

# UOF - General information

## Table of contents:

- [Timestamps in the documentation XML examples](#)
- [Access methods](#)
- [Working with the SDK](#)
- [Working with the API](#)
- [Rate limits](#)

## Additional content

- [UOF - Access restrictions for odds recovery](#)
- [UOF - AMQP topic filtering](#)
- [UOF - Generating access tokens for Unified Odds Feed](#)
- [UOF - Environments](#)
- [UOF - Language support](#)
- [UOF - Stage structure in Formula 1](#)

## Timestamps in the documentation XML examples

Every [message](#) in Unified Feed has a timestamp (when the message was created). Keep this in mind if some examples in the documentation are missing the timestamp. They have simply been removed from the examples in order to highlight the discussion topic.

In Unified Feed there are multiple types of messages that are produced. In some cases, there may be messages produced at the same time (two messages have the same timestamp). It is important to view this as a parallel process, and it may happen that i.e. a [fixture change](#) message ends up having the same timestamp as an [odds change](#) message. Even the same odds [producer](#) may generate multiple odds changes on the same millisecond, but this is unlikely to happen on the same sport event.

## Access methods

The preferred and recommended mechanism is the [UnifiedFeed SDK](#), as it helps simplify protocol handling; in particular, things like caching of descriptive localized meanings of markets, outcomes, players, teams, etc. Unless fine-grained control is a requirement, you can focus on the UnifiedFeed SDK and can safely disregard the protocol specification. Welcome to the Betradar Unified Odds Feed. The Unified Odds ensures you can uniformly access all odds for all events (matches, races, outrights) that Betradar supports in a consistent and fast manner. There are two ways to access Betradar Unified Odds-related information: either through the UnifiedFeed Software Development Kit (SDK) library in Java and C#, or directly at the protocol level. Throughout Unified Odds (whether you use the SDK or the protocols) Betradar upholds various unifying concepts: names and terms are the same across our different endpoints (odds, fixtures, teams or sports) and datatypes are consistent.

Betradar Unified Odds are provided through two protocol mechanisms: Messaging over Advanced Message Queuing Protocol (AMQP 0.9.1) and a HTTP/XML-based application program interface (API). The messages are designed to be lightweight and only include important changes – additional information, such as match-details (team names, player names, etc.). Localized versions of such information are obtained through the [API](#). This design enables Betradar to keep the messages as small as possible for best possible performance and latency.

## Working with the SDK

Betradar provides an extension to the Bookmaker SDK for accessing the Unified Odds in Java or C#. Read more about how to use the *Unified Feed SDK* using our online documentation available at <http://sdk.sportradar.com/>, in the “*UnifiedFeed SDK*” section. If you cannot work with the SDK, the protocols can be used instead. This will add more complexity.

## Working with the API

When programming directly against the protocol, you must handle additional elements and attributes where specified in the corresponding schemas. This enables Betradar to add new functionality when it becomes available, and for you to take advantage of such information when convenient. No existing required data is removed or renamed. If such changes are made, they do not show up in the existing feed, and the existing feed works for at least 24 months after such a change has been made.

- Unified Odds Messages provide fast real-time updates to key information, such as odds over AMQP (0.9.1)/XML.
- [The Unified Odds API](#) is a request-based RESTful API for additional information (player names, current scores, market descriptions, etc.).

## Rate limits

Most of our rate limits outside of our recovery endpoints are set high - in order to avoid them being reached with regular usage. There should not be a reason to pull i.e. the static market overview more than 600 times per hour, but this is currently possible.

### Note



Special rate limits apply to our recovery endpoints. For more specific information about how these rate limits work please visit [THIS](#) section about access restrictions and recovery.

If you are using the [SDK](#) - it will handle recovery automatically.

[Back to top](#)