACTIVE
WINNER	WINNER
LOSER	LOSER
PLACED	The runner was placed, applies to EACH_WAY marketTypes only.
REMOVED_VACANT	REMOVED_VACANT applies to Greyhounds. Greyhound markets always return a fixed number of runners (traps). If a dog has been removed, the trap is shown as vacant.
REMOVED	REMOVED
HIDDEN	The selection is hidden from the market. This occurs in Horse Racing markets were runners is hidden when it is doesn’t hold an official entry following an entry stage. This could be because the horse was never entered or because they have been scratched from a race at a declaration stage. All matched customer bet prices are set to 1.0 even if there are later supplementary stages. Should it appear likely that a specific runner may actually be supplemented into the race this runner will be reinstated with all matched customer bets set back to the original prices.

TimeGranularity

Value	Description
DAYS	
HOURS	
MINUTES	

Side

Value	Description
BACK	To back a team, horse or outcome is to bet on the selection to win. For LINE markets a Back bet refers to a SELL line. A SELL line will win if the outcome is LESS THAN the taken line (price)
LAY	To lay a team, horse, or outcome is to bet on the selection to lose. For LINE markets a Lay bet refers to a BUY line. A BUY line will win if the outcome is MORE THAN the taken line (price)

OrderStatus

Value	Description
PENDING	An asynchronous order is yet to be processed. Once the bet has been processed by the exchange (including waiting for any in-play delay), the result will be reported and available on the Exchange Stream API and API NG. Not a valid search criteria on MarketFilter
EXECUTION_COMPLETE	An order that does not have any remaining unmatched portion.
EXECUTABLE	An order that has a remaining unmatched portion.
EXPIRED	The order is no longer available for execution due to its time in force constraint. In the case of FILL_OR_KILL orders, this means the order has been killed because it could not be filled to your specifications. Not a valid search criteria on MarketFilter

OrderBy

Value	Description
BY_BET	@Deprecated Use BY_PLACE_TIME instead. Order by placed time, then bet id.
BY_MARKET	Order by market id, then placed time, then bet id.
BY_MATCH_TIME	Order by time of last matched fragment (if any), then placed time, then bet id. Filters out orders which have no matched date. The dateRange filter (if specified) is applied to the matched date.
BY_PLACE_TIME	Order by placed time, then bet id. This is an alias of to be deprecated BY_BET. The dateRange filter (if specified) is applied to the placed date.
BY_SETTLED_TIME	Order by time of last settled fragment (if any due to partial market settlement), then by last match time, then placed time, then bet id. Filters out orders which have not been settled. The dateRange filter (if specified) is applied to the settled date.
BY_VOID_TIME	Order by time of last voided fragment (if any), then by last match time, then placed time, then bet id. Filters out orders which have not been voided. The dateRange filter (if specified) is applied to the voided date.

SortDir

Value	**Description**
EARLIEST_TO_LATEST	Order from earliest value to latest e.g. lowest betId is first in the results.
LATEST_TO_EARLIEST	Order from the latest value to the earliest e.g. highest betId is first in the results.

OrderType

Value	**Description**
LIMIT	A normal exchange limit order for immediate execution
LIMIT_ON_CLOSE	Limit order for the auction (SP)
MARKET_ON_CLOSE	Market order for the auction (SP)

MarketSort

Value	Description
MINIMUM_TRADED	Minimum traded volume
MAXIMUM_TRADED	Maximum traded volume
MINIMUM_AVAILABLE	Minimum available to match
MAXIMUM_AVAILABLE	Maximum available to match
FIRST_TO_START	The closest markets based on their expected start time
LAST_TO_START	The most distant markets based on their expected start time

MarketBettingType

Value	Description
ODDS	Odds Market - Any market that doesn't fit any any of the below categories.
LINE	Line Market - LINE markets operate at even-money odds of 2.0. However, price for these markets refers to the line positions available as defined by the markets min-max range and interval steps. Customers either Buy a line (LAY bet, winning if outcome is greater than the taken line (price)) or Sell a line (BACK bet, winning if outcome is less than the taken line (price)). If settled outcome equals the taken line, stake is returned.
RANGE	Range Market - **Now Deprecated**
ASIAN_HANDICAP_DOUBLE_LINE	Asian Handicap Market - A traditional Asian handicap market. Can be identified by marketType ASIAN_HANDICAP
ASIAN_HANDICAP_SINGLE_LINE	Asian Single Line Market - A market in which there can be 0 or multiple winners. e,.g marketType TOTAL_GOALS
FIXED_ODDS	Sportsbook Odds Market. This type is deprecated and will be removed in future releases, when Sportsbook markets will be represented as ODDS market but with a different product type.

ExecutionReportStatus

Value	Description
SUCCESS	Order processed successfully
FAILURE	Order failed.
PROCESSED_WITH_ERRORS	The order itself has been accepted, but at least one (possibly all) actions have generated errors. This error only occurs for **replaceOrders**, **cancelOrders** and **updateOrders** operations. The **placeOrders** operation will not return PROCESSED_WITH_ERRORS status as it is an atomic operation.
TIMEOUT	Order timed out.

ExecutionReportErrorCode

Value	Description
ERROR_IN_MATCHER	The matcher is not healthy
PROCESSED_WITH_ERRORS	The order itself has been accepted, but at least one (possibly all) actions have generated errors
BET_ACTION_ERROR	There is an error with an action that has caused the entire order to be rejected. Check the instructionReports errorCode for the reason for the rejection of the order.
INVALID_ACCOUNT_STATE	Order rejected due to the account's status (suspended, inactive, dup cards)
INVALID_WALLET_STATUS	Order rejected due to the account's wallet's status
INSUFFICIENT_FUNDS	Account has exceeded its exposure limit or available to bet limit
LOSS_LIMIT_EXCEEDED	The account has exceed the self imposed loss limit
MARKET_SUSPENDED	Market is suspended
MARKET_NOT_OPEN_FOR_BETTING	Market is not open for betting. It is either not yet active, suspended or closed awaiting settlement.
DUPLICATE_TRANSACTION	Duplicate customer reference data submitted - **Please note**: There is a time window associated with the de-duplication of duplicate submissions which is 60 second
INVALID_ORDER	Order cannot be accepted by the matcher due to the combination of actions. For example, bets being edited are not on the same market, or order includes both edits and placement

INVALID_MARKET_ID	Market doesn't exist
PERMISSION_DENIED	Business rules do not allow order to be placed. You are either attempting to place the order using a Delayed Application Key or from a restricted jurisdiction (i.e. USA)
DUPLICATE_BETIDS	duplicate bet ids found
NO_ACTION_REQUIRED	Order hasn't been passed to matcher as system detected there will be no state change
SERVICE_UNAVAILABLE	The requested service is unavailable
REJECTED_BY_REGULATOR	The regulator rejected the order. On the **Italian Exchange** t his error will occur if more than 50 bets are sent in a single placeOrders request.
NO_CHASING	A specific error code that relates to Spanish Exchange markets only which indicates that the bet placed contravenes the Spanish regulatory rules relating to loss chasing.
REGULATOR_IS_NOT_AVAILABLE	The underlying regulator service is not available.
TOO_MANY_INSTRUCTIONS	The amount of orders exceeded the maximum amount allowed to be executed
INVALID_MARKET_VERSION	The supplied market version is invalid. Max length allowed for market version is 12.

PersistenceType

Value	**Description**
LAPSE	Lapse the order when the market is turned in-play
PERSIST	Persist the order to in-play. The bet will be place automatically into the in-play market at the start of the event.
MARKET_ON_CLOSE	Put the order into the auction (SP) at turn-in-play

InstructionReportStatus

Value	**Description**
SUCCESS	
FAILURE	
TIMEOUT	

InstructionReportErrorCode

Value	Description
INVALID_BET_SIZE	bet size is invalid for your currency or your regulator
INVALID_RUNNER	Runner does not exist, includes vacant traps in greyhound racing
BET_TAKEN_OR_LAPSED	Bet cannot be cancelled or modified as it has already been taken or has been cancelled/lapsed Includes attempts to cancel/modify market on close BSP bets and cancelling limit on close BSP bets. The error may be returned on placeOrders request if for example a bet is placed at the point when a market admin event takes place (i.e. market is turned in-play)
BET_IN_PROGRESS	No result was received from the matcher in a timeout configured for the system
RUNNER_REMOVED	Runner has been removed from the event
MARKET_NOT_OPEN_FOR_BETTING	Attempt to edit a bet on a market that has closed.
LOSS_LIMIT_EXCEEDED	The action has caused the account to exceed the self imposed loss limit
MARKET_NOT_OPEN_FOR_BSP_BETTING	Market now closed to bsp betting. Turned in-play or has been reconciled
INVALID_PRICE_EDIT	Attempt to edit down the price of a bsp limit on close lay bet, or edit up the price of a limit on close back bet
INVALID_ODDS	Odds not on price ladder - either edit or placement
INSUFFICIENT_FUNDS	Insufficient funds available to cover the bet action. Either the exposure limit or available to bet limit would be exceeded
INVALID_PERSISTENCE_TYPE	Invalid persistence type for this market, e.g. KEEP for a non in-play market.
ERROR_IN_MATCHER	A problem with the matcher prevented this action completing successfully
INVALID_BACK_LAY_COMBINATION	The order contains a back and a lay for the same runner at overlapping prices. This would guarantee a self match. This also applies to BSP limit on close bets
ERROR_IN_ORDER	The action failed because the parent order failed
INVALID_BID_TYPE	Bid type is mandatory
INVALID_BET_ID	Bet for id supplied has not been found

CANCELLED_NOT_PLACED	Bet cancelled but replacement bet was not placed
RELATED_ACTION_FAILED	Action failed due to the failure of a action on which this action is dependent
NO_ACTION_REQUIRED	the action does not result in any state change. eg changing a persistence to it's current value
TIME_IN_FORCE_CONFLICT	You may only specify a time in force on either the place request OR on individual limit order instructions (not both), since the implied behaviors are incompatible.
UNEXPECTED_PERSISTENCE_TYPE	You have specified a persistence type for a FILL_OR_KILL order, which is nonsensical because no umatched portion can remain after the order has been placed.
INVALID_ORDER_TYPE	You have specified a time in force of FILL_OR_KILL, but have included a non-LIMIT order type.
UNEXPECTED_MIN_FILL_SIZE	You have specified a minFillSize on a limit order, where the limit order's time in force is not FILL_OR_KILL. Using minFillSize is not supported where the time in force of the request (as opposed to an order) is FILL_OR_KILL.
INVALID_CUSTOMER_ORDER_REF	The supplied customer order reference is too long.
INVALID_MIN_FILL_SIZE	The minFillSize must be greater than zero and less than or equal to the order's size. The minFillSize cannot be less than the minimum bet size for your currency

RollupModel

Value	Description
STAKE	The volumes will be rolled up to the minimum value which is >= rollupLimit.
PAYOUT	The volumes will be rolled up to the minimum value where the payout(price * volume) is >= rollupLimit. On a LINE market, volumes will be rolled up where payout(2.0 * volume) is >= rollupLimit
MANAGED_LIABILITY	The volumes will be rolled up to the minimum value which is >= rollupLimit, until a lay price threshold. There after, the volumes will be rolled up to the minimum value such that the liability >= a minimum liability. Not supported as yet.
NONE	No rollup will be applied. However the volumes will be filtered by currency specific minimum stake unless overridden specifically for the channel.

GroupBy

Value	Description
EVENT_TYPE	A roll up of settled P&L, commission paid and number of bet orders, on a specified event type
EVENT	A roll up of settled P&L, commission paid and number of bet orders, on a specified event
MARKET	A roll up of settled P&L, commission paid and number of bet orders, on a specified market
SIDE	An averaged roll up of settled P&L, and number of bets, on the specified side of a specified selection within a specified market, that are either settled or voided
BET	The P&L, commission paid, side and regulatory information etc, about each individual bet order

BetStatus

Value	Description
SETTLED	A matched bet that was settled normally
VOIDED	A matched bet that was subsequently voided by Betfair, before, during or after settlement
LAPSED	Unmatched bet that was cancelled by Betfair (for example at turn in play).
CANCELLED	Unmatched bet that was cancelled by an explicit customer action.

marketType - Legacy Data

Value	Description
A	Asian Handicap
L	Line market
O	Odds market
R	Range market.
NOT_APPLICABLE	The market does not have an applicable marketType.

