

CLAIMS

What is claimed is:

1. A method for storing and querying time series data, comprising:
upon receiving data to be stored from an Internet of things (IoT) device, determining a
5 data type of the data to be stored, wherein the data to be stored is time series data, and the data
type is intended to indicate a change in the time series data over time;
obtaining compressed data by compressing the data to be stored by a data compression
method corresponding to the data type, wherein different data types correspond to different
data compression methods;
10 storing the compressed data to a data storage table corresponding to the data type,
wherein different data types correspond to different data storage tables;
receiving a query request, wherein the query request comprises a query data type and a
query time condition; and
querying target data that meets the query time condition from a data storage table
15 corresponding to the query data type.
2. The method according to claim 1, wherein the data to be stored is a state quantity,
wherein the state quantity is a physical quantity that is discontinuous over time and has a
value belonging to a preset enumerated data set, and obtaining the compressed data by
20 compressing the data to be stored by the data compression method corresponding to the data
type comprises:
reading values of n pieces of data to be stored that are continuously received, wherein n
is an integer greater than or equal to 2; and
obtaining the compressed data by compressing the data to be stored based on changes in
25 the values of the n pieces of data to be stored.
3. The method according to claim 2, wherein obtaining the compressed data by
compressing the data to be stored based on the changes in the values of the n pieces of data to

be stored comprises:

for an $(i+1)^{\text{th}}$ piece of data to be stored among the n pieces of data to be stored, retaining the $(i+1)^{\text{th}}$ piece of data to be stored if a value of the $(i+1)^{\text{th}}$ piece of data to be stored is different from that of an i^{th} piece of data to be stored, wherein i is an integer greater than or equal to 1; and

deleting the $(i+1)^{\text{th}}$ piece of data to be stored if the value of the $(i+1)^{\text{th}}$ piece of data to be stored is the same as that of the i^{th} piece of data to be stored.

4. The method according to claim 1, wherein the query data type is a state quantity, and querying the target data that meets the query time condition from the data storage table corresponding to the query data type comprises:

determining data corresponding to a second moment in the data storage table as the target data if the query time condition is a first moment and the data storage table corresponding to the state quantity does not store data corresponding to the first moment, wherein the second moment is before the first moment and is closest to the first moment.

5. The method according to claim 1, wherein the data to be stored is an analog quantity, wherein the analog quantity is a physical quantity that is continuous over time and has a value within a preset value range, and obtaining the compressed data by compressing the data to be stored by the data compression method corresponding to the data type comprises:

obtaining the compressed data by sampling and compressing the data to be stored based on an acquisition frequency of the data to be stored and a target sampling frequency, wherein the acquisition frequency is higher than the target sampling frequency.

6. The method according to claim 1, wherein the query data type is an analog quantity, and querying the target data that meets the query time condition from the data storage table corresponding to the query data type comprises:

acquiring data corresponding to at least one fourth moment and at least one fifth moment in the data storage table if the query time condition is a third moment and the data storage table corresponding to the analog quantity does not store data corresponding to the third

moment, wherein the fourth moment is before the third moment and closest to the third moment, and the fifth moment is after the third moment and closest to the third moment; and
calculating the target data corresponding to the third moment by an interpolation algorithm based on the data corresponding to the fourth moment and the fifth moment.

5

7. The method according to claim 1, wherein the data to be stored is a cumulative quantity, wherein the cumulative quantity is a physical quantity whose value increases progressively over time, and the data to be stored is compressed by a preset lossless compression algorithm in a time series database; and

10 the query data type is the cumulative quantity, and querying the target data that meets the query time condition from the data storage table corresponding to the query data type comprises:

calculating components corresponding to respective time periods based on adjacent cumulative quantities in the data storage table;

15 querying target components belonging to a target time period indicated by the query time condition; and

determining the target data based on the target components.

8. An apparatus for storing and querying time series data, comprising:

20 a determining module, configured to, upon receiving data to be stored from an Internet of things (IoT) device, determine a data type of the data to be stored, wherein the data to be stored is time series data, and the data type is intended to indicate a change in the time series data over time;

a processing module, configured to obtain compressed data by compressing the data to
25 be stored by a data compression method corresponding to the data type, wherein different data types correspond to different data compression methods;

a storing module, configured to store the compressed data to a data storage table corresponding to the data type, wherein different data types correspond to different data storage tables;

30 a receiving module, configured to receive a query request, wherein the query request

comprises a query data type and a query time condition; and

a querying module, configured to query target data that meets the query time condition from a data storage table corresponding to the query data type.

5 9. A server, comprising: a processor; and a memory storing at least one instruction, at least one program, a code set, or an instruction set, wherein the at least one instruction, the at least one program, the code set, or the instruction set, when loaded and executed by the processor, causes the processor to perform the method for storing and querying time series data as defined in any one of claims 1 to 7.

10

10. A computer-readable storage medium storing at least one instruction, at least one program, a code set, or an instruction set, wherein the at least one instruction, the at least one program, the code set, or the instruction set, when loaded and executed by a processor, causes the processor to perform the method for storing and querying time series data as defined in

15 any one of claims 1 to 7