Kaiko

MIGRATION GUIDE

# Moneyness Surfaces Constant & Floating

## Amberdata → Kaiko

2026

www.kaiko.com

## 1 At a glance

**What this endpoint gives you:** for BTC and ETH options on Deribit and OKX, this endpoint returns the **implied volatility at fixed moneyness levels** K/F of 0.6, 0.7, 0.8, 0.9, 1.0 (at the money), 1.1, 1.2, 1.3 and 1.4, at each snapshot of the requested period, together with the market references used (index price, forward price). The volatilities come from an SVI smile calibrated on the exchange mark implied volatilities, either at the constant maturities 7, 30, 60, 90 and 180 days or at every listed expiry. Amberdata's two endpoints, **Moneyness Surfaces Constant** and **Moneyness Surfaces Floating**, become one Kaiko endpoint, **Mark IV Moneyness Surfaces**, with a `maturity_type` parameter:

| Amberdata endpoint | Kaiko request |
|---|---|
| `.../volatility/moneyness-surfaces/constant` | `/mark-iv-moneyness-surfaces?maturity_type=constant` |
| `.../volatility/moneyness-surfaces/floating` | `/mark-iv-moneyness-surfaces?maturity_type=floating` |

### What changes for you

- You will call a new endpoint, `/analytics/volatility/mark-iv-moneyness-surfaces`, using your Kaiko API key in the `X-Api-Key` header (§2).
- The maturity type is now chosen with `maturity_type`, which defaults to `constant`: if you used the floating endpoint, set `maturity_type=floating`, otherwise you will receive the constant maturities without any error (§3).
- The underlying is now expressed with three lowercase parameters: `currency=BTC` becomes `exchange=drbt`, `base_asset=btc` and `quote_asset=usd`. The exchange and the quote asset are new required parameters (§3, §6).
- Kaiko serves every calibrated minute and has no `interval` parameter: for hourly data, you can keep the rows stamped `hh:00`. A request covers at most 24 hours, and the `end_time` is now excluded (§3).
- A snapshot is available in the API about 1.5 minutes after its `value_time`.
- The `timeInterval`, `daysToExpirationStart`, `daysToExpirationEnd` and `timeFormat` parameters are removed: you can filter the response on `days_to_expiry`, or on `expiry` for floating rows (§3).
- The fields keep their names in snake_case, for example `put40PercentOutOfMoneyVolatility` becomes `put_40_percent_out_of_money_volatility` and `daysToExpiration` becomes `days_to_expiry`. A new `expiry` field gives the listed expiry, `null` on constant rows (§5).
- Values are now returned as decimals, for example `0.3269` where Amberdata returned `32.69` (percent).
- Floating rows now include the listed expiries less than 1 day away, which Amberdata did not return. Constant maturities are never extrapolated (§7).
- Kaiko calibrates the SVI smile with its own published methodology and uses per-minute median index and forward prices: values change (§7).

### Coverage & distribution

**Data type:** Implied volatility at fixed moneyness K/F, one row per snapshot and maturity

**Constant:** Fixed tenors of 7, 30, 60, 90 and 180 days

**Floating:** Each listed expiry, including those expiring within the day

**Model:** SVI calibrated on the total implied variance of the exchange mark implied volatilities (Kaiko's published methodology); constant tenors interpolated in total variance, never extrapolated

**Snapshots:** Every minute (no `interval` parameter), available about 1.5 minutes after `value_time`

**Exchanges:** Deribit and OKX options, BTC/USD and ETH/USD (§6)

**History:** From 12 February 2021 on Deribit and 16 April 2024 on OKX

**Distribution:** REST API

## 2 Endpoint & authentication

| | Amberdata (today) | Kaiko (new) |
|---|---|---|
| **Method** | GET | GET |
| **Base URL** | `https://api.amberdata.com` | `https://api.kaiko.io` |
| **Path** | `/markets/derivatives/analytics /volatility/moneyness-surfaces /constant or /floating` | `/analytics/volatility /mark-iv-moneyness-surfaces + maturity_type` |
| **Auth** | API key header `x-api-key` | API key header `X-Api-Key` (your Kaiko key) |

Header names are not case-sensitive: `x-api-key` and `X-Api-Key` are the same header, so only the key itself changes. **Versioning:** Amberdata reports its version in `payload.metadata.api-version` ( `2023-09-30` ). Kaiko takes an optional `api_version` query parameter (default and current `2026-10-01` ) and returns it in `meta.api_version` .

