

Unabated API Reference

Updated 1/3/24

Overview

The Unabated API provides programmatic access to all of Unabated's Game Odds and Props market, game state, and scoring data. Some key features and points of the API are:

1. A "snapshot" endpoint to retrieve the current lines for all markets across all sports that Unabated offers. This includes all sportsbook, team, and player metadata.
2. A "changes" endpoint to retrieve market line changes that occurred since a specified timestamp.
3. No additional data latency beyond that which we have on our backend as we pull information from sportsbooks. These latency categories are specified in the sportsbook metadata, and can also be seen in our Game Odds and Props Odds UI as colored dots in the grid headers:
 - a. Green = Real-time
 - b. Yellow = Under 30 seconds
 - c. Orange = Over 30 seconds
4. There are no call limits or API throttling, but you should not call these endpoints more than 1 time per second as there will be no change in information during that time and would just be wasted compute time on our servers and your client.

You can leverage the endpoints to serve your needs. If you want to be staying on top of every market movement you would use a combination of the Snapshot Endpoint and Changes Endpoint as described in the section below. Or, if you are just interested in taking occasional snapshots of the markets you can simplify your implementation by just calling the Snapshot Endpoint. If you want to get the data more frequently than every 30 seconds then for performance reasons you should use the Snapshot Endpoint to get the initial odds and metadata coupled with Changes Endpoint to get ongoing changes. You can also use the Snapshot Endpoint as a refresh point for example in a case where your processing was interrupted and you need to get back to the current market state.

Accessing Unabated Game Odds

Snapshot Endpoint

A complete snapshot of current game odds can be retrieved from the following URL via a GET request. Note that authentication is required here; substitute the API key you received from the Unabated team in place of *YOUR_API_KEY*:

Game Odds Markets

Dev/Test: https://dev-partner-api.unabated.com/api/markets/gameOdds?x-api-key=YOUR_API_KEY

Production: https://partner-api.unabated.com/api/markets/gameOdds?x-api-key=YOUR_API_KEY

Props Markets

Dev/Test: https://dev-partner-api.unabated.com/api/markets/playerProps?x-api-key=YOUR_API_KEY

Production: https://partner-api.unabated.com/api/markets/playerProps?x-api-key=YOUR_API_KEY

The structure of the data returned from this URL is described in detail in the section of this document titled **Full Game Odds JSON Description**.

To identify which team is favored in your Game Odds calls, check the `gameOddsMarketSourcesLines` object key for values “si0” or “si1”. For example, “si1:ms1:an0” is the section of JSON for the Home team for DraftKings (ms1).

Changes Endpoint

After retrieving odds from this URL, you can retrieve incremental updates to those odds on an ongoing basis. The following URL will retrieve the subset of odds that have changed within the last several seconds. Note that authentication is required here as well:

Game Odds Markets

Dev/Test: https://dev-partner-api.unabated.com/api/markets/changes?x-api-key=YOUR_API_KEY

Production: https://partner-api.unabated.com/api/markets/changes?x-api-key=YOUR_API_KEY

Props Markets

Dev/Test: https://dev-partner-api.unabated.com/api/markets/playerProps/changes?x-api-key=YOUR_API_KEY

Production: https://partner-api.unabated.com/api/markets/playerProps/changes?x-api-key=YOUR_API_KEY

The structure of the data returned from this URL is described in detail in the section of this document titled **Incremental Game Odds JSON Description**.

One of the top-level fields in the response is a number, `lastTimestamp` (note: this number is *not* interpretable as a Unix timestamp). You should add this timestamp to the [changes](#) URL to get all the changes that have occurred since the last time you called it, like so:

Dev/Test: https://dev-partner-api.unabated.com/api/markets/changes/68327798229687600?x-api-key=YOUR_API_KEY

Production: https://partner-api.unabated.com/api/markets/changes/68327798229687600?x-api-key=YOUR_API_KEY

Putting this together, you can construct a program that continuously updates odds from Unabated by following these steps:

1. Call [gameOdds](#) to get an initial full snapshot of odds for all events currently on the calendar for that day and several upcoming days.
2. Call [changes](#) with no arguments to get an initial set of incremental changes.
3. Continue calling [changes](#) in a loop, adding the top-level `lastTimestamp` value seen in the previous response from [changes](#).
4. Repeat step 3 for as long as you wish to keep getting updated odds, no more than once per

second.

Notes:

- Unabated updates odds approximately once per second. Therefore, you should avoid calling [changes](#) more frequently than once per second from your program.
- If your program is paused for a while, the `lastTimestamp` value you have may become stale. In that event, [changes](#) will not be able to return incremental changes. It will fail and indicate this with a top-level `resultCode` value of "Failed". When this happens, you will need to restart the process from step 1.
- Generally, the timespan that Point 2 covers is 30s. However, due to some caching mechanisms we have in place on the backend to keep it performant that could be lower when market change volume increases. We recommend keeping the polling interval in Point 3 reasonably frequent (15s or less) so that you won't miss any changes.

