SERVICE DATA STORAGE METHOD AND APPARATUS, COMPUTER DEVICE, AND STORAGE MEDIUM

[0001] This application claims the priority to Chinese Patent Application No. 202310311147.1, titled "SERVICE DATA STORAGE METHOD AND APPARATUS, COMPUTER DEVICE, AND STORAGE MEDIUM", filed on March 14, 2023 with the China National Intellectual Property Administration, which is incorporated herein by reference in its entirety.

FIELD

[0002] The present disclosure relates to the field of computer technology, and in particular to a service data storage method, a service data storage apparatus, a computer device, a storage medium and a computer program product.

BACKGROUND

[0003] With the development of data storage technology, the heap storage technology emerges. According to the heap storage technology, massive amounts of data with different lengths are stored in heaps, and indexes corresponding to the data are recorded. The indexes include pointers corresponding to the heaps storing the data. By using a pointer, a corresponding heap may be access and then the data is obtained.

[0004] However, according to the conventional heap storage technology, data is blindly stored in heaps, resulting in low resource utilization.

SUMMARY

[0005] A service data storage method is provided according to the present disclosure. The method includes: obtaining an initial index sequence corresponding to target service data; determining target sub-data from sub-service data contained in the target service data, and determining an index space corresponding to the target sub-data in the initial index sequence as a target index space; writing the target sub-data in the target index space, and obtaining a

target index sequence, in a case that a target data length of the target sub-data is less than or equal to a space data length of the target index space; and writing the target sub-data in a target heap space corresponding to the target service data, obtaining a target pointer corresponding to the target sub-data, writing the target pointer in the target index space and obtaining the target index sequence, in a case that the target data length of the target sub-data is greater than the space data length of the target index space.

[0006] A service data storage apparatus is further provided according to the present disclosure. The apparatus includes an index sequence obtaining module, a target sub-data determination module, and a target sub-data storage module. The index sequence obtaining module is configured to obtain an initial index sequence corresponding to target service data. The target sub-data determination module is configured to determine target sub-data from sub-service data contained in the target service data, and determine an index space corresponding to the target sub-data in the initial index sequence as a target index space. The target sub-data storage module is configured to, in a case that a target data length of the target sub-data is less than or equal to a space data length of the target index space, write the target sub-data in the target index space and obtain a target index sequence. The target sub-data storage module is further configured to, in a case that the target data length of the target sub-data is greater than the space data length of the target index space, write the target sub-data in a target heap space corresponding to the target service data, obtain a target pointer corresponding to the target sub-data, write the target pointer in the target index space and obtain the target index sequence.

[0007] A computer device is further provided according to the present disclosure. The computer device includes a memory and a processor. The memory stores a computer program. The processor is configured, when executing the computer program, to perform the service data storage method described above.

[0008] A computer-readable storage medium is further provided according to the present disclosure. The computer-readable storage medium stores a computer program. The computer program, when executed by a processor, causes the processor to perform the service data storage method described above.

[0009] A computer program product is further provided according to the present disclosure. The computer program product includes a computer program. The computer program, when

executed by a processor, causes the processor to perform the service data storage method described above.

[0010] Details of one or more embodiments of the present disclosure are provided in the accompanying drawings and descriptions below. The other features, objectives, and advantages of the present disclosure are apparent in the specification, drawings, and claims.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] To illustrate the technical solutions in the embodiments of the present disclosure or in the conventional technology clearly, drawings to be used in the descriptions of the embodiments or the conventional technology are introduced briefly hereinafter. It is apparent that the drawings described below only illustrate the embodiments of the present disclosure. Those skilled in the art may obtain other drawings based on the provided drawings without any creative work.

[0012] FIG. 1 is a schematic diagram of an application environment of a service data storage method according to an embodiment of the present disclosure;

[0013] FIG. 2 is a flowchart of a service data storage method according to an embodiment of the present disclosure;

[0014] FIG. 3 is a flowchart of determining a space data length according to an embodiment of the present disclosure;

[0015] FIG. 4 is a schematic diagram of storing service data according to an embodiment of the present disclosure;

[0016] FIG. 5 is a schematic diagram of storing service data according to another embodiment of the present disclosure;

[0017] FIG. 6 is a schematic structural diagram of a service data storage apparatus according to an embodiment of the present disclosure;

[0018] FIG. 7 is a schematic structural diagram of a service data storage apparatus according to another embodiment of the present disclosure;

[0019] FIG. 8 is a schematic diagram of an internal structure of a computer device according to an embodiment of the present disclosure; and

[0020] FIG. 9 is a schematic diagram of an internal structure of a computer device according to another embodiment of the present disclosure.

DETAILED DESCRIPTION

[0021] To make the objectives, technical solutions, and advantages of the present disclosure clearer, the present disclosure is described in detail below in conjunction with the accompanying drawings and embodiments. It should be understood that the embodiments described herein are only used for explaining the present disclosure, and are not intended to limit the present disclosure.

[0022] A service data storage method according to an embodiment of the present disclosure may be applied to an application environment shown in FIG. 1. A terminal 102 communicates with a server 104 via a network. A data storage system may store data to be processed by the server 104. The data storage system may be integrated on the server 104 or may be configured on a cloud or other network servers. The terminal 102 may be, but is not limited to, various personal computers, laptops, smart phones, tablets, Internet of Things devices, and portable wearable devices. The Internet of Things devices may include smart TVs, smart vehicle-mounted devices, and the like. The portable wearable devices may include smart watches, smart bracelets, head mounted devices, and the like. The server 104 may be implemented by using an independent server or a server cluster including multiple servers. The terminal 102 and the server 104 may directly or indirectly communicate with each other in a wired communication manner or a wireless communication manner, which is not limited herein.