TimeInForce

Value	Description
FILL_OR_KILL	Execute the transaction immediately and completely (filled to size or between minFillSize and size) or not at all (cancelled). For LINE markets Volume Weighted Average Price (VWAP) functionality is disabled

BetTargetType

Value	Description
BACKERS_PROFIT	The payout requested minus the calculated size at which this LimitOrder is to be placed. BetTargetType bets are invalid for LINE markets
PAYOUT	The total payout requested on a LimitOrder

Betting Type Definitions

Type definitions

MarketFilter

Field name	Type	Required	Description
textQuery	String		Restrict markets by any text associated with the market such as the Name, Event, Competition, etc. You can include a wildcard (*) character as long as it is not the first character.
exchangeIds	Set<String>		**DEPRECATED**
eventTypeIds	Set<String>		Restrict markets by event type associated with the market. (i.e., Football, Hockey, etc)
eventIds	Set<String>		Restrict markets by the event id associated with the market.
competitionIds	Set<String>		Restrict markets by the competitions associated with the market.
marketIds	Set<String>		Restrict markets by the market id associated with the market.
venues	Set<String>		Restrict markets by the venue associated with the market. Currently only Horse Racing markets have venues.
bspOnly	boolean		Restrict to bsp markets only, if True or non-bsp markets if False. If not specified then returns both BSP and non-BSP markets
turnInPlayEnabled	boolean		Restrict to markets that will turn in play if True or will not turn in play if false. If not specified, returns both.
inPlayOnly	boolean		Restrict to markets that are currently in play if True or are not currently in play if false. If not specified, returns both.
marketBettingTypes	Set< MarketBettingType >		Restrict to markets that match the betting type of the market (i.e. Odds, Asian Handicap Singles, Asian Handicap Doubles or Line)
marketCountries	Set<String>		Restrict to markets that are in the specified country or countries
marketTypeCodes	Set<String>		Restrict to markets that match the type of the market (i.e., MATCH_ODDS, HALF_TIME_SCORE). You should use this instead of relying on the market name as the market type codes are the same in all locales
marketStartTime	TimeRange		Restrict to markets with a market start time before or after the specified date
withOrders	Set< OrderStatus >		Restrict to markets that I have one or more orders in these status.

MarketCatalogue

Information about a market

Field name	Type	Required	Description
marketId	String	✓	The unique identifier for the market. MarketId's are prefixed with '1.' or '2.' 1. = UK Exchange 2. = AUS Exchange.
marketName	String	✓	The name of the market
marketStartTime	Date		The time this market starts at, only returned when the MARKET_START_TIME enum is passed in the marketProjections
description	MarketDescription		Details about the market
totalMatched	Double		The total amount of money matched on the market
runners	List< RunnerCatalog >		The runners (selections) contained in the market
eventType	EventType		The Event Type the market is contained within
competition	Competition		The competition the market is contained within. Usually only applies to Football competitions
event	Event		The event the market is contained within

MarketBook

The dynamic data in a market

Field name	Type	Required	Description
marketId	String	✓	The unique identifier for the market. MarketId's are prefixed with '1.' or '2.' 1. = UK Exchange 2. = AUS Exchange.
isMarketDataDelayed	boolean	✓	True if the data returned by listMarketBook will be delayed. The data may be delayed because you are not logged in with a funded account or you are using an Application Key that does not allow up to date data.
status	MarketStatus		The status of the market, for example OPEN, SUSPENDED, CLOSED (settled), etc.
betDelay	int		The number of seconds an order is held until it is submitted into the market. Orders are usually delayed when the market is in-play
bspReconciled	boolean		True if the market starting price has been reconciled
complete	boolean		If false, runners may be added to the market
inplay	boolean		True if the market is currently in play
numberOfWinners	int		The number of selections that could be settled as winners
numberOfRunners	int		The number of runners in the market
numberOfActiveRunners	int		The number of runners that are currently active. An active runner is a selection available for betting
lastMatchTime	Date		The most recent time an order was executed
totalMatched	double		The total amount matched
totalAvailable	double		The total amount of orders that remain unmatched
crossMatching	boolean		True if cross matching is enabled for this market.
runnersVoidable	boolean		True if runners in the market can be voided
version	long		The version of the market. The version increments whenever the market status changes, for example, turning in-play, or suspended when a goal is scored.
runners	List< Runner >		Information about the runners (selections) in the market.

RunnerCatalog

Information about the Runners (selections) in a market

Field name	Type	Required	Description
selectionId	long	✓	The unique id for the selection.
runnerName	String	✓	The name of the runner
handicap	double	✓	The handicap
sortPriority	int	✓	The sort priority of this runner
metadata	Map<String,String>		Metadata associated with the runner. For a description of this data for Horse Racing, please see Runner Metadata Description

Runner

The dynamic data about runners in a market

Field name	Type	Required	Description
selectionId	long	✓	The unique id of the runner (selection)
handicap	double	✓	The handicap. Enter the specific handicap value (returned by RUNNER in listMaketBook) if the market is an Asian handicap market.
status	RunnerStatus	✓	The status of the selection (i.e., ACTIVE, REMOVED, WINNER, PLACED, LOSER, HIDDEN) Runner status information is available for 90 days following market settlement.
adjustmentFactor	double	✓	The adjustment factor applied if the selection is removed
lastPriceTraded	double		The price of the most recent bet matched on this selection
totalMatched	double		The total amount matched on this runner
removalDate	Date		If date and time the runner was removed
sp	StartingPrices		The BSP related prices for this runner
ex	ExchangePrices		The Exchange prices available for this runner
orders	List< Order >		List of orders in the market
matches	List< Match >		List of matches (i.e, orders that have been fully or partially executed)
matchesByStrategy	Map<String,Matches>		List of matches for each strategy, ordered by matched data

StartingPrices

Information about the Betfair Starting Price. Only available in BSP markets

Field name	Type	Required	Description
nearPrice	double		What the starting price would be if the market was reconciled now taking into account the SP bets as well as unmatched exchange bets on the same selection in the exchange. This data is cached and update every 60 seconds. **Please note:** Type Double may contain numbers, INF, -INF, and NaN.
farPrice	double		What the starting price would be if the market was reconciled now taking into account only the currently place SP bets. The Far Price is not as complicated but not as accurate and only accounts for money on the exchange at SP. This data is cached and updated every 60 seconds. **Please note:** Type Double may contain numbers, INF, -INF, and NaN.
backStakeTaken	List< PriceSize >		The total amount of back bets matched at the actual Betfair Starting Price.
layLiabilityTaken	List< PriceSize >		The lay amount matched at the actual Betfair Starting Price.
actualSP	double		The final BSP price for this runner. Only available for a BSP market that has been reconciled.

ExchangePrices

Field name	Type	Required	Description
availableToBack	List< PriceSize >		
availableToLay	List< PriceSize >		
tradedVolume	List< PriceSize >		

Event

Event

Field name	Type	Required	Description
id	String		The unique id for the event
name	String		The name of the event
countryCode	String		The ISO-2 code for the event. A list of ISO-2 codes is available via http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2
timezone	String		This is timezone in which the event is taking place.
venue	String		venue
openDate	Date		The scheduled start date and time of the event. This is Europe/London (GMT) by default

EventResult

Event Result

Field name	Type	Required	Description
event	Event		Event
marketCount	int		Count of markets associated with this event

Competition

Competition

Field name	Type	Required	Description
id	String		id
name	String		name

CompetitionResult

Competition Result

Field name	Type	Required	Description
competition	Competition		Competition
marketCount	int		Count of markets associated with this competition
competitionRegion	String		Region in which this competition is happening

EventType

EventType

Field name	Type	Required	Description
id	String		id
name	String		name

EventTypeResult

EventType Result

Field name	Type	Required	Description
eventType	EventType		The ID identifying the Event Type
marketCount	int		Count of markets associated with this eventType

MarketTypeResult

MarketType Result

Field name	Type	Required	Description
marketType	String		Market Type
marketCount	int		Count of markets associated with this marketType

CountryCodeResult

CountryCode Result

Field name	Type	Required	Description
countryCode	String		The ISO-2 code for the event. A list of ISO-2 codes is available via http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2
marketCount	int		Count of markets associated with this Country Code

VenueResult

Venue Result

Field name	Type	Required	Description
venue	String		Venue
marketCount	int		Count of markets associated with this Venue

TimeRange

TimeRange

Field name	Type	Required	Description
from	Date		from
to	Date		to

TimeRangeResult

TimeRange Result

Field name	Type	Required	Description
timeRange	TimeRange		TimeRange
marketCount	int		Count of markets associated with this TimeRange

Order

Field name	Type	Required	Description
betId	String	✓	
orderType	OrderType	✓	BSP Order type.
status	OrderStatus	✓	Either EXECUTABLE (an unmatched amount remains) or EXECUTION_COMPLETE (no unmatched amount remains).
persistenceType	PersistenceType	✓	What to do with the order at turn-in-play
side	Side	✓	Indicates if the bet is a Back or a LAY. For LINE markets customers either Buy a line (LAY bet, winning if outcome is greater than the taken line (price)) or Sell a line (BACK bet, winning if outcome is less than the taken line (price))
price	double	✓	The price of the bet. **Please note**: LINE markets operate at even-money odds of 2.0. However, price for these markets refers to the line positions available as defined by the markets min-max range and interval steps
size	double	✓	The size of the bet.
bspLiability	double	✓	Not to be confused with size. This is the liability of a given BSP bet.
placedDate	Date	✓	The date, to the second, the bet was placed.
avgPriceMatched	double		The average price matched at. Voided match fragments are removed from this average calculation. For MARKET_ON_CLOSE BSP bets this reports the matched SP price following the SP reconciliation process. This value is not meaningful for activity on LINE markets and is not guaranteed to be returned or maintained for these markets.
sizeMatched	double		The current amount of this bet that was matched.
sizeRemaining	double		The current amount of this bet that is unmatched.
sizeLapsed	double		The current amount of this bet that was lapsed.
sizeCancelled	double		The current amount of this bet that was cancelled.
sizeVoided	double		The current amount of this bet that was voided.
customerOrderRef	CustomerOrderRef		The customer order reference sent for this bet
customerStrategyRef	CustomerStrategyRef		The customer strategy reference sent for this bet

Match

An individual bet Match, or rollup by price or avg price. Rollup depends on the requested MatchProjection

Field name	Type	Required	Description
betId	String		Only present if no rollup
matchId	String		Only present if no rollup
side	Side	✓	Indicates if the bet is a Back or a LAY
price	double	✓	Either actual match price or avg match price depending on rollup. This value is not meaningful for activity on LINE markets and is not guaranteed to be returned or maintained for these markets.
size	double	✓	Size matched at in this fragment, or at this price or avg price depending on rollup
matchDate	Date		Only present if no rollup

MarketVersion

Market version

Field name	Type	Required	Description
version	long		A non-monotonically increasing number indicating market changes

MarketDescription

Market definition

Field name	Type	Required	Description
persistenceEnabled	boolean	✓	If 'true' the market supports 'Keep' bets if the market is to be turned in-play
bspMarket	boolean	✓	If 'true' the market supports Betfair SP betting
marketTime	Date	✓	The market start time
suspendTime	Date	✓	The market suspend time
settleTime	Date		settled time
bettingType	MarketBettingType	✓	See MarketBettingType
turnInPlayEnabled	boolean	✓	If 'true' the market is set to turn in-play
marketType	String	✓	Market base type
regulator	String	✓	The market regulator
marketBaseRate	double	✓	The commission rate applicable to the market
discountAllowed	boolean	✓	Indicates whether or not the user's discount rate is taken into account on this market. If ‘false’ all users will be charged the same commission rate, regardless of discount rate.
wallet	String		The wallet to which the market belongs (UK/AUS)
rules	String		The market rules.
rulesHasDate	boolean		
eachWayDivisor	double		The divisor is returned for the marketType EACH_WAY only and refers to the fraction of the win odds at which the place portion of an each way bet is settled
clarifications	String		Any additional information regarding the market

MarketRates

Market Rates

Field name	Type	Required	Description
marketBaseRate	double	✓	marketBaseRate
discountAllowed	boolean	✓	discountAllowed

MarketLicence

Market Licence

Field name	Type	Required	Description
wallet	String	✓	The wallet from which funds will be taken when betting on this market
rules	String		The rules for this market
rulesHasDate	boolean		The market's start date and time are relevant to the rules.
clarifications	String		Clarifications to the rules for the market

MarketLineRangeInfo

Market Line and Range Info

Field name	Type	Required	Description
maxUnitValue	double	✓	maxPrice - Maximum value for the outcome, in market units for this market (eg 100 runs)
minUnitValue	double	✓	minPrice - Minimum value for the outcome, in market units for this market (eg 0 runs)
interval	double	✓	interval - The odds ladder on this market will be between the range of minUnitValue and maxUnitValue, in increments of the inverval value.e.g. If minUnitValue=10 runs, maxUnitValue=20 runs, interval=0.5 runs, then valid odds include 10, 10.5, 11, 11.5 up to 20 runs.
marketUnit	String	✓	unit - The type of unit the lines are incremented in by the interval (e.g: runs, goals or seconds.

