Betradar Unified Odds

Migration guide - Moving from Live Odds to Unified Feed

August 31, 2018

Introduction

If you have been using the legacy Live Odds feed for a while, there are several differences when moving over to the new and improved Unified Odds feed. Unified Odds provides a more cleaned up version of the legacy XML feed, and in addition provides users with an API for requesting additional information not provided in the feed.

This document has been provided as a migration guide for clients moving over from the legacy Live Odds feed to Unified Feed. Down below you fill find most of the major differences in Unified Feed, coming from the legacy Live Odds feed. If you have any additional questions concerning our Unified Feed product, feel free to contact our customer support for additional information and assistance.

EMAIL: support@betradar.com
PHONE: +41 71 517 72 00

Table of content

Summary

Environments

The Unified feed provides clients with 3 different environments; *Production*, *integration* and a *replay server*.

Improved XML feed

Unified feed provides clients with messages through AMQP and an API to provide lightweight, fast and relevant messages. We also provide our clients with a Java and .Net SDK (Software development kit).

XML configurations as standard

Unified feed provides most of the legacy Live Odds configurations as standard, and additional information outside the feed can be accessed from the API when needed instead.

Cleaned up messages

Unified Feed provides a different message logic than the legacy Live Odds feed, providing easier to read and manage messages for our clients. This applies to how odds, markets, probabilities and more are represented in the feed. The same information is provided with improved performance, easier to understand logic and less configurations to worry about.

Multiple producers, one feed

Unified feed combines all our odds producers into a single feed, a single point of entry for all information about both pre-match odds and live odds. Unified Feed also includes, but is not limited to our virtual sport products, BetPal and Premium Cricket producers.

Mapping

With Unified Feed it is possible to get mappings to all our legacy Live Odds and Life cycle of odds IDs if needed.

SDK and API

Betradar Unified Odds are provided through two protocol mechanisms: Messaging over *Advanced Message Queuing Protocol* (AMQP), 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.), and 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.

The Unified Odds SDK (Software developer kit) provides a simple and efficient way for bookmakers to access Betradar's odds and sport information. It combines subscription of messages and API calls into a unified Java or .NET interface that hides most of the complexity, including recovery.

For more information about our SDK and what it offers, feel free to check out our dedicated SDK documentation. If you are unable to use our SDK, you will need to take advantage of our API and work on the protocol level.

Self-service documentation

All endpoints in Unified Feed are available for testing purposes at http://iodocs.betradar.com. From here you can see each of the endpoints, execute the different requests, and see sample responses as long as you have an access-token. For more detailed information and examples on how to use the self-service documentation and the API, consult the Unified Feed integration documentation.

The purpose of providing these endpoints is that our clients can access and request information when needed, instead of adjusting configurations in different ways to make it accessible in the feed. This ensures that only vital information will be provided in a clean and neat fashion through our Unified XML Feed, and additional information can be accessed when needed by each individual client from the API. This is also convenient when new features for requesting information is made available, as it can be accessed through the API with no more adjustments of configurations.

Changes to the root element logic in messages

The Betradar system sends out many different types of XML messages. The content of these messages can differ from odds updates to alive messages, but the root element always stayed the same in the legacy feed, <BetradarLiveOdds>. Which type of XML message is determined by the status attribute found inside this root element.

Incremental ID

In the legacy Live Odds feed there was a feature in each message that displayed an incremental ID, showing in what sequence messages would arrive in the XML feed. In the following example you can see this illustrated by the *msgnr* attribute inside of the <BetradarLiveOdds> element.

XML example 1 - Incremental ID

```
<BetradarLiveOdds status="betstart" time="0" timestamp="1199435558847">
<Match active="1" betstatus="started" matchid="661373" msgnr="5"
score="4:4" status="not_started"/>
</BetradarLiveOdds>
```

Because Unified Feed is a distributed system with multiple producers, there is no need for an incremental ID. Either you get all the messages you are interested in, in the correct sequence, or at some point you will stop receiving messages altogether.

Alive message

Below is an example of an alive message from the legacy feed, where the status of the <BetradarLiveOdds> message has been set to status=”alive” to indicate that this is an alive message.

