Kaiko

MIGRATION GUIDE

# Spot – OB Depth – LWAP

HTTP REST

Amberdata → Kaiko

2026

www.kaiko.com

## 1 At a glance

**What this endpoint gives you.** Liquidity-weighted average price (LWAP): the average execution price obtained by sweeping the resting orders of a spot order book up to a given depth from the best bid / best ask, with the related slippage and executed size, at four depth levels.

What changes for you

- **One row per snapshot, both sides as columns.** Amberdata returns one row per side ( `side` = `ask` or `bid` ); Kaiko returns both sides in the same row, with `ask_` and `bid_` prefixes (see §5).
- **Same four depth levels, new naming.** Amberdata suffixes in basis points ( `005` , `025` , `050` , `100` ) become percent suffixes on Kaiko ( `0_05` , `0_25` , `0_5` , `1` ).
- **Units change: adaptation required.** Slippage and spread are percentages on Amberdata and **decimal fractions** on Kaiko (divide Amberdata by 100). Executed size is in millions on Amberdata and **in units** on Kaiko (multiply Amberdata by 1,000,000).
- **Three Amberdata field groups have no Kaiko equivalent:** slippage in USD, depth size in base asset and order count (see §5).
- **Exchange and instrument move into the URL path**; dates and pagination are query parameters (see §2–3). Snapshots every 30 seconds on Kaiko, 1 minute on Amberdata.

### Coverage & distribution

**Data type:** Order book snapshot analytics (LWAP)

**Markets:** CEX spot

**Snapshot frequency:** Kaiko: point-in-time snapshots, at least one per minute (every 30 seconds in the sample). Amberdata: 1 minute

**Pagination:** Kaiko: 100 snapshots per page, newest first

**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/lwap` | `/v2/data/order_book_snapshots.v1/exchanges/{exchange}/{instrument_class}/{instrument}/snapshots/lwap` |
| **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. |
| `timeFormat` (optional) | — | REMOVED | Amberdata default `hr` (human-readable). Kaiko returns both Unix milliseconds and ISO 8601 (see §5). |
| — | `page_size` | NEW | Optional. Snapshots per page; default 100. |
| — | `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. |
| `payload.metadata.api-version` | `query.data_version` | RENAMED | Data version ( `v1` ), inside the echoed `query` object. |

| Amberdata field | Kaiko field | Change | Notes |
|---|---|---|---|
| — | `query` | NEW | Echo of the request, including the defaults applied ( `start_time` , `end_time` , `page_size` , `sort` ). |
| — | `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 snapshot. `{XXX}` = Amberdata level in basis points; `{L}` = Kaiko level in percent. All Kaiko values are decimal strings.

| 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` , `poll_date` | RENAMED | Snapshot time: Unix milliseconds and ISO 8601 on Kaiko. |
| `bestBidPrice` , `bestAskPrice` | `best_bid` , `best_ask` | RENAMED | Number → decimal string. |
| `midPrice` | `mid_price` | RENAMED | Number → decimal string. |
| `spreadPercent` | `spread_percent` | RENAMED | **Adaptation required.** Percentage on Amberdata, decimal fraction on Kaiko: Kaiko = Amberdata / 100. |
| `lwap{XXX}` | `ask_lwap{L}` , `bid_lwap{L}` | RENAMED | LWAP for each depth level; `null` when the book lacks the liquidity to reach it. Levels: `005` → `0_05` , `025` → `0_25` , `050` → `0_5` , `100` → `1` . |
| `lwap{XXX}` `ExecutionVersusBBO` | `ask_slippage{L}` , `bid_slippage{L}` | RENAMED | **Adaptation required.** Slippage of the LWAP versus the best ask (ask side) or best bid (bid side). Percentage on Amberdata, decimal fraction on Kaiko: Kaiko = Amberdata / 100. |

| Amberdata field | Kaiko field | Change | Notes |
|---|---|---|---|
| `lwap{XXX}UsdMillion` | `ask_lwap{L}_usd`, `bid_lwap{L}_usd` | RENAMED | **Adaptation required.** Executed size, **normalised to USD on Kaiko** (Amberdata: in quote currency, often USD): millions on Amberdata, units on Kaiko (Kaiko = Amberdata × 1,000,000). |
| `slippage{XXX}UsdVersusBBO` | — | REMOVED | No Kaiko field. |
| `basisPoints{XXX}` | — | REMOVED | Depth size in base asset; no Kaiko field on this endpoint. |
| `basisPoints{XXX}OrderCount` | — | REMOVED | Order count; no Kaiko field on this 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 sweep the order book from the best bid / best ask at the same four depth levels, so LWAP values should be close once rows are aligned on time and units are converted. Differences come from snapshot timing (30 seconds on Kaiko, 1 minute on Amberdata) and from the depth of order book each provider captures.

