**Understanding of TSDB**

—ChenZhongShan 2020141461133

1. Preliminary understanding of TSDB
2. Application scope of TSDB
3. Broad prospects of TSDB

**Preliminary understanding of TSDB**

**Basic concepts**

Time series database: a data management system that provides efficient access to time series data and statistical analysis functions. It is widely used in Internet of things (IOT) equipment monitoring system, enterprise energy management system (EMS), production safety monitoring system, power detection system and other industry scenarios. Opentsdb is an extensible distributed temporal database, and the bottom layer relies on HBase. As a typical representative of time series database based on general storage, it started relatively early and has a relatively high recognition in the time series market. Opentsdb is the top of HBase, a distributed and scalable time series database (TSDB). Opentsdb is written to meet a common demand: store, index and provide index systems and applications collected from computer systems (network devices, operating systems), and make these data easy to access and crawl.

Opentsdb defines that each time series data needs to contain the following attributes:

1. metric name

This is tailored to the continuous generation of massive data by sequential services. At present, to achieve high-throughput writing of the system, two basic technical requirements must be met: the system has horizontal scalability and single machine LSM architecture.

1. UNIX timestamp

This is a technical feature customized for the hot and cold properties of time series data. Hierarchical data storage requires the ability to put the most recent hour level data into memory, the most recent day level data into SSD, and the more distant data into cheaper HDD or directly use TTL to expire and eliminate.

1. Value (64 bit integer or single precision floating-point number)

There are two considerations in providing high compression ratio: on the one hand, cost savings; on the other hand, compressed data can be more easily stored in memory

1. tags.It is used to describe data attributes and contains at least one or more tags. Each tag is composed of tagkey and tagvalue, both of which are strings

Time series data usually have labels of multiple dimensions to describe a piece of data, which is the dimension column mentioned above. How to query efficiently according to several random dimensions is a problem that must be solved. This problem usually needs to consider bitmap index or inverted index technology.

**characteristic**

◇High throughput write

◇Hierarchical data storage

◇High compression ratio reduces storage space

◇Multi dimensional query capability

◇Efficient aggregation capability

This determines its advantages in use:

1. Using HBase storage, there is no single point of failure.

2. With HBase storage, the storage space is almost unlimited. Support permanent storage and capacity planning

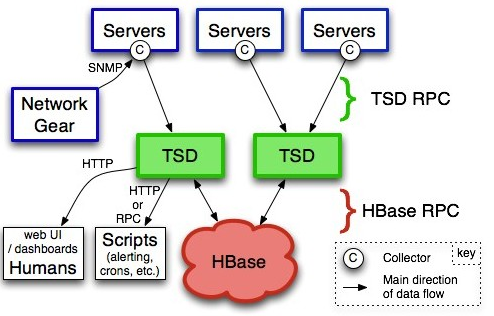
3. Easy to customize graphics

4. It can expand the data collection points to 10billion level.

5. It can expand the number of metrics to K level (such as CPU usage, which can be counted as a metric, that is, metric is a monitoring item)

6. Support second level data.

**Framework**

Opentsdb consists of a time series daemon (TSD) and a set of command line utilities. The interaction with opentsdb is mainly realized by running one or more TSDs. Each TSD is independent. There is no master server and no shared state, so you can run any number of TSDs as needed to handle any load you put into them. Each TSD uses HBase in the cloudtable cluster to store and retrieve time series data. The data pattern is highly optimized to quickly aggregate similar time series, thereby minimizing storage space. TSD users do not need to directly access the underlying storage. You can communicate with TSD through HTTP API. All communications occur on the same port (TSD determines the protocol of the client by looking at the first few bytes it receives).

**Storage optimization**

## As a widely used time series database, opentsdb has made a lot of optimization in storage, and the optimization choice is completely consistent with the storage model of HBase database on which it depends.

