Kaiko

MIGRATION GUIDE

Spot – OB Depth – Average Depth

HTTP REST

Amberdata → Kaiko

2026

www.kaiko.com

1 At a glance

What this endpoint gives you. Average order book depth per level over an interval: the average quote value and number of orders resting within given distances of the best price, on each side of a spot order book.

What changes for you

- `timeInterval` **becomes** `interval`: `hour` / `day` become `1h` / `1d`. Default `1d` on Kaiko, `hour` on Amberdata.
- **One row per interval, both sides as columns.** Amberdata returns one row per side (`side`); Kaiko returns both sides in the same row with `ask_` / `bid_` prefixes.
- **Same 20 depth levels** (5, 10, 15, 20, 25, 50, 75, 100, then every 25 up to 500 bps), with suffixes such as `_5bps` instead of `005`.
- **Units are unchanged:** quote value in millions (`usd_million`), size in base asset, order counts.
- **Reference price:** Kaiko measures levels from the mid price by default; add `slippage_ref`=`best` to match Amberdata (best bid / best ask).
- **Kaiko adds the average size in base asset** (`volume`) per level; order counts are whole numbers.
- **Exchange and instrument move into the URL path**; the market (`{instrument_class}`) becomes explicit (see §2–3).

Coverage & distribution

Data type: Order book aggregation (depth per level)

Markets: CEX spot

Granularity: Average per hour or day on Amberdata (`timeInterval`); per `interval` on Kaiko, default `1d`

Distribution: REST API

2 Endpoint & authentication

	Amberdata (today)	Kaiko (new)
Method	GET	GET
Base URL	`https://api.amberdata.com/markets/spot/analytics`	`https://us.market-api.kaiko.io` (or `eu.market-api.kaiko.io`)
Path	`/depth/average-time-series`	`/v2/data/order_book_snapshots.v1/exchanges/{exchange}/{instrument_class}/{instrument}/ob_aggregations/average_depth`
Auth	API key header `x-api-key`	API key header `X-Api-Key` (same mechanism, new key)

Kaiko API family: Classic (versioned)

This endpoint belongs to Kaiko's existing versioned API. Its URL structure and response envelope differ from Kaiko's New API grammar endpoints (`market-data`), such as OHLCV Batch and Tickers.

3 Request parameters

Amberdata param	Kaiko param	Change	Notes
`exchange` (query, required)	`{exchange}` (path)	RENAMED	Moves to the URL path. Kaiko exchange code, e.g. `gdax` → `cbse` (see §6).
—	`{instrument_class}` (path)	NEW	Market: `spot`.
`pair` (query, required)	`{instrument}` (path)	RENAMED	Moves to the URL path. Kaiko instrument code, e.g. `btc_usd` → `btc-usd` (see §6).
`startDate` (optional)	`start_time`	RENAMED	Inclusive lower bound on both sides. Amberdata: seconds, milliseconds or ISO 8601. Kaiko: ISO 8601. Optional; automatically included in continuation tokens.
`endDate` (optional)	`end_time`	RENAMED	Exclusive upper bound on both sides. ISO 8601 on Kaiko.
`timeInterval` (required)	`interval`	RENAMED	Amberdata: `hour` (default) or `day`. Kaiko: e.g. `1h`, `1d` (default). Any duration from 1 second to 1 day, suffixed `s`, `m`, `h` or `d` (e.g. `30s`, `15m`, `4h`, `1d`); maximum 1 day.
`timeFormat` (optional)	—	REMOVED	Kaiko returns Unix milliseconds.
—	`slippage_ref`	NEW	**Reference price of the depth levels.** Default `mid_price`; set `best` to measure levels from the best bid / best ask, as Amberdata does.
—	`slippage`	NEW	Shared order book aggregation parameter, echoed in `query` (default `0`); not used for depth.
—	`page_size`	NEW	Optional. Rows per page; default 10.
—	`sort`	NEW	Optional. `asc` or `desc` (default).
—	`continuation_token`	NEW	Pagination; use `next_url` as returned.

4 Response envelope & metadata

The fields around the data. The Amberdata envelope is **identical on every Amberdata endpoint**; the Kaiko envelope is the one shared by Kaiko's order book snapshot endpoints.

Amberdata field	Kaiko field	Change	Notes
`status` / `title` / `description`	`result`	RENAMED	Single string (`"success"`) instead of the HTTP-style status object. Error handling must be rewritten.
`payload.data`	`data`	RENAMED	Array holding the snapshot rows (see §5), now at the top level.

