Kaiko

MIGRATION GUIDE

OHLCV

Amberdata → Kaiko

2026

www.kaiko.com

1 At a glance

What this endpoint gives you. Historical OHLCV candles (open, high, low, close, volume) for one instrument over a date range and interval you choose. Kaiko offers it in two variants: `ohlcv`, with the same fields as Amberdata, and `count_ohlcv_vwap`, which also returns the trade count and VWAP of each candle.

What changes for you

- **One Kaiko endpoint replaces three Amberdata endpoints.** Spot, futures and options use the same path; the market is a path segment (`instrument_class`, see §3).
- **One exchange per call.** Amberdata accepts a comma-separated list of exchanges; on Kaiko the exchange is in the path, so a multi-exchange request becomes one call per exchange.
- **Default sort order is reversed.** Amberdata returns oldest first; Kaiko returns **newest first** unless you set `sort=asc`. Sorting also works on any date range, not only the last 24 hours.
- **No 2-year window cap per request.** Amberdata limits `startDate`–`endDate` to 731 days; on Kaiko you follow `next_url` through the pages instead.
- **Futures are split in two.** Amberdata's futures endpoint covers dated and perpetual futures; Kaiko separates them into `future` and `perpetual-future`.
- **Dates are ISO 8601 only.** `start_time` is inclusive and `end_time` exclusive, as on Amberdata.
- **Prices and volume are returned as strings**, not JSON numbers. Parse them as decimals (see §5).
- Exchange and instrument are no longer repeated on each row; they are echoed once in the `query` object (see §4, §5).

Coverage & distribution

Data type: OHLCV candles built from trades (optionally with trade count and VWAP)

Markets: CEX spot, dated futures, perpetual futures and options

Instruments per call: 1, on one exchange (Amberdata: 1 instrument, one or more exchanges)

Granularity: any interval from `1s` to `1d`, suffixed `s`, `m`, `h` or `d`; default `1d` (Amberdata: minutes, hours, days)

History: paginated with `next_url`, limited by your key's entitlement (Amberdata: 731 days per request, previous 12 months by default)

Distribution: REST API

2 Endpoint & authentication

	Amberdata (today)	Kaiko (new)
Method	GET	GET
Base URL	`https://api.amberdata.com/markets`	`https://us.market-api.kaiko.io` `https://eu.market-api.kaiko.io`

Path	`/spot/ohlcv/{instrument}`
	`/futures/ohlcv/{instrument}`
	`/options/ohlcv/{instrument}`
		`/v2/data/trades.v1/exchanges/{exchange}`
		`/{instrument_class}/{instrument}`
		`/aggregations/ohlcv`
		`(or /aggregations/count_ohlcv_vwap)`

	Amberdata (today)	Kaiko (new)
Auth	API key header `x-api-key`	API key header `X-Api-Key` (same mechanism, new key)

The three Amberdata endpoints take the same parameters and return the same fields; only the market segment in the path differs. On Kaiko, the exchange and the market move into the path, and you choose a `us` or `eu` host. End the path with `ohlcv` for Amberdata's field set, or with `count_ohlcv_vwap` to also get `count` and `price` (VWAP); parameters are the same.

3 Request parameters

Kaiko path segments (required): `exchange`, `instrument_class`, `instrument`. Query parameters (optional): `start_time`, `end_time`, `interval`, `sort`, `page_size`, `continuation_token`.