[0023] Each of the terminal and the server may individually performs the service data storage method according to the embodiment of the present disclosure.

[0024] For example, the terminal acquires an initial index sequence corresponding to target service data. The terminal determines target sub-data from sub-service data contained in the target service data, and determines an index space corresponding to the target sub-data in the initial index sequence as a target index space. In a case that a target data length of the target sub-data is less than or equal to a space data length of the target index space, the terminal writes the target sub-data in the target index space and obtains a target index sequence. In a case that the target data length of the target sub-data is greater than the space data length of

the target index space, the terminal writes the target sub-data in a target heap space corresponding to the target service data, obtains a target pointer corresponding to the target sub-data, writes the target pointer in the target index space, and obtains the target index sequence.

[0025] The terminal and the server may collaboratively perform the service data storage method according to the embodiment of the present disclosure.

[0026] For example, the terminal may transmit a service data storage request carrying a target service data identifier to the server, and the server acquires an initial index sequence corresponding to target service data based on the target service data identifier. The server determines target sub-data from sub-service data contained in the target service data, and determines an index space corresponding to the target sub-data in the initial index sequence as a target index space. In a case that a target data length of the target sub-data is less than or equal to a space data length of the target index space, the server writes the target sub-data in the target index space and obtains a target index sequence. In a case that the target data length of the target sub-data is greater than the space data length of the target index space, the server writes the target sub-data in a target heap space corresponding to the target service data, obtains a target pointer corresponding to the target sub-data, writes the target pointer in the target index space, and obtains the target index sequence. The server returns a service data storage result to the terminal.

[0027] In an embodiment, as shown in FIG. 2, a service data storage method is provided. Descriptions are provided by taking the service data storage method being applied to a computer device as an example. The computer device may be a terminal or a server. The service data storage method may be independently performed by the terminal or the server, or may be collaboratively performed by the terminal and the server. The service data storage method includes the following steps S202 to S208.

[0028] In step S202, an initial index sequence corresponding to target service data is obtained.

[0029] Service data refers to service objects containing a large amount of sub-service data to be stored. For example, the service object may be an information registration table, and data information in cells of the table is the sub-service data; or the service object may be a document, and to-be-stored data information contained in the document is the sub-service

data. The target service data is service data required to be stored. An index sequence is a sequence including index spaces corresponding to sub-service data in the target service data. The initial index sequence is an index sequence corresponding to the target service data before storing the target sub-data.

[0030] For example, the target service data is stored according to the heap storage technology. A corresponding storage space is allocated for target sub-data in a heap, then the target sub-data is written in the corresponding storage space in the heap, and a pointer corresponding to the storage space is written in a target index space corresponding to the target service data in the initial index sequence. However, in a case that the target service data includes a large amount of sub-service data with non-fixed lengths (that is, the data length of the sub-service data is not fixed) and there is a large amount of sub-service data with a data length less than or equal to the length of the index space, blindly storing the sub-service data in the heap may result in resource waste. Therefore, in storing the target service data, a computer device may firstly acquire the initial index sequence corresponding to the target service data, and then compares the space data length of the target index space in the initial index sequence with the target data length of the target sub-data to determine a storage mode corresponding to the target sub-data, so that the target sub-data may be stored in a corresponding storage mode.

[0031] In step S204, target sub-data is determined from sub-service data contained in the target service data, and an index space corresponding to the target sub-data in the initial index sequence is determined as a target index space.

[0032] The target sub-data refers to the sub-service data that is required to be stored and is determined from the sub-service data contained in the target service data. The index space refers to a storage space for storing a pointer corresponding to the sub-service data or directly storing the sub-service data. In performing data query, in a case that the data stored in the index space is a pointer, corresponding sub-service data may be obtained from the heap by using the pointer stored in the index space; and in a case that the data stored in the index space is sub-service data, the sub-service data may be directly obtained from the index space. The target index space is an index space corresponding to the target sub-data.

[0033] For example, in a case that a sub-service data storage order is configured for the target service data, the computer device determines the target sub-data from the sub-service

data contained in the target service data in the sub-service data storage order; and in a case that no sub-service data storage order is configured for the target service data, any sub-service data among the to-be-stored sub-service data contained in the target service data is determined as the target sub-data. Then, an index space corresponding to the target sub-data is determined in the initial index sequence, and the index space is determined as a target index space corresponding to the target sub-data.

[0034] In step S206, in a case that a target data length of the target sub-data is less than or equal to a space data length of the target index space, the target sub-data is written in the target index space, and a target index sequence is obtained.

[0035] The target data length refers to the data length of the target sub-data. For example, in a case that the target sub-data is character-type data, the target data length is 1 byte; and in a case that the target sub-data is integer-type data, the target data length is 4 bytes. The space data length refers to the size of the storage space occupied by the index space. For example, the space data length of the target index space may be 4 bytes, 8 bytes, and the like. The target index sequence is an index sequence obtained by updating the target index space in the initial index sequence based on the target sub-data after the target sub-data is stored.

[0036] For example, the target data length of the target sub-data is compared with the space data length of the target index space. In a case that the target data length is less than or equal to the space data length of the target index space, the target sub-data is directly written into the target index space, and an updated target index sequence is obtained. Then, target sub-data is determined from the sub-service data contained in the target service data, and an obtained initial index sequence corresponding to the target service data is returned, until all the sub-service data contained in the target service data are written in corresponding storage spaces.

