**TDengine introduction**

TDengine is an open source time series database launched by Taosi data. It is also an efficient platform for storing, querying and analyzing time series databases. It is applicable to the Internet of things, industrial Internet, Internet of vehicles, it operation and maintenance, energy, finance and other fields. In terms of use, TDengine provides an operation similar to the relational database mysql, which is convenient for users to get started quickly.

In the 1970s and 1980s, Oracle and other companies launched relational databases in order to solve the informatization problems of banks and insurance companies, so they designed structured databases. Later, with the rise of the Internet, people found that the characteristics of Internet data are very different from those of bank data. They have a lot of unstructured data, such as data crawled by web crawlers and social media data. They are all unstructured, so NoSQL databases (such as HBase, mongodb, etc.) appeared, and many Internet applications began to use NoSQL. In the 21st century, the Internet of things, the Internet of vehicles, and the industrial Internet have developed. The data characteristics of these scenes are "sequential" and will grow at a high speed. Both relational databases and NoSQL databases are not suitable for processing. Therefore, people have proposed a new way of "temporal database", and TDengine is one of the special tools for processing the temporal data of Internet of things scenes. It enables a large number of devices and data collectors to produce TB or even Pb level data every day, which can be processed efficiently and in real time, monitor and warn the operation status of the business in real time, and excavate business value from big data.

**Functional advantages**

One of the modules of TDengine is Time series database. At the same time, in order to reduce the complexity of research and development and the difficulty of system maintenance, TDengine also provides cache, message queue, subscription, streaming computing and other functions, and provides a full stack technical solution for the processing of big data in the Internet of things and industrial Internet. It is an efficient and easy-to-use big data platform of the Internet of things. Compared with Hadoop and other typical big data platforms, TDengine has the following distinct advantages:

* **More than 10 times of performance improvement:** an innovative data storage structure is defined. A single core can process at least 20000 requests per second, insert millions of data points, and read more than 10 million data points, which is more than 10 times faster than the existing general database.
* **The cost of hardware or cloud services is reduced to 1/5:** due to super performance, the computing resources are less than 1/5 of the general big data solution; Through column oriented and advanced compression algorithm, storage occupies less than 1/10 of the general database.
* **Full stack sequential data processing engine:** integrates database, message queue, cache, streaming computing and other functions. Applications do not need to integrate Kafka, redis, HBase, spark, HDFS and other software, significantly reducing the complexity and cost of application development and maintenance. Seamless connection with third-party tools: without a line of code, it can be integrated with telegraf, grafana, MATLAB, R. In the future, mqtt, OPC, Hadoop, spark, etc. will be supported, and Bi tools will also be seamlessly connected.
* **Powerful analysis function:** whether it is data ten years ago or one second ago, you can query it within a specified time range. Data can be aggregated on the timeline or on multiple devices. Ad hoc query can be carried out at any time through shell, python, R, MATLAB.
* **High availability and horizontal scalability:** through distributed architecture and consistency algorithms, and through multiple replication and clustering features, TDengine ensures high availability and horizontal scalability to support mission critical applications.
* **Zero operation and maintenance cost and zero learning cost:** the installation and cluster can be completed in a few seconds without any dependence. There is no need to divide the database and table. The system operation status monitoring can be seamlessly integrated with grafana or other operation and maintenance tools for real-time backup. Similar to standard SQL, it supports restful, python, Java, c/c++, c#, go, node JS and other programming languages, the use method is similar to MySQL, with zero learning cost.
* **Core open source:** in addition to some auxiliary functions, the core of TDengine is open source. Enterprises will no longer be bound by databases. This makes the ecosystem stronger, the products more stable, and the developer community more active.

**Application scenario**

As an Internet of things (IOT) big data platform, TDengine is applicable to almost all IOT application scenarios, as well as some time series data scenarios other than the Internet of things, such as CRM, ERP and other systems. As a basic software, TDengine has a wide range of applications. In principle, it can be used in all places where machines, equipment and sensors are used to collect data.

Some specific application scenarios are listed below:

Public safety: online records, call records, individual tracking, interval screening

Power industry: centralized monitoring of smart meters, power grids and power generation equipment

Communication industry: telephone bill details, user behavior, base station / communication equipment monitoring

Financial industry: transaction records, access records, ATM, POS machine monitoring

Travel tools: real time monitoring of train / car / taxi / plane / bicycle

Transportation industry: real time road conditions, intersection flow monitoring, checkpoint data;

Petroleum and petrochemical: real time monitoring of oil wells, transportation pipelines and transportation fleets

Internet: server / application monitoring, user access log, advertisement click log

Logistics industry: tracking and monitoring of vehicles and containers

Environmental monitoring: weather, air, hydrology, geological environment and other monitoring;

Internet of things: elevators, boilers, machinery, water meters, gas meters and other networking equipment

Military industry: data acquisition and storage of various military equipment

Manufacturing: production process control, process data, supply chain data collection and analysis

**Write storage policy of TDengine**

* Single point write: Based on the characteristics of time series data, unlike many time series databases, TDengine stores the data of each collection point as an independent table in the database. In this way, for the data of a collection point, whether in memory or hard disk, the data points are stored continuously on the medium TDengine, which greatly reduces random reading operations, reduces the number of IO operations, and improves the reading and query efficiency by an order of magnitude. Moreover, because the process of data generation by different data acquisition devices is completely independent, each device only generates its own data, and there is only one writer for a table. In this way, each table can be written without lock, and the writing speed can be greatly improved
* Continuous storage: secondly, for a data acquisition point, because the data it generates is sequential, it is a natural sort data structures. Therefore, the subsequent writing methods are implemented by orderly append, which can give full play to the performance of the hard disk and further improve the data writing speed.
* Super Table: TDengine each device corresponds to a table. However, there is no need to write information such as device address number to the disk. So they put many tables together to become a super table, and then directly use address number and other information to filter when querying
* Column compression: In order to improve the efficiency of compression and query, TDengine adopts column oriented. Because the content format of each column under column oriented is similar, it is conducive to compression and space saving. Moreover, different compression algorithms will be adopted for different types of data. After another targeted compression, another conventional compression. So the space occupied by the data written to the hard disk will be very small.

**Features of TDengine**

* It is designed for the data of the Internet of things, and uses the timing characteristics of the data of the Internet of things to realize the function that each collection point corresponds to a table. However, it is not suitable for processing general Internet data.
* Column storage + compression is adopted to save hardware cost. (high compression efficiency: using the characteristics of the Internet of things with little change in data, DIF interpolation compression, and then second-order compression, the efficiency is very high.)
* Support high availability, and divide each physical node into multiple virtual data nodes and virtual management nodes. The virtual data node stores data, and the virtual management node manages metadata. Virtual data nodes and virtual management nodes are distributed on different physical nodes to achieve high availability of dataset applications.
* In terms of storage structure, an independent table is created for each collection point to store. In this way, the data of each collection point can be stored continuously and the reading efficiency can be improved. Since each table has only one data source, it can realize lock free writing and improve the writing rate.
* For the changeable aggregation, the concept of super table is introduced. The same type of collection device can create a super table. When creating a super table, you can assign labels to such tables, and filter the tables in the database through labels when querying. In this way, even if there are many tables in the database, you can realize rapid multi table aggregation.
* The installation package is very small and easy to install and use. Support SQL, and the syntax is similar to MySQL.