PriceSize

Field name	Type	Required	Description
price	double	✓	The price available
size	double	✓	The stake available

ClearedOrderSummary

Summary of a cleared order.

Field name	Type	Required	Description
eventTypeId	EventTypeId		The id of the event type bet on. Available at EVENT_TYPE groupBy level or lower.
eventId	EventId		The id of the event bet on. Available at EVENT groupBy level or lower.
marketId	MarketId		The id of the market bet on. Available at MARKET groupBy level or lower.

selectionId	SelectionId		The id of the selection bet on. Available at RUNNER groupBy level or lower.
handicap	Handicap		The handicap. Enter the specific handicap value (returned by RUNNER in listMaketBook) if the market is an Asian handicap market. Available at MARKET groupBy level or lower.
betId	BetId		The id of the bet. Available at BET groupBy level.
placedDate	Date		The date the bet order was placed by the customer. Only available at BET groupBy level.
persistenceType	PersistenceType		The turn in play persistence state of the order at bet placement time. This field will be empty or omitted on true SP bets. Only available at BET groupBy level.
orderType	OrderType		The type of bet (e.g standard limited-liability Exchange bet (LIMIT), a standard BSP bet (MARKET_ON_CLOSE), or a minimum-accepted-price BSP bet (LIMIT_ON_CLOSE)). If the bet has a OrderType of MARKET_ON_CLOSE and a persistenceType of MARKET_ON_CLOSE then it is a bet which has transitioned from LIMIT to MARKET_ON_CLOSE. Only available at BET groupBy level.
side	Side		Whether the bet was a back or lay bet. Available at SIDE groupBy level or lower.
itemDescription	ItemDescription		A container for all the ancillary data and localised text valid for this Item
betOutcome	String		The settlement outcome of the bet. Tri-state (WIN/LOSE/PLACE) to account for Each Way bets where the place portion of the bet won but the win portion lost. The profit/loss amount in this case could be positive or negative depending on the price matched at. Only available at BET groupBy level.
priceRequested	Price		The average requested price across all settled bet orders under this Item. Available at SIDE groupBy level or lower. For LINE markets this is the line position requested. For LINE markets this is the line position requested.
settledDate	Date		The date and time the bet order was settled by Betfair. Available at SIDE groupBy level or lower.
lastMatchedDate	Date		The date and time the last bet order was matched by Betfair. Available on Settled orders only.
betCount	int		The number of actual bets within this grouping (will be 1 for BET groupBy)
commission	Size		The cumulative amount of commission paid by the customer across all bets under this Item, in the account currency. Available at EXCHANGE, EVENT_TYPE, EVENT and MARKET level groupings only.
priceMatched	Price		The average matched price across all settled bets or bet fragments under this Item. Available at SIDE groupBy level or lower. For LINE markets this is the line position matched at.
priceReduced	boolean		If true, then the matched price was affected by a reduction factor due to of a runner removal from this Horse Racing market.

sizeSettled	Size		The cumulative bet size that was settled as matched or voided under this Item, in the account currency. Available at SIDE groupBy level or lower.
profit	Size		The profit or loss (negative profit) gained on this line, in the account currency
sizeCancelled	Size		The amount of the bet that was available to be matched, before cancellation or lapsing, in the account currency
customerOrderRef	String		The order reference defined by the customer for the bet order
customerStrategyRef	String		The strategy reference defined by the customer for the bet order

ClearedOrderSummaryReport

A container representing search results.

Field name	Type	Required	Description
clearedOrders	List<ClearedOrderSummary>	✓	The list of cleared orders returned by your query. This will be a valid list (i.e. empty or non-empty but never 'null').
moreAvailable	boolean	✓	Indicates whether there are further result items beyond this page. Note that underlying data is highly time-dependent and the subsequent search orders query might return an empty result.

ItemDescription

This object contains some text which may be useful to render a betting history view. It offers no long-term warranty as to the correctness of the text.

Field name	Type	Required	Description
eventTypeDesc	String		The event type name, translated into the requested locale. Available at EVENT_TYPE groupBy or lower.
eventDesc	String		The eventName, or openDate + venue, translated into the requested locale. Available at EVENT groupBy or lower.
marketDesc	String		The market name or racing market type ("Win", "To Be Placed (2 places)", "To Be Placed (5 places)" etc) translated into the requested locale. Available at MARKET groupBy or lower.
marketType	String		The market type e.g. MATCH_ODDS, PLACE, WIN etc.
marketStartTime	Date		The start time of the market (in ISO-8601 format, not translated). Available at MARKET groupBy or lower.
runnerDesc	String		The runner name, maybe including the handicap, translated into the requested locale. Available at BET groupBy.
numberOfWinners	int		The number of winners on a market. Available at BET groupBy.
eachWayDivisor	double		The divisor is returned for the marketType EACH_WAY only and refers to the fraction of the win odds at which the place portion of an each way bet is settled

RunnerId

This object contains the unique identifier for a runner

Field name	Type	Required	Description
marketId	MarketId	✓	The id of the market bet on
selectionId	SelectionId	✓	The id of the selection bet on
handicap	Handicap		The handicap associated with the runner in case of asian handicap markets, otherwise returns '0.0'.

CurrentOrderSummaryReport

A container representing search results.

Field name	**Type**	**Required**	**Description**
currentOrders	List< CurrentOrderSummary >	✓	The list of current orders returned by your query. This will be a valid list (i.e. empty or non-empty but never 'null').
moreAvailable	boolean	✓	Indicates whether there are further result items beyond this page. Note that underlying data is highly time-dependent and the subsequent search orders query might return an empty result.

CurrentOrderSummary

Summary of a current order.

Field name	Type	Required	Description
betId	String	✓	The bet ID of the original place order.
marketId	String	✓	The market id the order is for.
selectionId	long	✓	The selection id the order is for.
handicap	double	✓	The handicap associated with the runner in case of Asian handicap markets, null otherwise.
priceSize	PriceSize	✓	The price and size of the bet.
bspLiability	double	✓	Not to be confused with size. This is the liability of a given BSP bet.
side	Side	✓	BACK/LAY
status	OrderStatus	✓	Either EXECUTABLE (an unmatched amount remains) or EXECUTION_COMPLETE (no unmatched amount remains).
persistenceType	PersistenceType	✓	What to do with the order at turn-in-play.
orderType	OrderType	✓	BSP Order type.
placedDate	Date	✓	The date, to the second, the bet was placed.
matchedDate	Date	✓	The date, to the second, of the last matched bet fragment (where applicable)
averagePriceMatched	double		The average price matched at. Voided match fragments are removed from this average calculation. The price is automatically adjusted in the event of non runners being declared with applicable reduction factors. **Please note:** T his value is not meaningful for activity on LINE markets and is not guaranteed to be returned or maintained for these markets.
sizeMatched	double		The current amount of this bet that was matched.
sizeRemaining	double		The current amount of this bet that is unmatched.
sizeLapsed	double		The current amount of this bet that was lapsed.
sizeCancelled	double		The current amount of this bet that was cancelled.
sizeVoided	double		The current amount of this bet that was voided.
regulatorAuthCode	String		The regulator authorisation code.
regulatorCode	String		The regulator Code.
customerOrderRef	String		The order reference defined by the customer for this bet
customerStrategyRef	String		The strategy reference defined by the customer for this bet

PlaceInstruction

Instruction to place a new order

Field name	Type	Required	Description
orderType	OrderType	✓	
selectionId	long	✓	The selection_id.
handicap	double		The handicap associated with the runner in case of Asian handicap markets (e.g. marketTypes ASIAN_HANDICAP_DOUBLE_LINE, ASIAN_HANDICAP_SINGLE_LINE) null otherwise.
side	Side	✓	Back or Lay
limitOrder	LimitOrder		A simple exchange bet for immediate execution
limitOnCloseOrder	LimitOnCloseOrder		Bets are matched if, and only if, the returned starting price is better than a specified price. In the case of back bets, LOC bets are matched if the calculated starting price is greater than the specified price. In the case of lay bets, LOC bets are matched if the starting price is less than the specified price. If the specified limit is equal to the starting price, then it may be matched, partially matched, or may not be matched at all, depending on how much is needed to balance all bets against each other (MOC, LOC and normal exchange bets)
marketOnCloseOrder	MarketOnCloseOrder		Bets remain unmatched until the market is reconciled. They are matched and settled at a price that is representative of the market at the point the market is turned in-play. The market is reconciled to find a starting price and MOC bets are settled at whatever starting price is returned. MOC bets are always matched and settled, unless a starting price is not available for the selection. Market on Close bets can only be placed before the starting price is determined
customerOrderRef	String		An optional reference customers can set to identify instructions.. No validation will be done on uniqueness and the string is limited to 32 characters. If an empty string is provided it will be treated as null.

PlaceExecutionReport

Field name	Type	Required	Description
customerRef	String		Echo of the customerRef if passed.
status	ExecutionReportStatus	✓	
errorCode	ExecutionReportErrorCode		
marketId	String		Echo of marketId passed
instructionReports	List< PlaceInstructionReport >		

LimitOrder

Place a new LIMIT order (simple exchange bet for immediate execution)

Field name	Type	Required	Description
size	double	✓	The size of the bet. **Please note**: For market type EACH_WAY. The total stake = size x 2
price	double	✓	The limit price. For LINE markets, the price at which the bet is settled and struck will always be 2.0 (Evens). On these bets, the Price field is used to indicate the line value which is being bought or sold
persistenceType	PersistenceType	✓	What to do with the order at turn-in-play
timeInForce	TimeInForce		The type of TimeInForce value to use. This value takes precedence over any PersistenceType value chosen. If this attribute is populated along with the PersistenceType field, then the PersistenceType will be ignored. When using FILL_OR_KILL for a Line market the Volume Weighted Average Price (VWAP) functionality is disabled
minFillSize	Size		An optional field used if the TimeInForce attribute is populated. If specified without TimeInForce then this field is ignored. If no minFillSize is specified, the order is killed unless the entire size can be matched. If minFillSize is specified, the order is killed unless at least the minFillSize can be matched. The minFillSize cannot be greater than the order's size. If specified for a BetTargetType and FILL_OR_KILL order, then this value will be ignored
betTargetType	BetTargetType		An optional field to allow betting to a targeted PAYOUT or BACKERS_PROFIT. It's invalid to specify both a Size and BetTargetType Matching provides best execution at the requested price or better up to the payout or profit. If the bet is not matched completely and immediately, the remaining portion enters the unmatched pool of bets on the exchange BetTargetType bets are invalid for LINE markets
betTargetSize	Size		An optional field which must be specified if BetTargetType is specified for this order The requested outcome size of either the payout or profit. This is named from the backer's perspective. For Lay bets the profit represents the bet's liability

LimitOnCloseOrder

Place a new LIMIT_ON_CLOSE bet

Field name	Type	Required	Description
liability	double	✓	The size of the bet.
price	double	✓	The limit price of the bet if LOC

MarketOnCloseOrder

Place a new MARKET_ON_CLOSE bet

Field name	Type	Required	Description
liability	double	✓	The size of the bet.