[0037] In an embodiment, a security level corresponding to the target sub-data is obtained. In a case that the target data length of the target sub-data is less than or equal to the space data length and the security level of the target sub-data is less than a predetermined security level, the target sub-data is stored in the target index space. In a case that the security level of the target sub-data is greater than or equal to the predetermined security level, the target sub-data is stored in a target heap space corresponding to the target service data. Security levels are configured for the sub-service data contained in the target service data, and the storage modes

are determined based on the security levels and the data lengths corresponding to different sub-service data, thereby improving the resource utilization and ensuring the security of data storage.

[0038] In an embodiment, an access level corresponding to the target sub-data is obtained. The access level indicates a frequency at which the target sub-data is accessed. In a case that target sub-data is frequently accessed, a high access level is configured for the target sub-data. The security level corresponding to the target sub-data is obtained. In a case that the target data length of the target sub-data is less than or equal to the space data length and the access level corresponding to the target sub-data is greater than or equal to a predetermined access level, the target sub-data is stored in the target index space, thereby improving the query speed and the resource utilization. In a case that the target data length of the target sub-data is less than or equal to the space data length, the access level corresponding to the target sub-data is less than the predetermined access level and the security level corresponding to the target sub-data is greater than or equal to the predetermined security level, the target sub-data is stored in the target heap space, thereby improving the security of the target sub-data. In a case that the target data length of the target sub-data is less than or equal to the space data length, the security level corresponding to the target sub-data is less than the predetermined security level and the access level corresponding to the target sub-data is less than the predetermined access level, the target sub-data is stored in the target index space, thereby improving the query speed and the resource utilization. Security levels and access levels are configured for the sub-service data contained in the target service data, and storage modes are determined based on the security levels, the access levels, and the data lengths corresponding to different sub-service data. In this way, the security requirements and the access requirements corresponding to the different sub-service data are comprehensively considered in data storage, thereby effectively improving the flexibility of data storage.

[0039] In step S208, in a case that the target data length of the target sub-data is greater than the space data length of the target index space, the target sub-data is written into a target heap space corresponding to the target service data, a target pointer corresponding to the target sub-data is obtained, the target pointer is written into the target index space, and the target index sequence is obtained.

[0040] The target heap space refers to a heap storage space allocated for the target service

data. The target pointer is a first address of a storage space corresponding to the target sub-data in the target heap space. For example, in a case that the target sub-data is the first sub-data stored in the target heap space, the target pointer corresponding to the target sub-data is 0, and the target pointer of the second sub-data stored in the target heap space is 4 if the data length of the target sub-data is 4 bytes.

[0041] For example, the target data length of the target sub-data is compared with the space data length of the target index space. In a case that the target data length is greater than the space data length of the target index space, the computer device allocates a storage space for the target sub-data in the heap as the target heap space corresponding to the target sub-data. The target sub-data is written into the target heap space, the target pointer corresponding to the target heap space is written into the target index space, and the target index sequence is obtained. Then, target sub-data is determined from the sub-service data contained in the target service data, and an obtained initial index sequence corresponding to the target service data is returned, until all the sub-service data contained in the target service data are written in corresponding storage spaces.

[0042] According to the service data storage method in the present disclosure, the initial index sequence corresponding to the target service data is obtained. The target sub-data is determined from sub-service data contained in the target service data, and an index space corresponding to the target sub-data in the initial index sequence is determined as a target index space. In a case that the target data length of the target sub-data is less than or equal to the space data length of the target index space, the target sub-data is written in the target index space, and the target index sequence is obtained. In a case that the target data length of the target sub-data is greater than the space data length of the target index space, the target sub-data is written in the target heap space corresponding to the target service data, the target pointer corresponding to the target sub-data is obtained, the target pointer is written in the target index space, and the target index sequence is obtained. In this way, in the case that the target data length of the target sub-data is less than or equal to the space data length of the target index space, the target sub-data is directly written in the target index space without allocating a heap space for the target sub-data, avoiding configuring unnecessary resource space and thereby effectively improving the resource utilization.

[0043] In an embodiment, the service data storage method further includes: determining

data lengths corresponding to the sub-service data contained in the target service data, and obtaining an expected data length corresponding to the target service data; and determining the space data length based on the expected data length.

[0044] The expected data length refers to a length of a storage space required for storing all the sub-service data contained in the target service data.

[0045] For example, the computer device determines the data lengths corresponding to the sub-service data contained in the target service data, and calculates a sum of the data lengths corresponding to the sub-service data to obtain an expected data length of the target service data. The space data length corresponding to the index space is calculated based on the expected data length. For example, the space data length corresponding to the index space of the target service data is usually 4 bytes or 8 bytes. In a case of a space data length of 4 bytes, a maximum addressing range is 2^{31}-1; and in a case of a space data length of 8 bytes, a maximum addressing range is 2^{63}-1. In a case that an expected data length of 2^{30} bytes, the space data length may be set to 4 bytes; and in a case of an expected data length of 2^{40} bytes, the space data length may be set to 8 bytes.

[0046] In an embodiment, the computer device determines candidate data lengths corresponding to the index space based on the expected data length. The candidate data lengths refer to candidate space data lengths corresponding to the index space. A maximum addressing range corresponding to the candidate data length is greater than the expected data length, and a maximum candidate data length is 8 bytes. To improve resource utilization, a minimum candidate data length in the candidate data lengths is determined as the space data length. For example, in a case of an expected data length of 2^{30} bytes, the candidate data lengths corresponding to the index space are 4 bytes, 5 bytes, 6 bytes, 7 bytes, and 8 bytes, and the space data length is set to 4 bytes; and in a case of an expected data length of 2^{50} bytes, the candidate data lengths corresponding to the index space are 7 bytes and 8 bytes, and the space data length is set to 7 bytes.