Amberdata param	Kaiko param	Change	Notes
`spot` | `futures` | `options` (path)	`instrument_class` (path)	RENAMED	Values `spot`, `future`, `perpetual-future`, `option`. Amberdata `futures` maps to both `future` and `perpetual-future`.
`instrument` (path, required)	`instrument` (path)	RENAMED	Same position, Kaiko instrument code (e.g. `btc_usd` → `btc-usd` for spot); see §6.
`exchange` (query, required)	`exchange` (path)	RENAMED	Moves from query to path, and takes **one** exchange instead of a comma-separated list. Kaiko exchange code (e.g. `gdax` → `cbse`); see §6.
`startDate` (optional)	`start_time`	RENAMED	Inclusive on both sides. Amberdata accepts seconds, milliseconds or ISO 8601; Kaiko takes ISO 8601 only (e.g. `2026-01-01T00:00:00Z`). Amberdata's 731-day maximum between start and end does not apply; long ranges are paginated.
`endDate` (optional)	`end_time`	RENAMED	**Exclusive on both sides.** ISO 8601 only on Kaiko.
`timeInterval` (optional)	`interval`	RENAMED	Amberdata: `minutes`, `hours`, `days` (default). Kaiko: any value from one second to one day, suffixed `s`, `m`, `h` or `d` (e.g. `30s`, `15m`, `4h`); default `1d`. Equivalents: `1m`, `1h`, `1d`.
`sortDirection` (optional)	`sort`	RENAMED	**Defaults differ:** Amberdata `asc`, Kaiko `desc` (newest first). Amberdata only allows sorting within the most recent 24 hours; Kaiko sorts any range. Values `asc` / `desc`.
`timeFormat` (optional)	—	REMOVED	Amberdata can return milliseconds (default), ISO 8601 or human-readable timestamps. Kaiko always returns Unix milliseconds.
`api-version` (header, optional)	—	REMOVED	Kaiko versions this endpoint in the path (`/v2/`, `trades.v1`).
`Accept-Encoding` (header, required)	`Accept-Encoding`	IDENTICAL	Required by Amberdata (`gzip, deflate, br`); supported by Kaiko (e.g. `curl --compressed`).
—	`page_size`	NEW	Optional. Number of candles per page.

Amberdata param	Kaiko param	Change	Notes
—	`continuation_token`	NEW	Pagination cursor. Already embedded in `next_url`, which also carries your time range, sort and page size; you rarely set it yourself.

4 Response envelope & metadata

The fields around the data, the wrapper your parser reads before reaching the numbers. This envelope is **identical on every Amberdata endpoint**.

Amberdata field	Kaiko field	Change	Notes
`status`	`result`	RENAMED	Integer `200` on Amberdata; string `"success"` on Kaiko. Rely on the HTTP status code.
`title`	—	REMOVED	"OK" label, no direct Kaiko equivalent.
`description`	—	REMOVED	Free-text status description.
`payload.data`	`data`	RENAMED	**Array** holding the candles (see §5), now at the top level.
`payload.metadata.next`	`next_url`	RENAMED	Full URL of the next page. Call it as-is until it is no longer returned.
`payload.metadata.api-version`	—	REMOVED	Version lives in the URL path on Kaiko; `query.data_version` echoes the data version (`v1`).
—	`continuation_token`	NEW	Cursor for the next page, also embedded in `next_url`.
—	`query`	NEW	Echo of the request: `exchange`, `instrument_class`, `instrument`, `interval`, `sort`, `page_size`, `request_time`, etc.
—	`time` / `timestamp`	NEW	When the **response** was produced (ISO 8601 / Unix ms). Not a candle time: do not confuse with `data[].timestamp`.
—	`access`	NEW	`access_range` : the period your key is entitled to; `data_range` : the period Kaiko holds for this instrument. Both as Unix ms start/end.

5 Response - data fields

One row per candle. Fields are the same for every market.

Amberdata field	Kaiko field	Type	Change	Notes
`instrument`	`query.instrument`	string	RENAMED	Not repeated on each row; echoed once in `query`. Kaiko code format (see §6).

Amberdata field	Kaiko field	Type	Change	Notes
`exchange`	`query.exchange`	string	RENAMED	Not repeated on each row; echoed once in `query`. Kaiko exchange code.
`exchangeTimestamp`	`timestamp`	integer (ms)	RENAMED	Unix milliseconds on both sides. Kaiko stamps each candle with its **open** time (start of the interval).
`open`	`open`	number → string	RENAMED	Same name; returned as a decimal string. `null` when no trades.
`high`	`high`	number → string	RENAMED	Same name; returned as a decimal string. `null` when no trades.
`low`	`low`	number → string	RENAMED	Same name; returned as a decimal string. `null` when no trades.
`close`	`close`	number → string	RENAMED	Same name; returned as a decimal string. `null` when no trades.
`volume`	`volume`	number → string	RENAMED	Same name; returned as a decimal string. `"0"` when no trades.
—	`price`	string	NEW	`count_ohlcv_vwap` only. Volume-weighted average price (VWAP) of the interval. `null` when no trades.
—	`count`	integer	NEW	`count_ohlcv_vwap` only. Number of trades in the interval. `0` when no trades.