PlaceInstructionReport

Response to a PlaceInstruction

Field name	Type	Required	Description
status	InstructionReportStatus	✓	whether the command succeeded or failed
errorCode	InstructionReportErrorCode		cause of failure, or null if command succeeds
orderStatus	OrderStatus		The status of the order, if the instruction succeeded. If the instruction was unsuccessful, no value is provided.
instruction	PlaceInstruction	✓	The instruction that was requested
betId	String		The bet ID of the new bet. Will be null on failure or if order was placed asynchronously.
placedDate	Date		Will be null if order was placed asynchronously
averagePriceMatched	Price		Will be null if order was placed asynchronously. This value is not meaningful for activity on LINE markets and is not guaranteed to be returned or maintained for these markets.
sizeMatched	Size		Will be null if order was placed asynchronously

CancelInstruction

Instruction to fully or partially cancel an order (only applies to LIMIT orders)

Field name	Type	Required	Description
betId	String	✓	The betId
sizeReduction	double		If supplied then this is a partial cancel. Should be set to 'null' if no size reduction is required.

CancelExecutionReport

Field name	Type	Required	Description
customerRef	String		Echo of the customerRef if passed.
status	ExecutionReportStatus	✓	
errorCode	ExecutionReportErrorCode		
marketId	String		Echo of marketId passed
instructionReports	List< CancelInstructionReport >		

ReplaceInstruction

Instruction to replace a LIMIT or LIMIT_ON_CLOSE order at a new price. Original order will be cancelled and a new order placed at the new price for the remaining stake.

Field name	Type	Required	Description
betId	String	✓	Unique identifier for the bet
newPrice	double	✓	The price to replace the bet at

ReplaceExecutionReport

Field name	Type	Required	Description
customerRef	String		Echo of the customerRef if passed.
status	ExecutionReportStatus	✓	
errorCode	ExecutionReportErrorCode		
marketId	String		Echo of marketId passed
instructionReports	List< ReplaceInstructionReport >		

ReplaceInstructionReport

Field name	Type	Required	Description
status	InstructionReportStatus	✓	whether the command succeeded or failed
errorCode	InstructionReportErrorCode		cause of failure, or null if command succeeds
cancelInstructionReport	CancelInstructionReport		Cancelation report for the original order
placeInstructionReport	PlaceInstructionReport		Placement report for the new order

CancelInstructionReport

Field name	Type	Required	Description
status	InstructionReportStatus	✓	whether the command succeeded or failed
errorCode	InstructionReportErrorCode		cause of failure, or null if command succeeds
instruction	CancelInstruction		The instruction that was requested
sizeCancelled	double	✓	
cancelledDate	Date		

UpdateInstruction

Instruction to update LIMIT bet's persistence of an order that do not affect exposure

Field name	Type	Required	Description
betId	String	✓	Unique identifier for the bet
newPersistenceType	PersistenceType	✓	The new persistence type to update this bet to

UpdateExecutionReport

Field name	Type	Required	Description
customerRef	String		Echo of the customerRef if passed.
status	ExecutionReportStatus	✓	
errorCode	ExecutionReportErrorCode		
marketId	String		Echo of marketId passed
instructionReports	List< UpdateInstructionReport >		

UpdateInstructionReport

Field name	Type	Required	Description
status	InstructionReportStatus	✓	whether the command succeeded or failed
errorCode	InstructionReportErrorCode		cause of failure, or null if command succeeds
instruction	UpdateInstruction	✓	The instruction that was requested

PriceProjection

Selection criteria of the returning price data

Field name	Type	Required	Description
priceData	Set< PriceData >		The basic price data you want to receive in the response.
exBestOffersOverrides	ExBestOffersOverrides		Options to alter the default representation of best offer prices Applicable to EX_BEST_OFFERS priceData selection
virtualise	boolean		Indicates if the returned prices should include virtual prices. Applicable to EX_BEST_OFFERS and EX_ALL_OFFERS priceData selections, default value is false.
rolloverStakes	boolean		Indicates if the volume returned at each price point should be the absolute value or a cumulative sum of volumes available at the price and all better prices. If unspecified defaults to false. Applicable to EX_BEST_OFFERS and EX_ALL_OFFERS price projections. Not supported as yet.

ExBestOffersOverrides

Options to alter the default representation of best offer prices

Field name	Type	Required	Description
bestPricesDepth	int		The maximum number of prices to return on each side for each runner. If unspecified defaults to 3. Maximum returned price depth returned is 10.
rollupModel	RollupModel		The model to use when rolling up available sizes. If unspecified defaults to STAKE rollup model with rollupLimit of minimum stake in the specified currency.
rollupLimit	int		The volume limit to use when rolling up returned sizes. The exact definition of the limit depends on the rollupModel. If no limit is provided it will use minimum stake as default the value. Ignored if no rollup model is specified.
rollupLiabilityThreshold	double		Only applicable when rollupModel is MANAGED_LIABILITY. The rollup model switches from being stake based to liability based at the smallest lay price which is >= rollupLiabilityThreshold.service level default (TBD). Not supported as yet.
rollupLiabilityFactor	int		Only applicable when rollupModel is MANAGED_LIABILITY. (rollupLiabilityFactor * rollupLimit) is the minimum liabilty the user is deemed to be comfortable with. After the rollupLiabilityThreshold price subsequent volumes will be rolled up to minimum value such that the liability >= the minimum liability.service level default (5). Not supported as yet.

MarketProfitAndLoss

Profit and loss in a market

Field name	Type	Required	Description
marketId	String		The unique identifier for the market
commissionApplied	double		The commission rate applied to P&L values. Only returned if netOfCommision option is requested
profitAndLosses	List<RunnerProfitAndLoss>		Calculated profit and loss data.

RunnerProfitAndLoss

Profit and loss if selection is wins or loses

Field name	Type	Required	Description
selectionId	SelectionId		The unique identifier for the selection
ifWin	double		Profit or loss for the market if this selection is the winner.
ifLose	double		Profit or loss for the market if this selection is the loser. Only returned for multi-winner odds markets.
ifPlace	double		Profit or loss for the market if this selection is placed. Applies to marketType EACH_WAY only.

Type Aliases

Alias	Type
MarketType	String
Venue	String
MarketId	String
SelectionId	long
Handicap	double
EventId	String
EventTypeId	String
CountryCode	String
ExchangeId	String
CompetitionId	String
Price	double
Size	double
BetId	String
MatchId	String
CustomerOrderRef	String
CustomerStrategyRef	String

Accounts API

Endpoints

Please find below the details for the current Accounts API endpoints.

Global Exchange

Interface	Endpoint	JSON-RPC Prefix	<methodname> Example
JSON-RPC	https://api.betfair.com/exchange/account/json-rpc/v1	<methodname>	AccountAPING/v1.0/getAccountFunds

JSON REST	https://api.betfair.com/exchange/account/rest/v1.0/		getAccountFunds/

Accounts Operations

Required Headers

Please note - although the majority of API-NG calls require both the **X-Authentication** (sessionToken) and **X-Application** (Application Key) in the request header, this isn't applicable for some API Account Operations that are available to Software Vendors Only. The applicable headers for each Vendor API operation are included in the below table

Summary

Type	Operation	Description	Available to Software Vendors Only	X-Authentication	X-Applicaiton
DeveloperApp	createDeveloperAppKeys (String appName)	Create 2 Application Keys for given user; one 'Delayed and the other 'Live'. You must apply to have your 'Live' App Key activated.		**Required**	
List< DeveloperApp >	getDeveloperAppKeys ( )	Get all application keys owned by the given developer/vendor		**Required**	
AccountFundsResponse	getAccountFunds ( )	Get available to bet amount.		**Required**	**Required**
TransferResponse	transferFunds (Wallet from, Wallet to, double amount)			**Required**	**Required**
AccountDetailsResponse	getAccountDetails ( )	Returns the details relating your account, including your discount rate and Betfair point balance.		**Required**	**Required**
String	getVendorClientId ( )	Returns the vendor client id for customer account which is a unique identifier for that customer.		**Required**	**Required**
String	getApplicationSubscriptionToken (intsubscriptionLength)	Used to create new subscription tokens for an application. Returns the newly generated subscription token which can be provided to the end user. **Available to owner managed (Vendor) App Keys Only**	**Y**	**Required**	**Required**
Status	activateApplicationSubscription (StringsubscriptionToken)	Activates the customers subscription token for an application		**Required**	
Status	cancelApplicationSubscription (StringsubscriptionToken)	Cancel the subscription token. The customers subscription will no longer be active once cancelled**. Available to owner managed (Vendor) App Keys Only**	**Y**	**Required**	**Required**
String	updateApplicationSubscription (String vendorClientId, int subscriptionLength)	Update an application subscription with a new expiry date. **Available to owner managed (Vendor) App Keys Only**	**Y**	**Required**	**Required**
List< ApplicationSubscription >	listApplicationSubscriptionTokens (SubscriptionStatus subscriptionStatus)	Returns a list of subscription tokens for an application based on the subscription status passed in the request.	**Y**	**Required**	**Required**
List< AccountSubscription >	listAccountSubscriptionTokens ( )	List of subscription tokens associated with the account. **Available to owner managed (Vendor) App Keys Only**	**Y**	**Required**	**Required**
List<SubscriptionHistory >	getApplicationSubscriptionHistory (String vendorClientId)	Returns a list of subscriptions tokens that have been associated with the customers account. **Available to owner managed (Vendor) App Keys Only**	**Y**	**Required**	**Required in request header OR request body**

AccountStatementReport	getAccountStatement (String locale, int fromRecord, int recordCount, TimeRange itemDateRange, IncludeItem includeItem,Walletwallet)	Get account statement - provides full audit trail of money moving to and from your account.	Not available via the Vendor Web API	**Required**	**Required**
List<CurrencyRate>	listCurrencyRates **(** String fromCurrency **)**	Returns a list of currency rates based on given currency.			
VendorAccessTokenInfo	**token (String client_id, GrantType grant_type**, String code, **String client_secret**, String refresh_token **)**	Generate web vendor session based on a standard session identifiable by auth code, vendor secret and app key	**Y**	**Required**	**Required**
VendorDetails	**getVendorDetails (String vendorId)**	Return details about a vendor from its identifier. Response includes Vendor Name and URL			
Status	**revokeAccessToWebApp (long vendorId)**	Remove the link between an account and a vendor web app. This will remove the refreshToken for this user-vendor pair subscription.			
List<VendorDetails>	**listAuthorizedWebApps ( )**	Retrieve all vendors applications currently subscribed to by the user making the request			
boolean	**isAccountSubscribedToWebApp (String vendorId)**	Return whether an account has authorised a web app.			
List<AffiliateRelation>	**getAffiliateRelation (List<String> vendorClientIds)**	Return relation between a list of users and an affiliate	**Y**	**Required**	**Required**

createDeveloperAppKeys

Operation

createDeveloperAppKeys

DeveloperApp createDeveloperAppKeys (String appName) throws AccountAPINGException

Create 2 Application Keys for given user; one 'Delayed and the other 'Live'. You must apply to have your 'Live' App Key activated.

Parameter name	Type	Required	Description
appName	String	✓	A Display name for the application.

Return type	Description
DeveloperApp	A map of application keys, one marked ACTIVE, and the other DELAYED

Throws	Description
AccountAPINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

getAccountDetails

Operation

getAccountDetails

AccountDetailsResponse getAccountDetails ( ) throws AccountAPINGException

Returns the details relating your account, including your discount rate and Betfair point balance.

Return type	Description
AccountDetailsResponse	Response for retrieving account details.

Throws	Description
AccountAPINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

Please note: The data returned by **getAccountDetails** relies on two underlying services. The **pointsBalance** is returned by a separate service from the other data.

As a consequence of this, in the event of a failure to a single underlying service, either the **pointsBalance** or the remaining data may not be included in the **getAccountDetails** response. If both services fail, the error UNEXPECTED_ERROR will be returned.

getAccountFunds

Operation

getAccountFunds

AccountFundsResponse getAccountFunds ( ) throws AccountAPINGException

Get available to bet amount. The getAccounts service will return the UK wallet balance by default from either the UK or AUS Accounts API endpoint if the wallet parameter is not specified.

