DATA QUERY METHOD AND SYSTEM, AND STORAGE MEDIUM

TECHNICAL FIELD

[0001] This specification relates to the field of data processing technologies, and in particular, to a data query method and system, and a storage medium.

BACKGROUND

[0002] In a rapidly developing information society, a data query is an essential step before data processing and analysis. The data query needs to be performed in many scenarios.

[0003] For example, in the payment field, a payment platform needs to query information (for example, logistics information corresponding to the target transaction) related to a target transaction from a data source, and verify authenticity of the target transaction based on the found information; or query information (for example, business information of a target merchant) related to the target merchant from a data source, and verify authenticity of the target merchant based on the found information.

[0004] Regardless of a scenario, efficiently finding required data from the data source is an important objective of the data query step, and determines accuracy of a subsequent data processing step. Therefore, a solution in which the required data can be efficiently found from the data source in any scenario needs to be provided.

[0005] It is worthwhile to note that, content of the related technology is merely information known to the inventor, and neither means that the information has entered a public domain before an application date of this specification, nor means that the information can become a conventional technology of this specification.

SUMMARY

[0006] This specification provides a data query method and system, and a storage medium, to avoid at least one of the above-mentioned technical problems.

[0007] According to a first aspect, this specification provides a data query method, including: obtaining indicator data of a plurality of data sources in response to a data query request, where indicator data of one data source represents query performance of the data source in a historical period; sorting the plurality of data sources based on the indicator data of the plurality of data sources, to obtain a target order; and performing a data query in at least some of the plurality of data sources based on the target order, to obtain a data query result.

[0008] According to a second aspect, this specification provides a data query system, including: at least one storage medium, storing at least one instruction set to perform a data query; and at least one processor, communicatively connected to the at least one storage medium, where when running, the at least one processor reads the at least one instruction set, and performs the following operations based on an indication of the at least one instruction set: obtaining indicator data of a plurality of data sources in response to a data query request, where indicator data of one data source represents query performance of the data source in a historical period; sorting the plurality of data sources based on the indicator data of the plurality of data sources, to obtain a target order; and performing a data query in at least some of the plurality of data sources based on the target order, to obtain a data query result.

[0009] According to a third aspect, this specification provides a computer-readable non-transitory storage medium. The computer-readable non-transitory storage medium stores at least one instruction set, and when the at least one instruction set is executed by at least one processor, the method according to the first aspect is implemented.

[0010] It can be learned from the above-mentioned technical solutions that, according to the data query method and system, and the storage medium that are provided in this specification, dynamic indicator data of the plurality of data sources is obtained, to dynamically sort the plurality of data sources based on the dynamic indicator data, and perform a data query based on the sorted plurality of data sources. A data source for a data query can be dynamically selected, instead of fixedly selecting a data source for a data query in the above-mentioned related technology. Flexibility and diversity of selecting the data source can be implemented, and data query experience of a user can also be improved.

[0011] Other functions of the data query method and system, and the storage medium provided in this specification are partially listed in the following descriptions. Creative aspects of the data query method and system, and the storage medium provided in this specification can be fully

explained by practice or by using the methods, apparatuses, and combinations described in the following detailed examples.

BRIEF DESCRIPTION OF DRAWINGS

[0012] To describe the technical solutions in the embodiments of this specification more clearly, the following briefly describes the accompanying drawings needed for describing the embodiments. Clearly, the accompanying drawings in the following descriptions show merely some embodiments of this specification, and a person of ordinary skill in the art can still derive other drawings from these accompanying drawings without creative efforts.

[0013] FIG. 1 is a schematic diagram of a fixed query sequence according to an embodiment of this specification;

[0014] FIG. 2 is a schematic diagram of an application scenario of a data query method according to an embodiment of this specification;

[0015] FIG. 3 is a diagram of a hardware structure of a data query system according to an embodiment of this specification;

[0016] FIG. 4 is a schematic flowchart of a data query method according to an embodiment of this specification;

[0017] FIG. 5 is a schematic principle diagram of a data query method according to an embodiment of this specification;

[0018] FIG. 6 is a schematic principle diagram of a data query method according to another embodiment of the present specification;

[0019] FIG. 7 is a schematic diagram of a dynamically changing query sequence according to an embodiment of this specification; and

[0020] FIG. 8 is a schematic diagram of a visual interface according to an embodiment of this specification.

