

Varve Fundamentals

US company fundamentals, as first filed · by Varve Data

Methodology & Schema Guide

Version	0.1.0
Built	2026-10-08
Source	SEC Financial Statement Data Sets, 2009q3–2026_08_notes
Companies	16,545 (12,972 with a ticker; ~9,300 no longer filing)
Values	9,204,226
Validation	32/32 golden checks pass

Every number in this guide is generated from the shipped files at build time. If the guide and the data ever disagree, the guide is out of date.

Contents

1. Overview	3
2. Source data	4
3. How the dataset is built	4
4. Point-in-time design	5
5. Derived values	6
6. Companies, identifiers and survivorship	7
7. Data quality and validation	7
8. Known limitations	9
9. Releases and versioning	9
Appendix A. Table reference	10
Appendix B. Field dictionary	16
Appendix C. Recipes	20

1. Overview

Varve Fundamentals is a point-in-time, survivorship-free database of fundamentals for every US company that has filed XBRL financial statements with the SEC since 2009. It is built for one purpose: letting you test an idea against the numbers investors could actually see on the day, for the companies that actually existed on the day.

Three properties distinguish it from most free and many paid sources:

- **As first filed.** Every value is the number as originally reported, dated by the filing that made it public. The later, restated number is kept alongside it, never in place of it.
- **Survivorship-free.** The source contains every filer in every quarter, so the ~9,300 companies that have since delisted, been acquired or gone bankrupt are present, with their historical tickers where they can be recovered.
- **Traceable.** Every value carries the accession number of its source filing, so any figure can be checked against the original document in one click.

What's included

Table	Grain	Rows	Formats
varve_fundamentals_quarterly	One row per company × fiscal quarter-end	491,263	parquet, csv.gz
varve_fundamentals_annual	One row per company × fiscal year-end	126,277	parquet, csv.gz
varve_fundamentals_facts	One row per company × field × period × duration	9,204,226	parquet, csv.gz
varve_fundamentals_fact_versions	One row per company × field × period × duration × version	8,314,926	parquet, csv.gz
varve_fundamentals_ticker_history	One row per company × ticker × interval	49,073	parquet, csv.gz
varve_fundamentals_company_names	One row per company × registrant name × interval	22,119	parquet, csv.gz
varve_fundamentals_companies	One row per company (CIK)	16,545	parquet, csv.gz
varve_fundamentals_filings	One row per periodic filing (10-K, 10-Q, 20-F, 40-F and amendments)	414,329	parquet, csv.gz

Most users need only **quarterly** (or **annual**) plus **companies**. The **facts** table is the fully provenanced long form that everything else is derived from; **filings** indexes the source documents.

What's not included

- Prices, returns and market capitalisation. Join your own prices; `shares_outstanding` is provided.
- Periods before 2009 (pre-XBRL), and financial statements reported in currencies other than USD.
- Segment-level, geographic and footnote detail; only consolidated totals.

2. Source data

All data comes from the SEC's **Financial Statement Data Sets**, published quarterly by the Division of Economic and Risk Analysis (sec.gov/dera/data/financial-statement-data-sets). Each quarterly release contains every XBRL financial-statement value submitted to EDGAR in that quarter, by every filer. Two of its files are used:

File	Contents	Used for
sub.txt	One row per submission: accession number, CIK, company name, SIC, form type, fiscal labels, filing date, acceptance timestamp	Filing dates, fiscal labels, the companies and filings tables
num.txt	One row per numeric value: XBRL tag, period end, duration in quarters, unit, segment and co-registrant markers, value	Every value in the dataset

This release uses the 2009q3 through 2026_08_notes data sets. XBRL tagging became mandatory for the largest filers in 2009 and for all operating companies by 2011, which sets the start of coverage. Data published by the SEC is a work of the US government and is not subject to copyright.

3. How the dataset is built

The pipeline runs end to end on every release; nothing is edited by hand.