Parameter name	Type	Required	Description
wallet	Wallet		Name of the wallet in question. **Please Note:** To return the the Australian Exchange wallet balance you must specify **AUSTRALIAN** as the Wallet parameter.

Return type	Description
AccountFundsResponse	Response for retrieving available to bet.

Throws	Description
AccountAPINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

getDeveloperAppKeys

Operation

getDeveloperAppKeys

List< DeveloperApp > getDeveloperAppKeys ( ) throws AccountAPINGException

Get all application keys owned by the given developer/vendor

Return type	Description
List< DeveloperApp >	A list of application keys owned by the given developer/vendor

Throws	Description
AccountAPINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

getAccountStatement

getAccountStatement

AccountStatementReport getAccountStatement (String locale, int fromRecord, int recordCount, TimeRange itemDateRange, IncludeItem includeItem, Wallet wallet**) throws AccountAPINGException**

Get account statement

Parameter name	Type	Required	Description
locale	String		The language to be used where applicable. If not specified, the customer account default is returned.
fromRecord	int		Specifies the first record that will be returned. Records start at index zero. If not specified then it will default to 0.
recordCount	int		Specifies the maximum number of records to be returned. Note that there is a page size limit of 100.
itemDateRange	TimeRange		Return items with an itemDate within this date range. Both from and to date times are inclusive. If from is not specified then the oldest available items will be in range. If to is not specified then the latest items will be in range. nb. This itemDataRange is currently only applied when includeItem is set to ALL or not specified, else items are NOT bound by itemDate.
includeItem	IncludeItem		Which items to include, if not specified then defaults to ALL.
wallet	Wallet		Which wallet to return statementItems for. If unspecified then the UK wallet will be selected

Return type	Description
AccountStatementReport	List of statement items chronologically ordered plus moreAvailable boolean to facilitate paging

Throws	Description
AccountAPINGException	Generic exception that is thrown if this operation fails for any reason.

listCurrencyRates

listCurrencyRates

List<CurrencyRate> listCurrencyRates (String fromCurrency **) throws AccountAPINGException**

Returns a list of currency rates based on given currency. **Please note:** the currency rates are updated once every hour a few seconds after the hour.

Parameter name	Type	Required	Description
fromCurrency	String		The currency from which the rates are computed. **Please note:** GBP is currently the only based currency support

Return type	Description
List<CurrencyRate>	List of currency rates

Throws	Description
AccountAPINGException	Generic exception that is thrown if this operation fails for any reason.

transferFunds

Operation

transferFunds

TransferResponse transferFunds (Wallet from, Wallet to, double amount) throws AccountAPINGException

Transfer funds between the UK Exchange and other wallets.

This operatio is currently deprecated due to the removal of the AUS wallet

Parameter name	Type	Required	Description
from	Wallet	✓	Source wallet
to	Wallet	✓	Destination wallet
amount	double	✓	Amount to transfer

Return type	Description
TransferResponse	Response for transfer funds action

Throws	Description
AccountAPINGException	Generic exception that is thrown if this operation fails for any reason.

Since 1.0.0

Accounts Exceptions

Exceptions

AccountAPINGException

This exception is thrown when an operation fails

Error code	Description
INVALID_INPUT_DATA	Invalid input data
INVALID_SESSION_INFORMATION	The session token hasn't been provided, is invalid or has expired.
UNEXPECTED_ERROR	An unexpected internal error occurred that prevented successful request processing.
INVALID_APP_KEY	The application key passed is invalid or is not present
SERVICE_BUSY	The service is currently too busy to service this request
TIMEOUT_ERROR	Internal call to downstream service timed out
DUPLICATE_APP_NAME	Duplicate application name
APP_KEY_CREATION_FAILED	Creating application key version has failed
APP_CREATION_FAILED	Application creation has been failed
NO_SESSION	A session token header ('X-Authentication') has not been provided in the request
NO_APP_KEY	An application key header ('X-Application') has not been provided in the request
SUBSCRIPTION_EXPIRED	An application key is required for this operation
INVALID_SUBSCRIPTION_TOKEN	The subscription token provided doesn't exist
TOO_MANY_REQUESTS	Too many requests
INVALID_CLIENT_REF	Invalid length for the client reference
WALLET_TRANSFER_ERROR	There was a problem transferring funds between your wallets
INVALID_VENDOR_CLIENT_ID	The vendor client ID is not subscribed to this application key

Other parameters	Type	Required	Description
errorDetails	String		the stack trace of the error
requestUUID	String		

Accounts Enums

Enums

SubscriptionStatus

Value	Description
ALL	Any subscription status
ACTIVATED	Only activated subscriptions
UNACTIVATED	Only unactivated subscriptions
CANCELLED	Only cancelled subscriptions
EXPIRED	Only expired subscriptions

Status

Value	Description
SUCCESS	Sucess status

ItemClass

Value	Description
UNKNOWN	Statement item not mapped to a specific class. All values will be concatenated into a single key/value pair. The key will be 'unknownStatementItem' and the value will be a comma separated string.

Wallet

Value	Description
UK	The UK Exchange wallet
AUSTRALIAN	The Australian Exchange wallet - **THIS IS NOW DEPRECATED**

IncludeItem

Value	Description
ALL	Include all items
DEPOSITS_WITHDRAWALS	Include payments only
EXCHANGE	Include exchange bets only
POKER_ROOM	Include poker transactions only

winLose

Value	Description
RESULT_ERR	Internal error
RESULT_FIX	Result has been updated after an initial state. i.e. your account history has been changed to reflect this.
RESULT_LOST	Loss
RESULT_NOT_APPLICABLE	Include poker transactions only
RESULT_WON	Won
COMMISSION_REVERSAL	Betfair have restored the funds to your account that it previously received from you in commission.

GrantType

Value	Description
AUTHORISATION_CODE	
REFRESH_TOKEN	

TokenType

Value	Description
BEARER	

AffiliateRelationStatus

Value	Description
INVALID_USER	Provided vendor client ID is not valid
AFFILIATED	Vendor client ID valid and affiliated
NOT_AFFILIATED	Vendor client ID valid but not affiliated

Accounts TypeDefinitions

Type definitions

TransferResponse

Transfer operation response

Field name	Type	Required	Description
transactionId	String	✓	The id of the transfer transaction that will be used in tracking the transfers between the wallets

ApplicationSubscription

Application subscription details

Field name	Type	Required	Description
subscriptionToken	String	✓	Application key identifier
expiryDateTime	Date		Subscription Expiry date
expiredDateTime	Date		Subscription Expired date
createdDateTime	Date		Subscription Create date
activationDateTime	Date		Subscription Activation date
cancellationDateTime	Date		Subscription Cancelled date
subscriptionStatus	SubscriptionStatus		Subscription status
clientReference	String		Client reference
vendorClientId	String		Vendor client Id

Subscription History

Application subscription history details

Field name	Type	Required	Description
subscriptionToken	String	✓	Application key identifier
expiryDateTime	Date		Subscription Expiry date
expiredDateTime	Date		Subscription Expired date
createdDateTime	Date		Subscription Create date
activationDateTime	Date		Subscription Activation date
cancellationDateTime	Date		Subscription Cancelled date
subscriptionStatus	SubscriptionStatus		Subscription status
clientReference	String		Client reference

AccountSubscription

Application subscription details

Field name	Type	Required	Description
subscriptionTokens	List< SubscriptionTokenInfo >	✓	Lis t of subscription token details
applicationName	String		Application name
applicationVersionId	String		Application version Id

SubscriptionTokenInfo

Subscription token information

Field name	Type	Required	Description
subscriptionToken	String	✓	Subscription token
activatedDateTime	Date		Subscription Activated date
expiryDateTime	Date		Subscription Expiry date
expiredDateTime	Date		Subscription Expired date
cancellationDateTime	Date		Subscription Cancelled date
subscriptionStatus	SubscriptionStatus		Subscription status

DeveloperApp

Describes developer/vendor specific application

Field name	Type	Required	Description
appName	String	✓	The unique name of the application
appId	long	✓	A unique id of this application
appVersions	List< DeveloperAppVersion >	✓	The application versions (including application keys)

DeveloperAppVersion

Describes a version of an external application

Field name	Type	Required	Description
owner	String	✓	The user who owns the specific version of the application
versionId	long	✓	The unique Id of the application version
version	String	✓	The version identifier string such as 1.0, 2.0. Unique for a given application.
applicationKey	String	✓	The unqiue application key associated with this application version
delayData	boolean	✓	Indicates whether the data exposed by platform services as seen by this application key is delayed or realtime.
subscriptionRequired	boolean	✓	Indicates whether the application version needs explicit subscription
ownerManaged	boolean	✓	Indicates whether the application version needs explicit management by the software owner. A value of false indicates, this is a version meant for personal developer use.
active	boolean	✓	Indicates whether the application version is currently active
vendorId	String		Public unique string provided to the Vendor that they can use to pass to the Betfair API in order to identify themselves.
vendorSecret	String		Private unique string provided to the Vendor that they pass with certain calls to confirm their identity. Linked to a particular App Key.

AccountFundsResponse

Response for retrieving available to bet.

Field name	Type	Required	Description
availableToBetBalance	double		Amount available to bet.
exposure	double		Current exposure.
retainedCommission	double		Sum of retained commission.
exposureLimit	double		Exposure limit.
discountRate	double		User Discount Rate.
pointsBalance	int		The Betfair points balance

AccountDetailsResponse

Response for Account details.

Field name	Type	Required	Description
currencyCode	String		Default user currency Code. See Currency Parameters for minimum bet sizes relating to each currency.
firstName	String		First Name.
lastName	String		Last Name.
localeCode	String		Locale Code.
region	String		Region based on users zip/postcode (ISO 3166-1 alpha-3 format). Defaults to GBR if zip/postcode cannot be identified.
timezone	String		User Time Zone.
discountRate	double		User Discount Rate.
pointsBalance	int		The Betfair points balance.
countryCode	String		The customer's country of residence (ISO 2 Char format)

AccountStatementReport

A container representing search results.

Field name	Type	Required	Description
accountStatement	List<StatementItem >	✓	The list of statement items returned by your request.
moreAvailable	boolean	✓	Indicates whether there are further result items beyond this page.

StatementItem

Summary of a cleared order.

Field name	Type	Required	Description
refId	String		An external reference, eg. equivalent to betId in the case of an exchange bet statement item.
itemDate	Date	✓	The date and time of the statement item, eg. equivalent to settledData for an exchange bet statement item. (in ISO-8601 format, not translated)
amount	double		The amount of money the balance is adjusted by
balance	double		Account balance.
itemClass	ItemClass		Class of statement item. This value will determine which set of keys will be included in itemClassData
itemClassData	Map<String,String>		Key value pairs describing the current statement item. The set of keys will be determined by the itemClass
legacyData	StatementLegacyData		Set of fields originally returned from APIv6. Provided to facilitate migration from APIv6 to API-NG, and ultimately onto itemClass and itemClassData

StatementLegacyData

Summary of a cleared order.