XML example 2 - Legacy alive message

```
<BetradarLiveOdds status="alive" timestamp="1413791030423">
<Match active="1" betstatus="started" gamescore="0:0" matchid="6263978"
msgnr="126" server="1" score="0:0" setscore1="3:4" status="1set
" tiebreak="false"/>
<Match active="1" betstatus="started" gamescore="0:30" matchid="6264696"
msgnr="188" server="1" score="0:0" setscore1="5:3" status="1
set" tiebreak="false"/>
</BetradarLiveOdds>
```

In Unified Odds the whole concept of messages has been changed, and each message uses the type of message as its root element in the message. Here is an example of an alive message sent in Unified feed:

XML example 3 - Unified feed alive message

```
<alive timestamp="1234579" product="1" subscribed="1"/>
```

As you can see, the root element is now simply called *alive*, this indicates that this is an alive message, with some attached information. This logic applies to all the different XML messages provided in Unified Feed.

Because Unified Feed receives input from multiple odds producers, the alive messages come with a product attribute that indicates which producer is sending the messages. This comes in the form of a numerical value (product=”1” is the Live Odds producer). Finally, the alive message has an attribute called subscribed, this attribute indicates whether or not the producer is available for sending additional messages.

Odds change message

Another example of how this change looks can be seen in the messages that indicate a change in odds. Again the legacy feed provides this message with the <BetradarLiveOdds> as its root element, illustrated in the example below:

XML example 4 - Legacy odds change messsage

```
<BetradarLiveOdds status="change" timestamp="1199777659034">
<Match active="1" betstatus="started" earlybetstatus="stopped"
matchid="5984472" matchtime="38" msgnr="155" score="1:0" setscores="1:0"
status="1p" >

<Odds active="1" changed="true" combination="0" freetext="Total Corners"
id="57178832" mostbalanced="0" specialoddsvalue="7.5"
subtype="126" type="ftnw" typeid="8">
<OddsField active="1" type="under" typeid="373">2.6</OddsField>
<OddsField active="1" type="over" typeid="374">1.45</OddsField>
</Odds>
<Odds active="1" changed="false" combination="0" freetext="Asian handicap
first half" ftr="0.25" id="57178580" mostbalanced="1"
specialoddsvalue="1.25" subtype="36" type="ft2w" typeid="7">
<OddsField active="1" type="1" typeid="17">2.8</OddsField>
<OddsField active="1" type="2" typeid="18">1.4</OddsField>
</Odds>

<!-- Additional odds possible -->
</Match>
</BetradarLiveOdds>
```

Here we can see the type of XML message is again decided by the status attribute found inside the <BetradarLiveOdds> element in the top of the message. The status=”change” means that this XML message indicates an odds change message.

As mentioned before, Unified Feed uses the name of the XML message type as the root element, in this case the XML message is called odds_change.

XML example 5 - Unified feed odds change message

```
<odds_change event_id="sr:match:1234" timestamp="1234" product="2">
<sport_event_status status=”1" reporting="1" match_status="1"
home_score="2” away_score=”0">
     <clock match_time=”10:00” remaining_time=”50:00” stopped=”true”/>
   </sport_event_status>
  <odds>
    <market id="47" specifiers="score=41.5" favourite="1" status="1">
      <outcome id="1" odds="1.12" active="1"/>
      <outcome id="2" odds="1.92" active="1"/>
    </market>
  </odds>
</odds_change>
```

As you can see the XML message looks different, and not just the changed root element. There are multiple changes from the legacy Live Odds feed, which will also be covered.

Sport event status

The element <sport_event_status> is important, and is provided in the odds_change message. This attribute describes the current status of the sport-event itself (not started, live, ended, closed, etc.). Additional attributes are live-only attributes, and only provided while the match is live; additionally, which attributes are provided depends on the sport.

XML example 6 - sport_event_status element

```
<sport_event_status match_status_name="1st quarter" away_score="0"
home_score="0" match_status="13" reporting="1" status="1">
</sport_event_status>

Odds and market information comes here…
```

In the above example you will also see that Unified feed uses both a numerical and text representation for the match_status attribute. Match statuses use the same numbering as they did in the legacy feed, in this case match_status=”13” means “1st quarter”.

In legacy Live Odds feed the change message would be sent out as a part of the odds change message examples used earlier.

