# 《The Design and Application of Time-series Database》

# Course Report



**Subject name：Why do we need a time-series database?**

Name of the project leader：李煦辰 Li Xuchen

Student ID: 2020141461103

Teacher Advisor： Tao Jianhui

2022 年 7月 7日

**Why do we need a time-series database?**

Software Engineer Major

**Student** Li Xuchen **Teacher Advisor** Tao Jianhui

First, it shows that our world is changing at a very fast pace, and it also demonstrates our ability to collect and analyze more and more data. However, if you look closely, you'll see that these applications all require a special type of data: · Self-driving cars need to constantly collect data on changes in their surroundings and adjust it to weather conditions, potholes, and countless other variables. · Automated trading algorithms need to constantly collect data about market changes in order to optimize short-term and long-term return on investment. · Smart homes need to regulate temperature, identify hazards, etc., and handle human-computer interaction by monitoring indoor conditions. · The retail industry accurately and efficiently monitors the transportation of every item. Compared to stock market trends, self-driving cars, and accurate predicting when you'll make your next online purchase, there have been many examples in recent years of how time series data collection and analysis can impact an individual's daily lives. For the first time ever, the world's interest in time series data peaked in the most unexpected way. The COVID-19 pandemic has turned billions of people around the world into consumers of time-series data and demanded accurate and timely understanding of daily TRENDS in COVID-19. In our information-hungry world, access to detailed, feature-rich time series data has become one of the most valuable commodities. Businesses, governments, schools, and communities, large and small, are looking for ways to extract value from analyzing time series data. The usage patterns of software developers already reflect the same trend. Over the past two years, the Time Series Database (tsdb) has been the fastest growing database:

A time series database is a specific type of database that is primarily used to store time series data. As 5G technology continues to mature, Internet of Things technology will make everything interconnected. Before the era of the Internet of Things, only mobile phones and computers can be networked, and all devices will be networked in the future, and these devices will spit out a large amount of time-organized data every moment, which needs to be stored for query, statistics and analysis. Time series data is very different from ordinary business data in every way.

Previously, our view of time series data was static, with daily temperatures, opening and closing prices of the stock market, and even daily or cumulative hospitalizations due to COVID-19. However, we tend to overlook the subtle differences caused by potential changes in these static data.

Let's look at some examples. If I give you $10, the bank's traditional database will have an expense on my account and a receipt on your account. Then, if you give me $10, the same process goes backwards. In the end, our bank balances still look the same, and for banks, nothing has changed this month. However, with the time series database, the bank will perceive that the two people have been transferring 10 yuan to each other, and there may be a deeper reason. If you track this nuance, the month-end account balance makes more sense.

Another example is the average temperature in a location for several consecutive days. Over the past few decades, average temperatures have been used as a major reference for energy efficiency in buildings. The average daily temperature in the same location may be only slightly different during any given week, but at the same time, factors affecting the environment can change dramatically. Conversely, understanding the temperature changes at each hour of the day, combined with precipitation, cloud cover, and wind speed during that time, can greatly improve the ability of property modeling and optimizing energy efficiency. Similarly, while it is valuable to know the total number of COVID-19 hospitalizations per day in a community, this number alone does not provide a good picture of the details. For example, a hospital might disclose that 20 people were hospitalized on Monday, while the number of hospitalizations increased slightly throughout the week, bringing the total number of hospitalizations to 23 by Friday. At first glance, hospitalizations increased by 15% this week. But if we count the detailed records (increasing the frequency of collection), we may see that there was a net increase of 3 cases this week, but in reality, 10 people were discharged, 13 new, and in the last 5 days, the number of new admissions increased by 65%. Tracking various aspects of patient data over time (e.g., patient age, admission or discharge, days of recovery, etc.) helps us understand how to derive daily statistics, allowing us to better analyze trends, accurately report totals, and take action that can even influence government policies.