Field name	Type	Required	Description
avgPrice	double		The average matched price of the bet (null if no part has been matched)
betSize	double		The amount of the stake of your bet. (0 for commission payments or deposit/withdrawals)
betType	String		Back or lay
betCategoryType	String		Exchange, Market on Close SP bet, or Limit on Close SP bet.
commissionRate	String		Commission rate on market
eventId	long		**Please note:** this is the Id of the market without the associated exchangeId
eventTypeId	long		Event Type
fullMarketName	String		Full Market Name. For card payment items, this field contains the card name
grossBetAmount	double		The winning amount to which commission is applied.
marketName	String		Market Name. For card transactions, this field indicates the type of card transaction (deposit, deposit fee, or withdrawal).
marketType	marketType		Market type. For account deposits and withdrawals, marketType is set to NOT_APPLICABLE.
placedDate	Date		Date and time of bet placement
selectionId	long		Id of the selection (this will be the same for the same selection across markets)
selectionName	String		Name of the selection
startDate	Date		Date and time at the bet portion was settled
transactionType	String		Debit or credit
transactionId	long		The unique reference Id assigned to account deposit and withdrawals.
winLose	winLose		Win or loss

TimeRange

TimeRange

Field name	Type	Required	Description
from	Date		from, format: ISO 8601)
to	Date		to, format: ISO 8601

CurrencyRate

Currency rate

Field name	Type	Required	Description
currencyCode	String		Three letter ISO 4217 code
rate	double		Exchange rate for the currency specified in the request

AuthorisationResponse

AuthorisationResponse

Wrapper object containing authorisation code and redirect URL for web vendors

Field name	Type	Required	Description
authorisationCode	String	✓	The authorisation code
redirectUrl	String	✓	URL to redirect the user to the vendor page

SubscriptionOptions

SubscriptionOptions

Wrapper object containing details of how a subscription should be created

Field name	Type	Required	Description
subscription_length	int		How many days should a created subscription last for. Open ended subscription created if value not provided. Relevant only if createdSubscription is true.
subscription_token	String		An existing subscription token that the caller wishes to be activated instead of creating a new one. Ignored is createSubscription is true.
client_reference	String		Any client reference for this subscription token request.

VendorAccessTokenInfo

Wrapper object containing UserVendorSessionToken, RefreshToken and optionally a Subscription Token if one was created

Field name	Type	Required	Description
access_token	String	✓	Session token used by web vendors
token_type	TokenType	✓	Type of the token
expires_in	long	✓	How long until the token expires
refresh_token	String	✓	Token used to refresh the session token in future
application_subscription	ApplicationSubscription	✓	Object containing the vendor client id and optionally some subscription information

VendorDetails

Wrapper object containing vendor name and redirect url

Field name	Type	Required	Description
appVersionId	long	✓	Internal id of the application
vendorName	String	✓	Vendor name
redirectUrl	String		URL to be redirected to

AffiliateRelation

Wrapper object containing affiliate relation details

Field name	Type	Required	Description
vendorClientId	String	✓	ID of user
status	AffiliateRelationStatus	✓	The affiliate relation status

Heartbeat API

- Detailed documentation
- Operations
- Type definitions
- Exceptions
- Typical Interaction

This Heartbeat operation is provided to allow customers to automatically cancel their unmatched bets in the event of their API client/s losing connectivity with the Betfair API.

UK Exchange

Interface	Endpoint	<method name> Example
JSON-RPC	https://api.betfair.com/exchange/heartbeat/json-rpc/v1	HeartbeatAPING/v1.0/heartbeat

Italian Exchange

Interface	Endpoint	<method name> Example
JSON-RPC	https://api.betfair.it/exchange/heartbeat/json-rpc/v1	HeartbeatAPING/v1.0/heartbeat

Spanish Exchange

Interface	Endpoint	<method name> Example
JSON-RPC	https://api.betfair.es/exchange/heartbeat/json-rpc/v1	HeartbeatAPING/v1.0/heartbeat

Operation summary

HeartbeatReport	**heartbeat (int preferredTimeoutSeconds)**

Detailed documentation

Heartbeat

- Operations
  - heartbeat
- Events
- Type definitions
  - HeartbeatReport
- Enums
  - ActionPerformed
- Exceptions
  - APINGException

Operations

heartbeat

HeartbeatReport heartbeat (int preferredTimeoutSeconds) throws APINGException

This heartbeat operation is provided to help customers have their positions managed automatically in the event of their API clients losing connectivity with the Betfair API. If a heartbeat request is not received within a prescribed time period, then Betfair will attempt to cancel all 'LIMIT' type bets for the given customer on the given exchange. There is no guarantee that this service will result in all bets being cancelled as there are a number of circumstances where bets are unable to be cancelled. Manual intervention is strongly advised in the event of a loss of connectivity to ensure that positions are correctly managed. If this service becomes unavailable for any reason, then your heartbeat will be unregistered automatically to avoid bets being inadvertently cancelled upon resumption of service. you should manage your position manually until the service is resumed. Heartbeat data may also be lost in the unlikely event of nodes failing within the cluster, which may result in your position not being managed until a subsequent heartbeat request is received.

Parameter name	Type	Required	Description
preferredTimeoutSeconds	int	✓	Maximum period in seconds that may elapse (without a subsequent heartbeat request), before a cancellation request is automatically submitted on your behalf. **The minimum value is 10, the maximum value permitted is 300**. Passing 0 will result in your heartbeat being unregistered (or ignored if you have no current heartbeat registered). You will still get an actionPerformed value returned when passing 0, so this may be used to determine if any action was performed since your last heartbeat, without actually registering a new heartbeat. Passing a negative value will result in an error being returned, INVALID_INPUT_DATA. Any errors while registering your heartbeat will result in a error being returned, UNEXPECTED_ERROR. Passing a value that is less than the minimum timeout will result in your heartbeat adopting the minimum timeout. Passing a value that is greater than the maximum timeout will result in your heartbeat adopting the maximum timeout. The minimum and maximum timeouts are subject to change, so your client should utilise the returned actualTimeoutSeconds to set an appropriate frequency for your subsequent heartbeat requests.

Return type	Description
HeartbeatReport	Response from heartbeat operation

Throws	Description
APINGException	Thrown if the operation fails

Events

This interface does not define any events.

Type definitions

HeartbeatReport

Response from heartbeat operation

Field name	Type	Required	Description
actionPerformed	ActionPerformed	✓	The action performed since your last heartbeat request.
actualTimeoutSeconds	int	✓	The actual timeout applied to your heartbeat request, see timeout request parameter description for details.

Enums

ActionPerformed

Value	Description
NONE	No action was performed since last heartbeat, or this is the first heartbeat
CANCELLATION_REQUEST_SUBMITTED	A request to cancel all unmatched bets was submitted since last heartbeat
ALL_BETS_CANCELLED	All unmatched bets were cancelled since last heartbeat
SOME_BETS_NOT_CANCELLED	Not all unmatched bets were cancelled since last heartbeat
CANCELLATION_REQUEST_ERROR	There was an error requesting cancellation, no bets have been cancelled
CANCELLATION_STATUS_UNKNOWN	There was no response from requesting cancellation, cancellation status unknown

Exceptions

APINGException

This exception is thrown when an operation fails

Error code	Description
INVALID_INPUT_DATA	Invalid input data
INVALID_SESSION_INFORMATION	The session token passed is invalid
NO_APP_KEY	An application key is required for this operation
NO_SESSION	A session token is required for this operation
INVALID_APP_KEY	The application key passed is invalid
UNEXPECTED_ERROR	An unexpected internal error occurred that prevented successful request processing.

Other parameters	Type	Required	Description
errorDetails	String		Specific error details
requestUUID	String		

Typical Interaction

SET UP

```
[{"jsonrpc": "2.0", "method": "HeartbeatAPING/v1.0/heartbeat", "params": {"preferredTimeoutSeconds":"10"}, "id": 1}]
```

```
[{"jsonrpc":"2.0","result":{"actualTimeoutSeconds":10,"actionPerformed":"NONE"},"id":1}]
```

RESET

```
[{"jsonrpc": "2.0", "method": "HeartbeatAPING/v1.0/heartbeat", "params": {"preferredTimeoutSeconds":"0"}, "id": 1}]
```

```
[{"jsonrpc":"2.0","result":{"actualTimeoutSeconds":0,"actionPerformed":"NONE"},"id":1}]
```

RE-SET UP HEARTBEAT

```
[{"jsonrpc": "2.0", "method": "HeartbeatAPING/v1.0/heartbeat", "params": {"preferredTimeoutSeconds":"10"}, "id": 1}]
```

```
[{"jsonrpc":"2.0","result":{"actualTimeoutSeconds":10,"actionPerformed":"NONE"},"id":1}]
```

You should be able to reset the heartbeat by passing a value of actualTimeoutSeconds":0 and then restarting it by setting the required value.

EXAMPLE OF RESPONSE IF HEARTBEAT ISN'T RECEIVED WITHIN SPECIFIED TIME

```
[{"jsonrpc": "2.0", "method": "HeartbeatAPING/v1.0/heartbeat", "params": {"preferredTimeoutSeconds":"10"}, "id": 1}]
```

```
[{"jsonrpc":"2.0","result":{"actualTimeoutSeconds":10,"actionPerformed":"ALL_BETS_CANCELLED"},"id":1}]
```

Race Status API

The **listRaceDetails** operation is provided to allow customers to establish the status of a horse or greyhound race market both prior to and after the start of the race. This information is available for UK and Ireland races only.

- listRaceDetails
- Operation summary
- Operations
- Events
- Type definitions
- Type aliases
- Enums
- Exceptions

listRaceDetails

Interface	Endpoint	JSON-RPC Prefix	<methodname> Example
JSON-RPC	https://api.betfair.com/exchange/scores/json-rpc/v1	<methodname>	ScoresAPING/v1.0/listRaceDetails

Operation summary

List<RaceDetails>	**listRaceDetails (** Set<MeetingId> meetingIds, Set<RaceId> raceIds **)**

- Operations

Operations

listRaceDetails

List<RaceDetails> listRaceDetails (Set<MeetingId> meetingIds, Set<RaceId> raceIds **) throws APINGException**

Search for races to get their details.

Parameter name	Type	Required	Description
meetingIds	Set<MeetingId>		Optionally restricts the results to the specified meeting IDs. The unique Id for the meeting equivalent to the eventId for that specific race as returned by listEvents. Optionally restricts the results to the specified meeting IDs.
raceIds	Set<RaceId>		Optionally restricts the results to the specified race IDs. The unique Id for the race in the format meetingid.raceTime (hhmm). raceTime is in GMT. Optionally restricts the results to the specified race IDs. The raceID field is returned within the Race node of the Navigation Data For Applications service.

Return type	Description
List<RaceDetails>	List of retrieved race details

Throws	Description
APINGException	

Since 1.0.0

Events

This interface does not define any events.

Type definitions

RaceDetails

Race Details

Field name	Type	Required	Description
meetingId	MeetingId	✓	The unique Id for the meeting equivalent to the eventId for that specific race as returned by listEvents. Optionally restricts the results to the specified meeting IDs.
raceId	RaceId	✓	The unique Id for the race in the format meetingid.raceTime (hhmm). Optionally restricts the results to the specified race IDs.
raceStatus	RaceStatus	✓	The current status of the race.
lastUpdated	LastUpdated		This is the time the data was last updated
responseCode	ResponseCode	✓	

Type aliases

Alias	Type
UpdateSequence	long
EventId	String
EventTypeId	String
EventTime	String
UpdateType	String
LastUpdated	Date
MeetingId	String
RaceId	String

Enums

RaceStatus

Value	Description
DORMANT	There is no data available for this race.
DELAYED	The start of the race has been delayed
PARADING	The horses/greyhounds are in the parade ring
GOINGDOWN	The horses are going down to the starting post
GOINGBEHIND	The horses are going behind the stalls
APPROACHING	The greyhounds are approaching the traps
GOINGINTRAPS	The greyhounds are being put in the traps
HARERUNNING	The hare has been started
ATTHEPOST	The horses are at the post
OFF	The race has started
FINISHED	The race has finished
FINALRESULT	The result has been declared (Greyhounds only)
FALSESTART	There has been a false start
PHOTOGRAPH	The result of the race is subject to a photo finish
RESULT	The result of the race has been announced
WEIGHEDIN	The jockeys have weighed in
RACEVOID	The race has been declared void
NORACE	The race has been declared a no race
MEETINGABANDONED	The meeting has been abandoned
RERUN	The race will be rerun
ABANDONED	The race has been abandoned

ResponseCode

Value	Description
OK	Data returned successfully
NO_NEW_UPDATES	No updates since the passes UpdateSequence
NO_LIVE_DATA_AVAILABLE	Event scores are no longer available or are not on the schedule
SERVICE_UNAVAILABLE	Data feed for the event type (tennis/football etc) is currently unavailable
UNEXPECTED_ERROR	An unexpected error occurred retrieving score data
LIVE_DATA_TEMPORARILY_UNAVAILABLE	Live Data feed for this event/match is temporarily unavailable, data could potentially be stale

Exceptions

APINGException

This exception is thrown when an operation fails

Error code	Description
UNEXPECTED_ERROR	The operation failed with an unexpected error
INVALID_INPUT_DATA	Invalid input data
INVALID_SESSION_INFORMATION	The session token passed is invalid or expired
INVALID_APP_KEY	The application key passed is invalid
SERVICE_BUSY	The service is currently too busy to service this request
TIMEOUT_ERROR	Internal call to downstream service timed out
NO_SESSION	A session token is required for this operation
NO_APP_KEY	An application key is required for this operation
TOO_MANY_REQUESTS	Too many requests
SERVICE_UNAVAILABLE	Service is currently unavailable

Other parameters	Type	Required	Description
errorDetails	String		The stack trace of the error
requestUUID	String		

Interface Definition Documents

The below documents provide a machine readable interface description for API-NG in XML format.