XML example 7 - Legacy feed match information

```
<BetradarLiveOdds status="change" timestamp="1413846115107">
<Match active="1" betstatus="started" earlybetstatus="stopped"
matchid="5650450" matchtime="1" matchtime_extended="0:00" msgnr="6"
score="0:0" setscores="0:0" status="1p">

<!-- Possible odds information goes here -->
</Match>
</BetradarLiveOdds>
```

In the above example you can see the top of an odd change message in the legacy Live Odds feed, providing match information like the <sport_event_status> element in Unified Feed.

Score information

In the legacy Live Odds feed the score information was made available with a message having the status attribute set to change, with an attached <Match> element displaying the score. The below illustration of how this looked in the legacy feed:

XML example 8 - Legacy feed score information

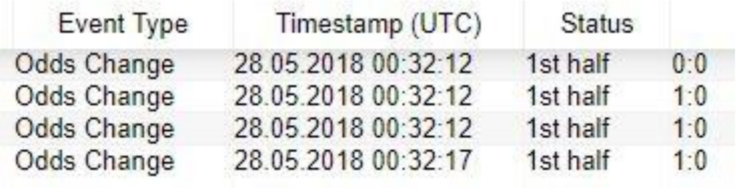
```
<BetradarLiveOdds xmlns="http://www.betradar.com/BetradarLiveOdds"
status="change" timestamp="1414364604992">
<Match active="1" betstatus="started" matchid="5867680" matchtime="1"
msgnr="5" score="0:0" setscores="0:0" status="1p">
<!-- Odds information goes here -->
</Match>
</BetradarLiveOdds>
```

In Unified Feed the same information is available as a sub element <period_scores> in the <sport_event_status> element, found in the <odds_change> element/messages. Here is an example of how this looks in Unified Feed:

XML example 9 - Period scores in unified feed

```
<sport_event_status status="closed" home_score="2" away_score="2"
status_code="4" match_status_code="100">
    <period_scores>
        <period_score home_score="2" away_score="2" type="regular_period"
number="1"/>
        <period_score home_score="0" away_score="0" type="regular_period"
number="2"/>
    </period_scores>
</sport_event_status>
```

Here you can find the <period_score> element showing the home and away score. It is however important to note that individual sporting events (like a soccer goal, yellow/red cards, etc.) are not sent in individual messages in Unified Feed. The change in score will be indicated by changes made to the next odds change message in the XML feed. The message sequence in the XML feed can be seen in the following image:



Event Type	Timestamp (UTC)	Status	
Odds Change	28.05.2018 00:32:12	1st half	0:0
Odds Change	28.05.2018 00:32:12	1st half	1:0
Odds Change	28.05.2018 00:32:12	1st half	1:0
Odds Change	28.05.2018 00:32:17	1st half	1:0

Figure 1 - Message flow in Unified feed

Here you can see the score changes (in the right most column) from 0:0 to 1:0.

Fixtures and entities

In the legacy Live Odds feed, information about fixtures, player information (competitors), venues and so on, was provided via different XML configurations and added to the meta message for each match. Most of the XML configurations available in the legacy Live Odds feed have been made standard in the Unified Feed, but some have been made available via an API self-service. This enables clients to get the desired information whenever they want, without the additional hassle of making sure a certain configuration is enabled for a certain sport or match.

The Unified Feed provides a simplified fixtures change message that provides the necessary information on when/if fixtures change for an event, match, tournament happen. For additional information about fixtures outside of this i.e. Where the event is taking place, who are the competitors, league information, what coverage is included and so forth, the API is available for all such requests.

Here is an example of the simplified XML fixtures message available in the feed:

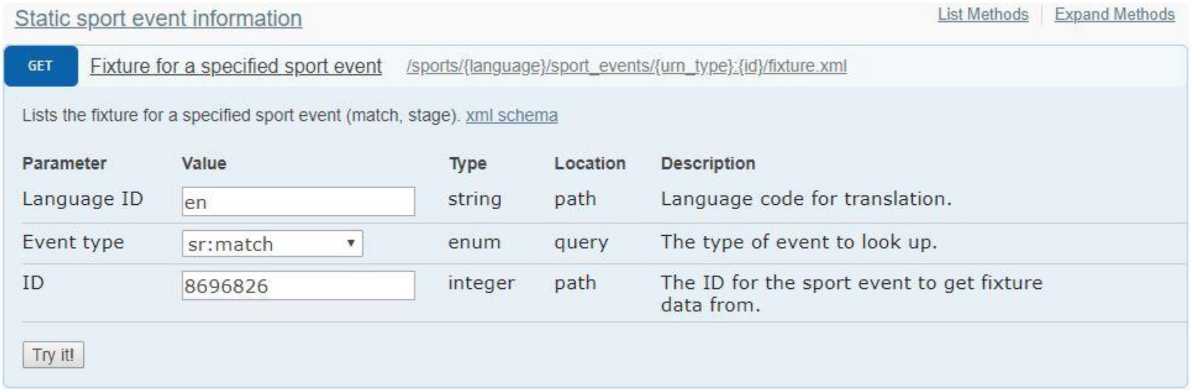
XML example 10 - Fixture change message in Unified feed

```
<fixture_change event_id="sr:match:14469596" product="1"
start_time="1527979500000" timestamp="1527291076723"/>
```

The above message includes the event ID for the match, which producer is delivering the odds for this patch (product attribute), and relevant timestamps. A minimal message with the essential information.

As mentioned before, additional information is available in our API. Getting access to our API is easy, simply follow the provided link at the top of this document, in the self-service documentation section, or click HERE.

In order to get information about fixtures, find the appropriate endpoint (/fixture.xml), and make your request by providing information in the fields:



Static sport event information

Fixture for a specified sport event /sports/{language}/sport_events/{urn_type}:{id}/fixture.xml

Lists the fixture for a specified sport event (match, stage). xml schema

Parameter	Value	Type	Location	Description
Language ID	en	string	path	Language code for translation.
Event type	sr:match ▾	enum	query	The type of event to look up.
ID	8696826	integer	path	The ID for the sport event to get fixture data from.

Figure 2 - fixture.xml endpoint

The next example will illustrate what an expanded fixtures information message could look like, with the provided information in the image above.

XML example 11 - Fixtures API response in Unified feed

```
<fixtures_fixture>
<fixture id="sr:match:8696826" scheduled="2016-10-31T18:00:00+00:00"
next_live_time="2016-10-31T18:00:00+00:00" start_time="2016-10-
31T18:00:00+00:00" start_time_confirmed="true">
    <tournament_round betradar_id="4301" type="group" number="25"
group_long_name="Division 1, Södra"/>
    <season id="sr:season:12346" name="Div 1, Sodra 2016" start_date="2016-
04-16" end_date="2016-11-05" year="2016" tournament_id="sr:tournament:68"/>
    <tournament id="sr:tournament:68" name="Div 1 Sodra">
        <sport id="sr:sport:1" name="Soccer"/>
        <category id="sr:category:9" name="Sweden" country_code="SWE"/>
    </tournament>
    <competitors>
        <competitor id="sr:competitor:1860" name="IK Oddevold"
country="Sweden" country_code="SWE" abbreviation="ODD" qualifier="home">
            <reference_ids>
                <reference_id name="betradar" value="1449434"/>
            </reference_ids>
        </competitor>
        <competitor id="sr:competitor:22356" name="Tvaakers IF"
country="Sweden" country_code="SWE" abbreviation="TVA" qualifier="away">
            <reference_ids>
                <reference_id name="betradar" value="7281378"/>
            </reference_ids>
        </competitor>
    </competitors>
    <extra_info>
        <info key="neutral_ground" value="false"/>
        <info key="period_length" value="45"/>
        <info key="overtime_length" value="15"/>
    </extra_info>
    <coverage_info level="bronze" live_coverage="false">
        <coverage includes="basic_score"/>
    </coverage_info>
    <product_info>
        <is_in_hosted_statistics/>
    </product_info>
    <reference_ids>
        <reference_id name="BetradarCtrl" value="11428313"/>
    </reference_ids>
</fixture>
</fixtures_fixture>
```

As you can see, when fixtures information is requested from the API it will provide more detailed information about the event, including sport-specific information, venue information, and more.

Markets, outcomes and specifiers

The odds change message is an important part of the XML feed, and this message looks fundamentally different in Unified Feed than it does in the legacy Live Odds feed.

