

SCIENCE (HTTPS://DATASCIENC

ENGLISHARTIFICIAL INTELLIGENCE(HTTPS://DATASCIENCE.EU/CATEGORY/ARTIFICIAL-

DATA SCIENCE (HTTPS://DATAS

(<https://click.linksynergy.com/fs-bin/click?id=sM3/YRO9V9Y&offerid=916798.562&subid=0&type=4>)

WIKI (HTTPS://DATASCIENCE.EU/CATEGORY/WIKI/)

What the heck is time-series data (and why do I need a time-series database)?

Subscribe

* indicates required

Email Address *

First Name

Last Name

Language

SUBSCRIBE

(<https://datascience.eu/author/data-science-team/>) by Data Science Team (<https://datascience.eu/author/data-science-team/>)

• 2 years ago

(<https://click.linksynergy.com/fs-bin/click?id=sM3/YRO9V9Y&offerid=916798.562&subid=0&type=4>)

(<https://click.linksynergy.com/fs-bin/click?id=sM3/YRO9V9Y&offerid=916798.562&subid=0&type=4>)

A primer on time-series data, and why you'll not

Categories

Select Categor

English

(<https://datascience.eu/wiki/what-the->

want to use a “normal” database to store it.

Here’s a riddle: what do self-driving Teslas, autonomous Wall Street trading algorithms, smart homes, transportation networks that fulfill lightning-fast same-day deliveries, and an open-data-publishing NYPD have in common?

For one, they’re signs that our world is changing at warp speed, because of our ability to capture and analyze more and more data in faster and faster ways than before.

However, if you look closely, you’ll notice that every of those applications requires a special quite data:

Self-driving cars continuously collect data about how their local environment is changing around them.

Autonomous trading algorithms continuously collect data on how the markets are changing.

Our smart homes monitor what’s happening inside them to manage temperature, identify intruders, and answer our beck-and-call (“Alexa, play some relaxing music”).

Our retail industry monitors how their assets are moving with such precision and efficiency that cheap same-day delivery may be a luxury that a lot of folks deem granted.

The NYPD tracks its vehicles to permit us to carry them more accountable (e.g., for analyzing 911 response times).

These applications believe a sort of data that

heck-is-time-series-data-and-why-do-i-need-a-time-series-database/)

Deutsch

(<https://datascience.eu/de/wiki-de/was-zum-teufel-sind-zeitreihendaten-und-warum-brauche-ich-eine-zeitreihendatenbank/>)

Français

(<https://datascience.eu/fr/wiki-fr/quest-ce-que-cest-que-des-donnees-chronologiques-et-pourquoi-ai-je-besoin-dune-base-de-donnees-chronologiques/>)

Español

(<https://datascience.eu/es/wiki-es/que-diablos-son-los-datos-de-series-temporales-y-por-que-necesito-una-base-de-datos-de-series-temporales/>)

Português

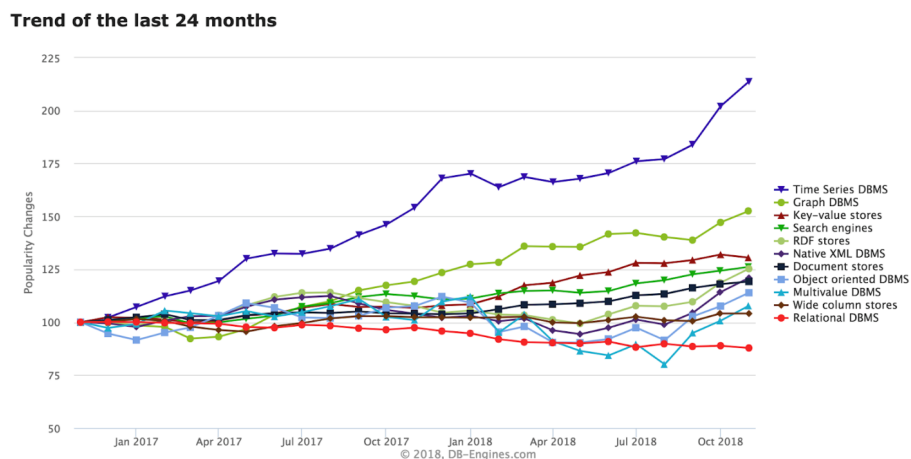
(<https://datascience.eu/pt/wiki-pt/o-que-diabos-sao-dados-de-series-temporais-e-por-que-preciso-de-um-banco-de-dados-de-series-temporais/>)