Updated 4th April 2017

SportsAPING.xml

AccountAPING.xml

HeartbeatAPING.xml

Additional Information

Betfair Price Increments

Below is a list of price increments per price 'group'. Placing a bet outside of these increments will result in an **INVALID_ODDS** error

Odds Markets

Price	Increment
1.01 2	0.01

2 3	0.02
3 4	0.05
4 6	0.1
6 10	0.2
10 20	0.5
20 30	1
30 50	2
50 100	5
100 1000	10

BettingType ASIAN_HANDICAP_SINGLE_LINE & ASIAN_HANDICAP_DOUBLE_LINE only

Price	Increment
1.01 1000	0.01

Currency Parameters

Guide to available currencies and minimum bet sizes.

Currency name	Symbol	Currency Code	Min Bet Size	Min Deposit Size	Minimum BSP Liability	Minimum Bet Payout
UK Sterling	£	GBP	2	10	10	10
Euro	€	EUR	2	15	20	20
US Dollar	US$	USD	4	15	20	20
Hong Kong Dollars	HK$	HKD	25	150	125	125
Australian Dollar	AUD	AUD	5	30	30	30
Canadian Dollar	CAD	CAD	6	25	30	30
Danish Kroner	DKK	DKK	30	150	150	150
Norwegian Kronor	NOK	NOK	30	150	150	150
Swedish Krona	SEK	SEK	30	150	150	150
Singapore Dollar	SGD	SGD	6	30	30	30

Racecourse Abbreviations

The racecourse abbreviations lists for Horse Racing and Greyhounds is available via horsegreyhoundcourseabbreviations.xls

Runner Metadata Description

The RUNNER_METADATA returned by listMarketCatalogue for **Horse Racing** (when available) is described in the table below.

Parameter	Description
WEIGHT_UNITS	The unit of weight used.

ADJUSTED_RATING	Adjusted ratings are race-specific ratings which reflect weights allocated in the race and, in some circumstances, the age of the horse. Collectively they represent the chance each runner has on form. https://www.timeform.com/Racing/Articles/How_the_ratings_for_a_race_are_calculated Please note: this data is only returned for those with a Premium Timeform subscription
DAM_YEAR_BORN	The year the horse’s mother's birth
DAYS_SINCE_LAST_RUN	The number of days since the horse last ran
WEARING	Any extra equipment the horse is wearing
DAMSIRE_YEAR_BORN	The year in which the horse's grandfather was born on its mothers side
SIRE_BRED	The country were the horse's father was bred
TRAINER_NAME	The name of the horse's trainer
STALL_DRAW	The stall number the horse is starting from
SEX_TYPE	The sex of the horse
OWNER_NAME	The owner of the horse
SIRE_NAME	The name of the horse's father
FORECASTPRICE_NUMERATOR	The forecast price numerator
FORECASTPRICE_DENOMINATOR	The forecast price denominator
JOCKEY_CLAIM	The reduction in the weight that the horse carries for a particular jockey were applicable.
WEIGHT_VALUE	The weight of the horse
DAM_NAME	The name of the horse's mother
AGE	The age of the horse
COLOUR_TYPE	The colour of the horse
DAMSIRE_BRED	The country were the horse's grandfather was born
DAMSIRE_NAME	The name of the horse's grandfather
SIRE_YEAR_BORN	The year the horse's father was born
OFFICIAL_RATING	The horses official rating
FORM	The horses recent form
BRED	The country in which the horse was born
runnerId	The runnerId for the horse
JOCKEY_NAME	The name of the jockey. **Please note**: This field will contain '**Reserve**' in the event that the horse has been entered into the market as a reserve runner. Any reserve runners will be withdrawn from the market once it has been confirmed that they will not run.
DAM_BRED	The country where the horse's mother was born
COLOURS_DESCRIPTION	The textual description of the jockey silk
COLOURS_FILENAME	A relative URL to an image file corresponding to the jockey silk. You must add the value of this field to the base URL: http://content-cache.betfair.com/feeds_images/Horses/SilkColours/ **Please note** - silk cloth images aren't provided for US Racing. The saddlecloth images used for US racing can be view via https://sn4.cdnbf.net/exchange/plus/images/app/common/assets/images/saddlecloths-sprite_2608_.gif
CLOTH_NUMBER	The number on the saddle-cloth
CLOTH_NUMBER ALPHA	The number on the saddle-cloth. For US Racing were the runner is paired, this field will display the cloth number of the paired runner e.g. "1A"

Terms of use: Please refer to http://form.timeform.betfair.com/termsofuse regarding the above data.

Time Zones & Time Format

All times are returned in GMT and are in ISO 8601 (http://en.wikipedia.org/wiki/ISO_8601) format. They can be converted to your local timezone using the timezone field returned by the getAccountDetails operation or into the local market timezone using the timezone returned for the event by listMarketCatalogue

To synchronize with the Betfair server time we recommend that you use the pool of NTP servers listed via http://www.pool.ntp.org/zone/europe

The following table lists the time zones returned be the API along with their meaning.

Location	Abbreviation	Notes
Africa/Johannesburg	RSA	
America/Costa_Rica	SJMT	
America/Indiana/Indianapolis	IEST	North America Indiana East
America/Santiago	SMT	
Asia/Bangkok	THAI	
Asia/Calcutta	INT	
Asia/Dubai	UAE	
Australia/Adelaide	ACST	
Australia/Darwin	ANST	
Australia/Perth	AWST	
Australia/Queensland	AQST	
Australia/Sydney	AEST	
Brazil/East	BRT	
Brazil/West	AMT	
CET	CET	Central European Time
EET	EET	Eastern European Time
Etc/GMT-5	PKT	
Europe/London	UKT	
Europe/Moscow	MSK	
GMT/UTC	GMT/UTC	Greenwich Mean Time/Coordinated Universal Time
Hongkong	HK	
Jamaica	KMT	
Japan	JPT	
NZ	NZT	New Zealand
US/Alaska	AKST	
US/Arizona	AST	
US/Central	CST	
US/Eastern	EST	
US/Hawaii	HST	

US/Mountain	MST	
US/Pacific	PST	

Common Error Codes

Meaning	FaultCode	Client/Server	Associated HTTP Transport Response Code	Comments
DSC-0008	JSONDeserialisationParseFailure	Client	400	
DSC-0009	ClassConversionFailure	Client	400	Invalid format for parameter, for example passing a string where a number was expected. Can also happen when a value is passed that does not match any valid enum.
DSC-0018	MandatoryNotDefined	Client	400	A parameter marked as mandatory was not provided
DSC-0019	Timeout	Server	504	The request has timed out
DSC-0021	NoSuchOperation	Client	404	The operation specified does not exist
DSC-0023	NoSuchService	Client	404	
DSC-0024	RescriptDeserialisationFailure	Client	400	Exception during deserialization of RESCRIPT request
DSC-0034	UnknownCaller	Client	400	A valid and active App Key hasn't been provided in the request. Please check that your App Key is active. Please see Application Keys for further information regarding App Keys.
DSC-0035	UnrecognisedCredentials	Client	400	
DSC-0036	InvalidCredentials	Client	400	
DSC-0037	SubscriptionRequired	Client	403	The user is not subscribed to the App Key provided
DSC-0038	OperationForbidden	Client	403	The App Key sent with the request is not permitted to access the operation

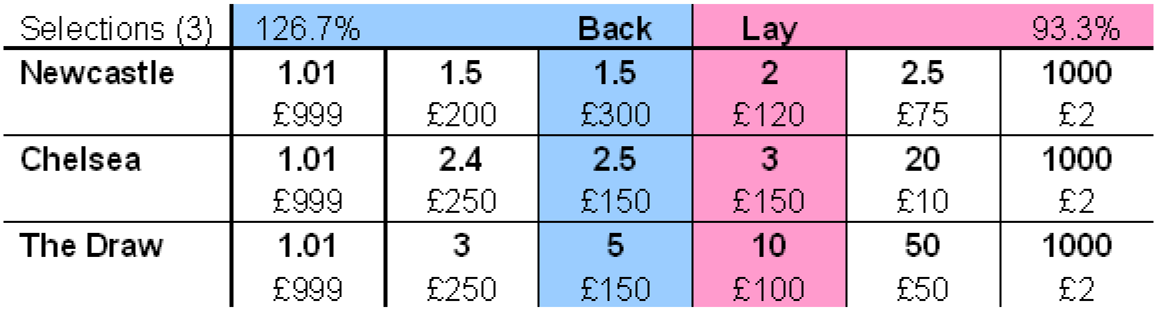
Virtual Bets

The Betfair Exchange uses a 'cross matching' algorithm to display the best possible prices (bets) available by taking into account the back and lay offers (unmatched bets) across all selections.

> You can return virtual bets in the response when using API-NG by including the virtualise":"true" in the listMarketBook request e.g. [{"jsonrpc": "2.0", "method": "SportsAPING/v1.0/listMarketBook", "params": {"marketIds":["1.114101556"],"priceProjection":{"priceData":["EX_BEST_OFFERS"],"virtualise":"true"}}, "id": 1}]
>
> You should subscribe to the EX_BEST_OFFERS_DISP MarketDataFlter if using the **Exchange Stream API**

One of the easiest ways to understand how we generate virtual bets for cross matching is to work through a couple of examples.

Consider the following market and what would happen if we placed a very large back bet at 1.01 on The Draw:



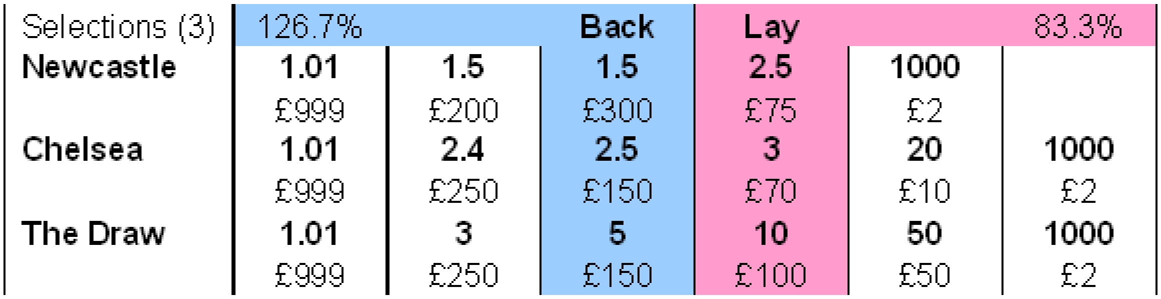
Selections (3)	126.7%		Back	Lay		93.3%
Newcastle	1.01 £999	1.5 £200	1.5 £300	2 £120	2.5 £75	1000 £2
Chelsea	1.01 £999	2.4 £250	2.5 £150	3 £150	20 £10	1000 £2
The Draw	1.01 £999	3 £250	5 £150	10 £100	50 £50	1000 £2

Without cross matching, this bet would be matched in three portions;

1) £150 at 5.0,
2) £250 at 3.0,
and £999 at 1.01 with anything remaining being left unmatched.

With cross matching we can do better.

We have a back bet on Newcastle for £120 at 2.0 and a back bet on Chelsea for £150 at 3.0 (shown in pink on the available to lay side of the market).



