Kaiko

MIGRATION GUIDE

# Spot – Trade Analytics – Trade Frequency

HTTP REST

Amberdata → Kaiko

2026

www.kaiko.com

## 1 At a glance

**What this endpoint gives you.** Trade statistics of a spot pair per interval: trade counts, buy and sell volumes, total volume, average trade size and interval prices.

What changes for you

- **One exchange and one pair per call.** Exchange and instrument move into the URL path; `fuzzyMatch` is removed.
- **No trade-size filter.** `orderSizeCategoryUsd` has no equivalent: Kaiko values correspond to Amberdata `All`.
- **Numbers become decimal strings** (counts stay integers); timestamps are Unix milliseconds.
- **The Kaiko endpoint is** `trade_stats`; `netBuyVolumeCoin` is not returned (compute it from buy and sell volume).

### Coverage & distribution

**Data type:** Trade aggregations

**Markets:** CEX spot

**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/frequency` | `/v2/data/trades.v1/exchanges/{exchange}/{instrument_class}/{instrument}/aggregations/trade_stats` |
| **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, optional) | `{exchange}` (path, required) | RENAMED | **Required on Kaiko.** Kaiko exchange code, e.g. `gdax` → `cbse` (see §6). |

| Amberdata param | Kaiko param | Change | Notes |
|---|---|---|---|
| — | `{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). |
| `fuzzyMatch` (required) | — | REMOVED | No equivalent: one instrument per call on Kaiko. To cover all pairs of an asset, call once per pair. |
| `orderSizeCategoryUsd` (optional) | — | REMOVED | Trade-size filter (`0`, `1k`, `10k`, `100k`, `All`). No equivalent on this endpoint: Kaiko uses all trades, i.e. Amberdata `All`. |
| `startDate` (optional) | `start_time` | RENAMED | Inclusive on both sides. Kaiko: ISO 8601. Optional; automatically included in continuation tokens. |
| `endDate` (optional) | `end_time` | RENAMED | Exclusive on both sides. Kaiko: ISO 8601. |
| `timeInterval` (optional) | `interval` | RENAMED | `minute` / `hour` / `day` → `1m` / `1h` / `1d` (default). |
| `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`). |
| — | `time`, `timestamp` | NEW | Server response time (ISO 8601 and Unix milliseconds). |

| Amberdata field | Kaiko field | Change | Notes |
|---|---|---|---|
| — | `continuation_token`, `next_url` | NEW | Pagination: call `next_url` for the next page. |
| — | `access` | NEW | Time range your API key can access. |

## 5 Response - data fields

One row per interval. Buy and sell are measured by aggressor side on both providers.

| Amberdata field | Kaiko field | Change | Notes |
|---|---|---|---|
| `exchange`, `pair`, `orderSizeCategoryUsd` | — | REMOVED | Not repeated per row; exchange and instrument are in the request path. |
| `timestamp` | `timestamp` | RENAMED | Human-readable string by default on Amberdata; Unix milliseconds on Kaiko. Start of the interval. |
| `open`, `high`, `low`, `close` | same names | RENAMED | Interval prices; number → decimal string. |
| `totalTradeCount` | `count` | RENAMED | Number of trades; integer. |
| `buyTradeCount`, `sellTradeCount` | `buy_count`, `sell_count` | RENAMED | Buy and sell trade counts; integers. |
| `buyVolumeCoin`, `sellVolumeCoin` | `buy_volume`, `sell_volume` | RENAMED | Buy and sell volume, base asset. |
| `netBuyVolumeCoin` | — | REMOVED | Not returned; compute `buy_volume` – `sell_volume` (or use the Trade Pressure endpoint). |
| — | `volume` | NEW | Total volume, base asset. |
| `totalVolumeUsd` | `volume_usd` | RENAMED | Total volume, converted to USD. |
| `avgTradeSizeUsd` | `average_trade_size_usd` | RENAMED | `volume_usd` / `count` (checked on the sample). |
| `vwap` | `price` | RENAMED | VWAP (volume-weighted average price) of the interval, computed by Kaiko; number → decimal string. |

## 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**

Counts and volumes use the same aggressor-based definitions on both sides and should be close for the same pair and interval (Amberdata `orderSizeCategoryUsd` = `All` ).

- **Net buy volume.** Compute it from `buy_volume` and `sell_volume` .
- **Number representation.** Kaiko returns decimal strings; compare as decimals.

## 8 Before / after example

### Request

**Amberdata**

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

**Kaiko**

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

### Response

Amberdata: example from its documentation (shortened). Kaiko: real sample ( `btc-usdt` ). Different pairs and dates.

**Amberdata**


```
{
  "status": 200,
  "title": "OK",
  "description": "Successful request",
  "payload": {
    "data": [
      {
        "avgTradeSizeUsd": 2430.788381711542,
        "buyTradeCount": 248726,
        "buyVolumeCoin": 4368.776665650001,
        "close": 96495.11,
        "exchange": "gdax",
        "high": 97546.99,
        "low": 94136.79,
        "netBuyVolumeCoin": 403.52652067,
        "open": 96165.11,
        "orderSizeCategoryUsd": "All",
        "pair": "btc_usd",
        "sellTradeCount": 207596,
        "sellVolumeCoin": 3965.2501449799993,
        "timestamp": "2025-05-01 00:00:00 000",
        "totalTradeCount": 456322,
        "totalVolumeUsd": 801971245.3019549,
        "vwap": 96235.22440469854
      }
    ]
  }
}
```


**Kaiko**


```
{
  "query": {
    "interval": "1d",
    "aggregation": "trade_stats",
    "exchange": "cbse",
    "instrument_class": "spot",
    "instrument": "btc-usdt",
    "sort": "desc",
    "page_size": 100,
    ...
  },
  "data": [
    {
      "timestamp": 1793577600000,
      "count": 440,
      "open": "78389.2",
      "high": "78832.25",
      "low": "77635.04",
      "close": "77666.28",
      "volume": "2.5879071899999984",
      "volume_usd": "202170.7621765902",
      "price": "78223.3153257396",
      "buy_count": 344,
      "sell_count": 96,
      "buy_volume": "1.8568125800000026",
      "sell_volume": "0.7310946100000002",
      "average_trade_size_usd":
"459.4790049467959"
    },
    ...
  ],
  "result": "success",
  ...
}
```