Odds and markets in the change messages

When an odds change occurred in the legacy feed, it could look like the following example:

XML example 12 - Legacy markets

```
<BetradarLiveOdds status="change" timestamp="1287056116518">
<Match active="1" betstatus="started" clock_stopped="0" matchid="1467300"
matchtime="32" msgnr="31" remaining_time="8:49" score="13:10"
setscores="0:0 - 5:5 - 3:5" status="4q">

<Odds active="1" changed="false" combination="0" freetext="Asian Handicap"
id="748814" specialoddsvalue="3.5" subtype="34" type="
ft2w" typeid="7">
<OddsField active="1" type="1">1.75</OddsField>
<OddsField active="1" type="2">1.9</OddsField>
</Odds>

<Odds active="0" changed="true" combination="0" id="748905"
specialoddsvalue="87.5" type="to" typeid="5"/>
<Odds active="0" changed="false" combination="0" freetext="Asian handicap
for fourth period" id="748888" specialoddsvalue="0.5" subtype
="54" type="ft2w" typeid="7"/>
<Odds active="1" changed="true" combination="0" freetext="Asian handicap
for whole match, including overtime" id="748867"
specialoddsvalue="4.5" subtype="38" type="ft2w" typeid="7">
<OddsField active="1" type="1">1.85</OddsField>
<OddsField active="1" type="2">1.8</OddsField>
</Odds>

<!-- Additional markets... -->
</Match>
</BetradarLiveOdds>
```

In the code above we can see the same root element as before, followed by some match information (<match> element), and market/odds information found in the <Odds> and <OddsField> elements.

Betradar Unified Odds utilizes markets and market lines differently. Instead of the <Odds> and <OddsField> elements, each market is a bet type identified with a unique ID, and within a <market> element. Multiple different lines are often provided.

Each of these lines is uniquely identified by additional specifiers. Total Goals 2.5 is the same market as Total Goals 1.5, but it is two different market lines. The market ID for both are the same, but the first one has the specifier (total=3.5) and the other one has the specifier (total=2.5) that uniquely identifies them.

In the following example you can see the same total market with the same name and ID, but with different outcomes listed.

XML example 13 - Unified feed markets with market lines

```
<market name="Total" id="18" specifiers="total=3.5" status="1">
    <outcome name="over 3.5" active="1" id="12" odds="2.3"/>
    <outcome name="under 3.5" active="1" id="13" odds="1.55"/>
</market>

<market name="Total" id="18" specifiers="total=2.5" status="1">
    <outcome name="under 2.75" active="1" id="13" odds="2.1"/>
    <outcome name="over 2.75" active="1" id="12" odds="1.65"/>
</market>
```

In the feed this information is available in the odds change message just like it would be in the legacy Live Odds feed. A more complete odds change message example can be seen below:

XML example 14 - Odds change with outcomes and specifiers

```
<odds_change event_id="sr:match:1234" timestamp="1234" product="2">
<sport_event_status status=”1" reporting="1" match_status="1"
home_score="2” away_score=”0">
     <clock match_time=”10:00” remaining_time=”50:00” stopped=”true”/>
   </sport_event_status>
  <odds>
<market name="Total" id="18" specifiers="total=3.5" status="1">
    <outcome name="over 3.5" active="1" id="12" odds="2.3"/>
    <outcome name="under 3.5" active="1" id="13" odds="1.55"/>
</market>

<market name="Total" id="18" specifiers="total=2.75" status="1">
    <outcome name="under 2.75" active="1" id="13" odds="2.1"/>
    <outcome name="over 2.75" active="1" id="12" odds="1.65"/>
</market>
  </odds>
</odds_change>
```

Specifiers

In the legacy Live Odds feed there is a concept called Special Odds Value. The equivalent in Unified Odds are called specifiers. The Unified Odds specifiers is a cleaned up version of the special odds values that provides a uniform and descriptive way of specifying additional parameters that uniquely identifies a market (as seen in the examples above).

The differences here are that legacy Live Odds only sends the values separated with a single separator (usually a / symbol), while the Unified Feed sends a key/value pair separated by the | symbol. The following example is from a cricket match, where the over/under bet of 2.5 (total=2.5) is set to the 5th over in the cricket match, represented by the “overnr=5” value.