Selections (3)	126.7%		Back	Lay		83.3%
Newcastle	1.01 £999	1.5 £200	1.5 £300	2.5 £75	1000 £2	
Chelsea	1.01 £999	2.4 £250	2.5 £150	3 £70	20 £10	1000 £2
The Draw	1.01 £999	3 £250	5 £150	10 £100	50 £50	1000 £2

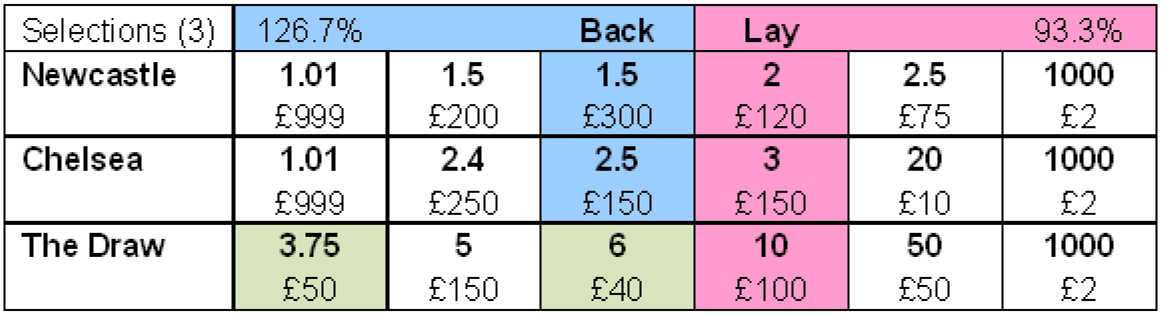
These two bets can be matched against a back bet on The Draw at a price of 6.0, since 2.0, 3.0, and 6.0 form a 100% book. To ensure that the book is balanced, we choose the stakes to be inversely proportional to the prices.

This means that we take the full £120 at 2.0 on Newcastle, only £80 at 3.0 on Chelsea, and £40 at 6.0 on The Draw, which is the first virtual bet

We now have a back bet on Newcastle for £75 at 2.5 and a back bet on Chelsea for £70 at 3.0 (again shown in pink on the available to lay side of the market).

These two bets can be matched against a back bet on The Draw at a price of 3.75, since 2.5, 3.0, and 3.75 also form a 100% book.

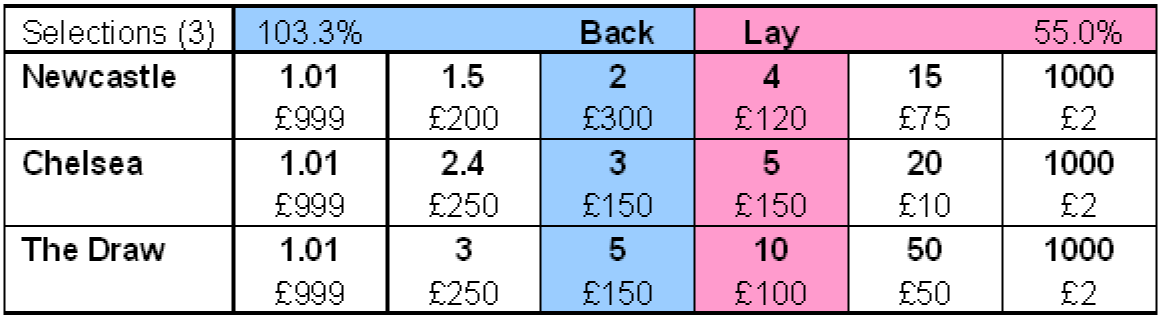
Balancing the stakes means that we need to take the full £75 at 2.5 on Newcastle, only £62.50 at 3.0 on Chelsea, and £50 at 3.75 on The Draw, which is the second virtual bet. Since 3.75 is less than 5.0, the £150 at 5.0 would be matched first followed by £50 at 3.75. If we continued this process we would get further matching at 1.50 and 1.05, but for the purposes of displaying the market view we have the best 3 prices for the available to back bets on The Draw, and so we can stop calculating the virtual bets. The virtual bets are just the bets that would have been matched had we received a sufficiently large back bet at 1.01; in this example, £40 at 6.0 and £50 at 3.75. **We take these virtual bets and merge them with the existing bets on the market to generate the following market view (with the virtual bets shown in green)**



Selections (3)	126.7%		Back	Lay		93.3%
Newcastle	1.01 £999	1.5 £200	1.5 £300	2 £120	2.5 £75	1000 £2
Chelsea	1.01 £999	2.4 £250	2.5 £150	3 £150	20 £10	1000 £2
The Draw	3.75 £50	5 £150	6 £40	10 £100	50 £50	1000 £2

The process is repeated to obtain the virtual lay bets (available to back bets) for Newcastle and Chelsea.

Here we have a slightly different market (as before, chosen to make the numbers nice) and consider what would happen if we placed a very large lay bet at 1000 on The Draw.

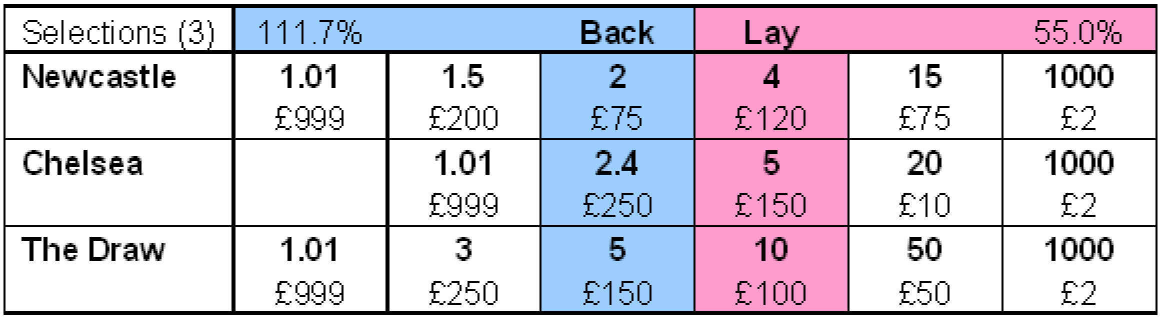


Selections (3)	103.3%		Back	Lay		55.0%
Newcastle	1.01 £999	1.5 £200	2 £300	4 £120	15 £75	1000 £2
Chelsea	1.01 £999	2.4 £250	3 £150	5 £150	20 £10	1000 £2
The Draw	1.01 £999	3 £250	5 £150	10 £100	50 £50	1000 £2

Without cross matching, this bet would be matched in three portions; 1) £100 at 10.0, 2) £50 at 50.0, and £2 at 1000 with anything remaining being left unmatched. With cross matching we can do better. We have a lay bet on Newcastle for £300 at 2.0 and a lay bet on Chelsea for £150 at 3.0 (shown in blue on the available to back side of the market). These two bets can be matched against a lay bet on The Draw at a price of

6.0, since 2.0, 3.0, and 6.0 form a 100% book. To ensure that the book is balanced, we choose the stakes to be inversely proportional to the prices. This means that we take only £225 at 2.0 on Newcastle, the full £150 at 3.0 on Chelsea, and £75 at 6.0 on The Draw, which is the first virtual bet.

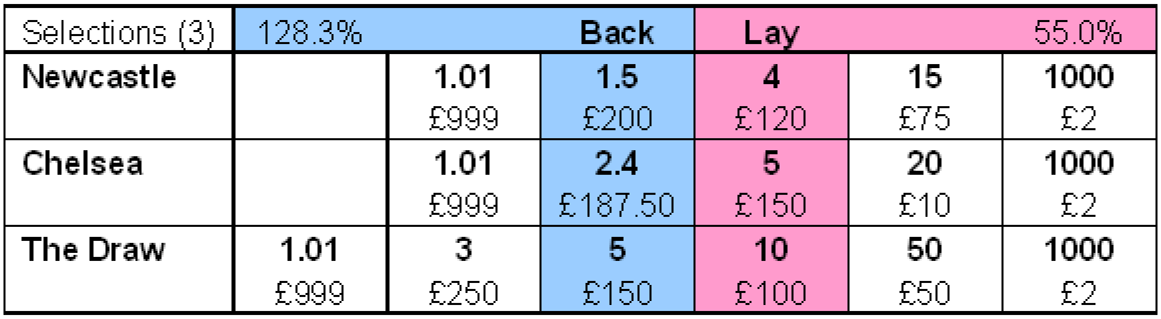
Assuming these bets got matched, the market would look like this:



Selections (3)	111.7%		Back	Lay		55.0%
Newcastle	1.01 £999	1.5 £200	2 £75	4 £120	15 £75	1000 £2
Chelsea		1.01 £999	2.4 £250	5 £150	20 £10	1000 £2
The Draw	1.01 £999	3 £250	5 £150	10 £100	50 £50	1000 £2

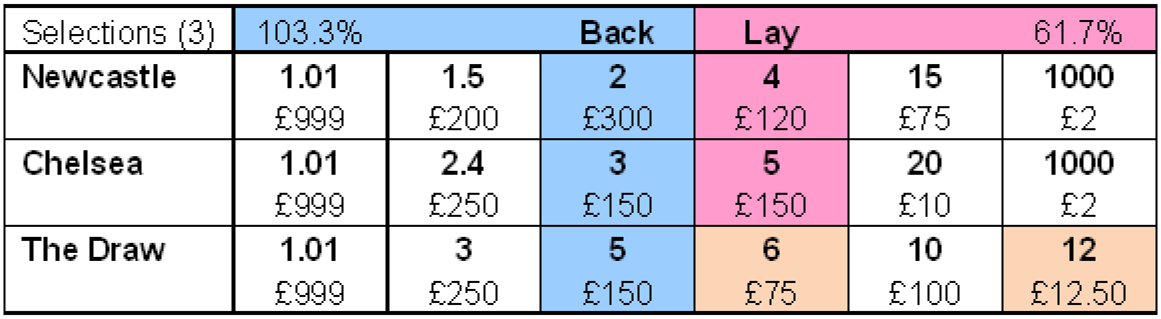
We now have a lay bet on Newcastle for £75 at 2.0 and a lay bet on Chelsea for £250 at 2.4 (again shown in blue on the available to back side of the market). These two bets can be matched against a lay bet on The Draw at a price of 12.0, since 2.0, 2.4, and 12.0 also form a 100% book. Balancing the stakes means that we need to take the full £75 at 2.0 on Newcastle, only £62.50 at 2.4 on Chelsea, and £12.50 at 12.0 on The Draw, which is the second virtual bet.

This leaves the following market:



Selections (3)	128.3%		Back	Lay		55.0%
Newcastle		1.01 £999	1.5 £200	4 £120	15 £75	1000 £2
Chelsea		1.01 £999	2.4 £187.50	5 £150	20 £10	1000 £2
The Draw	1.01 £999	3 £250	5 £150	10 £100	50 £50	1000 £2

This time we can't continue the process since there is no valid price for a virtual bet on The Draw that would result in a 100% book, and so we can stop calculating the virtual bets. Again, the virtual bets are just the bets that would have been matched had we received a sufficiently large lay bet at 1000; in this example, £75 at 6.0 and £12.50 at 12.0. **We take these virtual bets and merge them with the existing bets on the market to generate the following market view (with the virtual bets shown in orange):**



Selections (3)	103.3%		Back	Lay		61.7%
Newcastle	1.01 £999	1.5 £200	2 £300	4 £120	15 £75	1000 £2
Chelsea	1.01 £999	2.4 £250	3 £150	5 £150	20 £10	1000 £2
The Draw	1.01 £999	3 £250	5 £150	6 £75	10 £100	12 £12.50

Locale Specification

The locale specification determines the language returned for names of sports and markets. It is an optional parameter you can specify when you want to retrieve names in a language that differs from the language specified for the account. For example, if the account language is specified as English, you can use the locale parameter to retrieve non-English sport or market names.

The language code is based on the ISO 639-1 standard which defines two-letter codes, such as "en" and "fr".

The follow languages are available, but please be aware not all markets in all languages are fully translated:

Language	Locale Code

English	en
Danish	da
Swedish	sv
German	de
Italian	it
Greek	el
Spanish	es
Turkish	tr
Korean	ko
Czech	cs
Bulgarian	bg
Russian	ru
French	fr
Thai	th