#	Stage	What happens	Why
1	Select filings	Keep 10-K, 10-Q, 20-F, 40-F and their amendments.	Periodic financial statements only; excludes registration statements and 8-Ks.
2	Consolidated totals only	Drop values carrying a segment or co-registrant marker.	In a typical quarter more than half of all rows are segment breakdowns; keeping them counts a company's revenue several times.
3	Map tags to fields	27 fields, each fed by a priority-ordered list of us-gaap tags (Appendix B).	Companies use different tags for the same concept, and revenue changed tags under ASC 606 in 2018. Series are merged, not truncated.
4	Unit and date sanity	Unit must suit the field (USD, shares, USD per share). Drop period ends outside 2008–today and any period ending after its own filing.	The source contains typo'd dates (years 1994 and 2923) and currency mismatches.
5	First-reported and latest	For each company, field, period and duration, keep the earliest-filed value and, separately, the latest-filed value.	The first is what was knowable at the time; the latest reflects restatements.
6	Quarterise	Turn year-to-date and annual figures into fiscal quarters (Section 5).	Most filers never tag a standalone Q4, and cash-flow items are reported year-to-date.
7	Label and identify	Attach fiscal labels, company metadata and every known ticker (Section 6).	Makes the data joinable to prices and readable by people.
8	Publish and validate	Write the tables, a manifest with SHA-256 checksums, and the validation report (Section 7).	Every release states how it was checked and how often checks fail.

4. Point-in-time design

A backtest that uses a number before it was public is not a backtest. The dataset is designed so that the default way of using it is look-ahead-free.

First-reported vs latest values

Companies re-report prior periods in later filings, sometimes with different numbers: restatements, reclassifications, discontinued operations. Most fundamentals sources overwrite history with the latest figure, so a 2016 backtest silently uses numbers first published in 2018. Here, `value` is always the first-reported number and `value_latest` the most recent; restated flags the difference.

Example. Berkshire Hathaway Inc (BRK-B) first reported fiscal-2016 revenue of \$223.60 billion on 27 Feb 2017. A filing on 25 Feb 2019 restated it to \$215.11 billion (-3.8%). A dataset holding only the restated figure would hand a backtest that number two years early.

Restatement is common: 10.7% of revenue values and 10.5% of net income values were later re-reported with a different number (full table in the validation report).

Two kinds of time, and full histories

Every value has a **valid time** (the period it describes, `period_end`) and a **known time** (when it became public, filed). Attributes that change, such as tickers, names, CUSIPs and restated values, are stored as intervals, so any date can be answered exactly:

- `fact_versions`: every distinct value reported for a period, valid from `known_from` until `known_to`. 74,660 facts were restated two or more times; a first/latest pair alone would lose the middle versions.
- `ticker_history`: which symbol a company traded under, `valid_from` to `valid_to`, observed from Form 3/4/5 filings (2006+) and SEC ticker lists (2017+). Handles renames (FB→META) and reused symbols (BBBY).
- `company_names`: the registrant name on each filing, as intervals.
- `13f_cusip_map` (Varve Events): `first_seen` and `last_seen` quarters for each CUSIP, which changes with mergers and reorganizations.

A change in precision only (4,843,000 in one filing, 4,843,349 in the next) is not counted as a restatement or a new version.

Which date to use

Column	Meaning	Use
<code>period_end</code>	The end of the fiscal period the value describes	Never as the availability date: values are not public until weeks later
<code>filed</code>	The date the value became public	Point-in-time joins (default)
<code>accepted</code>	EDGAR acceptance timestamp, to the second	Intraday or event studies; filings accepted after 17:30 ET carry the next business day's filing date

The median value is filed 45 days after its period ends.

```
import pandas as pd
q = pd.read_parquet("varve_fundamentals_quarterly.parquet")
```

```
# everything known on 31 March 2020: latest filed quarter per company
asof = (q[q.filed <= "2020-03-31"]
        .sort_values("period_end")
        .groupby("cik").tail(1))
```

One conservative caveat. `filed` is the first appearance of a value in an *XBRL* filing. For 2008–2010 periods of smaller companies that only began *XBRL* tagging in 2011, and for comparatives of fields a company started tagging later, the number was public earlier in an *HTML* filing. 147,249 values have a lag of more than two years for this reason. The error can delay a value, never leak it early.

5. Derived values

Three kinds of value are computed rather than read directly. Each is labelled in the `source` column of the facts table, so you can exclude them.

source	Applies to	Method	Share of values
reported	All fields	Read directly from the filing.	82.9%
ytd_derived	Cash-flow fields	Cash-flow statements report year-to-date. Quarter k = YTD(k) – YTD($k-1$), matched on fiscal quarter ends.	11.1%
q4_derived	Additive income-statement fields	Most filers report the fiscal year in the 10-K but never a standalone Q4. $Q4 = FY - (Q1 + Q2 + Q3)$, only when all three quarters exist.	4.1%
q4_ratio_derived	EPS and weighted share counts	Per-share figures are not additive. $Q4 \text{ shares} = 4 \times FY \text{ weighted shares} - (Q1 + Q2 + Q3)$, kept only within 0.5–2× of Q3 shares; $Q4 \text{ EPS} = Q4 \text{ net income} \div Q4 \text{ shares}$, falling back to Q3 shares.	1.91%

A derived value's `filed` date is the latest filing date among its inputs: it could not have been computed until all of them were public.