## 3 Request parameters

Both Amberdata endpoints take the same parameters.

| Amberdata param | Kaiko param | Change | Notes |
|---|---|---|---|
| Path `/constant` or `/floating` | `maturity_type` (optional) | NEW | `constant` (default) or `floating` . Replaces the end of the Amberdata path. |
| None | `exchange` (required) | NEW | `drbt` (Deribit), the exchange Amberdata served, or `okex` (OKX) (§6). |
| `currency` (required) | `base_asset` + `quote_asset` (both required) | RENAMED | `BTC` → `btc` + `usd` , `ETH` → `eth` + `usd` . Lowercase only. |
| `startDate` (optional) | `start_time` (optional) | RENAMED | Included on both sides. ISO 8601 with a time zone. |
| `endDate` (optional) | `end_time` (optional) | RENAMED | **Excluded** at Kaiko. Amberdata also returned the snapshot stamped exactly at `endDate` . |

| Amberdata param | Kaiko param | Change | Notes |
|---|---|---|---|
| `timeInterval` (optional) | None | REMOVED | Amberdata defaulted to one snapshot per hour. Kaiko serves every minute: keep the rows stamped `hh:00` for one per hour. |
| `daysToExpirationStart` (optional) | None | REMOVED | Kept the maturities at or above this number of days. Filter the response on `days_to_expiry` instead. |
| `daysToExpirationEnd` (optional) | None | REMOVED | Upper bound of the same filter. Filter the response on `days_to_expiry` instead. |
| `timeFormat` (optional) | None | REMOVED | Kaiko times are always ISO 8601 strings, Amberdata's default format here. |
| None | `sort` (optional) | NEW | `desc` (default, newest first, as on Amberdata) or `asc`. Maturities ascending within a snapshot. |
| None | `api_version` (optional) | NEW | Pins the response format. Default `2026-10-01`. |

### Window, snapshots and limits

- **Boundary:** the window is `[start_time, end_time)`, so the snapshot at `start_time` is included and the one at `end_time` is not. Amberdata included both: a request from 12:00 to 13:00 returned the 12:00 and the 13:00 snapshots, while Kaiko returns the minutes from 12:00 to 12:59.
- **Every minute:** you receive one snapshot per calibrated minute. When a minute has no new calibration, Kaiko repeats the latest surface from up to 5 minutes before, stamped with that minute.
- **Defaults:** if you include no dates, you receive the latest snapshot, as on Amberdata. With only `start_time`, the window runs until now. With only `end_time`, you receive the minute before it. A `start_time` equal to `end_time` returns a 400 error, so for one snapshot, simply send `end_time` alone.
- **Limits:** a request covers up to 24 hours (1,440 snapshots). A wider window returns a 400 error naming the cap, so you can split a longer history into several requests. The whole window comes in one response, without pagination.
- **Time format:** ISO 8601 with a time zone and at most millisecond precision, for example `2026-10-29T12:00:00.000Z`. Amberdata also accepted a time without a zone; Kaiko returns a 400 error in that case, and for a precision finer than the millisecond.

## 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` | None | REMOVED | Use the HTTP status code (200). |
| `title` | None | REMOVED | No equivalent on success. |
| `description` | None | REMOVED | No equivalent on success. |

| Amberdata field | Kaiko field | Change | Notes |
|---|---|---|---|
| `payload.data` | `data` | RENAMED | Top level. **Array**, one row per snapshot and maturity. |
| `payload.metadata` | `meta` | RENAMED | Top level. |
| `api-version` | `api_version` | RENAMED | `2026-10-01` (Amberdata: `2023-09-30`). |
| None | `count` | NEW | Rows in `data`. |
| None | `next_url` | NEW | Always `null` (one page). |
| None | `url` | NEW | Your request URL, with `api_version`. |

At Amberdata, `api-version` sits inside `payload.metadata`. At Kaiko, `api_version`, `count`, `next_url` and `url` sit inside `meta`.

### Errors and empty results

- **Underlying not served:** Amberdata returned a 200 response with an empty `data` array, while Kaiko returns a **404** error with the code `unknown_instrument` (for example "drbt does not serve doge-usd").
- **Uppercase values or Amberdata codes:** Kaiko returns a **400** error (for example for `BTC`, `deribit` or `maturity_type=FLOATING`).
- **No snapshot in the window:** both APIs return a 200 response with an empty `data` array.
- **Missing key:** Kaiko returns a **401** error with the code `missing_api_key`.
- **Missing or invalid parameter:** both APIs return a 400 error. Kaiko errors are `application/problem+json` (RFC 9457) bodies with `status`, `title`, a `detail` message naming the parameter, and the code `invalid_parameter`.

