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

In the context of big data, if we subdivide the data type, each subdivision type of data will have a certain storage optimisation space, mainly to see if this optimisation has added a new set of subdivision type data processing scheme to be more valuable (of course, if our system only has a certain subdivision type of data, Then it will be much better to choose the subdivision type).

A time-series database is a subdivision of data types under the background of big data. Due to the advent of the era of the Internet of Things, the total amount of time series data is increasing, and processing schemes specifically for a time-series database types are constantly emerging. They are called a time-series databasebases.

The era of big data has come for many years, big data solutions are basically mature, and Hadoop cluster processing solutions have basically become a best practice for processing big data. The data he processes includes structured, semi-structured and unstructured data, collecting data through Sqoop, Flume, kafka, storing data through hbase, hdfs, calculating data through mapreduce, sparkstreaming, etc., the most Then use hive as a data warehouse to provide the required data for the application layer.This is a universal and comprehensive big data solution.

If you subdivide the data type and aim at a large amount of time series data, in short, the time series data is the data indexed according to the time dimension, such as vehicle trajectory data and sensor temperature data. With the advent of the Internet of Things era, the amount of data in time series data has burst out, and the optimised storage for this data segmentation is becoming more and more important.

I briefly and immaturely summarised the characteristics and solutions of the a time-series databasebase, as follows:

General a time-series database will have these attributes:

The measured data set is similar to the table in a relational database;

A data point, similar to row in a relational database;

Timestamp, characterisation of the time when the data is collected;

The dimension column represents the attribution and attributes of the data, indicating which device/module is generated. Generally, it does not change over time for query use;

The indicator column represents the measured value of the data and fluctuates smoothly over time.

For a time-series database, we have summarised the following characteristics:

1. Data characteristics: the amount of data is large, the data increases over time, the value is repeated in the same dimension, and the indicator changes smoothly (the trajectory coordinates of a smooth change are uploaded on a device of a vehicle).

2. Write features: High concurrency writing and will not be updated (the trajectory will not be updated).

3. Query characteristics: Statistical analysis of indicators according to different dimensions, there are obvious hot and cold data, and generally only query recent data (generally, we only care about recent trajectory data).

What are the advantages of time-series database?

(1) Structured data is stored. We all know that the data to be stored in traditional big data schemes includes structured, semi-structured and unstructured data, which determines that we can't decide which fields and define the data types of each field. For example, hbase is stored uniformly through byte types, that is to say, put into hba. The data in se is all byte arrays. Converting from ordinary types to byte arrays needs to be done by ourselves. We don't know how to convert them to bytes. Its storage efficiency will be higher. However, the data generated by a time-series database is structured data. We can define the fields and types of data in advance, so that the database system can choose the optimal compression method according to different field types, and greatly improve the utilisation rate of storage.

(2) The analysis aggregates structured data. Since Analytical Aggregation Is Structured Data, we don't need to use complex computing tools such as Mapreduce, or generally do not need data warehouses such as Hive, but only polymerate computing tools similar to Sum and avg at the Database Storage Level, or even doing some simple streaming calculations provides the foundation for 'hyper-convergence' (hyper-convergence means converging multiple components similar to the previous big data processing scheme into one component, mainly because the structured data is too simple, and the collection and calculation are relatively simple, which is also the development trend of subsequent time series databases. Reduce system complexity).