Why EPS is not derived by subtraction

Subtracting three quarters of EPS from annual EPS ignores the fact that the share count changes every quarter. The ratio method was chosen by testing both against filer-years that do report Q4 EPS directly, re-measured on every release:

Method	n	Exact to the cent	Within 5%	Within 20%	Wrong sign
ratio (used)	22,780	72.5%	78.8%	89.4%	0.6%
annual minus 3 quarters	27,895	70.9%	72.7%	88.4%	2.9%

Sign errors matter most: an earnings-surprise signal built on a Q4 that flips from profit to loss is not merely noisy but wrong.

Worked example

Apple does not tag a standalone fiscal Q4. Its quarter ending September 2023 is therefore derived: \$89,498M, labelled `q4_derived`, dated 03 Nov 2023 (the 10-K), from accession 0000320193-23-000106. Apple's own press release reported \$89,498M.

6. Companies, identifiers and survivorship

The primary key for a company is the SEC's **CIK**. Tickers are a poor key: they change (PCLN became BKNK), and they are recycled (a delisted company's symbol can later be issued to an unrelated company).

The SEC's bulk data carries no tickers at all, and the SEC's current ticker file lists only companies that still file, so on its own it can never name a company that delisted. Tickers are therefore recovered by combining ~73 archived snapshots of the SEC's own ticker file, August 2017 to present:

- **All symbols are kept.** `companies.tickers_all` lists every symbol seen for a CIK, so both old and new symbols resolve.
- **Recycled symbols stay with their first owner.** Snapshots are processed oldest-first, so a reused symbol points to the company that held it historically, which is what a backtest needs.
- **Coverage.** 78.4% of companies carry a ticker. Most of the rest stopped filing before August 2017 or never had a listed common stock (debt-only issuers, private filers).

Survivorship. 55.9% of the companies in the dataset no longer file. Because the source is organised by filing quarter rather than by today's company list, none of them needs special handling: each simply appears in the quarters in which it filed.

7. Data quality and validation

Every release ships with a validation report. Problems in the source filings are measured and disclosed rather than silently "fixed".

Golden checks: 32 of 32 pass

Values compared with figures in the companies' own earnings releases and annual reports, within $\pm 0.5\%$ ($\pm \$0.01$ for EPS). The set deliberately includes derived Q4s and a company that no longer files.

Check	Expected	Dataset	Source	Result
AAPL FY23 revenue	383,285M	383,285M	reported	pass
AAPL FY23 net income	96,995M	96,995M	reported	pass
AAPL FY23 diluted EPS	6.13	6.13	reported	pass
AAPL FY22 revenue	394,328M	394,328M	reported	pass
AAPL Q4 FY23 revenue (derived)	89,498M	89,498M	q4_derived	pass
AAPL Q4 FY23 net income (derived)	22,956M	22,956M	q4_derived	pass
AAPL Q4 FY23 diluted EPS (ratio-derived)	1.46	1.46	q4_ratio_derived	pass
MSFT Q4 FY23 diluted EPS (ratio-derived)	2.69	2.69	q4_ratio_derived	pass
AAPL Q1 FY24 revenue	119,575M	119,575M	reported	pass
AAPL Q1 FY24 net income	33,916M	33,916M	reported	pass
AAPL total assets 9/23	352,583M	352,583M	reported	pass
AAPL equity 9/23	62,146M	62,146M	reported	pass
MSFT FY23 revenue	211,915M	211,915M	reported	pass

