Kaiko

MIGRATION GUIDE

Term Structures Richness, TradFi

Amberdata → Kaiko

2026

www.kaiko.com

1 At a glance

What this endpoint gives you: Amberdata's **Term Structures Richness** for US-listed options becomes Kaiko's **TradFi Term Structure Richness**. For one US-listed equity or ETF (IBIT, MSTR, COIN, ...), one row per snapshot: the at-the-money implied volatility at the 10, 21, 63, 84 and 189-day constant maturities, and `richness`, the mean short-over-long ratio of those volatilities over the ten tenor pairs. Above 1, short-dated volatility is richer than long-dated (backwardation); below 1, the curve slopes up (contango). The volatilities are read off the constant-maturity delta surface of the underlying, on both sides.

What changes for you

- New endpoint and key: call `/analytics/volatility/tradfi-term-structure-richness` with your Kaiko API key in the `X-Api-Key` header (§2).
- The ticker is lowercase: `currency=MSTR` becomes `base_asset=mstr`; `quote_asset` is optional and always `usd` (§3, §6).
- `timeInterval=day` becomes `interval=1d`, the default; `5m` and `1h` are also available (§3).
- The window boundary changes: Kaiko serves `[start_time, end_time)` where Amberdata served `(startDate, endDate]`. With the same dates, you now receive the row at the start of the window rather than the one at its end (§3).
- ATM volatilities stay decimals, now rounded to 5 decimals: Kaiko returns `0.69627` where Amberdata returned `0.6962699890136719`. `ratio`, `counter` and `richness` are unchanged (§5, §7).
- Field names move to snake_case, for example `atm10days` becomes `atm_10_days`, and `timestamp` becomes `value_time` (§5).
- The same 23 underlyings are served (§6).

Coverage & distribution

Data type: Time series, one row per snapshot, newest first

Sampling: Every 5 minutes during the US session (13:30 to 20:00 UTC on US daylight time), the bucket on the hour, or one bucket per trading day (default); one snapshot each, not averaged

Metrics: ATM implied volatility at 10, 21, 63, 84 and 189 days · ratio, counter, richness

Underlyings: 23 US-listed crypto ETFs and crypto-exposed equities (§6)

Range: Up to 1, 90 or 3,660 days per request (5m, 1h, 1d)

History: From 14 October 2026; a backfill to extend the available history is in progress

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/term-structures/richness/tradfi`	`/analytics/volatility/tradfi-term-structure-richness`

	Amberdata (today)	Kaiko (new)
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, only the key 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

Amberdata param	Kaiko param	Change	Notes
`currency` (required)	`base_asset` (required)	RENAMED	The ticker, written in lowercase: `MSTR` becomes `mstr` (§6).
None	`quote_asset` (optional)	NEW	Always `usd` , the default; you can omit it.
`startDate` (optional)	`start_time` (optional)	RENAMED	Now **inclusive**; Amberdata excluded it. ISO 8601 with a time zone.
`endDate` (optional)	`end_time` (optional)	RENAMED	Now **exclusive**; Amberdata included it. ISO 8601 with a time zone.
`timeInterval` (optional)	`interval` (optional)	RENAMED	`minute` → `5m` , `hour` → `1h` , `day` → `1d` (default on both sides).
`timeFormat` (optional)	None	REMOVED	No longer needed: Kaiko always returns times as ISO 8601 strings.
None	`sort` (optional)	NEW	`desc` (default, newest first, as with Amberdata) or `asc` .
None	`api_version` (optional)	NEW	Pins the response format to a version. Default `2026-10-01` .