Italiano

(<https://datascience.eu/it/wiki-it/che-diavolo-sono->

measures how things change over time. Where time isn't just a metric, but a primary axis. This is often time-series data and it's beginning to play a bigger role in our world.

Software developer usage patterns already reflect this. In fact, over the past 24 months time-series databases (TSDBs) have steadily remained the fastest growing category of databases:



What is time-series data?

Some consider “time-series data” as a sequence of knowledge points, measuring an equivalent thing over time, stored in time order. That’s true, but it just scratches the surface.

Others might imagine of a series of numeric values, each paired with a timestamp, defined by a reputation and a group of labeled dimensions (or “tags”). This is often perhaps a method to model time-series data, but not a definition of the info itself.

Here’s a basic illustration. Imagine sensors collecting data from three settings: a city, farm, and factory. during this example, each of those

i-dati-delle-serie-temporali-e-perche-ho-bisogno-di-una-banca-dati-delle-serie-temporali/) Netherlands (https://datascience.eu/nl/wiki-nl/wat-is-in-hemelsnaam-tijdreeksgegevens-en-waarom-heb-ik-een-tijdreeksendatabase-nodig/) Polski (https://datascience.eu/pl/wiki-pl/co-to-jest-do-cholery-za-dane-z-serii-czasowej-i-po-co-mi-baza-danych-z-serii-czasowej/) Русский (https://datascience.eu/ru/wiki-ru/%d1%87%d1%82%d0%be-%d1%82%d0%b0%d0%ba%d0%be%d0%b5-%d0%b4%d0%b0%d0%bd%d0%bd%d1%8b%d0%b5-%d0%b2%d1%80%d0%b5%d0%bc%d0%b5%d0%bd%d0%bd%d1%8b%d1%85-%d1%80%d1%8f%d0%b4%d0%be%d0%b2-%d0%b8-%d0%b7%d0%b0%d1%87/)

sources periodically sends new readings, creating a series of measurements collected over time.

Here's another example, with real data from the town of latest York, showing taxicab rides for the primary few seconds of 2018. As you'll see, each row may be a "measurement" collected at a selected time:

tpep_pickup_datetime	tpep_dropoff_datetime	fare_amount	passenger_count	trip_distance
2018-01-01 00:00:17	2018-01-01 00:10:55	12	1	3.76
2018-01-01 00:00:16	2018-01-01 00:00:49	55	1	0
2018-01-01 00:00:15	2018-01-01 00:14:17	10.5	1	2.06
2018-01-01 00:00:15	2018-01-01 00:08:21	7	2	1.2
2018-01-01 00:00:14	2018-01-01 00:11:38	14	1	4
2018-01-01 00:00:14	2018-01-01 00:04:32	5	1	0.9
2018-01-01 00:00:13	2018-01-01 00:07:03	6	1	0.9
2018-01-01 00:00:11	2018-01-01 00:06:05	7	1	1.7
2018-01-01 00:00:06	2018-01-01 00:24:34	23.5	1	6.9
2018-01-01 00:00:04	2018-01-01 00:08:13	8	1	1.59
2018-01-01 00:00:04	2018-01-01 00:13:24	13.5	1	3.6
2018-01-01 00:00:03	2018-01-01 00:03:52	5.5	3	0.99
2018-01-01 00:00:03	2018-01-01 00:21:06	20.5	1	6.1
2018-01-01 00:00:02	2018-01-01 00:08:48	7.5	1	1.36
2018-01-01 00:00:00	2018-01-01 00:00:00	27	1	9.14

There are many other forms of time-series data. To call a few: DevOps monitoring data, mobile/web application event streams, industrial machine data, scientific measurements.