Check	Expected	Dataset	Source	Result
MSFT FY23 net income	72,361M	72,361M	reported	pass
MSFT FY23 diluted EPS	9.68	9.68	reported	pass
MSFT FY22 revenue	198,270M	198,270M	reported	pass
MSFT Q4 FY23 revenue	56,189M	56,189M	q4_derived	pass
AMZN FY23 revenue	574,785M	574,785M	reported	pass
AMZN FY23 net income	30,425M	30,425M	reported	pass
AMZN FY22 net income (loss)	-2,722M	-2,722M	reported	pass
GOOGL FY23 revenue	307,394M	307,394M	reported	pass
GOOGL FY23 net income	73,795M	73,795M	reported	pass
KO FY23 revenue	45,754M	45,754M	reported	pass
KO FY23 net income	10,714M	10,714M	reported	pass
NVDA FY24 revenue	60,922M	60,922M	reported	pass
NVDA FY24 net income	29,760M	29,760M	reported	pass
TSLA FY23 revenue	96,773M	96,773M	reported	pass
TSLA FY23 net income	14,997M	14,997M	reported	pass
JNJ FY23 revenue	85,159M	85,159M	reported	pass
JNJ FY23 net income	35,153M	35,153M	reported	pass
Twitter FY21 revenue (delisted 2022)	5,078M	5,077M	reported	pass
Twitter FY21 net loss (delisted 2022)	-221M	-221M	reported	pass

Accounting identities

Identity	n	Within 1%	Within 5%
assets = liabilities + equity	318,102	82.6%	89.1%
gross profit = revenue - cost	139,579	95.7%	96.5%
FY revenue = sum of 4 reported quarters	31,027	89.0%	93.2%
FY net_income = sum of 4 reported quarters	36,802	89.7%	93.2%

The balance-sheet identity is the loosest by design: noncontrolling interests and temporary (mezzanine) equity sit outside both `total_liabilities` and `equity`. Mismatches in the quarterly-sum check are mostly companies whose annual and quarterly figures use different tags or were restated in between.

Known anomalies

- **Negative revenue** in 1.27% of quarterly values. Those marked reported are as filed: mostly micro-caps booking sales reversals and brokers with trading losses. Derived negatives usually mean the annual and quarterly figures came from different tags.
- **Scale errors** in 0.12% of quarterly revenue values: one quarter more than 100× both its neighbours, the signature of a filer tagging thousands as units. Left as filed; screen with a rolling-median filter if your method is sensitive to outliers.

8. Known limitations

- **Coverage starts in 2009** and is thin until 2011, when XBRL became mandatory for all operating companies.
- **USD only.** Foreign private issuers reporting in other currencies are excluded.
- **Tag choice is mechanical.** A field takes the highest-priority tag a filer used. A company that switches tags mid-year can produce an odd derived Q4; filter on `source = reported` when precision matters more than coverage.
- **Financial companies** have no single revenue line; revenue uses the closest top-line tag and is not comparable to an industrial company's revenue.
- **Tickers for companies that stopped filing before August 2017** are often missing. Use CIK.
- **Cover-page share counts** (`dei:EntityCommonStockSharesOutstanding`) are not in the source; `shares_outstanding` is the balance-sheet count.
- **Garbage in.** Tagging mistakes in filings are reproduced, not corrected. Every value names its source filing, so anything important can be verified.

9. Releases and versioning

- **Complete snapshots.** Each release is the full history, rebuilt from scratch. There is no patch file to apply and no state to carry.
- **Cadence.** A new release follows each quarterly SEC data set, normally within a week of its publication.
- **History is stable.** A first-reported value only changes if a pipeline error is corrected; such fixes bump the version and are listed in the changelog. New filings add rows and update `value_latest`.
- **Integrity.** `manifest.json` lists every file with its size and SHA-256 checksum, the source quarters used and the row counts.
- **Versions** follow semantic versioning: major for schema changes that break existing code, minor for new fields or tables, patch for data fixes.

Appendix A. Table reference

varve_fundamentals_quarterly

First-reported values only, pivoted wide: each quarter's flow fields (Q) alongside the balance sheet at that quarter-end (I). **Grain:** One row per company × fiscal quarter-end. **Primary key:** cik, period_end. **Rows:** 491,263. **Formats:** parquet, csv.gz.

Column	Type	Description
cik	integer	SEC Central Index Key: the permanent company identifier. Tickers change and get recycled; CIKs never do. Join on this.
ticker	string	Display ticker: the company's current symbol if it still has one, otherwise the earliest symbol recovered for it. NOT point-in-time; for every known symbol see companies.tickers_all.
name	string	Company name (latest).
period_end	date	Fiscal period end, normalised to month-end by the SEC (a 52/53-week year ending Saturday 30 Dec 2023 appears as 2023-12-31).
fiscal_year	integer	The company's own fiscal-year label for the period, from the filing that reported it.
fiscal_period	string (categorical)	Fiscal quarter label: Q1–Q4.
filed	date	Latest `filed` date across the row's values: the first date on which the WHOLE row was public. Use this for point-in-time joins.
27 field columns	float	One column per field, first-reported value; see Appendix B.

varve_fundamentals_annual

First-reported fiscal-year flow values (FY) with the year-end balance sheet (I). **Grain:** One row per company × fiscal year-end. **Primary key:** cik, period_end. **Rows:** 126,277. **Formats:** parquet, csv.gz.

