Kaiko

MIGRATION GUIDE

Spot – Trade Analytics – VWAP / TWAP

HTTP REST

Amberdata → Kaiko

2026

www.kaiko.com

1 At a glance

What this endpoint gives you. Volume-weighted (VWAP) and time-weighted (TWAP) average prices of a spot pair on one exchange, per interval.

What changes for you

- **Two Kaiko endpoints replace one Amberdata endpoint.** Amberdata returns `vwap` and `twap` in the same row; Kaiko serves them separately (`aggregations/vwap` and `aggregations/twap`), each returning a single `price` . Call both and join on `timestamp` .
- **No traded volume.** Amberdata's `totalVolumeUSD` is not returned; use Kaiko OHLCV for volume.
- **No trade-size filter.** Amberdata's `orderSizeCategoryUsd` (0, 1k, 10k, 100k, All) has no Kaiko equivalent: Kaiko computes over all trades.
- `timeInterval` **becomes** `interval` : `minute` / `hour` / `day` become `1m` / `1h` / `1d` (default `1d` on both sides).
- **Exchange and instrument move into the URL path**; the market (`{instrument_class}`) becomes explicit (see §2–3).
- **Also available beyond spot.** Amberdata offers VWAP / TWAP on spot only; the same Kaiko endpoints also serve futures and perpetuals (e.g. `drbt/perpetual-future/btc-usd`).

Coverage & distribution

Data type: Trade aggregations (VWAP, TWAP)

Markets: CEX spot (Amberdata scope); Kaiko also covers futures and perpetuals

Granularity: per `interval` (`1m` , `1h` , `1d` ...), 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	`/trade/vwap` (VWAP and TWAP)	`/v2/data/trades.v1/exchanges/{exchange}/{instrument_class}/{instrument}/aggregations/vwap` `/v2/data/trades.v1/exchanges/{exchange}/{instrument_class}/{instrument}/aggregations/twap`
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`, `binance` → `binc` (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 Kaiko; ISO 8601.
`timeInterval` (optional)	`interval`	RENAMED	`minute` / `hour` / `day` (default) become `1m` / `1h` / `1d` (default).
`orderSizeCategoryUsd` (required)	—	REMOVED	Trade-size filter (`0`, `1k`, `10k`, `100k`, `All`). No Kaiko equivalent: Kaiko uses all trades, i.e. Amberdata `All`.
`timeFormat` (optional)	—	REMOVED	Kaiko returns Unix milliseconds.
—	`page_size`	NEW	Optional. Rows 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 versioned trade aggregation 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 rows (see §5), now at the top level.
`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 (`interval`, `sort`, `page_size`).

Amberdata field	Kaiko field	Change	Notes
—	`time`, `timestamp`	NEW	Server response time (ISO 8601 and Unix milliseconds).
—	`continuation_token`, `next_url`	NEW	Pagination: call `next_url` for the next page.

5 Response - data fields

One row per interval. Amberdata returns VWAP and TWAP in the same row; Kaiko returns one `price` per row on each endpoint.

Amberdata field	Kaiko field	Change	Notes
`exchange`, `pair`	—	REMOVED	Not repeated per row; recoverable from the request path or `query`.
`timestamp`	`timestamp`	RENAMED	Human-readable string by default on Amberdata (`2025-06-08 23:00:00.000`); Unix milliseconds on Kaiko. Start of the interval (midnight UTC for `1d` in the sample).
`vwap`	`price` (aggregations/vwap)	RENAMED	Volume-weighted average price of the trades in the interval; number → decimal string.
`twap`	`price` (aggregations/twap)	RENAMED	Time-weighted average price; number → decimal string.
`totalVolumeUSD`	—	REMOVED	Traded volume in quote terms. Not on these endpoints; use Kaiko OHLCV (`aggregations/ohlcv`).

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 compute VWAP and TWAP from the same exchange trades, so values should be close for the same pair, interval and trade set.

- **Trade-size filter.** Only Amberdata `orderSizeCategoryUsd` = `All` is comparable with Kaiko.

- **Trade capture.** A trade missing or corrected on one side changes the VWAP and TWAP of that interval.
- **Number representation.** Kaiko returns decimal strings; compare as decimals.

8 Before / after example

Request

Amberdata

```
GET https://api.amberdata.com/markets/spot/
analytics
    /trade/vwap
    ?exchange=gdax
    &pair=btc_usd
    &timeInterval=day
    &orderSizeCategoryUsd=All
x-api-key: <your key>
```

Kaiko (two calls)

```
GET https://us.market-api.kaiko.io
    /v2/data/trades.v1
    /exchanges/cbse/spot/btc-usd
    /aggregations/vwap
    ?interval=1d

GET https://us.market-api.kaiko.io
    /v2/data/trades.v1
    /exchanges/cbse/spot/btc-usd
    /aggregations/twap
    ?interval=1d
X-Api-Key: <your key>
```

Response

Amberdata: example from its documentation (shortened). Kaiko: real samples. Different dates, so values are not comparable.

Amberdata

```
{
  "status": 200,
  "title": "OK",
  "description": "Successful request",
  "payload": {
    "data": [
      {
        "exchange": "gdax",
        "pair": "btc_usd",
        "timestamp": "2025-06-08 23:00:00 000",
        "totalVolumeUSD": 17077624.10253145,
        "twap": 105726.12016666665,
        "vwap": 105724.81842395054
      },
      ...
    ]
  }
}
```

Kaiko

```
// aggregations/twap
{
  "query": {
    "interval": "1d",
    "aggregation": "twap",
    "exchange": "cbse",
    "instrument_class": "spot",
    "instrument": "btc-usd",
    "sort": "desc",
    "page_size": 100,
    ...
  },
  "data": [
    {
      "timestamp": 1793491200000,
      "price": "77774.77994117646"
    },
    ...
  ],
  "result": "success",
  ...
}

// aggregations/vwap (binc, btc-usdt)
"data": [
  {
    "timestamp": 1793491200000,
    "price": "77952.65573007488"
  },
  ...
]
```