These examples illustrate how different modern time series data is from what we used to know. Time series data analysis is much deeper than pie charts or Excel. This data doesn't just use time as a measure, the key is to help us analyze data and obtain valuable information. In addition, there are many kinds of time series data, but regardless of the scenario or user case, all time series datasets have 3 things in common: 1. The data written is almost new 2. Data is usually written in chronological order 3. Time is a spindle (time intervals can be regular or irregular) In other words, time series data is usually written as "appended." While data may need to be corrected, processed with delays, out of order, etc. after the fact, these are anomalies, not the norm. TSDB's biggest application scenario is to monitor the service (sentinel), taking sentry as an example, sentinel will deploy various script clients on the business server to collect server indicator data (IO indicators, CPU indicators, bandwidth memory indicators, etc.), business-related data (method call anomalies, response delay, JVM GC-related data, etc.), database-related data (read latency, write latency, etc.), Obviously, these data are time series related. After the client collects it, it is sent to the sentry server, which stores this data and provides a page for the user to query.

In fact, the potential of TSDB has not yet exploded, at least not yet. In the foreseeable next 3 to 5 years, with the advent of the Internet of Things and Industry 4.0, all devices will carry sensors and network, and the time series data collected by sensors will rely heavily on TSDB's real-time analysis capabilities, storage capabilities and query statistics capabilities.

The feature of TSDB.

High throughput write capability. This is tailored for the characteristics of the continuous generation of massive data in the time series business, and at present, to achieve high throughput write to the system, it is necessary to meet two basic technical requirements: the system has horizontal scalability and a stand-alone LSM architecture. The system has horizontal scalability is easy to understand, the stand-alone machine must not be able to hold, the system must be clustered, and it must be easy to add node expansion, in the end, it is the expansion of the time to have no perception of the business, the current Hadoop ecosystem can basically do this; The LSM architecture is used to ensure high-throughput writes to a single machine, and the LSM structure only needs to write memory and append write logs, so that there is no need to write data to disk randomly, HBase, Kudu and Druid and other systems that require write performance are currently using this structure.

Data tiered storage/TTL. This is a technical feature tailored to the hot and cold nature of time series data. Data hierarchical storage requires the ability to put the data of the most recent hour level into memory, the data of the most recent day level into the SSD, and the data of the older generation to the cheaper HDD or directly use the TTL expiration to retire.

High compression ratio. There are two aspects of providing high compression ratio, on the one hand, cost savings, which is easy to understand, compressing 1T data to 100G can reduce the hard disk overhead of 900G, which is a great temptation for business. Another aspect is that the compressed data can be more easily guaranteed to be stored in memory, such as the last 3 hours of data is 1T, I now only have 100G of memory, if not compressed, there will be 900G of data forced to put on the hard disk, so that the query overhead will be very large, and the use of compression will put this 1T data into memory, query performance will be very good.

Multi-dimensional query capability. Time series data often has labels of multiple dimensions to depict a single piece of data, which is the dimension column mentioned above. How to query efficiently based on a random number of dimensions is a problem that must be solved, which usually requires considering bitmap indexing or inverted indexing techniques.

Efficient aggregation capabilities. A common requirement of time series business is to aggregate statistical report queries, such as the total number of times an interface has been abnormal in the last day in the sentry system, or the maximum time it takes for an interface to execute. Such aggregation is actually a simple count and max, the problem is how to efficiently query out and aggregate the raw data that meets the conditions on the basis of such a large amount of data, to know that the original value of the statistics may not be in memory because of the long time, so this may be a very time-consuming operation. At present, the more mature scheme in the industry is to use pre-aggregation, that is, to complete the basic aggregation operation when the data is written in.

Future technology points: real-time detection of anomalies, future prediction, etc.

TSDB will be a very marketable and challenging database in the future, and although there are already such and such services, most of them have such and such problems, and it is difficult to talk about maturity now. In order to occupy a certain position in the era of the Internet of Things and the era of Industry 4.0, TSDB is a technology that must be expanded.