Column	Type	Description
cik	integer	SEC Central Index Key: the permanent company identifier. Tickers change and get recycled; CIKs never do. Join on this.
ticker	string	Display ticker: the company's current symbol if it still has one, otherwise the earliest symbol recovered for it. NOT point-in-time; for every known symbol see companies.tickers_all.
name	string	Company name (latest).
period_end	date	Fiscal period end, normalised to month-end by the SEC (a 52/53-week year ending Saturday 30 Dec 2023 appears as 2023-12-31).
fiscal_year	integer	The company's own fiscal-year label for the period, from the filing that reported it.
filed	date	Latest `filed` date across the row's values: the first date on which the WHOLE row was public. Use this for point-in-time joins.
27 field columns	float	One column per field, first-reported value; see Appendix B.

varve_fundamentals_facts

The long, fully-provenanced table every other table is derived from. **Grain:** One row per company × field × period × duration. **Primary key:** cik, field, duration, period_end. **Rows:** 9,204,226. **Formats:** parquet, csv.gz.

Column	Type	Description
cik	integer	SEC Central Index Key: the permanent company identifier. Tickers change and get recycled; CIKs never do. Join on this.
field	string (categorical)	Field name; see the field dictionary.
duration	string (categorical)	Q = fiscal quarter, FY = fiscal year (both for flow fields), I = instant (balance-sheet fields).
period_end	date	Fiscal period end, normalised to month-end by the SEC (a 52/53-week year ending Saturday 30 Dec 2023 appears as 2023-12-31).
fiscal_year	integer	The company's own fiscal-year label for the period, from the filing that reported it.
fiscal_period	string (categorical)	Q1–Q4 for quarters and balance sheets, FY for fiscal years.
value	float	The value AS FIRST REPORTED, in the field's unit.
source	string (categorical)	How the value was obtained: reported, ytd_derived, q4_derived or q4_ratio_derived (see Derivations).
filed	date	Date the value became public: the filing date, or for a derived value the latest filing date among its inputs.
accepted	date	EDGAR acceptance timestamp of that filing (Eastern time). Filings accepted after 17:30 are dated the next business day.
accession	string	EDGAR accession number of the filing, e.g. 0000320193-23-000106. Filing index: sec.gov/Archives/edgar/data/{cik}/{accession without dashes}/.
value_latest	float	The most recent value reported for the same company, field and period (after any restatements).
filed_latest	date	Filing date of value_latest.
accession_latest	string	Accession number of the filing that reported value_latest.
restated	boolean	True when value_latest differs from value.

varve_fundamentals_fact_versions

Every distinct value a company reported for a period, with the interval during which it was the current number. Repeats of the same number in later filings extend a version; a change in precision only (4,843,000 vs 4,843,349) is not a new version. Covers values filers report directly (Q, FY, I); derived Q4 and year-to-date values are in facts. **Grain:** One row per company × field × period × duration × version. **Primary key:** cik, field, duration, period_end, version. **Rows:** 8,314,926. **Formats:** parquet, csv.gz.

Column	Type	Description
cik	integer	SEC Central Index Key: the permanent company identifier. Tickers change and get recycled; CIKs never do. Join on this.
field	string (categorical)	Field name; see the field dictionary.

Column	Type	Description
duration	string (categorical)	Q = fiscal quarter, FY = fiscal year, I = instant (balance sheet).
period_end	date	Fiscal period end, normalised to month-end by the SEC (a 52/53-week year ending Saturday 30 Dec 2023 appears as 2023-12-31).
version	integer	1 = first reported, 2 = first restatement, ...
n_versions	integer	How many distinct values this company-field-period has had.
value	float	The value as reported in this version.
known_from	date	Filing date from which this value was the current public number.
known_to	date	Filing date of the next version (exclusive); null while this version is current.
accepted	date	EDGAR acceptance timestamp of the filing that introduced this version.
accession	string	Accession number of the filing that introduced this version.
n_reports	integer	How many filings reported this exact value (originals plus comparatives).
last_reported	date	Filing date of the last filing that reported this value.

varve_fundamentals_ticker_history

When each company traded under each symbol, observed from SEC filings: the issuer symbol on every Form 3/4/5 (2006+) and archived SEC ticker lists (2017+). Dates are first/last observation, not official listing dates. A company can hold several tickers at once (share classes) and a symbol can pass between companies (recycled tickers). **Grain:** One row per company × ticker × interval. **Primary key:** cik, ticker, valid_from. **Rows:** 49,073. **Formats:** parquet, csv.gz.