DESCRIPTION OF EMBODIMENTS

[0021] Example embodiments are described in detail herein, and examples of the example embodiments are presented in the accompanying drawings. When the following description relates to the accompanying drawings, unless specified otherwise, same numbers in different

accompanying drawings represent same or similar elements. Implementations described in the following example embodiments do not represent all implementations consistent with this specification. On the contrary, the embodiments are merely examples of apparatuses and methods that are described in the appended claims in details and consistent with some aspects of this specification.

[0022] It should be understood that the terms "include" and "have" in the embodiments of this specification and any variants thereof are intended to cover the non-exclusive inclusion, for example, a product or device that includes a series of components are not necessarily limited to those components that are clearly listed, but can include other components that are not clearly listed or inherent to such product or device.

[0023] In the embodiments of this specification, the term "and/or" describes an association relationship between associated objects and indicates that three relationships can exist. For example, "A and/or B" can indicate the following three cases: Only A exists, both A and B exist, and only B exists. The character "/" usually indicates an "or" relationship between the associated objects.

[0024] In the embodiments of this specification, the term "a plurality of" refers to two or more, and other quantifiers are similar to this term.

[0025] The terms "first", "second", "third", etc. in this specification are used to distinguish between similar or same objects or entities, and do not necessarily mean limiting a specific order or sequence, unless otherwise indicated. It should be understood that the terms used in such a way is interchangeable in appropriate cases, for example, can be implemented in an order other than those illustrated or described in the embodiments of this specification.

[0026] The term "unit/module" used in this specification means any known or later developed hardware, software, firmware, artificial intelligence, fuzzy logic, or a combination of hardware code or/and a combination of software code, and can perform functions related to the element.

[0027] To facilitate readers' understanding of this specification, application scenarios of this specification are described.

[0028] The technical solutions provided in this specification are applicable to a scenario in which a data query needs to be performed from a plurality of data sources. For example, the technical solutions provided in this specification can be applied to a merchant entity authenticity verification scenario, a logistics query scenario, a transaction authenticity verification scenario,

etc., which are not listed one by one herein.

[0029] The merchant entity authenticity verification scenario is used as an example

[0030] A merchant can initiate a joining request to an e-commerce platform, and the joining request can carry an identifier of the merchant (for example, a name, a website domain name, an address, etc. of the merchant). The e-commerce platform needs to query business information of the merchant, to ensure authenticity of a merchant entity. A plurality of data sources are used to query the business information. The e-commerce platform can execute the technical solutions of this specification to query one or more data sources to obtain the business information.

[0031] The logistics query scenario is used as an example:

[0032] A user can send a logistics query request to a logistics query platform. The logistics query request can carry a logistics tracking number. A plurality of data sources support a logistics query. For example, the plurality of data sources can be provided by different logistics suppliers. A logistics platform can execute the technical solutions of this specification to query one or more data sources to obtain logistics information (for example, a logistics status) corresponding to the logistics tracking number.

[0033] The transaction authenticity verification scenario is used as an example:

[0034] The authenticity verification scenario can be specifically an authenticity verification scenario under trade risk control. For example, a payment platform usually needs to verify authenticity of a transaction to prevent and control a risky transaction. Specifically, a participant of the transaction can initiate a payment request to the payment platform (when a user initiates a payment request or when a merchant initiates a collection request), and can further provide the payment platform with a transaction proof material, for example, a logistics tracking number. Similarly, a plurality of data sources support a logistics query. The payment platform can execute the technical solutions of this specification to query one or more data sources to obtain a logistics status corresponding to the logistics tracking number. Based on this, the payment platform can further verify the authenticity of the transaction based on the logistics status, to avoid a compliance or fraud risk.

[0035] With reference to the above-mentioned analysis, it can be learned that in actual applications, a plurality of data sources may be available for a data query. When a data query is performed for a specific object, a query in some data sources may fail, or only partial information may be found from some data sources. Therefore, in some embodiments, to ensure integrity of a

query result, a query sequence of a plurality of data sources can be preconfigured, and a data query is sequentially performed in the plurality of data sources in the query sequence, until a complete data query result is obtained.

[0036] For example, as shown in FIG. 1, the plurality of data sources include a data source 1, a data source 2, and a data source 3. It is assumed that the preconfigured query sequence is: data source 1>data source 2>data source 3.