- **Unit conversion.** Divide Amberdata slippage and spread by 100; multiply Amberdata executed size by 1,000,000. Without this, values look 100 times or a million times off.
- **Row layout.** Merge Amberdata's ask and bid rows for the same timestamp to compare with one Kaiko row.
- **Snapshot timing.** Kaiko snapshots are 30 seconds apart; align on the nearest time.
- **Insufficient liquidity.** When the order book does not hold enough liquidity to reach a depth level, Kaiko returns `null` for that level, as on its other order book snapshot endpoints. Handle `null` explicitly.
- **Number representation.** Kaiko returns decimal strings; compare as decimals.

## 8 Before / after example

### Request

**Amberdata**

```
GET https://api.amberdata.com/markets/spot/
analytics
    /depth/lwap
    ?exchange=gdax
    &pair=btc_usd
    &startDate=2025-03-15
    &endDate=2025-03-16
x-api-key: <your key>
```

**Kaiko**

```
GET https://us.market-api.kaiko.io
    /v2/data/order_book_snapshots.v1
    /exchanges/cbse/spot/btc-usd
    /snapshots/lwap
    ?start_time=2025-03-15T00:00:00Z
    &end_time=2025-03-16T00:00:00Z
X-Api-Key: <your key>
```

### Response

Amberdata: example from its documentation (shortened). Kaiko: real sample, request without parameters. Different dates, so values are not comparable.

**Amberdata**


```
{
  "status": 200,
  "title": "OK",
  "description": "Successful request",
  "payload": {
    "data": [
      {
        "exchange": "gdax",
        "pair": "btc_usd",
        "side": "ask",
        "timestamp": "2025-03-15 00:27:00 000",
        "bestAskPrice": 84160.01,
        "bestBidPrice": 84160,
        "midPrice": 84160.005,
        "spreadPercent": 0.0000118821,
        "lwap005": 84182.63687387163,
        "lwap005ExecutionVersusBBO": 0.0268855408,
        "lwap005UsdMillion": 0.4667357172,
        "slippage005UsdVersusBBO": 12548.4421928,
        "basisPoints005": 5.54432285,
        "basisPoints005OrderCount": 61,
        ... (same fields for 025, 050, 100)
      }
    ]
  }
}
```


**Kaiko**


```
{
  "query": {
    "exchange": "cbse",
    "instrument_class": "spot",
    "instrument": "btc-usd",
    "start_time": "2026-10-01T11:22:03.152Z",
    "end_time": "2026-11-01T11:22:03.152Z",
    "page_size": 100,
    "sort": "desc",
    "aggregation": "lwap",
    "data_version": "v1",
    ...
  },
  "time": "2026-11-01T11:22:03.314851167Z",
  "timestamp": 1793532123314,
  "data": [
    {
      "poll_timestamp": 1793532093263,
      "poll_date": "2026-11-01T11:21:33.263Z",
      "best_bid": "77853.05",
      "best_ask": "77853.06",
      "mid_price": "77853.055",
      "spread_percent": "0.000000128447",
      "ask_lwap0_05": "77870.65796179963",
      "ask_slippage0_05":
"0.00022604072080961886",
      "ask_lwap0_05_usd": "819733.2053252384",
      ... (ask 0_25, 0_5, 1)
      "bid_lwap0_05": "77824.96077336988",
      "bid_slippage0_05":
"0.00036079802435644396",
      "bid_lwap0_05_usd": "590701.4667858299",
      ... (bid 0_25, 0_5, 1)
    },
    ...
  ],
  "result": "success",
  "continuation_token": "...",
  "next_url": "https://us.market-
api.kaiko.io/...",
  "access": { ... }
}
```