Column	Type	Description
cik	integer	SEC Central Index Key: the permanent company identifier. Tickers change and get recycled; CIKs never do. Join on this.
ticker	string	Ticker symbol.
valid_from	date	First date the company was observed under this symbol.
valid_to	date	Last date observed (null while the symbol is current). When the company's insider filings switch symbols, the old interval ends at the switch.
current	boolean	True if the symbol is in SEC's current ticker list for this company.
last_seen	date	Last observation date, also for current symbols.
superseded_by	string	The symbol the company switched to, when the interval ended in a rename.
sources	string	insider (Form 3/4/5 issuer symbol), sec_list (SEC ticker list), or both.
n_observations	integer	Number of filings / list snapshots supporting the interval.

varve_fundamentals_company_names

The registrant name printed on each periodic filing, collapsed into intervals (2009+). **Grain:** One row per company × registrant name × interval. **Primary key:** cik, valid_from. **Rows:** 22,119. **Formats:** parquet, csv.gz.

Column	Type	Description
cik	integer	SEC Central Index Key: the permanent company identifier. Tickers change and get recycled; CIKs never do. Join on this.
name	string	Registrant name as filed.
valid_from	date	Filing date of the first filing under this name.
valid_to	date	Filing date of the first filing under the next name (null = current name).
last_seen	date	Filing date of the last filing under this name.
n_filings	integer	Number of periodic filings under this name.

varve_fundamentals_companies

Every company with at least one value in the dataset, active or not. **Grain:** One row per company (CIK). **Primary key:** cik. **Rows:** 16,545. **Formats:** parquet, csv.gz.

Column	Type	Description
cik	integer	SEC Central Index Key: the permanent company identifier. Tickers change and get recycled; CIKs never do. Join on this.
ticker	string	Display ticker: the company's current symbol if it still has one, otherwise the earliest symbol recovered for it. NOT point-in-time; for every known symbol see companies.tickers_all.
ticker_current	string	Symbol in SEC's current ticker file; empty if the company no longer has one.
tickers_all	string	Every symbol ever recovered for this company, pipe-separated (e.g. BKNG PCLN).
name	string	Latest registered company name.
former_names	string	Earlier names reported to the SEC, pipe-separated.
sic	integer	Standard Industrial Classification code assigned by the SEC.
country_inc	string	Country of incorporation (ISO code).
state_inc	string	State of incorporation, for US companies.
fiscal_year_end_mmdd	string	Fiscal year end as MMDD (e.g. 0930).
first_filed	date	Date of the company's first 10-K/10-Q/20-F/40-F in the source.
last_filed	date	Date of its most recent one. A date well in the past means the company stopped filing (delisted, acquired, bankrupt, deregistered).
n_filings	integer	Number of periodic filings in the source.

varve_fundamentals_filings

Index of every filing the dataset draws on. **Grain:** One row per periodic filing (10-K, 10-Q, 20-F, 40-F and amendments). **Primary key:** accession. **Rows:** 414,329. **Formats:** parquet, csv.gz.

Column	Type	Description
accession	string	EDGAR accession number (primary key).
cik	integer	SEC Central Index Key: the permanent company identifier. Tickers change and get recycled; CIKs never do. Join on this.
form	string	Form type: 10-K, 10-Q, 20-F, 40-F, or an amendment (/A).
period_end	date	Balance-sheet date of the report, normalised to month-end.
fiscal_year	integer	Fiscal year the filer assigned to the report.
fiscal_period	string	Fiscal period the filer assigned: Q1–Q3, FY (annual report).
filed	date	Filing date.
accepted	date	EDGAR acceptance timestamp (Eastern time).
superseded	boolean	True if the filer later amended this submission.

How the tables join

Every table joins on **cik**, the SEC Central Index Key. Tickers change and get reused; CIKs never do.

accession links a value or event to its source filing, and **cusip** links 13F holdings to companies through the CUSIP map. The same diagram ships as `join_diagram.svg`.