## 5 Response data fields

Both Amberdata endpoints and both maturity types use the same fields; only their meaning differs where noted. You get one row per snapshot and constant maturity (constant), or per snapshot and listed expiry (floating).

| Amberdata field | Kaiko field | Type | Change |
|---|---|---|---|
| `timestamp` | `value_time` | string | RENAMED |
| `exchange` | None | string | REMOVED |
| `currency` | None | string | REMOVED |
| None | `expiry` | string or null | NEW |
| `daysToExpiration` | `days_to_expiry` | number | RENAMED |
| `indexPrice` | `index_price` | number or null | RENAMED |
| `underlyingPrice` | `underlying_price` | number or null | RENAMED |

| Amberdata field | Kaiko field | Type | Change |
|---|---|---|---|
| `putXXPercentOutOfMoneyVolatility` (XX = 40, 30, 20, 10) | `put_XX_percent_out_of_money_volatility` | number or null | RENAMED |
| `atm` | `atm` | number or null | IDENTICAL |
| `callXXPercentOutOfMoneyVolatility` (XX = 10, 20, 30, 40) | `call_XX_percent_out_of_money_volatility` | number or null | RENAMED |

- **value_time:** the snapshot time, in ISO 8601 ( `"2026-10-29T12:00:00.000Z"` ), the same format as Amberdata's default here.
- **exchange, currency:** these are no longer repeated in the rows. They are in your request and in `meta.url` , together with `maturity_type` . If you store or merge rows from several requests (for example constant and floating), you can add them yourself.
- **expiry:** new, the listed expiry on floating rows (ISO 8601, `"2026-10-31T08:00:00.000Z"` ). It is always `null` on constant rows, because a tenor is not a listed expiry.
- **days_to_expiry:** the tenor on constant rows ( `7` , `30` , `60` , `90` or `180` ), and the fractional days from `value_time` to `expiry` on floating rows (for example `0.8333` ), below 1 for an expiry within the day.
- **index_price:** the exchange index price in `quote_asset` , now the median of the index prices reported across the options within the minute. It is the same on every row of a snapshot.
- **underlying_price:** the forward price F that the moneyness K/F is measured against, in `quote_asset` . On floating rows, it is the forward of the expiry, now the median of the underlying prices reported across its options within the minute. On constant rows, it is interpolated linearly in time to maturity between the forwards of the two surrounding listed expiries (Amberdata interpolated the squared forward).
- **put_XX / call_XX_percent_out_of_money_volatility:** these keep Amberdata's levels, the put XX% below the forward and the call XX% above it. Put 40, 30, 20 and 10 are K/F = 0.6, 0.7, 0.8 and 0.9; `atm` is K/F = 1.0; call 10, 20, 30 and 40 are K/F = 1.1, 1.2, 1.3 and 1.4. A level where the calibrated smile has no point is `null` .

**Units:** volatilities are annualized. Amberdata used percent ( `32.69` = 32.69%), while Kaiko uses decimals ( `0.3269` = 32.69%): divide Amberdata's values by 100, or multiply Kaiko's by 100. `index_price` and `underlying_price` keep their units.

## 6 Exchange & venue names

Amberdata's moneyness surfaces had no exchange parameter: they were Deribit's, and every row carried `exchange: "deribit"` . Kaiko asks for the exchange in the request and does not repeat it in the rows. Map the currency you send as follows (the same for constant and floating):

| Amberdata currency | Kaiko exchange | base_asset | quote_asset | History from |
|---|---|---|---|---|
| BTC | drbt | btc | usd | Feb 2021 |
| ETH | drbt | eth | usd | Feb 2021 |
| None | okex | btc | usd | Apr 2024 |
| None | okex | eth | usd | Apr 2024 |

`drbt` and `okex` are Kaiko's codes for Deribit and OKX. Uppercase values and Amberdata's code `deribit` return a 400 error.

## 7 Will your values change?

**Short answer**

Yes. The volatilities come from an SVI smile calibrated with a different methodology. They are read exactly at the requested moneyness, from forwards that are per-minute medians (per expiry on floating rows, interpolated between the surrounding expiries on constant rows), and they are expressed as decimals instead of percent.