[0047] In an embodiment, the space data length of the target service data may be configured manually. Specifically, a service type of the target service data is determined, and then the space data length required for the target service data is determined. For example, in a case that a maximum storage space required for the service type of the target service data is less than 2^{31}-1 bytes, the space data length corresponding to the target service data may be manually

configured to 4 bytes; and in a case that a maximum storage space required for the service type of the target service data is greater than 2^{31}-1 bytes, the space data length corresponding to the target service data may be manually configured to 8 bytes.

[0048] According to the above embodiments, the data lengths corresponding to the sub-service data are determined, the expected data length corresponding to the target service data is determined, and then the space data length corresponding to the index space is determined based on the expected data length, thereby effectively saving the storage resources of the computer device and improving the resource utilization.

[0049] In an embodiment, as shown in FIG. 3, the space data length is determined based on the expected data length by performing the following step S302 to step S310.

[0050] In step S302, binary conversion is performed on the expected data length to obtain a conversion result, and an initial data length is obtained based on the conversion result.

[0051] In step S304, a reference data length is obtained, and an expanded length range is determined based on the reference data length and the initial data length.

[0052] In step S306, sub-service data having a data length within the expanded length range is determined from the sub-service data contained in the target service data to obtain an expanded sub-data amount.

[0053] In step S308, a target sub-data amount is obtained based on the sub-service data contained in the target service data.

[0054] In step S310, in a case that a ratio between the expanded sub-data amount and the target sub-data amount is greater than a predetermined ratio, the initial data length is updated based on the reference data length to obtain the space data length.

[0055] The conversion result refers to a result obtained by performing binary conversion on the expected data length.

[0056] The initial data length refers to an initial space data length corresponding to an index space determined based on the expected data length. The reference data length refers to an expanded initial data length. The reference data length is greater than or equal to the initial data length. For example, the reference data length may usually be set to 8 bytes. The expanded length range refers to a data length range determined based on the initial data length and the reference data length. The initial data length is a lower limit of the expanded length

range, and the reference data length is an upper limit of the expanded length range. The expanded length range includes the reference data length. For example, in a case of an initial data length of 2 bytes and a reference data length of 4 bytes, the expanded length range is (2,4]. The expanded sub-data amount refers to the amount of sub-service data having a data length within the expanded length range. The target sub-data amount refers to the amount of the sub-service data contained in the target service data.

[0057] For example, binary conversion is performed on the expected data length to obtain a conversion result, and an initial data length corresponding to the index space is determined based on the number of bits of the conversion result. For example, in a case of an expected data length of 255 bytes, binary conversion is performed on 255 to obtain a conversion result of 11111111, the number of bits of the conversion result is 8, so that the initial data length should be greater than or equal to 1 byte. For example, the initial data length may be set to 1 byte or 2 bytes. A reference data length is obtained. In a case that the initial data length is less than the reference data length, an expanded length range is determined based on the initial data length and the reference data length. Among the sub-service data contained in the target service data, sub-service data having a data length within the expanded length range is determined to obtain an expanded sub-data amount. A target sub-data amount is obtained based on the sub-service data contained in the target service data. A ratio between the expanded sub-data amount and the target sub-data amount is calculated. In a case that the ratio is greater than or equal to a predetermined ratio, the initial data length is updated with the reference data length to obtain the space data length. For example, the predetermined ratio may be set to 0.5. That is, in a case that the data lengths of 50% or more of the sub-service data in the target service data are within the expanded length range, the initial data length is updated with the reference data length. For example, in a case of a reference data length of 8 bytes and an initial data length of 4 bytes, the initial data length is updated to 8 bytes.

[0058] According to the above embodiments, the amount of the sub-service data within the expanded length range is determined, and then the amount of the sub-service data to be expanded is determined. In a case that the ratio of the amount of the sub-service data to be expanded and the target sub-data amount is greater than the predetermined threshold, it indicates that there is still a large amount of sub-service data having a short data length that is required to be stored in the heap space. The initial data length is updated based on the reference data length, so that the sub-service data having a data length within the expanded

length range may be directly written in the index space, thereby further improving the resource utilization.

[0059] In an embodiment, the service data storage method further includes: updating a target marker bit corresponding to the target index space based on a storage mode corresponding to the target sub-data in the target index sequence. The storage mode includes a heap space storage mode and an index space storage mode. The target marker bit indicates the storage mode corresponding to the target sub-data.

[0060] The marker bit indicates the storage mode corresponding to the sub-data in the index sequence. For example, as shown in FIG. 4, the marker bit may be arranged before the index space and occupy 1 byte. In a case that the marker bit corresponding to the sub-data is 1, it indicates that the storage mode corresponding to the sub-data is an index space storage mode, that is, a pointer space storage mode; and in a case that the marker bit corresponding to the sub-data is 0, it indicates that the storage mode corresponding to the sub-data is a heap space storage mode, that is, a heap storage mode. The target marker bit refers to a marker bit corresponding to the target sub-data in the target index sequence. In the heap space storage mode, a storage space in a heap is allocated for the sub-data as a heap space corresponding to the sub-data, and the sub-data is stored in the heap space. In the index space storage mode, the sub-data is directly stored in the corresponding index space.

[0061] For example, the computer device updates the target marker bit corresponding to the target index space based on the storage mode corresponding to the target sub-data. For example, the target marker bit corresponding to the target sub-data may be set to 1 in a case that the storage mode corresponding to the target sub-data is the index space storage mode, and the target marker bit corresponding to the target sub-data may be set to 0 in a case that the storage mode corresponding to the target sub-data is the heap space storage mode.

