Engineer

Dataset

a collection of data published as one unit

What it is

A dataset, also written data set, is a bounded body of data that people handle as one thing: it has a name, a source, a period it covers and a shape its records share, and those four facts say where it starts and stops. A month of meter readings in one file is a dataset; so is a nightly export of three related tables. The word says nothing about size or format, only that the data inside belongs together.

The pages on this site use the word in that working sense. One speaks of a historical dataset in JSONL format and of 1-day and 30-day datasets used for benchmarks, another of curated datasets and tables exported for loading elsewhere, a third of vast datasets generated by smart meters. Each time, the word draws a line around data before something is done to it.

The W3C's Data Catalog Vocabulary, DCAT, gives the word a formal frame. Version 3 became a W3C Recommendation on 22 August 2024, and it defines a dataset as a "collection of data, published or curated by a single agent, and available for access or download in one or more serializations or formats". The notion is broad on purpose: numbers, text, imagery and sound all qualify. DCAT then separates three things that everyday speech runs together:

- **Dataset**: the data as an idea, such as hourly consumption per zone for 2024.
- **Distribution**: one accessible form of it, such as a downloadable CSV file; one dataset may have several.
- **Dataset series**: datasets published separately that share enough to be grouped, such as one file per month.

So two files in different formats may be one dataset, and two monthly files under one name may be two members of a series.

The word has a second, narrower meaning in Google BigQuery. There a dataset is a container for tables, views and other resources, and it sits inside a project. A table has to belong to a dataset, so a dataset is created before any data is loaded, and it is addressed as `project.dataset` in GoogleSQL and as `project:dataset` in the bq command-line tool. Context decides which meaning is in play.

Use the concept in a decision

Put “What is a dataset?” to work in a real decision. We can help relate the concept to the system you are building or buying. Dashboards and reports showing the handful of figures a decision actually turns on, refreshed without being asked.

Explore our services Get in touch →

Test the assumptions behind the term

Use the explanation and linked work to frame questions for your project. We can examine the assumptions with you before a tool is chosen. Brand communication informed by audience response — brand communication informed by audience response

Brand communication informed by audience response Get in touch →

Why it matters

A boundary makes data comparable. Two sets of readings can be compared only when both cover the same period, the same places and the same spacing between records, and naming a dataset is how those terms get fixed. DCAT gives the spacing its own property, temporal resolution, meaning the smallest gap between the dataset’s items, written as a duration such as `P1D` for one day. A dataset that states its resolution tells the reader whether its rows are hours, quarter-hours or something irregular before the first sum.

Description has to travel with the data. DCAT lists what a distribution can say about itself: a media type, a format, a compression format, a packaging format, a byte size and a checksum, the last of which lets a receiver verify that the contents have stayed unchanged. The specification’s own example of a compression format is gzip. A file that arrives with those facts can be checked and opened; a file that arrives bare has to be guessed at.

The format of a distribution decides who can read it. Three are common:

- **CSV.** RFC 4180, from 2005, records the layout most implementations seem to follow: one record per line, an optional header line, double quotes around any field that holds a comma, a line break or a double quote, and a double quote inside a field written twice. Almost any tool opens it.
- **JSON Lines.** One valid JSON value per line, UTF-8, with `\n` as the line separator, saved as `.jsonl` or, gzipped, `.jsonl.gz`. A final newline is strongly recommended, not required. Programs read it one record at a time, and Unix-style text tools and shell pipes handle it well.
- **Parquet.** An open-source data file format that keeps the values of each column together. It is built to store and retrieve data efficiently, and its compression and encoding schemes let it handle complex data in bulk.