- **Calibration methodology:** Kaiko calibrates the SVI model (Gatheral and Jacquier, Arbitrage-free SVI volatility surfaces) on the **total implied variance** $\omega(\tau, k) = \sigma^2(k) \times \tau$, at each snapshot t and for each listed expiry T, where $\tau = T - t$ is the time to maturity, $k = \ln(K / F)$ the forward log-moneyness and $\sigma$ the exchange mark implied volatility. The calibration follows Kaiko's published methodology (Kaiko quantitative white papers, paper on HAL, hal-03715921), with Kaiko's own data filters and initialisation of the SVI parameters. Amberdata calibrated on the squared volatility $\sigma^2$ instead of the total variance $\sigma^2\tau$.
- **Constant maturities:** for each constant maturity, Kaiko interpolates the calibrated smiles of the surrounding listed expiries in total variance, at constant k. The forward of a constant maturity is interpolated linearly in time to maturity between the forwards of the two surrounding listed expiries, whereas Amberdata interpolated the squared forward. There is no extrapolation: a maturity below the closest listed expiry or above the farthest one gets no implied volatility.
- **Expiries below 1 day:** on floating rows, the listed expiries less than 1 day away are calibrated and returned, whereas Amberdata only returned the expiries more than 1 day away.
- **Index and forward prices:** within a minute, the exchange reports slightly different index and underlying (future) prices from one option to another. Kaiko takes their median within the minute.
- **Units:** volatilities are decimals instead of percent, so `32.69` becomes `0.3269`.

## 8 Before / after examples

### Constant: request

**Amberdata**

```
GET https://api.amberdata.com
    /markets/derivatives/analytics
    /volatility/moneyness-surfaces
    /constant
    ?currency=BTC
    &startDate=2026-11-05T12:00:00
    &endDate=2026-11-05T13:00:00
x-api-key: <your-amberdata-key>
```

**Kaiko**

```
GET https://api.kaiko.io
    /analytics/volatility
    /mark-iv-moneyness-surfaces
    ?exchange=drbt
    &base_asset=btc
    &quote_asset=usd
    &start_time=2026-11-05T12:00:00.000Z
    &end_time=2026-11-05T13:00:00.000Z
    &api_version=2026-10-01
X-Api-Key: <your-kaiko-key>
```

No `maturity_type` here: `constant` is the default. Kaiko returns every minute of the window, 60 snapshots from 12:00 to 12:59, where Amberdata returned one snapshot per hour.

## Constant: response (real responses, 12:00 snapshot, 7-day tenor; values rounded, fields reordered)

### Amberdata


```
{
  "status": 200,
  "title": "OK",
  "description": "Successful request",
  "payload": {
    "data": [
    … 5 rows stamped 13:00
      (endDate is included), then
    {
      "timestamp": "2026-11-05T12:00:00.000Z",
      "exchange": "deribit",
      "currency": "BTC",
      "daysToExpiration": 7,
      "indexPrice": 80301.29,
      "underlyingPrice": 80398.36,
      "put40PercentOutOfMoneyVolatility": 125.01,
      "put30PercentOutOfMoneyVolatility": 95.88,
      "put20PercentOutOfMoneyVolatility": 67.76,
      "put10PercentOutOfMoneyVolatility": 44.04,
      "atm": 33.37,
      "call10PercentOutOfMoneyVolatility": 39.50,
      "call20PercentOutOfMoneyVolatility": 51.80,
      "call30PercentOutOfMoneyVolatility": 64.29,
      "call40PercentOutOfMoneyVolatility": 75.62
    },
    … 4 more tenors, 30 to 180 days
    ],
    "metadata": {
      "api-version": "2023-09-30"
    }
  }
}
```


### Kaiko


```
{
  "data": [
  … 59 newer snapshots first
    (12:59 down to 12:01), then
  {
    "value_time": "2026-11-05T12:00:00.000Z",
    "expiry": null,
    "days_to_expiry": 7,
    "index_price": 80306.38,
    "underlying_price": 80404.11,
    "put_40_percent_out_of_money_volatility":
      1.0560,
    "put_30_percent_out_of_money_volatility":
      0.8487,
    "put_20_percent_out_of_money_volatility":
      0.6392,
    "put_10_percent_out_of_money_volatility":
      0.4405,
    "atm": 0.3349,
    "call_10_percent_out_of_money_volatility":
      0.3958,
    "call_20_percent_out_of_money_volatility":
      0.5118,
    "call_30_percent_out_of_money_volatility":
      0.6225,
    "call_40_percent_out_of_money_volatility":
      0.7191
  },
  … 4 more tenors, 30 to 180 days
  ],
  "meta": {
    "api_version": "2026-10-01",
    "count": 300,
    "next_url": null,
    "url": "https://api.kaiko.io/analytics/
volatility/mark-iv-moneyness-surfaces?
exchange=drbt&base_asset=btc&quote_asset=usd&start
_time=2026-11-05T12%3A00%3A00.000Z&end_time=2026-1
1-05T13%3A00%3A00.000Z&api_version=2026-10-01"
  }
}
```