[0062] In an embodiment, the target marker bit corresponding to the target index space may be stored outside the target index sequence. For example, a marker bit sequence may be configured, and the marker bit sequence stores marker bits corresponding to different index spaces.

[0063] According to the above embodiments, the target marker bit corresponding to the target sub-data is updated based on the storage mode corresponding to the target sub-data, so that the storage mode corresponding to the target sub-data may be determined based on the

target marker bit in performing data query, and then the target sub-data may be queried from the corresponding storage space based on the storage mode corresponding to the target sub-data, thereby ensuring the reliability of data query.

[0064] In an embodiment, the service data storage method further includes: obtaining a sub-data query request, where the sub-data query request carries a data identifier corresponding to to-be-queried sub-data; obtaining, based on the data identifier, a current index sequence corresponding to service data to which the to-be-queried sub-data belongs; determining, based on the data identifier, a to-be-queried index space corresponding to the to-be-queried sub-data in the current index sequence; determining a storage mode corresponding to the to-be-queried sub-data based on a to-be-queried marker bit corresponding to the to-be-queried index space; and obtaining the to-be-queried sub-data from a corresponding storage space based on the storage mode corresponding to the to-be-queried sub-data, and transmitting the to-be-queried sub-data to a query initiator corresponding to the sub-data query request.

[0065] The sub-data query request is a query request corresponding to the to-be-queried sub-data in the target service data. The to-be-queried sub-data is the sub-service data queried by the sub-data query request. The data identifier corresponds to the to-be-queried sub-data. The data identifier indicates a current index sequence corresponding to the to-be-queried sub-data, and indicates a to-be-queried index space corresponding to the to-be-queried sub-data in the current index sequence. The current index sequence is an index sequence corresponding to the service data to which the to-be-queried sub-data belongs. The to-be-queried index space is an index space corresponding to the to-be-queried sub-data in the current index sequence. The to-be-queried marker bit is a marker bit of the to-be-queried index space corresponding to the to-be-queried sub-data. The query initiator corresponding to the sub-data query request is a computer device that issues the sub-data query request, and the query initiator may be a terminal or a server.

[0066] For example, the computer device acquires the sub-data query request from the query initiator, determines the current index sequence corresponding to the service data to which the to-be-queried sub-data belongs based on the data identifier carried in the sub-data query request, and then determines, based on the data identifier, the to-be-queried index space corresponding to the to-be-queried sub-data in the current index sequence. Based on the

to-be-queried marker bit corresponding to the to-be-queried index space, a storage mode corresponding to the to-be-queried sub-data is determined. Based on the storage mode and the to-be-queried index space corresponding to the to-be-queried sub-data, a storage space corresponding to the to-be-queried sub-data is determined, the to-be-queried sub-data is obtained from the corresponding storage space, and the queried to-be-queried sub-data is transmitted to the query initiator corresponding to the sub-data query request.

[0067] According to the above embodiments, the sub-data query request carries the data identifier corresponding to the to-be-queried sub-data. The to-be-queried index space corresponding to the to-be-queried sub-data is determined based on the data identifier, then the storage mode corresponding to the to-be-queried sub-data is determined based on the to-be-queried marker bit corresponding to the index space, and then the to-be-queried sub-data is obtained from the corresponding storage space based on the storage mode corresponding to the to-be-queried sub-data, thereby ensuring the reliability of data query.

[0068] In an embodiment, the to-be-queried sub-data is obtained from the corresponding storage space based on the storage mode corresponding to the to-be-queried sub-data by: determining a to-be-queried storage space in a to-be-queried heap space corresponding to the to-be-queried sub-data based on a pointer stored in the to-be-queried index space, and obtaining the to-be-queried sub-data from the to-be-queried storage space, in a case that the storage mode corresponding to the to-be-queried sub-data is a heap space storage mode; and obtaining the to-be-queried sub-data from the to-be-queried index space in a case that the storage mode corresponding to the to-be-queried sub-data is an index space storage mode.

[0069] The to-be-queried heap space is a heap storage space allocated for the target service data to which the to-be-queried sub-data belongs. The to-be-queried storage space is a storage space in which the to-be-queried sub-data is stored in the to-be-queried heap space.

[0070] For example, in a case that the storage mode corresponding to the to-be-queried sub-data is the heap space storage mode, a pointer corresponding to the to-be-queried sub-data is obtained from the to-be-queried index space, a to-be-queried storage space corresponding to the to-be-queried sub-data is determined in the to-be-queried heap space corresponding to the to-be-queried sub-data based on the pointer, and then the to-be-queried sub-data is obtained from the to-be-queried storage space. In a case that the storage mode corresponding to the to-be-queried sub-data is the index space storage mode, the to-be-queried sub-data is obtained

from the to-be-queried index space corresponding to the to-be-queried sub-data.

[0071] According to the above embodiments, the to-be-queried sub-data is directly obtained from the to-be-queried index space corresponding to the to-be-queried sub-service data in a case that the storage mode corresponding to the to-be-queried sub-data is the index space storage mode, avoiding the time cost of locating the to-be-queried storage space corresponding to the to-be-queried data based on a pointer, and thereby improving the efficiency of data query.

[0072] In an embodiment, the target service data is a target table, and the target sub-data is target cell data.

[0073] The target table is a table of which data is required to be stored, and the target table contains a large amount of cell data with variable length. The target cell data is the data stored in cells of the target table.

[0074] In an embodiment, the service data storage method according to the present disclosure may be applied for storing data in a table. The service data storage method includes the following steps 1 to 3.