Exchange and instrument are carried by the request

Rows carry only the candle values. Each call targets a single exchange, market and instrument, all echoed in `query`, so every row of a response belongs to them. If you merge results from several calls (e.g. one per exchange), add the exchange and instrument to each row yourself.

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

Candles should be close but may differ slightly, because they are produced differently: **Amberdata uses the candles as reported by each exchange**, while **Kaiko builds its candles from the trades it collects**.

- **Trade capture.** Because Kaiko aggregates the trades it collects, a trade that is missed, arrives late or is later corrected by the exchange can change that candle's volume, and possibly its high or low, compared with the exchange-reported candle.
- **Daily candle boundaries.** Amberdata notes that exchanges compute daily candles differently and do not all use midnight UTC. Kaiko daily candles start at 00:00 UTC (e.g. `1605139200000` = 2020-11-12T00:00:00Z), so daily values can differ on those venues.
- **Number representation.** Kaiko returns decimal strings, which keep the exact digits; Amberdata returns JSON numbers (e.g. `85604` on Amberdata, `"85604"` on Kaiko). Compare as decimals, not as text.
- **Empty intervals.** When the instrument did not trade in an interval, Kaiko returns the row with `null` prices and a `volume` of `"0"` (and, with `count_ohlcv_vwap`, a `null` VWAP and a `count` of `0`).

8 Before / after example

Request

Kraken BTC/USD, daily candle of 2026-09-30.

Amberdata

```
GET https://api.amberdata.com/markets
    /spot/ohlcv/btc_usd
    ?exchange=kraken
    &startDate=2026-09-30T00:00:00
    &endDate=2026-10-01T00:00:00
    &timeInterval=days
x-api-key: <your key>
Accept-Encoding: gzip, deflate, br
```

Kaiko

```
GET https://us.market-api.kaiko.io
    /v2/data/trades.v1/exchanges/krkn
    /spot/btc-usd
    /aggregations/ohlcv
    ?end_time=2026-10-01T00:00:00.000Z
    &interval=1d
    &page_size=1
X-Api-Key: <your key>
```

With the default `sort=desc`, Kaiko returns the newest candle before `end_time` first, so `page_size=1` gives the 2026-09-30 candle. Add `sort=asc` to keep Amberdata's oldest-first order.

Response (real samples: the same candle on both sides)

Amberdata


```
{
  "status": 200,
  "title": "OK",
  "description": "Successful request",
  "payload": {
    "metadata": {
      "next": null,
      "api-version": "2023-09-30"
    },
    "data": [
      {
        "instrument": "btc_usd",
        "exchange": "kraken",
        "exchangeTimestamp": 1790726400000,
        "open": 83633.5,
        "high": 85604,
        "low": 82920.1,
        "close": 83564.7,
        "volume": 2715.90924755
      }
    ]
  }
}
```


Kaiko


```
{
  "query": {
    "interval": "1d",
    "aggregation": "ohlcv",
    "end_time": "2026-10-01T00:00:00.000Z",
    "end_timestamp": 1790812800000,
    "exchange": "krkn",
    "instrument_class": "spot",
    "instrument": "btc-usd",
    "sort": "desc",
    "page_size": 1,
    "request_time": "2026-11-08T19:53:30.394Z",
    "data_version": "v1",
    "commodity": "trades",
    ...
  },
  "time": "2026-11-08T19:53:30.394211881Z",
  "timestamp": 1794167610394,
  "data": [
    {
      "timestamp": 1790726400000,
      "open": "83633.5",
      "high": "85604",
      "low": "82920.1",
      "close": "83564.7",
      "volume": "2715.90924755"
    }
  ],
  "result": "success",
  "continuation_token": "41NtDg6Q...",
  "next_url": "https://us.market-api.kaiko.io
    /v2/.../aggregations/ohlcv
    ?continuation_token=41NtDg6Q...",
  "access": {
    "access_range": {
      "start_timestamp": 1696377600000,
      "end_timestamp": null
    },
    "data_range": {
      "start_timestamp": 946684800000,
      "end_timestamp": null
    }
  }
}
```