Window, sampling and limits

- **Boundary, changed:** Kaiko serves `[start_time, end_time)`, meaning that the bucket at `start_time` is included and the one at `end_time` is not. Amberdata did the opposite, `(startDate, endDate]`. In practice, with `interval=1h`, a request from 14:00 to 15:00 returns the 14:00 bucket on Kaiko where Amberdata returned the 15:00 one. To receive exactly the rows Amberdata returned, shift both dates forward by one interval.
- **Sampling:** with `5m` you receive every bucket of the US session. With `1h` you receive the bucket on each hour (`hh:00` UTC), as with Amberdata's `hour`. With `1d` you receive one row per New York trading day, taken from the 15:00 America/New York bucket (19:00 UTC on US daylight time, 20:00 UTC otherwise). When that bucket is missing, Kaiko serves the latest bucket available before it on that day, so that the day still has a row; for the day in progress, Kaiko serves the latest bucket so far, a provisional value that is replaced once the 15:00 bucket exists. Amberdata's `day` row was always the 20:00 UTC bucket: during US standard time (winter) this is the same 15:00 New York bucket, but during US daylight time (summer) the session closes before 20:00 UTC, so Amberdata had no daily row at all. Kaiko serves a daily row all year.
- **Defaults, the same as Amberdata:** without dates, both sides return the full window the interval allows, ending now (1 day at `5m`, 90 days at `1h`, 3,660 days at `1d`). With `start_time` alone, Kaiko returns everything from that time until now. With `end_time` alone, Kaiko returns only the one bucket before it, where Amberdata returned the full window before it; we recommend always sending both bounds.
- **Limits, the same as Amberdata:** up to 1 day per request at `5m`, 90 days at `1h`, 3,660 days at `1d`. A wider window returns a 400 error whose message names the maximum. The whole window comes back in a single response; there is no pagination to handle.
- **Time format:** ISO 8601 with a time zone, e.g. `2026-10-22T18:20:00.000Z`. Amberdata also accepted a time without a time zone; Kaiko requires one and returns a 400 error otherwise. Please note that Kaiko does not recognise Amberdata's values either: `interval=minute` or an uppercase ticker also return a 400 error.

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.
`payload.data`	`data`	RENAMED	Top level. **Array**, one row per snapshot.
`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).

Amberdata field	Kaiko field	Change	Notes
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

- **Ticker not served:** where Amberdata returned 200 with an empty `data` array, Kaiko returns **404** with the code `unknown_instrument` (for example "spy-usd is not a served US-listed option underlying"), so that a wrong ticker is caught immediately.
- **Uppercase ticker:** Amberdata accepted any case; Kaiko expects lowercase and returns **400** for `MSTR`, for example.
- **Start equal to or after end:** Kaiko returns **400**.
- **No data in the window:** outside the session or on a weekend, both sides return 200 with an empty `data` array.
- **Missing key:** a request without an API key returns **401** with the code `missing_api_key`.
- **Missing or invalid parameter:** both sides return 400. Kaiko error bodies follow the `application/problem+json` standard (RFC 9457): they carry a `status`, a `title`, a `detail` message that names the parameter at fault, and the code `invalid_parameter`.

5 Response data fields

Amberdata field	Kaiko field	Type	Change
`timestamp`	`value_time`	string	RENAMED
`exchange`	None	string	REMOVED
`currency`	None	string	REMOVED
`atm10days`	`atm_10_days`	number	RENAMED
`atm21days`	`atm_21_days`	number	RENAMED
`atm63days`	`atm_63_days`	number	RENAMED
`atm84days`	`atm_84_days`	number	RENAMED
`atm189days`	`atm_189_days`	number	RENAMED
`ratio`	`ratio`	number	IDENTICAL
`counter`	`counter`	integer	IDENTICAL
`richness`	`richness`	number	IDENTICAL

- **value_time** is the snapshot time, ISO 8601 (`"2026-10-22T18:20:00.000Z"`); Amberdata returned it as epoch milliseconds by default (`1792693200000`).

- **exchange, currency** are no longer repeated in the rows: the ticker is in your request and in `meta.url`.
- **atm_10_days to atm_189_days** are the at-the-money implied volatilities at the five constant maturities, annualised, decimal on both sides (`0.69627` = 69.627%). A tenor without a usable volatility is `null`, on both sides.
- **ratio** is the sum of the short-over-long ratios over the tenor pairs whose two legs are present, **counter** the number of those pairs (10 when every tenor is present) and **richness** is `ratio / counter`, 0 when no pair is present; unit-free, the same on both sides.

Units: decimal volatilities on both sides. Kaiko rounds them to 5 decimals (`0.69627`) where Amberdata returned the full floating-point expansion (`0.6962699890136719`); `ratio` and `richness` are returned at full precision on both sides.

6 Exchange & venue names

For this endpoint there is no exchange parameter: the options are US-listed, and the venue is fixed on both sides (Amberdata returned `exchange: "tradfi"` in every row, Kaiko does not repeat it).