Appendix B. Field dictionary

Monetary values are in USD, not thousands. Duration fields appear as fiscal quarters (Q) and fiscal years (FY); balance-sheet fields as instants (I). **Q4 method**: how a quarter is produced when the filer reports only the year (Section 5). **Companies**: number with at least one value.

Income statement

Field	Unit	Q4 method	Companies	Definition
revenue	USD	subtraction	13,377	Total revenue / net sales. For banks and insurers, which report no single revenue line, the closest top-line tag is used (interest and dividend income, net interest income, premiums earned); see tag priority.
cost_of_revenue	USD	subtraction	8,478	Cost of goods and/or services sold.
gross_profit	USD	subtraction	7,105	Gross profit as reported by the filer, never computed. Many service companies and financials do not report it; compute revenue – cost_of_revenue yourself where you need it.
sga	USD	subtraction	5,068	Selling, general and administrative expense.
rnd	USD	subtraction	5,231	Research and development expense.
operating_income	USD	subtraction	12,880	Operating income (loss).
interest_expense	USD	subtraction	10,874	Interest expense. Positive = expense.
income_tax	USD	subtraction	11,660	Income tax expense (benefit). Positive = expense, negative = benefit.
net_income	USD	subtraction	16,370	Net income (loss) attributable to the parent company. Falls back to consolidated profit including noncontrolling interests, then to income available to common shareholders.
eps_diluted	USD/shares	ratio	13,914	Diluted earnings per share, USD per share. Includes filers that report a single combined basic-and-diluted figure.
eps_basic	USD/shares	ratio	11,495	Basic earnings per share, USD per share.
shares_diluted	shares	ratio	12,905	Weighted-average diluted shares outstanding over the period: the denominator for per-share maths, not for market cap.
shares_basic	shares	ratio	11,464	Weighted-average basic shares outstanding over the period.

Cash flow

Field	Unit	Q4 method	Companies	Definition
operating_cash_flow	USD	ytd	15,510	Net cash provided by (used in) operating activities.
capex	USD	ytd	11,250	Payments to acquire property, plant and equipment. A positive number is a cash outflow. Free cash flow = operating_cash_flow – capex.

Field	Unit	Q4 method	Companies	Definition
depreciation_amortization	USD	ytd	10,078	Depreciation, depletion and amortization, from the cash-flow statement.
stock_comp	USD	ytd	10,789	Share-based compensation: the non-cash add-back in the cash-flow statement.
dividends_paid	USD	ytd	4,489	Cash dividends paid. A positive number is a cash outflow.
buybacks	USD	ytd	6,246	Payments for repurchase of common stock. A positive number is a cash outflow.

Balance sheet

Field	Unit	Q4 method	Companies	Definition
cash	USD	—	15,886	Cash and cash equivalents. Where only the fallback tag exists, includes restricted cash (the presentation required after ASU 2016-18).
current_assets	USD	—	13,147	Total current assets.
total_assets	USD	—	16,324	Total assets.
current_liabilities	USD	—	13,105	Total current liabilities.
total_liabilities	USD	—	14,431	Total liabilities. Many filers never tag this total (they report only liabilities-and-equity), so coverage is lower than total_assets.
long_term_debt	USD	—	7,029	Long-term debt, non-current portion where tagged; the fallback tags may include current maturities.
equity	USD	—	15,413	Stockholders' equity attributable to the parent; falls back to total equity including noncontrolling interests.
shares_outstanding	shares	—	13,719	Common shares outstanding at the balance-sheet date. Falls back to shares issued (which includes treasury shares). Use for market cap with a price dated on or after `filed`.

XBRL tag priority

When a filing contains several tags for the same field, the first in this list wins. Across filings, tags are merged into one series.

Field	us-gaap tags, highest priority first
revenue	RevenueFromContractWithCustomerExcludingAssessedTax RevenueFromContractWithCustomerIncludingAssessedTax Revenues SalesRevenueNet SalesRevenueGoodsNet RevenuesNetOfInterestExpense InterestAndDividendIncomeOperating InterestIncomeExpenseNet PremiumsEarnedNet