[0037] A data query system can determine a primary-secondary relationship between different data sources based on the fixed query sequence. For example, the data query system determines the data source 1 as a primary data source, and determines the data source 2 and the data source 3 as secondary data sources. When there are a plurality of secondary data sources, the data query system can further identify different secondary data sources based on a fixed query sequence. As shown in FIG. 1, the data source 2 whose query sequence is earlier is identified as a secondary data source 1, and the data source 3 whose query sequence is later is identified as a secondary data source 2.

[0038] The data query system sequentially performs a data query in the three data sources based on the fixed query sequence, until a complete data query result is obtained. That is, each time the data query system performs a data query (from the 1^{st} data query to the T^{th} data query in FIG. 1, where T is an integer greater than 1), the data query is first performed based on a primary data source (namely, the data source 1). If the query fails, the data query continues to be performed based on the data source 2 in the secondary data sources. If the query still fails, the data query continues to be performed based on the data source 3 in the secondary data sources.

[0039] With reference to the above-mentioned analysis, it can be learned that, in the above-mentioned embodiment, the query sequence between the plurality of data sources is fixed. However, in the fixed query sequence, the data query system cannot meet query requirements of different scenarios. For example, an accuracy requirement for a data query result is relatively high in some scenarios, a data query speed requirement is relatively high in some scenarios, and data query costs need to be as low as possible in some scenarios. It can be learned that according to the above-mentioned solution, the data query system cannot adapt to a scenario change in a timely manner, and query experience of a user is reduced.

[0040] Therefore, to avoid at least one of the above-mentioned problems, this specification proposes a creative technical concept: dynamically sorting a plurality of data sources based on

dynamic indicator data of the plurality of data sources, and performing a data query in the plurality of data sources in a sorted sequence.

[0041] For example, query performance of each data source dynamically changes with time. The data query system can determine dynamically changing indicator data of the data source based on the dynamically changing query performance, to sort the plurality of data sources based on the dynamically changing indicator data, so that such sorting is also dynamic. Correspondingly, the data query system performs a data query in at least some of the plurality of data sources in a dynamically sorted sequence, to obtain a data query result.

[0042] It can be learned from the above-mentioned analysis that, in the above-mentioned technical solutions, the plurality of data sources are dynamically sorted based on the dynamically changing indicator data, and the data query is performed in the plurality of data sources in the dynamically sorted sequence. Compared with a case of performing the data query in the plurality of data sources in the fixed sequence, in the above-mentioned technical solutions, there is relatively high flexibility and diversity in terms of a query sequence of the data sources, to help efficiently find required data in a plurality of scenarios, and improve data query experience of the user.

[0043] FIG. 2 is a schematic diagram of an application scenario of a data query method according to an embodiment of this specification. The data query method in this specification can be applied to an application scenario 200 shown in FIG. 2. As shown in FIG. 2, the application scenario 200 can include a target user 201, a client 202, a server 203, and a network 204.

[0044] The target user 201 can be a user who initiates a data query request. For example, the target user 201 can perform a target operation on the client 202, to initiate a data query request. It can be learned from the above-mentioned examples that the technical solutions in this specification can be applied to different application scenarios. A manner in which the target user 201 initiates the data query request varies with the application scenario.

[0045] For example, for a joining scenario, the target user 201 can initiate a joining request on the client 202. The joining request can trigger a data query request. For a logistics scenario, the target user 201 can initiate a logistics query request on the client 202. The logistics query request can be specifically a data query request used to query a logistics status. For a transaction scenario, the target user 201 can initiate a transaction request on the client 202. The transaction request can trigger a data query request.

[0046] The client 202 can be an electronic device that provides an interaction function for the

target user 201. For example, the client 202 can provide an interaction interface for the target user 201, and the target user 201 can perform an interaction operation in the interaction interface. In some embodiments, the client 202 performs the data query method described in this specification in response to detecting a data query operation triggered by the target user 201. In this case, the client 202 can store data or instructions for performing the data query method described in this specification, and can execute or be configured to execute the data or the instructions. In some embodiments, the client 202 can include a hardware device with a data information processing function and a necessary program needed to drive the hardware device to work, to perform the data query method described in this specification.