Accessing Unabated Game Odds

You should use the Dev/Test endpoints listed above while initially developing your application, and switch to the Production endpoints when you are ready to do user acceptance testing prior to taking your application live.

Note that the Dev/Test endpoints may not offer odds for all of the same games available from the Production environment. Also, these environments use different databases, so database IDs will differ. Therefore, your application should not mix data received from both of these environments.

Full Game Odds JSON Description

Here is a more complete description of the JSON object returned from [_gameOdds](#).

At the highest level the Game Odds JSON has sections for:

- Market Sources
- Teams
- Game Odds Events

Also implied in this JSON are Market and Market Line objects. Following is a description of all these sections/objects.

Market Object

The concept of a Market is a unique combination of the following things:

- Event
- Side (Away/Home, Over/Under)
- Bet Type
- Period Type

- Alternate Number (for future use, not used at this time)
- Pregame/Live

Market Line Object

A Market Line is a unique combination of:

- Market
- Market Source (typically a sportsbook or exchange)

A detailed field description of the JSON associated with a market line is provided below.

Market Sources Section

Market Sources are typically sportsbooks or exchanges, but could be a consensus line as well (the Unabated Line, for example, which is included in the product). The detailed market source data can be found in the `marketSources` section.

Teams Section

The `teams` section includes detailed team data that is referenced by team id in the `gameOddsEvents` section.

Game Odds Events Section

The `gameOddsEvents` section is where you can find all the market data for all the market sources. It is a keyed object of this format:

```
lg{league-id}:pt{period-type-id}:{pregame/live}
```

For example, all the NFL first half pre-game lines would be under the key `lg1:pt2:pregame`. And all the NBA in-game full game lines would be under the key `lg3:pt1:live`.

Within each of these keys is an array of Game Odds Event objects which hold the information for every game and market line.

Game Odds Event Object Fields

Field	Type	Description
eventId	Integer	The unique static key of the event/game
eventStart	Datetime	The scheduled start time of the game (in UTC)
eventEnd	Datetime	The end time of the game (in UTC). For future use.

statusId	Integer	<p>Enumeration describing the state of the event:</p> <p>1: Pre-game 2: Live 3: Final 4: Delayed 5: Postponed 6: Cancelled</p>
gameClock	String	<p>Contains a description of the game state. Typically the game clock.</p> <p>Example: 12:45 1H</p>
periodTypeId	Integer	Period Type (see Period Types table for valid values)
overtimeNumber	Integer	<p>A number indicating how many overtime periods the game went to.</p> <p>Examples: An MLB game that goes to the 12th inning will have the number 12. A CFB game which goes into 3 overtimes will have a 3.</p>
eventTeams	Keyed Object	Key: Side Index (see Side Index table for valid values)

		<p>Value: An Event Team object, which has the following fields:</p> <p><code>id</code>: The team id from the Teams section of the top-level JSON.</p> <p><code>rotationNumber</code>: The standard Don Best id that is used by brick-and-mortar sportsbooks to identify a betting line.</p> <p><code>score</code>: The number of points that team scored in the game.</p>
<code>gameOddsMarketSourceLines</code>	Keyed Object	<p>Key: Format described below</p> <p>Value: A Market Line object (see description in “Market Line Object Fields” below)</p>

The market line information is in the `gameOddsMarketSourceLines` section as a keyed object with the following key structure:

```
si{side-index}:ms{market-source-id}:an{alternate-line-index}
```

Within that is another keyed object that is formatted as `bt{bet-type-id}`. For example, a Pinnacle spread line for the home team would look something like this:

```
"si1:ms7:an0": {
  "bt2": {
    Market Line Object here
  }
}
```

Note that the alternate line index is not in use at this time, and the only relevant key right now is `an0`. If you find any others in the data you can ignore it.

Market Line Object Fields

Field	Type	Description
-------	------	-------------

sourceFormat	Integer	Enumeration of odds format types: 1 = American 2 = Decimal 3 = Percent 4 = Probability 5 = Sporttrade (0 to 100)
alternateNumber	Integer	For future use. Should always be 0.
statusId	Integer	Describes if the market line is on/off the board: 1: Available 2: Unavailable
sequenceNumber	Integer (64-bit)	Unique key based on timestamp indicating the order

		of when the market line change came in.
overrideType	String	For internal use only
includePeg	Boolean	For internal use only
stn	Decimal	For internal use only
cr	Decimal	For internal use only
bacr	Decimal	For internal use only
ib	Boolean	For internal use only
tm	Timestamp	For internal use only
createdOn	Datetime	When the market line was first created (in UTC)

createdBy Integer For internal use only When the market line was last
modifiedOn Datetime modified (in UTC)
modifiedBy Integer For internal use only id Integer For internal use only