[0075] In step 1, an index sequence corresponding to a target table is obtained.

[0076] The computer device obtains an initial index sequence corresponding to the target table, and obtains a space data length of an index space in the initial index sequence.

[0077] In step 2, target cell data and a target index space are determined.

[0078] The computer device determines the target cell data from cell data contained in the target table, and determines an index space corresponding to the target cell data in the initial index sequence as the target index space.

[0079] In step 3, the target cell data is stored.

[0080] As shown in FIG. 5, in a case that the data length corresponding to the target cell data is less than or equal to the space data length of the index space, the target cell data is written in the target index space (that is, storing in the pointer space storage mode), and the marker bit corresponding to the target index space is updated to obtain the target index sequence. In a case that the data length corresponding to the target cell data is greater than the space data length of the index space, the target cell data and information such as the data

length corresponding to the target cell data are written in the target heap space corresponding to the target table, a target pointer corresponding to the target cell data is obtained and written in the target index space, and a target index sequence is obtained.

[0081] According to the above embodiments, in a case that the target table contains a large amount of cell data having a data length less than or equal to the length of the index space, the cell data is directly stored in the corresponding index space. The index space is fully utilized, and a marker bit is used for indicating whether the data is directly stored in the index space or is stored in a heap. Thus, the resource utilization is improved significantly, and the waste of resources is reduced. In addition, the cell data is stored in the index space, thus the consumption of determining storage space based on a pointer may be reduced in querying data, that is, the time overhead of locating the storage space corresponding to the cell data in the heap based on the pointer may be avoided, thereby improving data obtaining performance and achieving an efficient program.

[0082] It should be understood that although the steps in the flowcharts involved in the above embodiments are displayed in an order indicated by arrows, it may be unnecessary to perform the steps in the order indicated by the arrows. Unless otherwise specified in this specification, the order of performing the steps is not strictly limited, and the steps may be performed in other orders. Moreover, at least a part of the steps in the flowcharts involved in the embodiments described above may include multiple steps or multiple stages, which are unnecessarily performed at the same time and may be performed at different time instants. These steps or stages are unnecessarily performed sequentially, and may be performed alternately with other steps or at least a part of steps or stages in the other steps.

[0083] Based on the same inventive concept, a service data storage apparatus for performing the service data storage method is further provided according to an embodiment of the present disclosure. The implementation solutions provided by the service data storage apparatus are similar to the implementation solutions described in the service data storage method. Hence, the limitations in the following embodiments of the service data storage apparatus may refer to the limitations of the service data storage method described above, and are not repeated herein.

[0084] In an embodiment, as shown in FIG. 6, a service data storage apparatus is provided. The service data storage apparatus includes: an index sequence obtaining module 602, a target

sub-data determination module 604, and a target sub-data storage module 606.

[0085] The target sub-data determination module is configured to determine target sub-data from sub-service data contained in target service data, and determine an index space corresponding to the target sub-data in the initial index sequence as a target index space.

[0086] The target sub-data storage module is configured to, in a case that a target data length of the target sub-data is less than or equal to a space data length of the target index space, write the target sub-data in the target index space and obtain a target index sequence.

[0087] The target sub-data storage module is further configured to, in a case that the target data length of the target sub-data is greater than the space data length of the target index space, write the target sub-data in a target heap space corresponding to the target service data, obtain a target pointer corresponding to the target sub-data, write the target pointer in the target index space and obtain the target index sequence.

[0088] According to the service data storage apparatus in the present disclosure, the initial index sequence corresponding to the target service data is obtained. The target sub-data is determined from sub-service data contained in the target service data, and an index space corresponding to the target sub-data in the initial index sequence is determined as a target index space. In a case that the target data length of the target sub-data is less than or equal to the space data length of the target index space, the target sub-data is written in the target index space, and the target index sequence is obtained. In a case that the target data length of the target sub-data is greater than the space data length of the target index space, the target sub-data is written in the target heap space corresponding to the target service data, the target pointer corresponding to the target sub-data is obtained, the target pointer is written in the target index space, and the target index sequence is obtained. In this way, in the case that the target data length of the target sub-data is less than or equal to the space data length of the target index space, the target sub-data is directly written in the target index space without allocating a heap space for the target sub-data, avoiding configuring unnecessary resource space and thereby effectively improving the resource utilization.

[0089] In an embodiment, as shown in FIG. 7, the service data storage apparatus further includes: a space data length determination module 702, a target marker bit updating module 704, and a query request response module 706.

[0090] The space data length determination module 702 is configured to: determine data lengths corresponding to the sub-service data contained in the target service data, and obtain an expected data length corresponding to the target service data; and determine the space data length based on the expected data length.

[0091] The target marker bit updating module 704 is configured to update a target marker bit corresponding to the target index space based on a storage mode corresponding to the target sub-data in the target index sequence. The storage mode includes a heap space storage mode and an index space storage mode. The target marker bit indicates the storage mode corresponding to the target sub-data.

[0092] The query request response module 706 is configured to: obtain a sub-data query request, where the sub-data query request carries a data identifier corresponding to to-be-queried sub-data; obtain, based on the data identifier, a current index sequence corresponding to service data to which the to-be-queried sub-data belongs; determine, based on the data identifier, a to-be-queried index space corresponding to the to-be-queried sub-data in the current index sequence; determine a storage mode corresponding to the to-be-queried sub-data based on a to-be-queried marker bit corresponding to the to-be-queried index space; and obtain the to-be-queried sub-data from a corresponding storage space based on the storage mode corresponding to the to-be-queried sub-data, and transmit the to-be-queried sub-data to a query initiator corresponding to the sub-data query request.