## Floating: request

### Amberdata

```
GET https://api.amberdata.com
    /markets/derivatives/analytics
    /volatility/moneyness-surfaces
    /floating
    ?currency=BTC
    &startDate=2026-11-05T12:00:00
    &endDate=2026-11-05T13:00:00
x-api-key: <your-amberdata-key>
```

### Kaiko

```
GET https://api.kaiko.io
    /analytics/volatility
    /mark-iv-moneyness-surfaces
    ?exchange=drbt
    &base_asset=btc
    &quote_asset=usd
    &maturity_type=floating
    &start_time=2026-11-05T12:00:00.000Z
    &end_time=2026-11-05T13:00:00.000Z
    &api_version=2026-10-01
X-Api-Key: <your-kaiko-key>
```

## Floating: response (real responses, 12:00 snapshot, expiry 7 November 2026 08:00 UTC; values rounded, fields reordered)

The 7 November expiry is Amberdata's closest one. Kaiko also returns the 6 November expiry, 0.83 days away. Amberdata's floating rows carry no expiry field.

**Amberdata**


```
{
  "status": 200,
  "title": "OK",
  "description": "Successful request",
  "payload": {
    "data": [
    … 11 rows stamped 13:00
      (endDate is included), then
    {
      "timestamp": "2026-11-05T12:00:00.000Z",
      "exchange": "deribit",
      "currency": "BTC",
      "daysToExpiration": 1.83,
      "indexPrice": 80301.29,
      "underlyingPrice": 80327.29,
      "put40PercentOutOfMoneyVolatility": 170.13,
      "put30PercentOutOfMoneyVolatility": 135.11,
      "put20PercentOutOfMoneyVolatility": 98.16,
      "put10PercentOutOfMoneyVolatility": 59.20,
      "atm": 34.96,
      "call10PercentOutOfMoneyVolatility": 54.90,
      "call20PercentOutOfMoneyVolatility": 82.20,
      "call30PercentOutOfMoneyVolatility": 104.95,
      "call40PercentOutOfMoneyVolatility": 123.66
    },
    … 10 more expiries, to 2027-09-24
    ],
    "metadata": {
      "api-version": "2023-09-30"
    }
  }
}
```


**Kaiko**


```
{
  "data": [
  … 59 newer snapshots first
    (12:59 down to 12:01), then the
    2026-11-06 expiry (0.83 days away,
    not in Amberdata), then
  {
    "value_time": "2026-11-05T12:00:00.000Z",
    "expiry": "2026-11-07T08:00:00.000Z",
    "days_to_expiry": 1.8333,
    "index_price": 80306.38,
    "underlying_price": 80334.59,
    "put_40_percent_out_of_money_volatility":
      1.4533,
    "put_30_percent_out_of_money_volatility":
      1.1870,
    "put_20_percent_out_of_money_volatility":
      0.9028,
    "put_10_percent_out_of_money_volatility":
      0.5820,
    "atm": 0.3501,
    "call_10_percent_out_of_money_volatility":
      0.5288,
    "call_20_percent_out_of_money_volatility":
      0.7342,
    "call_30_percent_out_of_money_volatility":
      0.8959,
    "call_40_percent_out_of_money_volatility":
      1.0269
  },
  … 10 more expiries, to 2027-09-24
  ],
  "meta": {
    "api_version": "2026-10-01",
    "count": 720,
    "next_url": null,
    "url": "https://api.kaiko.io/analytics/
volatility/mark-iv-moneyness-surfaces?
exchange=drbt&base_asset=btc&quote_asset=usd&start
_time=2026-11-05T12%3A00%3A00.000Z&end_time=2026-1
1-05T13%3A00%3A00.000Z&maturity_type=floating&api_
version=2026-10-01"
  }
}
```