Here is one record in the first two, where the comma inside the note forces quotes in the CSV:

```
id,read_at,kwh,note
M-17,2024-01-08T10:00:00Z,3.21,"late, manual"

{"id":"M-17","read_at":"2024-01-08T10:00:00Z","kwh":3.21,"note":"late, manual"}
```

What it looks like in practice

Shaping one dataset out of many sources was the task when electricity data arrived with each source's own reporting frequency, from hourly intervals to 15-minute intervals and in some cases irregular timestamps, which made a coherent, comparable dataset hard to produce. The work took raw event data from a historical JSONL dataset and turned it into hourly consumption, one row per hour and per zone, summing the production mix in each interval, adding imports and subtracting exports. Three variants were written, in Python, JQ and SQL. The Python pipeline parsed the JSONL files into data frames, resampled the time series to hourly intervals, and exported the results as CSV or loaded them into a lightweight SQLite database for inspection. Benchmarked on 1-day and 30-day historical datasets, it was the fastest of the three, at 34 ms for one day and 180 ms for thirty. The work between input and output is a change of grain, from each source's own spacing to one fixed row per hour and zone.

Moving a dataset is a different job. A set of six ETL/ELT pipelines carried data from a remote warehouse behind an IPsec VPN into Google BigQuery. The data arrived as compressed archives containing Parquet, CSV and .bak files, and Bash and Python scripts classified each file by type and schema. Once procedures had

cleaned, normalised and enriched the rows, curated datasets and tables were exported to intermediate formats, compressed with gzip and transferred to a Google Cloud Storage bucket, from which the bq CLI and Python scripts loaded them into BigQuery. Getting those datasets there by hand had taken 3+ hours; automated, the trip took under 20 minutes, and the tables were refreshed every day where they had been refreshed once a week.

Reading one is the third. Smart meters across multiple geographic regions produced vast datasets behind more than 500 analysis reports written in pure SQL on PostgreSQL and BigQuery. The data held geolocation coordinates, meter IDs, timestamped energy usage and environmental metadata. Each report carried annotated SQL so that colleagues could reproduce and audit it, and troubleshooting notes recorded common data quality issues, such as missing GPS coordinates or corrupted meter values, with recommended handling. Datasets that colleagues can reproduce and audit are datasets planners will trust, and the work contributed to a 10–15% gain in operational planning efficiency.

Where it goes wrong

Most dataset trouble is a fact about the data that stayed in one person's head.

The grain goes unstated. Hourly rows added to quarter-hour rows count some intervals four times over. Write the grain down in one sentence, such as one row per hour per zone in UTC, and align every source to it before totalling.

A gap reads as zero. An empty field, a null and a 0 mean three different things, and a total quietly treats a missing reading as nothing used. Missing GPS coordinates and corrupted meter values were common enough on the smart-meter project to earn documented handling. Decide per column whether a gap is dropped, filled or flagged, and say which in the dataset's notes.

The CSV dialect drifts. RFC 4180 documents the common layout, and real files stray from it: another separator, a missing header, a field with a comma and its quotes left off. The RFC leaves the header optional, signalled only by a MIME parameter that a file copied to disk leaves behind. Check each new file's first line against the expected header before loading.

A copy forgets where it came from. A curated extract loaded into a second system looks just like the original a month later. Record the source, the period, the extraction time and a checksum next to every copy so that a report can say which extract it read.

A BigQuery dataset lands in the wrong place. A dataset's location is fixed at creation; moving it means copying or recreating the dataset. Every table in one query must sit in datasets in the same location, so choose the location before the first load.

Further reading

- Data Catalog Vocabulary (DCAT) - Version 3 — the W3C definitions of dataset, distribution and dataset series, with the properties that describe each.
- BigQuery: Introduction to datasets — the container sense of the word: projects, locations, naming and limits.
- JSON Lines — the three rules of the format and its file-naming conventions.
- RFC 4180: Common Format and MIME Type for CSV Files — the CSV layout most tools follow, its quoting rules and the text/csv type.
- Apache Parquet: Overview — what the column-oriented file format is and where its specification lives.



Glossary

Choose for your actual constraints

Bring a representative dataset, the questions it must answer and the current bottleneck. We can agree how to measure quality, throughput and recovery.

See what it costs Get in touch →

Make the terminology practical

Ask how “What is a dataset?” applies to your organisation. A concrete example from your current system is enough to start the discussion.

Explore our services

welcome@engineer.company +45 50 10 61 84 Copenhagen, Denmark

Take this page with you

Contact card

Carefully Crafted With Love by D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – present · Engineer ApS · All rights reserved.

Printed from https://engineer.company/glossary/dataset/