Amberdata field	Kaiko field	Change	Notes
`payload.metadata.api-version`	`query.data_version`	RENAMED	Data version (`v1`), inside the echoed `query` object.
—	`query`	NEW	Echo of the request, including the defaults applied (`sort`, `page_size`, `slippage_ref`...).
—	`time`, `timestamp`	NEW	Server response time (ISO 8601 and Unix milliseconds).
—	`continuation_token`, `next_url`	NEW	Pagination: call `next_url` for the next page.
—	`access`	NEW	Time range your API key can access (`access_range`) and data range (`data_range`).

5 Response - data fields

One row per interval. `{X}` = level in basis points, Kaiko suffix `_{X}bps` (e.g. `_5bps`), Amberdata suffix on three digits (e.g. `005`). Levels: 5, 10, 15, 20, 25, 50, 75, 100, then every 25 up to 500 bps.

Amberdata field	Kaiko field	Change	Notes
`exchange`, `pair`	—	REMOVED	Not repeated per row; recoverable from the request path or `query`.
`side`	—	REMOVED	Kaiko returns both sides in each row, prefixed `ask_` / `bid_`.
`timestamp`	`poll_timestamp`	RENAMED	Unix milliseconds on Kaiko; timestamp of the aggregation interval (midnight UTC for `1d` in the sample).
`basisPoints{XXX}UsdMillion`	`ask_usd_million_{X}bps`, `bid_usd_million_{X}bps`	RENAMED	Average depth in quote value over the interval, **in millions on both sides**; decimal string.
`basisPoints{XXX}OrderCount`	`ask_order_count_{X}bps`, `bid_order_count_{X}bps`	RENAMED	Average number of orders. **Decimal on Amberdata, integer on Kaiko.**
—	`ask_volume_{X}bps`, `bid_volume_{X}bps`	NEW	Average depth size in base asset; decimal string. Not returned by Amberdata's average endpoint.

6 Exchange & instrument names

Kaiko identifies venues and instruments with its own normalized codes, which differ from Amberdata's names. Codes can also differ by market for the same venue (e.g. Binance spot and options).

Reference

The mapping rules and the full code lists apply to every endpoint and are described in the **Exchange, Asset & Instrument Mapping** migration guide (`exchange-instrument-asset-codes.pdf`).

7 Will your values change?

Short answer

Both providers average order book depth over the chosen interval at the same 20 levels and in the same units, so values should be close for matching intervals.

- **Interval.** Amberdata defaults to `hour`, Kaiko to `1d`. Set `interval` explicitly (`1h` or `1d`).
- **Order counts.** Kaiko returns whole numbers; Amberdata returns decimal averages.
- **Reference price.** Kaiko measures levels from the mid price by default, Amberdata from the best bid / best ask. Use `slippage_ref` = `best` to compare like with like; the narrow levels change most.
- **Book depth captured.** Depth is aggregated within 0–10% of the best bid or ask. Kaiko cannot collect the full 10% from every exchange it covers: on some exchanges, depth figures do not fully portray the order book volume.
- **Number representation.** Kaiko returns sizes and values as decimal strings.

8 Before / after example

Request

Amberdata

```
GET https://api.amberdata.com/markets/spot/
analytics
    /depth/average-time-series
    ?exchange=gdax
    &pair=btc_usd
    &timeInterval=hour
x-api-key: <your key>
```

Kaiko

```
GET https://us.market-api.kaiko.io
    /v2/data/order_book_snapshots.v1
    /exchanges/cbse/spot/btc-usd
    /ob_aggregations/average_depth
    ?interval=1h
    &slippage_ref=best
X-Api-Key: <your key>
```

Response

Amberdata: example from its documentation (shortened). Kaiko: real sample with the default `1d` interval. Different dates, so values are not comparable.

Amberdata


```
{
  "status": 200,
  "title": "OK",
  "description": "Successful request",
  "payload": {
    "data": [
      {
        "basisPoints005OrderCount":
50.11585807385952,
        "basisPoints005UsdMillion":
0.8164277240123664,
        ... (010 to 500)
        "exchange": "gdax",
        "pair": "btc_usd",
        "side": "ask",
        "timestamp": "2025-03-11 00:00:00 000"
      },
      ...
    ]
  }
}
```


Kaiko


```
{
  "query": {
    "interval": "1d",
    "aggregation": "average_depth",
    "exchange": "cbse",
    "instrument_class": "spot",
    "instrument": "btc-usd",
    "slippage": 0,
    "slippage_ref": "mid_price",
    "sort": "desc",
    "page_size": 10,
    ...
  },
  "data": [
    {
      "ask_order_count_5bps": 57,
      "ask_usd_million_5bps":
"0.8317582629377657",
      "ask_volume_5bps": "10.693793164306156",
      "bid_order_count_5bps": 54,
      "bid_usd_million_5bps":
"0.8403206932929316",
      "bid_volume_5bps": "10.809089352070483",
      ... (all 20 levels, ask and bid)
      "poll_timestamp": 1793491200000
    },
    ...
  ],
  "result": "success",
  ...
}
```