## ◇Optimization 1: shorten row key

a. Save storage space

b. Improve query efficiency: reduce the time of key matching

c. Improve transmission efficiency: it not only saves the bandwidth read from the system, but also saves the bandwidth occupied by data return, and improves the speed of data writing and reading.

d. Alleviate the memory pressure of Java programs: GC is a difficult problem for Java programs. Try to save memory. The metric name, tag key or tag value previously stored in string can now be replaced by a byte array of 3 bytes, which saves memory.

## ◇Optimization 2: reduce the number of key values

1.Merge multiple columns into multiple columns, and change multiple single columns into single multiple columns.

2.Merge multiple columns into a single column, and a single column becomes a single column.

## ◇Optimization 3: concurrent write optimization

Opentsdb merges data with the same tsuid (the same metric name, and the same tags) belonging to the same time period into one row for storage. The default time period in opentsdb is one hour, that is, all data points belonging to this hour will be merged into one row for storage, as shown in the figure. After merging into a row, the timestamp in the rowkey of the row will be specified as the start time of the hour (the base time of the time cycle), and the column name of each column will record the difference between the timestamp of the real data point and the start time (base) of the time cycle.

**Application scope of TSDB**

Time series database is written for a large amount of data, which is mainly used to record time series data, and is used to monitor the scene of recording; Write more and read less; Time series data is a series of data based on time.

As a time series database, opentsdb can not only provide the query of original data, but also support the aggregation ability of original data, and support the aggregation calculation after filtering.

It supports downsampling queries. For example, the original data is one data point per minute. If I want to display one data point per hour, it can also be supported.

It supports grouping queries based on dimensions. For example, I have data of a Chinese city. Now I want to group queries based on provinces, which can also be supported.

**Precautions and some suggestions**

A perfect monitoring system should include four parts: collection, storage, exhibition and alarm.The main function of opentsdb is also a time database to store the collected monitoring data. In addition, it also comes with some common monitoring scripts. Opentsdb also has a "solo exhibition" page, but it is often simple to do. Exhibition and alarm also need to find other components to cooperate. Recommend grafana.

1. First aggregate the full amount of data according to 1s, 1min, 5min, 10min, etc. This greatly reduces the number of query data points and improves the query efficiency

2. If you need more than one monitoring cluster, it is recommended to build multiple hbaes clusters, ⼀ HBase clusters specifically store the original monitoring data, and ⼀ HBase clusters store the aggregated monitoring data. It can greatly improve the query efficiency.

3.The default character set of opentsdb is iso-8859-1

4.When opentsdb is used, only 8 tags can be stored in tags by default. This situation is far from meeting the actual development needs. Here we need to manually open tsdb Conf add configuration tsd storage. max\_ Tags = maximum number of tags

5.When pushing data in actual use, some data push failures will be returned. Here, you can locate the reason for the failure according to the returned error information. For example, when encountering special characters, opentsdb cannot have special characters in the data by default. At this time, TSD is also added to the configuration core. tag. allow\_ Specialchars= special characters.

**Conclusion**: industrial data with typical characteristics: fast generation frequency (each monitoring point can generate multiple pieces of data in one second), heavy dependence on the collection time (each piece of data requires a unique time), multiple monitoring points and large amount of information (there are thousands of monitoring points in the conventional real-time monitoring system, and the monitoring points generate data every second and a large amount of data every day). Opentsdb is mainly used as a monitoring system. On the one hand, it can store and retrieve parameter data and save it for a long time. On the other hand, if you need to add functions, you can also add various new parameters. Finally, openstdb analyzes the data stored in HBase and graphically displays the status of network devices, operating systems and applications in the cluster.

**Broad prospects of TSDB**

Now in the field of time series database, there are many companies specializing in time series database all over the world. The competition is very fierce both at home and abroad. All companies are trying to make users use it more convenient and efficient. On the contrary, it is very beneficial for users. At present, time series database (TSDB) has become the fastest-growing database category. With the advent of 5g, time series databases will become more popular. Modeling time series data includes three important parts: subject, time point and measured value. Applying this model, we find that in our daily work and life, we are exposed to such data all the time. The birth of time series database is to solve the shortcomings of traditional relational database in time series data storage and analysis. It will play an increasingly important role in our lives in the future.