[0047] In some embodiments, the client 202 can include a mobile device, a tablet computer, a laptop computer, a built-in device in a motor vehicle or similar content, or any combination thereof. In some embodiments, the mobile device can include a smart home device, a smart mobile device, a virtual reality device, an augmented reality device or a similar device, or any combination thereof. In some embodiments, the smart home apparatus can include a smart TV, a desktop computer, or any combination. In some embodiments, the smart mobile device can include a smartphone, a personal digital assistance, a game device, a navigation device, or any combination thereof. In some embodiments, the built-in apparatus in the motor vehicle can include a vehicle-mounted computer, a vehicle-mounted television, etc.

[0048] In some embodiments, one or more applications (app) can be installed on the client 202. The app can provide the target user 201 with a capability and an interface for interacting with the outside through the network 204. The app includes but is not limited to a web browser app program, a search app program, a chat app program, a shopping app program, a video app program, a financial app program, an instant messaging tool, an email client, social platform software, etc. In some embodiments, a target app can be installed on the client 202. The target app can collect the data query request for the client 202.

[0049] As shown in FIG. 2, the client 202 can be communicatively connected to the server 203. The server 203 can be communicatively connected to one client 202, or can be communicatively connected to a plurality of clients 202. In some embodiments, the client 202 can interact with the server 203 through the network 204, to receive or send a message, etc. For example, the client 202 can interact with the server 203 through the network 204, to send the data query request to the server 203.

[0050] The server 203 can be a server that provides various services. For example, the server 203 can be a cloud server or a local server. The server 203 can be communicatively connected to one client 202 and receive data sent by the client 202, or can be communicatively connected to a plurality of clients 202, and receive data sent by each client 202.

[0051] In some embodiments, the data query method described in this specification can be performed on the server 203. In this case, the server 203 can store data or instructions for performing the data query method described in this specification, and can execute or be configured to execute the data or the instructions. The server 203 can include a hardware device with a data information processing function and a necessary program needed for driving the hardware device to work.

[0052] The network 204 is a medium configured to provide a communication connection between the client 202 and the server 203. The network 204 can facilitate exchange of information or data. As shown in FIG. 2, the client 202 and the server 203 can be separately connected to the network 204, and transmit information or data to each other through the network 204.

[0053] In some embodiments, the network 204 can be any type of wired or wireless network, or a combination thereof. For example, the network 204 can include a cable network, a wired network, an optical fiber network, a telecommunication network, an intranet, the internet, a local area network (LAN), a wide area network (WAN), a wireless local area network (WLAN), a metropolitan area network (MAN), a public switched telephone network (PSTN), a Bluetooth network™, a short-range wireless network (ZigBee™), a near field communication (NFC) network, or a similar network.

[0054] In some embodiments, the network 204 can include one or more network access points. For example, the network 204 can include a wired or wireless network access point, for example, a base station or an Internet switching point. Through the access point, one or more components of the client 202 and the server 203 can be connected to the network 204 to exchange data or information.

[0055] It is worthwhile to note that quantities of clients 202, servers 203, and networks 204 in FIG. 2 are merely examples. There can be any quantity of clients 202, servers 203, and networks 204 based on implementation needs. In addition, the data query method provided in this specification can be completely performed on the client 202, can be completely performed on the server 203, or can be partially performed on the client 202 and partially performed on the server

203.

[0056] In other words, FIG. 2 and the above-mentioned descriptions for FIG. 2 are merely used to describe an example of an application scenario to which the data query method in this specification may be applied, and cannot be understood as a limitation on the application scenario.

[0057] FIG. 3 is a diagram of a hardware structure of a data query system 300 according to an embodiment of this specification. The data query system 300 can perform the data query method described in this specification. The data query method is described in other parts of this specification. When the data query method is performed on a client 202, the data query system 300 can be the client 202. When the data query method is performed on a server 203, the data query system 300 can be the server 203. When the data query method can be partially performed on the client 202 and partially performed on the server 203, the data query system 300 can be a system including the client 202 and the server 203.

[0058] As shown in FIG. 3, the data query system 300 can include at least one storage medium 303 and at least one processor 302. In some embodiments, the data query system 300 can further include a communication port 304 and an internal communication bus 301. The data query system 300 can further include an I/O component 305.

[0059] The internal communication bus 301 can be connected to different system components. For example, the internal communication bus 301 can be connected to the storage medium 303, the processor 302, the communication port 304, and the I/O component 305.

[0060] The I/O component 305 supports an input/output between the data query system 300 and another component.