- Legacy special odds value: "5/2.5"
- Unified feed specifiers: "overnr=5|total=2.5"

If we look at the XML example provided earlier, we can see that the specifier identifies a market line within a market:

XML example 15 - Same market, different market lines

```
<market name="Total" id="18" specifiers="total=3.5" status="1">
    <outcome name="over 3.5" active="1" id="12" odds="2.3"/>
    <outcome name="under 3.5" active="1" id="13" odds="1.55"/>
</market>

<market name="Total" id="18" specifiers="total=2.5" status="1">
    <outcome name="under 2.75" active="1" id="13" odds="2.1"/>
    <outcome name="over 2.75" active="1" id="12" odds="1.65"/>
</market>
```

Here the total market is first identified with the name=”total” and id=”18”, but the specifier adds an additional identifier on top of this for each market line, separating the total=”2.5” from the total=”3.5”, even though they belong to the same market.

Bet stop

The concept of how bet stop and early bet stop works has been changed in Unified Feed. In the Legacy Live Odds service, the client systems got notified when a bet stop occurs, and once the markets were open again in the XML feed:

XML example 16 - Bet start and bet stop

```
Betstart
<BetradarLiveOdds xmlns="http://www.betradar.com/BetradarLiveOdds"
status="betstart" time="0" timestamp="1199435558847">
<Match active="1" betstatus="started" matchid="661373" msgnr="5" score="-:-
" status="not_started"/>
</BetradarLiveOdds>

Betstop
<BetradarLiveOdds xmlns="http://www.betradar.com/BetradarLiveOdds"
status="betstop" time="0" timestamp="1199435635925">
<Match active="1" betstatus="stopped" matchid="661373" msgnr="6" score="-:-
" status="not_started"/>
</BetradarLiveOdds>
```

In Unified Feed the same function is handled, but information about bet stop is sent in a bet_stop message. The bet_stop message is an optimized signal to indicate that all, or a set of markets should be instantly suspended (continue to display odds, but don't accept tickets).

XML example 17 - Betstop in Unified Feed

```
<bet_stop timestamp="12345" product="3" event_id="sr:match:471123"
groups="all”/>
```

Here we can see that a match has entered bet stop, and all markets are suspended, indicated by the groups=”all” attribute. This attribute should be a group-name, and “all” is special keyword simply saying that all markets should be suspended.

After the bet stop message has been received, the following odds change message will include the reason for the bet stop, and the status attribute for all effected markets will change to -1 (suspended):

XML example 18 - odds_change message after a betstop message

```
    <odds _betstop_reason_name="POSSIBLE_GOAL_HOME" betstop_reason="4">
     <market _name="2nd half - both teams to score" favourite="1" id="95"
status="-1">
            <outcome _name="yes" active="1" id="74" odds="5.0"/>
            <outcome _name="no" active="1" id="76" odds="1.14"/>
        </market>
        ... All markets listed ...
    </odds>
```

When the match returns to bet start the *betstop_reason* attribute will disappear from the odds element, and the markets will have the status attribute returned to 1 (active).

If you are in an early bet start situation the *betting_status* attribute will be removed from the odds element when going to normal bet start.

Bet cancel vs bet settlement messages

In Unified odds we have a direction to use *bet_settlement* messages with *void_factor = 1* when there is no error, but due to the results bets should be refunded. Whereas in Legacy Live Odds we normally do this with a *bet_cancel* message. So for Unified odds we are focusing on only using the *bet_cancel* messages for error conditions, when something is wrong and it needs correction.

Bet clearing

In the legacy Live Odds feed when an event happened in a match which defines the outcome of a specific odds type, the Betradar system sends out a clear bet message. Clear bet messages contain the information about the outcome of bets, and in some cases also whether the bet was voided or not. If the bet is partly voided, the voided part would (should) be refunded to the customer.

The XML example below illustrates a clear bet from the legacy Live Odds feed:

XML example 19 - Legacy feed clearbet

```
<BetradarLiveOdds status="clearbet" time="2" timestamp="1266483496355">
<Match active="1" betstatus="stopped" clearedscore="0:1" matchid="935448"
msgnr="19" status="paused">
<Odds active="1" combination="0" freetext="Asian total first half "
id="311538" specialoddsvalue="1.25" subtype="35" type="ftnw" typeid="
8">
<OddsField active="1" outcome="1" type="under" voidfactor="0.5"/>
<OddsField active="1" outcome="0" type="over" voidfactor="0.5"/>
</Odds>
</Match>
</BetradarLiveOdds>
```

In the above example you can see the same root element is used, the <BetradarLiveOdds>, but the status is set to status=”clearbet”, showing that this is a clear bet message for the following odds/market.