Reference Tables

Leagues

League Id	League Name
1	NFL
2	CFB
3	NBA
4	CBB
5	MLB
6	NHL
7	WNBA

Period Types

Period Type Id	Period Type Name
1	Full Game

2 First Half

3	Second Half
4	First Quarter
5	Second Quarter
6	Third Quarter

7	Fourth Quarter
8	First Period
9	Second Period
10	Third Period
11	First Inning
12	Second Inning
13	Third Inning
14	Fourth Inning
15	Fifth Inning
16	Sixth Inning
17	Seventh Inning
18	Eighth Inning

19 Ninth Inning

20 First Five Innings

Bet Types

Bet Type Id	Bet Type Name	Market Type
1	Moneyline	Game Odds
2	Spread	Game Odds
3	Total	Game Odds
4	Team Total	Game Odds
5	Division Winner	Futures
6	Conference Winner	Futures

7	Super Bowl Winner	Futures
---	-------------------	---------

8 Most Wins Futures

9 Number Of Wins Futures

10 Make Playoffs Futures

11	Rushing Attempts	Props
12	Rushing Yards	Props
13	Passing Completions	Props
14	Passing Yards	Props
15	Receptions	Props
16	Receiving Yards	Props
17	Pitcher Strikeouts	Props
18	Home Runs	Props
19	Total Bases	Props
20	Conference Top Seed	Futures
21	Conference Wild Card	Futures
22	Fewest Wins	Futures
23	Win All Games (17-0)	Futures
24	Lose All Games (0-17)	Futures
25	Last Undefeated	Futures
26	Most Points Scored	Futures

27	Fewest Points Scored	Futures
28	NFL MVP	Futures
29	NFL Superbowl MVP	Futures
30	NFL Offensive Player of the Year	Futures
31	NFL Defensive Player of the Year	Futures
32	NFL Comeback Player of the Year	Futures
33	NFL Offensive Rookie of the Year	Futures
34	NFL Defensive Rookie of the Year	Futures

35	NFL Most Passing Yards	Futures
36	NFL Most Passing Touchdowns	Futures
37	NFL Most Rushing Yards	Futures
38	NFL Most Rushing Touchdowns	Futures
39	NFL Most Receiving Yards	Futures
40	NFL Most Receiving Touchdowns	Futures

41	NFL Most Passing Interceptions	Futures
42	NFL Most Sacks	Futures
43	NFL Season Passing Yards	Futures
44	NFL Season Passing Touchdowns	Futures
45	NFL Season Rushing and Receiving Yards	Futures
46	NFL Season Receiving Yards	Futures
47	NFL Season Receiving Touchdowns	Futures
48	NFL Season Receptions	Futures
49	NFL Season Rushing Yards	Futures
50	NFL Season Rushing Touchdowns	Futures
51	NFL Season Interceptions	Futures
52	NFL Season Sacks	Futures
53	NFL Most Receiving Receptions	Futures

54	NFL Season Passing Interceptions	Futures
----	----------------------------------	---------

55	Defensive Interceptions	Props
56	Sacks	Props
57	Tackles and Assists	Props
58	Extra Points Made	Props
59	Field Goals Made	Props
60	Total Kicking Points	Props
61	Passing Attempts	Props
62	Interceptions Thrown	Props
63	Longest Pass Completion	Props
64	Passing and Rushing	Props
65	Yards Passing	Props
	Touchdowns	
66	Longest Reception	Props
67	Longest Rush	Props
68	Rushing and Receiving Yards	Props
69	Three Pointers Made	Props
70	Assists	Props
71	Blocks	Props
72	Double Double	Props

73	Points	Props
74	Points Assists	Props
75	Points Rebounds	Props
76	Points Rebounds Assists	Props
77	Rebounds	Props
78	Rebounds Assists	Props

79	Score First Field Goal	Props
80	Score Most Points	Props
81	Steals	Props
82	Steals Blocks	Props
83	Triple Double	Props
84	Turnovers	Props

Side Index

Side Id	Side Description
0	Away Team or Over
1	Home Team or Under

Incremental Game Odds JSON Description

Here is a more complete description of the JSON object returned from [changes](#).

Field	Type	Description		
lastTimestamp	Number	A value that can be used to poll for the next set of changes.		
resultCode	String ("Success" or "Failed")	Indicates whether the request was successful.		
results	Array	Contains recent changes to game odds. The elements of the array are ordered from oldest to most recent, so they should be processed in the order they were returned.		
		Each element of this array is a JSON object with the following fields (you will see other fields besides these, but they can be ignored):		
		Field	Type	Description
		marketSources	Array	Any market sources that have recently changed. It is relatively rare for this array to be non-empty. When it is, it is usually because a market source has been temporarily disabled or re-enabled by Unabated.
		events	Array	Any events that have recently changed. The most common changes to events are to their pregame/live/final status and score/time remaining.
gameOdds	Object	Any odds that have recently changed. This object has the same structure as detailed in the <u>Game Odds Events Section</u> portion of this document.		