These datasets primarily have 3 things in common:

The data that arrives is nearly always recorded as a replacement entry

The data typically arrives in time order

Time may be a primary axis (time-intervals are often either regular or irregular)

In other words, time-series data workloads are generally "append-only." While they'll got to correct

erroneous data after the very fact , or handle delayed or out-of-order data, these are exceptions, not the norm.

You may ask: How is that this different than simply having a time-field during a dataset? Well, it depends: how does your dataset track changes? By updating the present entry, or by inserting a replacement one?

When you collect a replacement reading for `sensor_x`, does one overwrite your previous reading, or does one create a fresh reading during a separate row? While both methods will provide you the present state of the system, only by writing the new reading during a separate row will you be ready to track all states of the system over time.

Simply put: time-series datasets track changes to the general system as INSERTs, not UPDATEs.

This practice of recording each and each change to the system as a replacement , different row is what makes time-series data so powerful. It allows us to live change: analyze how something changed within the past, monitor how something is changing within the present, predict how it's going to change within the future.

Put simply, here's how i prefer to define time-series data: data that collectively represents how a system/process/behavior changes over time.

This is quite just a tutorial distinction. By centering our definition around "change", we will start to spot time-series datasets that we aren't collecting today, but that we should always be collecting down the road . In fact, often people have time-

series data but don't know it .

Imagine you maintain an internet application. Whenever a user logs in, you'll just update a "last_login" timestamp for that user during a single row in your "users" table. But what if you treated each login as a separate event, and picked up them over time? Then you could: track historical login activity, see how usage is (in-/de-)creasing over time, bucket users by how often they access the app, and more.

This example illustrates a key point: by preserving the inherent time-series nature of our data, we are ready to preserve valuable information on how that data changes over time. Another point: event data is additionally time-series data.

Of course, storing data at this resolution comes with a clear problem: you finish up with tons of knowledge , rather fast. So that's the catch: time-series data piles up very quickly.

Having tons of knowledge creates problems when both recording it and querying it during a performant way, which is why people are now turning to time-series databases.

Why do i want a time-series database?

You might ask: Why can't I just use a "normal" (i.e., non-time-series) database?

The truth is that you simply can, and a few people do. Yet why are TSDBs the fastest growing category of databases today? Two reasons: (1) scale and (2) usability.

Scale: Time-series data accumulates very quickly. (For example, one connected car will collect 4,000 GB of knowledge per day.) And normal databases aren't designed to handle that scale. Relational databases fare poorly with very large datasets; NoSQL databases fare better at scale, but can still be outperformed by a database fine-tuned for time-series data. In contrast, time-series databases (which are often supported relational or NoSQL databases) handle scale by introducing efficiencies that are only possible once you treat time as a primary class citizen. These efficiencies end in performance improvements, including higher ingest rates, faster queries at scale (although some support more queries than others), and better data compression.

Usability: TSDBs also typically include functions and operations common to time-series data analysis like data retention policies, continuous queries, flexible time aggregations, etc. albeit scale it not a priority at the instant (e.g., if you're just beginning to collect data), these features can still provide a far better user experience and make your life easier.

This is why developers are increasingly adopting time-series databases and using them for a spread of use cases:

Monitoring software systems: Virtual machines, containers, services, applications

Monitoring physical systems: Equipment, machinery, connected devices, the environment, our homes, our bodies

Asset tracking applications: Vehicles, trucks,
physical containers, pallets

Financial trading systems: Classic securities, newer
cryptocurrencies

Eventing applications: Tracking user/customer
interaction data

Business intelligence tools: Tracking key metrics
and therefore the overall health of the business

(and more)

Even then, you'll got to pick a time-series database
that most closely fits your data model and
write/read patterns.