[0093] In an embodiment, the space data length determination module 702 is further configured to: perform binary conversion on the expected data length to obtain a conversion result, and obtain an initial data length based on the conversion result; obtain a reference data length, and determine an expanded length range based on the reference data length and the initial data length; determine sub-service data having a data length within the expanded length range, from the sub-service data contained in the target service data, to obtain an expanded sub-data amount; obtain a target sub-data amount based on the sub-service data contained in the target service data; and update the initial data length based on the reference data length to obtain the space data length in a case that a ratio between the expanded sub-data amount and the target sub-data amount is greater than a predetermined ratio.

[0094] In an embodiment, the query request response module 706 is further configured to: determine a to-be-queried storage space in a to-be-queried heap space corresponding to the

to-be-queried sub-data based on a pointer stored in the to-be-queried index space, and obtain the to-be-queried sub-data from the to-be-queried storage space, in a case that the storage mode corresponding to the to-be-queried sub-data is a heap space storage mode; and obtain the to-be-queried sub-data from the to-be-queried index space in a case that the storage mode corresponding to the to-be-queried sub-data is an index space storage mode.

[0095] The modules in the service data storage apparatus may be entirely or partially implemented by software, hardware, or a combination thereof. The modules may be embedded in or independent of the processor in the computer device in a hardware form, or may be stored in the memory in the computer device in a software form, so that the processor may call and execute the operations corresponding to the modules.

[0096] In an embodiment, a computer device is provided. The computer device may be a server. FIG. 8 shows an internal structure of the computer device. The computer device includes a processor, a memory, an input/output (I/O) interface, and a communication interface. The processor, the memory and the input/output interface are connected to each other via a system bus, and the communication interface is connected to the system bus via the input/output interface. The processor of the computer device is configured to perform computing and controlling. The memory of the computer device includes a non-volatile storage medium and an internal memory. The non-volatile storage medium stores an operating system, a computer program and a database. The internal memory provides an environment for the operation of the operating system and the computer program stored in the non-volatile storage medium. The database of the computer device stores data such as the initial index sequence and the target sub-data. The input/output interface of the computer device is configured to exchange information between the processor and external devices. The communication interface of the computer device is configured to communicate with external terminals through a network connection. The computer program, when executed by the processor, causes the processor to perform the service data storage method.

[0097] In an embodiment, a computer device is provided. The computer device may be a terminal. FIG. 9 shows an internal structure of the computer device. The computer device includes a processor, a memory, an input/output interface, a communication interface, a display unit, and an input device. The processor, the memory, and the input/output interface are connected via a system bus. The communication interface, the display unit and the input

device are connected to the system bus via the input/output interface. The processor of the computer device is configured to perform computing and controlling. The memory of the computer device includes a non-volatile storage medium and an internal memory. The non-volatile storage medium stores an operating system and a computer program. The internal memory provides an environment for the operation of the operating system and the computer program stored in the non-volatile storage medium. The input/output interface of the computer device is configured to exchange information between the processor and external devices. The communication interface of this computer device is configured to perform wired communication or wireless communication with external terminals. The wireless communication may be performed through WIFI, mobile cellular networks, NFC (near field communication), or other technologies. The computer program, when executed by the processor, causes the processor to perform the service data storage method. The display unit of the computer device is configured to form visible images, and may be a display screen, a projection device, or a virtual-reality imaging device. The display screen may be a liquid crystal display screen or an electronic ink display screen. The input device of the computer device may be a touch layer covered on the display screen, may also be a key, a trackball or a touchpad that are arranged on a shell of the computer device, and may be an external keyboard, touchpad, mouse, or the like.

[0098] It should be understood by those skilled in the art that, FIG. 8 and FIG. 9 merely show block diagrams of a portion of structures related to the solutions of the present disclosure, and do not limit the computer device to which the solutions of the present disclosure are applied. The computer device may include more or fewer components than shown in FIG. 8 and FIG. 9, or combine some components, or have a different component arrangement.

[0099] In an embodiment, a computer device is provided. The computer device includes a memory and a processor. The memory stores a computer program. The processor is configured, when executing the computer program, to perform the method described in the above embodiments.

[0100] In an embodiment, a computer-readable storage medium is provided. The computer-readable storage medium stores a computer program. The computer program, when executed by a processor, causes the processor to perform the method described in the above

embodiments.

[0101] In an embodiment, a computer program product or a computer program is provided. The computer program product or the computer program includes computer instructions. The computer instructions are stored in a computer-readable storage medium. A processor of a computer device reads the computer instructions from the computer-readable storage medium, and executes the computer instructions to perform the method described in the above embodiments.

[0102] It should be noted that the user information (including but not limited to user device information of user and user personal information) and data (including but not limited to data for analysis, stored data and displayed data) involved in the present disclosure are authorized by the user or fully authorized by all parties. The collection, usage, and processing of the relevant data comply with relevant laws, regulations, and standards of relevant countries and regions.

[0103] It should be understood by those skilled in the art that all or parts of the processes in the method in the embodiments as described above may be implemented by instructing relevant hardware through the computer program. The computer program may be stored in a non-volatile computer-readable storage medium. The methods in the embodiments may be performed when the computer program is executed. The memory, the database or other medium used in the embodiments of the present disclosure may include at least one of a non-volatile memory and a volatile memory. The non-volatile memory may include a read-only memory (ROM), a tape, a floppy disk, a flash memory, an optical memory, a high-density embedded non-volatile memory, a resistive random access memory (ReRAM), a magnetoresistive random access memory (MRAM), a ferroelectric random access memory (FRAM), a phase change memory (PCM), a graphene memory, and the like. The volatile memory may include a random access memory (RAM), an external cache memory, or the like. As an illustration rather than a limitation, the RAM may be in many forms, such as a static random access memory (SRAM) or a dynamic random access memory (DRAM). The database involved in the embodiments of the present disclosure may include at least one of a relational database and a non-relational database. The non-relational database may include, but is not limited to, a distributed database based on blockchain and the like. The processor involved in the embodiments of the present disclosure may include, but is not limited to, a

general-purpose processor, a central processor unit, a graphics processor, a digital signal processor, a programmable logic unit, and a data processing logic unit based on quantum computing.

