

# LD - General information



## Table of contents

- [Bookmaker SDK](#)
- [Access method](#)
- [Access restrictions](#)
- [What is expected of the client system](#)
- [Terms and agreement](#)
- [Live Data on the Cloud \(DLD\): ID to UUID migration FAQ](#)
  - [How to get UUID on the LD feed?](#)
  - [Which version and variant of UUID does Live Data use?](#)
  - [Can I use UUID to track the sequence of events?](#)
  - [How we suggest you keep track of "sequence of events" once ID is removed](#)
  - [How to update my code?](#)

## Bookmaker SDK

The bookmaker SDK is a client library that enables easier integration with the Betradar XML feeds. The SDK exposes the XML feed service interface in a more user-friendly way, and isolates the client from having to do XML feed parsing, proper connection handling, error recovery, event queuing and dispatching. It also makes a client solution more stable and robust when it comes to feed handling, especially with the release of new and updated XML feed versions. The Bookmaker SDK is available for Java and .NET:

- Bookmaker SDK for Java: <http://sdk.sportradar.com/bookmaker/java>
- Bookmaker SDK for .NET: <http://sdk.sportradar.com/bookmaker/net2>

## Access method

XML messages describing events happening during the match are sent through a socket connection to the client application. The system supports SSL connections only. Each account is able to have 2 open connections simultaneously. Connection information:

**Hostname:** [scout.betradar.com](http://scout.betradar.com)  
**Port:** 2017 (SSL only)

## Access restrictions

For security reasons we only allow connections from IP addresses that have been added to our whitelist. It is therefore necessary that you provide us with the relevant IP addresses from which you are accessing our server before using the production server. Otherwise your login requests will not be successful. The replay server has a separate whitelist. Testing should be done via the replay server before going live.

### Maximum 500 request in a 2 minute period.

If you exceed these limits, your account will be automatically suspended. The account can be reactivated by our support department immediately, once the spamming has been fixed. In this case you will receive automatically an email alerting you about the account suspension.

To ensure that your connection to our server is receiving data fast enough, some security mechanisms were added to avoid an XML message queue on the Betradar side:

### Connection will be disconnected as soon as the queue size has exceeded 700 files

There are also limitations to the time span for matchlist requests:

1. The maximum time span is 300 hours. If the timepan is greater than 300 hours, it will automatically be adjusted to "hoursforward" + 300 hours
2. If the optional attribute includeavailable="yes" is set, the maximum values of "hoursback" and "hoursforward" are 48 hours and 240 hours respectively

## What is expected of the client system

- Ability to establish a socket connection to the server (SSL)
- Automatic reconnection if connection to server for some reason is lost
- Automatically resubscribe to matches after a reconnect
- Functionality for both sending and receiving XML messages over the socket
- Need to send alive-messages periodically to let the server know the client is alive

## Terms and agreement

When purchasing the product, your Betradar account will be granted access to the Live Data XML feed. Username and password for logging in to the feed is the same as for logging in to [www.betradar.com](http://www.betradar.com). Each account is allowed to have two open subscriptions to the same match simultaneously. Each user within the account is allowed to subscribe to a match once. If a user subscribes to the same match a second time, the old subscription will be cancelled. Please note that the scout flash client (accessible through [www.betradar.com](http://www.betradar.com)) connects to the Live Data XML feed, using the client this will consume one of the available subscriptions.

## Live Data on the Cloud (DLD): ID to UUID migration FAQ

Below you will find information about the setting to include a reference to the "Universally Unique Identifier" (UUID) of the deleted event to all <event.../> messages in the feed. More information about this setting can be found [HERE](#).

### Please note



*The UUID was introduced in July 2019 as optional (XML Config ID 140) but from October 2020 it will be mandatory and will replace the ID for all events (ID will be removed).*

*To uniquely identify data events, ensure that you are using the UUID field and not the ID field in data you receive from us.*

*The problem with relying on ID is that it is not guaranteed to be unique should your connection be diverted to a different processing region.*

*If your code uses ID field, you will need to update your code to use UUID instead. This provides additional information about the UUID and intends to support customers while adapting their systems.*

## How to get UUID on the LD feed?

1. Activate **XML Config ID 140** (Include `uuid` attribute to all events). Note a `uuid` will be added.

### Example

```
<event type="1029" stime="1590607070215" side="away" mtime="17:07" info="
Dangerous attack [T2]" id="2565202555" uuid="136814be-76b0-4283-96d3-
16d89246049d" posx="10" posy="64" matchscore="0:1" ballspotting="false"
matchstatus="FIRST_HALF" />
```

2. Activate **XML Config ID 141** (Include `refuuid` to DELETE\_EVENT\_ALERT event). Note a `refuuid` will be added to events with `type=1044` (DELETE\_EVENT\_ALERT), its value represents the UUID of the event deleted.

### Example

```
<event type="1044" stime="1590590593442" side="away" mtime="00:37:36"
info="Event deleted : Tennis point (originally entered at 14:42:51 UTC)"
id="2564731569" refuuid="5b7ef738-b7ae-4731-88c4-5da02dbb6a96" uuid="
8ef9dbbc-20e4-4426-8e79-ab68a5dd7411" extrainfo="2564731021" gamenumber="
6" setnumber="1" gamescore="15:40" setscore="4:1" matchscore="0:0"
matchstatus="FIRST_SET" />
```

## Which version and variant of UUID does Live Data use?

Version 4, Variant 2

## Can I use UUID to track the sequence of events?

No. UUID should be seen as random string. [https://en.wikipedia.org/wiki/Universally\\_unique\\_identifier#Format](https://en.wikipedia.org/wiki/Universally_unique_identifier#Format)

## How we suggest you keep track of "sequence of events" once ID is removed

The attribute `id` was never intended to be used to track sequence of events. For sequences please use the `stime` attribute.

## How to update my code?

1. Make sure your code is parsing the new elements `uuid` and `refuuid`
2. Use `uuid` as unique internal identifier of events.
3. Use `refuuid` when handling `DELETE_EVENT_ALERT` (1044)

[Back to top](#)