Field	us-gaap tags, highest priority first
cost_of_revenue	CostOfRevenue CostOfGoodsAndServicesSold CostOfGoodsSold CostOfServices
gross_profit	GrossProfit
sga	SellingGeneralAndAdministrativeExpense
rnd	ResearchAndDevelopmentExpense ResearchAndDevelopmentExpenseExcludingAcquiredInProcessCost
operating_income	OperatingIncomeLoss
interest_expense	InterestExpense InterestExpenseNonoperating
income_tax	IncomeTaxExpenseBenefit
net_income	NetIncomeLoss ProfitLoss NetIncomeLossAvailableToCommonStockholdersBasic
eps_diluted	EarningsPerShareDiluted EarningsPerShareBasicAndDiluted
eps_basic	EarningsPerShareBasic
shares_diluted	WeightedAverageNumberOfDilutedSharesOutstanding WeightedAverageNumberOfShareOutstandingBasicAndDiluted
shares_basic	WeightedAverageNumberOfSharesOutstandingBasic
operating_cash_flow	NetCashProvidedByUsedInOperatingActivities NetCashProvidedByUsedInOperatingActivitiesContinuingOperations
capex	PaymentsToAcquirePropertyPlantAndEquipment PaymentsToAcquireProductiveAssets PaymentsForCapitalImprovements
depreciation_amortization	DepreciationDepletionAndAmortization DepreciationAndAmortization DepreciationAmortizationAndAccretionNet
stock_comp	ShareBasedCompensation
dividends_paid	PaymentsOfDividends PaymentsOfDividendsCommonStock
buybacks	PaymentsForRepurchaseOfCommonStock
cash	CashAndCashEquivalentsAtCarryingValue CashCashEquivalentsRestrictedCashAndRestrictedCashEquivalents Cash
current_assets	AssetsCurrent
total_assets	Assets
current_liabilities	LiabilitiesCurrent
total_liabilities	Liabilities
long_term_debt	LongTermDebtNoncurrent LongTermDebt LongTermDebtAndCapitalLeaseObligations

Field	us-gAAP tags, highest priority first
equity	StockholdersEquity StockholdersEquityIncludingPortionAttributableToNoncontrollingInterest
shares_outstanding	CommonStockSharesOutstanding CommonStockSharesIssued

Appendix C. Recipes

Load a company's history

```
import pandas as pd
q = pd.read_parquet("varve_fundamentals_quarterly.parquet")
c = pd.read_parquet("varve_fundamentals_companies.parquet")

cik = c.loc[c.tickers_all.str.split("|").apply(lambda s: "PCLN" in s), "cik"].iloc[0]
bkng = q[q.cik == cik].set_index("period_end")
bkng[["revenue", "net_income", "operating_cash_flow", "capex", "filed"]].tail(8)
```

Point-in-time join to prices

```
# prices: DataFrame[cik, date, close] (your own source)
q = q.sort_values("filed")
prices = prices.sort_values("date")
panel = pd.merge_asof(prices, q, left_on="date", right_on="filed",
                      by="cik", direction="backward") # only values already filed
panel["market_cap"] = panel.close * panel.shares_outstanding
```

Trailing twelve months

```
q = q.sort_values(["cik", "period_end"])
for f in ["revenue", "net_income", "operating_cash_flow", "capex"]:
    q[f + "_ttm"] = q.groupby("cik")[f].transform(lambda s: s.rolling(4).sum())
# require four consecutive quarters: check period_end spacing before trusting a TTM
```

Use first-reported or restated values

```
f = pd.read_parquet("varve_fundamentals_facts.parquet")
rev = f[(f.field == "revenue") & (f.duration == "FY")]
rev[["cik", "period_end", "value", "filed", "value_latest", "filed_latest", "restated"]]
```

Open the source filing for any value

```
row = rev.iloc[-1]
print(f"https://www.sec.gov/Archives/edgar/data/{row.cik}/"
      f"{row.accession.replace('-', '')}")
```

As-of value from the full version history

```
v = pd.read_parquet("varve_fundamentals_fact_versions.parquet")
d = pd.Timestamp("2018-06-30")
asof = v[(v.known_from <= d) & (v.known_to.isna() | (v.known_to > d))]
# exactly one row per company-field-period: the number the market knew on d
```

Point-in-time ticker ☐ CIK for joining prices

```
t = pd.read_parquet("varve_fundamentals_ticker_history.parquet")
on = lambda d: t[(t.valid_from <= d) & (t.valid_to.isna() | (t.valid_to >= d))]
on(pd.Timestamp("2017-06-30")).query("ticker == 'PCLN'") # Priceline (cik 1075531)
on(pd.Timestamp("2024-06-30")).query("ticker == 'BBBY'") # nobody: the symbol was idle
```

Exclude derived values

```
strict = f[f.source == "reported"] # no Q4 or YTD derivations
```

Provided as-is, without warranty of accuracy. Not investment advice. Source data: U.S. Securities and Exchange Commission.