[0104] The technical features in the above embodiments may be combined arbitrarily. To make the descriptions concise, not all possible combinations of the technical features in the embodiments are described herein. All combinations of these technical features are regarded as falling within the scope of the present disclosure, as long as there is no conflict.

[0105] The above embodiments, described specifically and in detail, are only provided for illustrating the present disclosure, and are not intended to limit the scope of the present disclosure. It should be noted that modifications and improvements may be made by those skilled in the art without departing from the concept of the present disclosure. These modifications and improvements fall within the protection scope of the present disclosure. Therefore, the protection scope of the present disclosure is subject to the attached claims.

CLAIMS

1. A service data storage method, comprising:

obtaining an initial index sequence corresponding to target service data;

determining target sub-data from sub-service data contained in the target service data, and determining an index space corresponding to the target sub-data in the initial index sequence as a target index space;

writing the target sub-data in the target index space, and obtaining a target index sequence, in a case that a target data length of the target sub-data is less than or equal to a space data length of the target index space; and

writing the target sub-data in a target heap space corresponding to the target service data, obtaining a target pointer corresponding to the target sub-data, writing the target pointer in the target index space and obtaining the target index sequence, in a case that the target data length of the target sub-data is greater than the space data length of the target index space.

2. The service data storage method according to claim 1, further comprising:

determining data lengths corresponding to the sub-service data contained in the target service data, and obtaining an expected data length corresponding to the target service data; and

determining the space data length based on the expected data length.

3. The service data storage method according to claim 2, wherein the determining the space data length based on the expected data length comprises:

performing binary conversion on the expected data length to obtain a conversion result, and obtaining an initial data length based on the conversion result;

obtaining a reference data length, and determining an expanded length range based on the reference data length and the initial data length;

determining sub-service data having a data length within the expanded length range,

from the sub-service data contained in the target service data, to obtain an expanded sub-data amount;

obtaining a target sub-data amount based on the sub-service data contained in the target service data; and

updating the initial data length based on the reference data length to obtain the space data length in a case that a ratio between the expanded sub-data amount and the target sub-data amount is greater than a predetermined ratio.

4. The service data storage method according to claim 1, further comprising:

updating a target marker bit corresponding to the target index space based on a storage mode corresponding to the target sub-data in the target index sequence, wherein the storage mode comprises a heap space storage mode and an index space storage mode, and the target marker bit indicates the storage mode corresponding to the target sub-data.

5. The service data storage method according to claim 1, further comprising:

obtaining a sub-data query request, wherein the sub-data query request carries a data identifier corresponding to to-be-queried sub-data;

obtaining, based on the data identifier, a current index sequence corresponding to service data to which the to-be-queried sub-data belongs;

determining, based on the data identifier, a to-be-queried index space corresponding to the to-be-queried sub-data in the current index sequence;

determining a storage mode corresponding to the to-be-queried sub-data based on a to-be-queried marker bit corresponding to the to-be-queried index space; and

obtaining the to-be-queried sub-data from a corresponding storage space based on the storage mode corresponding to the to-be-queried sub-data, and transmitting the to-be-queried sub-data to a query initiator corresponding to the sub-data query request.

6. The service data storage method according to claim 5, wherein the obtaining the

to-be-queried sub-data from a corresponding storage space based on the storage mode corresponding to the to-be-queried sub-data comprises:

determining a to-be-queried storage space in a to-be-queried heap space corresponding to the to-be-queried sub-data based on a pointer stored in the to-be-queried index space, and obtaining the to-be-queried sub-data from the to-be-queried storage space, in a case that the storage mode corresponding to the to-be-queried sub-data is a heap space storage mode; and

obtaining the to-be-queried sub-data from the to-be-queried index space in a case that the storage mode corresponding to the to-be-queried sub-data is an index space storage mode.

7. The service data storage method according to any one of claims 1 to 6, wherein the target service data is a target table, and the target sub-data is target cell data.

8. A service data storage apparatus, comprising:

an index sequence obtaining module, configured to obtain an initial index sequence corresponding to target service data;

a target sub-data determination module, configured to determine target sub-data from sub-service data contained in the target service data, and determine an index space corresponding to the target sub-data in the initial index sequence as a target index space;

a target sub-data storage module, configured to, in a case that a target data length of the target sub-data is less than or equal to a space data length of the target index space, write the target sub-data in the target index space and obtain a target index sequence; and

the target sub-data storage module, further configured to, in a case that the target data length of the target sub-data is greater than the space data length of the target index space, write the target sub-data in a target heap space corresponding to the target service data, obtain a target pointer corresponding to the target sub-data, write the target pointer in the target index space and obtain the target index sequence.

9. A computer device, comprising:

a memory, storing a computer program; and

a processor, wherein

the processor is configured, when executing the computer program, to perform the service data storage method according to any one of claims 1 to 7.

10. A computer-readable storage medium, storing a computer program, wherein the computer program, when executed by a processor, causes the processor to perform the service data storage method according to any one of claims 1 to 7.