In the Unified Odds feed there are also clear bet messages, but these are now called bet settlements. Void factor still applies to these messages, but as with all the XML messages in Unified Feed, it uses the message type as the root element:

XML example 20 - Bet settlement in Unified feed

```
<bet_settlement event_id="sr:match:3432" product="1" certainty=”1”
timestamp="1235">
   <outcomes>
      <market id="13">
        <outcome id="1" result="1"/>
        <outcome id="2" result="0"/>
        <outcome id="3" result="0" void_factor="1"/>
      </market>
   </outcomes>
</bet_settlement>
```

In the above example you can see the <bet_settlement> root element coming from product=”1” (this is the Live Odds producer), followed by the outcomes with results and void factor.

Please note that all the possible combinations of outcome and void factor are still the same in Unified Feed as they were in the legacy Live Odds feed:

- outcome="0" and no voidfactor: Lose entire bet
- outcome="1" and no voidfactor: Win entire bet
- outcome="0" and voidfactor="1": Refund entire bet
- outcome="1" and voidfactor="0.5": Refund half bet and win other half
- outcome="0" and voidfactor="0.5": Refund half bet and lose other half

Market mapping

If a market description is requested from the API, it is possible to include the market mappings that say what Live Odds or Life-cycle of Odds market IDs that are the same as in Unified Odds (or closest to).

This is reflected by adding an additional parameter to the market description request: *include_mappings=”true”*

XML configurations

The legacy Live Odds feed provided a wide range of manual configurable options for the XML feed. Most of these configurations have been made standard in Unified Feed, and require little to no time investment to manage on the client side. Below you can find some examples of commonly used XML configurations that have been made standard in Unified Feed.

Match time information

In multiple sports in the legacy Live Odds feed there are XML configurations that have to be enabled in order to receive information about match time (remaining time in period, remaining time of match, etc.). These configurations have been merged together into a <Clock> element in Unified Feed. This <Clock> element can be found inside the <sport_event_status> element, illustrated below:

XML example 21 - Match time information in Unified feed

```
<sport_event_status _match_status_name="1st half" away_score="0"
away_suspend="0" home_score="6" home_suspend="0" match_status="6"
reporting="1" status="1">

        <clock match_time="3:56" remaining_time="56:04"
remaining_time_in_period="26:04" stopped="true"/>

</sport_event_status>
```

This element can also include optional information about stoppage times and when they are announced in i.e. Soccer.

Period scores

In order to get period scores in the legacy Live Odds feed, an XML configuration would have to be enabled in the configuration. This feature is now standard across Unified Feed, and is found in the <sport_event_status> element as illustrated below:

XML example 22 - Period scores

```
<sport_event_status _match_status_name="1st half" away_score="0"
away_suspend="0" home_score="6" home_suspend="0" match_status="6"
reporting="1" status="1">

        <period_scores>
            <period_score away_score="0" home_score="6"
match_status_code="6" number="1"/>
        </period_scores>

</sport_event_status>
```

Mark most balanced odds

Because of the way the odds change message has changed in Unified Feed, so has the concept of marking the most balanced odds for a market. In the legacy feed this was a configuration that had to be enabled by the client, and it would add an addition attribute to the <Odds> element.

This has been changed in Unified Odds, using a new attribute called *favorite*, available in the <market> element found in *odds_change* messages. This attribute indicates which market line is currently the most balanced. One reason for this change is that it is not always the most balanced line that is preferred. In some cases, our customers prefer another line than the most balanced line, hence the name change to *favorite*.

XML example 23 - Marked most balanced odds in Unified feed

```xml
<market id="47" specifiers="score=41.5" favorite="1" status="1">
  <outcome id="1" odds="1.12" active="1"/>
  <outcome id="2" odds="1.92" active="1"/>
</market>
<market id="48" specifiers="score=42.5">
  <outcome id="1" odds="1.12" active="1"/>
  <outcome id="2" odds="1.92" active="1"/>
</market>
```

Because of the way markets and outcomes are made available in Unified Feed, the previous legacy feed configuration “Send multiple bets of the same type” is also made standard.