Same coverage, lowercase ticker: Kaiko serves the same 23 underlyings as Amberdata, with `quote_asset=usd` for all of them. The only change is the case of the ticker you send: Amberdata's uppercase `currency` becomes Kaiko's lowercase `base_asset`, `MSTR` → `mstr`, `IBIT` → `ibit`. Please note that an uppercase ticker returns a 400 error.

Served underlyings: ARKB, BITB, BITO, BITQ, BITX, BMNR, COIN, CRCL, EETH, ETHA, ETHE, ETHU, FBTC, FETH, GBTC, GME, IBIT, MARA, MSTR, MSTU, MSTY, SATO, SBET. History on Kaiko starts on 14 October 2026 for every underlying; the Kaiko teams are working on a backfill to extend the available history.

7 Will your values change?

Short answer

The calculation is the same, on the same constant-maturity surface: `ratio`, `counter` and `richness` keep their scale and the ATM volatilities match once rounded. What changes is which rows a date range returns, and the daily row on US daylight time.

- **Same methodology:** ATM volatility at the 10, 21, 63, 84 and 189-day constant maturities of the vendor's surface; `richness` is the mean of the short-over-long ratios over the pairs present, with the same treatment of missing tenors and the same 0 when no pair is present.
- **Window boundary:** with the same dates, you now receive the row at the start of the window rather than the one at its end (§3). To receive exactly the rows Amberdata returned, shift both dates forward by one interval.
- **Daily rows:** Kaiko's `1d` row is the 15:00 America/New York bucket all year, with a provisional row for the day in progress (§3). Amberdata's `day` row was the 20:00 UTC bucket: the same bucket during US standard time, and no row at all during US daylight time, so you now receive daily rows in summer as well.
- **Snapshot within the bucket:** the vendor refreshes its surface several times per 5-minute bucket. Kaiko reads the first refresh it receives in the bucket; Amberdata read the last. On an active day `richness` can differ by a few thousandths on some buckets, with no systematic sign.
- **Precision:** ATM volatilities rounded to 5 decimals instead of the raw floating-point expansion.

8 Before / after example

Request (the same 18:20 snapshot: note the dates shifted by one interval)

Amberdata

```
GET https://api.amberdata.com
    /markets/derivatives/analytics
    /volatility/term-structures
    /richness/tradfi
    ?currency=MSTR
    &timeInterval=minute
    &startDate=2026-10-22T18:15:00
    &endDate=2026-10-22T18:20:00
x-api-key: <your-amberdata-key>
```

Kaiko

```
GET https://api.kaiko.io
    /analytics/volatility
    /tradfi-term-structure-richness
    ?base_asset=mstr
    &interval=5m
    &start_time=2026-10-22T18:20:00.000Z
    &end_time=2026-10-22T18:25:00.000Z
X-Api-Key: <your-kaiko-key>
```

Response (real responses, the same snapshot on both sides; fields reordered)

Amberdata


```
{
  "status": 200,
  "title": "OK",
  "description": "Successful request",
  "payload": {
    "data": [ {
      "timestamp": 1792693200000,
      "exchange": "tradfi",
      "currency": "MSTR",
      "atm10days": 0.6962699890136719,
      "atm21days": 0.6883299946784973,
      "atm63days": 0.6813200116157532,
      "atm84days": 0.6841899752616882,
      "atm189days": 0.7026399970054626,
      "ratio": 9.977246172792663,
      "counter": 10,
      "richness": 0.9977246172792663
    } ],
    "metadata": {
      "api-version": "2023-09-30"
    }
  }
}
```


Kaiko


```
{
  "data": [ {
    "value_time": "2026-10-22T18:20:00.000Z",
    "atm_10_days": 0.69627,
    "atm_21_days": 0.68833,
    "atm_63_days": 0.68132,
    "atm_84_days": 0.68419,
    "atm_189_days": 0.70264,
    "ratio": 9.977246162402107,
    "counter": 10,
    "richness": 0.9977246162402107
  } ],
  "meta": {
    "api_version": "2026-10-01",
    "count": 1,
    "next_url": null,
    "url": "https://api.kaiko.io/analytics/
volatility/tradfi-term-structure-richness?
base_asset=mstr&interval=5m&start_time=2026-10-22T
18:20:00.000Z&end_time=2026-10-22T18:25:00.000Z&ap
i_version=2026-10-01"
  }
}
```


Questions? Kaiko support · support@kaiko.com

docs.kaiko.com