A parting thought: Is all data time-series data?

For the past decade approximately , we've lived
within the era of "Big Data ([https://datascience.eu
/computer-programming/what-is-big-data/](https://datascience.eu/computer-programming/what-is-big-data/))",
collecting massive amounts of data about our
world and applying computational resources to
form sense of it.

Even though this era started with modest
computing technology, our ability to capture, store,
and analyze data has improved at an exponential
pace, because of major macro-trends: Moore's law,
Kryder's law, cloud computing, a whole industry of
"big data" technologies.

Now we'd like more. We are not any longer content
to only observe the state of the planet , but we
now want to live how our world changes over time,
right down to sub-second intervals. Our "big data"

datasets are now being dwarfed by another sort of data, one that relies heavily on time to preserve information about the change that's happening.

Does all data begin as time-series data? Recall the sooner web application example: we had time-series data but didn't know it . Or consider any "normal" dataset. Say, the present accounts and balances at a serious retail bank. Or the ASCII text file for a software project. Or the text for this text .

Typically we elect to store the newest state of the system, but instead, what if we stored every change and computed the newest state at query time? Isn't a "normal" dataset just a view on top of an inherently time-series dataset (cached for performance reasons)? Don't banks have transaction ledgers? (And aren't blockchains just distributed, immutable time-series logs?) Wouldn't a software project have version control (e.g., git commits)? Doesn't this text have revision history? (Undo. Redo.)

Put differently: Don't all databases have logs?

We recognize that a lot of applications may never require time-series data (and would be better served by a "current-state view"). But as we continue along the graph of technological progress, it might seem that these "current-state views" subsided necessary. Which by storing more and more data in its time-series form, we could also be ready to know it better.

(<https://datascience.eu/computer-programming/cap-theorem-and-distributed-database-management-systems/>)

(<https://datascience.eu/author/data-science-team/>)

Data Science Team (<https://datascience.eu/author/data-science-team/>)

You may also like

Semantics

**noun,
plural in
form but
singular or
plural in
construction**

([https://datascience.eu/wiki/semantics-](https://datascience.eu/wiki/semantics-noun-plural-in-form-but-singular-or-plural-in-construction/)

**noun-
plural-in-
form-but-
singular-or-
plural-in-
construction/**)

by Data Science Team (<https://datascience.eu/author/data-science-team/>)

Semantics

([https://datascience.eu/wiki/semantics-](https://datascience.eu/wiki/semantics-noun-plural-in-form-but-singular-or-plural-in-construction/)

by Data Science Team (<https://datascience.eu/author/data-science-team/>)



(<https://datascience.eu/wiki/a-simple-explanation-of-the-internet-of-things/>)

**What You
Need to
Know
About IoT
(Internet of
Things)**

(<https://datascience.eu/wiki/a-simple-explanation-of-the-internet-of-things/>)

by Data Science Team (<https://datascience.eu/author/data-science-t>

**How to
convert**

**Grain and
noise**

**Significant
adjective**

**morse code
to text
(<https://datascience.eu/wiki/convert-morse-code-to-text/>)**

**texture
(<https://datascience.eu/wiki/grain-and-noise-texture/>)**

(<https://datascience.eu/wiki/significant-adjective/>)

by Data Science Team (<https://datascience.eu/author/data-science-t>)

by Data Science (<https://datascience.eu/author/data-science/>)

by Data Science Team (<https://datascience.eu/author/data-science-team/>)

Leave a comment

You must be logged in (https://datascience.eu/wp-login.php?redirect_to=https%3A%2F%2Fdatascience.eu%2Fwiki%2Fwhat-the-heck-is-time-series-data-and-why-do-i-need-a-time-series-database%2F) to post a comment.

PREVIOUS ARTICLE

DATA SCIENCE . EU

The #1 Multilingual Source for DataScience

 (<https://www.facebook.com/datascience.eu/>)

 (<https://www.instagram.com/datascience.eu/>)

 (<https://twitter.com/DatascienceEu>)