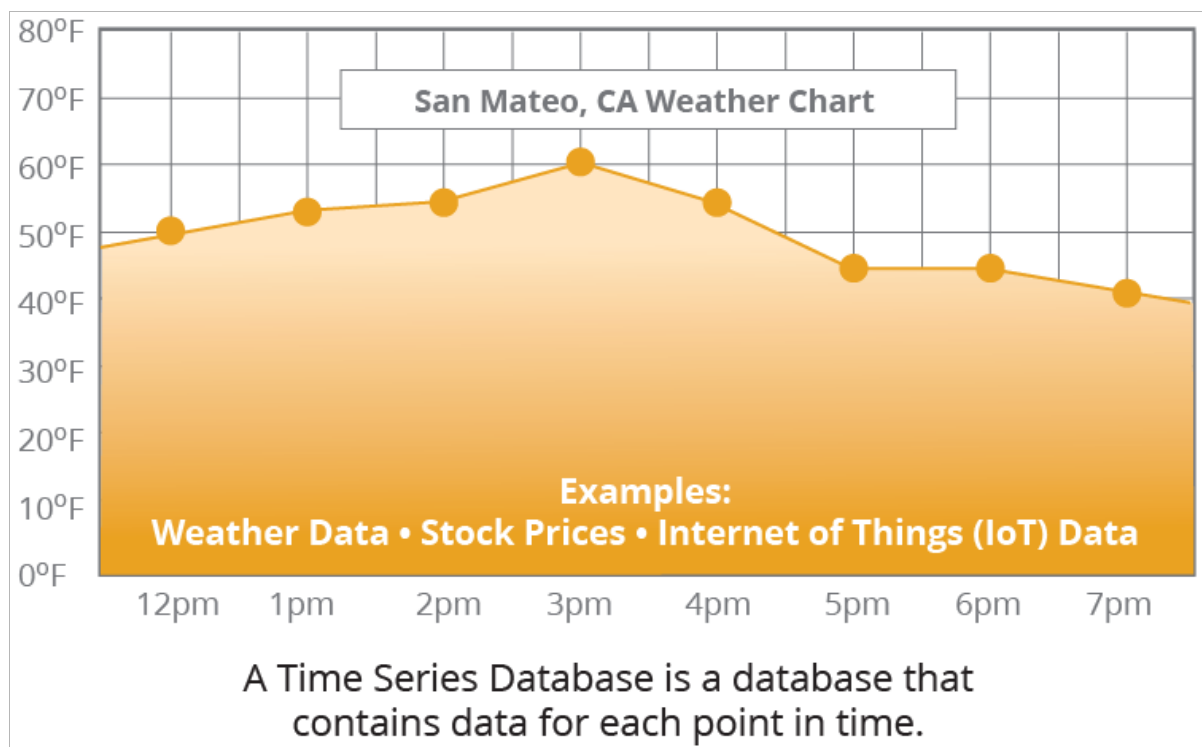


[Get Hazelcast](#)

Time Series Database

A **time-series database (TSDB)** is a computer system that is designed to store and retrieve data records that are part of a “time series,” which is a set of data points that are associated with timestamps. The timestamps provide a critical context for each of the data points in how they are related to others. Time series data is often a continuous flow of data like measurements from sensors and intraday stock prices. A time-series database lets you store large volumes of timestamped data in a format that allows fast insertion and fast retrieval to support complex analysis on that data.




A Time Series Database is a database that contains data for each point in time.

How Does a Time Series Database Work?

TSDBs work by capturing a set of fixed values along with a set of dynamic values. As a simple example, in an oil well where many metrics of the rig are captured, one set of data points might have the label “Oil Pressure Rig #1” and the associated

timestamp would be the measurement along with the timestamp. This



dynamic values would be the pressure measurement along with the timestamp. This example time series data is useful for tracking trends in the oil pressure which, when analyzed along with other metrics, could lead to predictions on maintenance needs as well as decisions on the abandonment of the well. These records are written to a storage medium in a format that allows fast time-based reads and writes.

Since all records are timestamped, the order of the data points is a native characteristic of the data. This order can be used to deliver the data to a stream processing engine that can treat the ordered data as if it were a data stream. Since one main goal of TSDBs is to be fast, leveraging a fast stream processing engine is generally ideal. Hazelcast Jet is an example of a fast **stream processing** engine which gets its performance boost from its in-memory architecture. **Hazelcast Jet** can integrate with TSDBs so that the time series data sourced by the TSDB can be processed at extreme speeds.

Example Use Cases

A broad example use case would be an **IoT database** for Internet of things (IoT) environments where remote devices are continually capturing metrics for analytical purposes. The aforementioned oil well example is a common IoT use case where the analysis of numerous metrics from an oil well can help with predictive maintenance, in which the analysis can lead to a prediction on when the equipment will fail due to trends and factors that are represented in the data. A TSDB would be used to capture the vast amount of collected data, then applications are run on the database to deliver the analytics.

Another TSDB use case is computer system metrics analysis. In this situation, readings from computer systems are stored in the TSDB which allow IT professionals to monitor the status of the various systems. Metrics such as memory utilization or process count can be tracked to see if new computer resources need to be deployed, or if applications need to be reallocated.

Alternatives to Time Series Databases

Relational database management systems (RDBS), which are often considered general-purpose database systems, can be used to store and retrieve time series data. With the flexibility of RDBMSs, they can store the same data as a TSDB, with one key difference being how the data is written to the storage medium. Since RDBMSs have different design goals than TSDBs, RDBMSs they are not optimized for time series data and tend to be slower for inserting and retrieving time series

For time series data and tend to be slower for inserting and retrieving time series data.

[Get Hazelcast](#)

Another type of database, NoSQL, are also often used to store time series data. Since NoSQL databases are more flexible in terms of the data format for each record, they are good for capturing time series data from a number of distinct sources. An implementation of a NoSQL database for time series data is often a very good alternative to TSDBs, and at the same time can provide capabilities that apply beyond time series data.

Related Topics

[In-Memory Data Grid](#)[In-Memory Database](#)[In-Memory Computing](#)

Relevant Resources

[White Paper](#)

[**The Crucial Role of Streaming Technology for Business**](#)

[Download](#)[Webinar](#) | [Video](#) | [60 minutes](#)

[**The Evolution of Stream Processing and Top Use Cases**](#)

[Watch Now](#)



Get Hazelcast



[Brochure](#) | [PDF](#) | [4 pages](#)

Hazelcast Jet Product Brochure

Download

[View All Resources](#)

[Back to Glossary Terms](#)