[0061] The communication port 304 is used for data communication between the data query system 300 and the outside. For example, the communication port 304 can be used for data communication between the query system 300 and the network 204. The communication port 304 can be a wired communication port, or can be a wireless communication port.

[0062] The storage medium 303 can include a data storage apparatus. The data storage apparatus may be a non-transitory storage medium, or may be a transitory storage medium. For example, the data storage apparatus can include one or more of a disk 3031, a read-only storage medium (ROM) 3032, or a random access storage medium (RAM) 3033. The storage medium 303 further includes at least one instruction set stored in the data storage apparatus. The instruction set includes computer program code. The computer program code can include a program, a routine,

an object, a component, a data structure, a process, a module, etc. for performing the data query method provided in this specification.

[0063] The at least one processor 302 can be communicatively connected to the at least one storage medium 303. The at least one processor 302 is configured to execute the at least one instruction set. When the data query system 300 runs, the at least one processor 302 reads the at least one instruction set and performs, based on an instruction of the at least one instruction set, the data query method provided in this specification. The processor 302 can perform all steps included in the data query method. The processor 302 can be in a form of one or more processors. In some embodiments, the processor 302 can include one or more hardware processors, for example, a microcontroller, a microprocessor, a reduced instruction set computer (RISC), an application-specific integrated circuit (ASIC), an application-specific instruction set processor (ASIP), a central processing unit (CPU), a graphics processing unit (GPU), a physical processing unit (PPU), a microcontroller unit, a digital signal processor (DSP), a field programmable gate array (FPGA), an advanced RISC machine (ARM), a programmable logic device (PLD), any circuit or processor that can perform one or more functions, or any combination thereof.

[0064] To merely describe a problem, only one processor 302 is shown in the data query system 300 in the accompanying drawings. However, it is worthwhile to note that, the data query system 300 in this specification can further include multiple processors. Therefore, the operations and/or method steps disclosed in this specification can be performed by one processor, or can be jointly performed by multiple processors. For example, if the processor 302 in the data query system 300 in this specification performs step A and step B, it should be understood as that step A and step B can be jointly or separately performed by two different processors 302 (for example, a first processor performs step A, and a second processor performs step B, or a first processor and a second processor jointly perform step A and step B).

[0065] FIG. 4 is a schematic flowchart of a data query method according to an embodiment of this specification. As shown in FIG. 4, the method includes S401 to S403.

[0066] S401: Obtain indicator data of a plurality of data sources in response to a data query request. Indicator data of one data source represents query performance of the data source in a historical period.

[0067] For example, the data source can be understood as a source from which data comes. One data source can be understood as one source from which data comes. The plurality of data

sources can be understood as a plurality of sources from which data comes. For example, the plurality of data sources can correspond to a plurality of different data providers or correspond to a plurality of different databases.

[0068] A data source has corresponding indicator data, which reflects/represents query performance of the data source in the historical period. Query performance of different data sources in the historical period may be different. Therefore, indicator data of the different data sources may also be different.

[0069] The historical period can be determined by a data query system based on a requirement, a test, etc. This is not limited in this embodiment.

[0070] With reference to FIG. 5, it can be learned that the query performance can be understood as indicator data obtained by the data query system by collecting statistics and analyzing data query activities of the data source in the historical period. The indicator data reflects data query performance of the data source in a past period of time. That is, data performance can be understood as various performance indicators and performance of a data source when a data query is performed in a historical period. The data query performance can include at least one of the following: a success rate, a recall rate, a price, an average response time, a throughput, etc.

[0071] It should be noted that indicator data of a data source represents query performance of the data source in a historical period. As time goes on, the historical period changes, and the query performance changes with the data query. Therefore, in this embodiment, the indicator data dynamically changes. In other words, the indicator data has dynamic changeability.

[0072] In some embodiments, the indicator data of one data source includes indicator values of a plurality of dimensions. S401 can include: for any data source in the plurality of data sources, obtaining historical query information of the any data source in the historical period; and collecting statistics in the plurality of dimensions based on the historical query information, to obtain the indicator values of the plurality of dimensions.

[0073] For example, the plurality of data sources include a data source 1, a data source 2, and a data source 3. For each data source, for example, the data source 1: The data query system collects historical query information of the data source 1 in the historical period, and collects statistics in the plurality of dimensions based on the historical query information, to obtain an indicator value that is of the data source 1 and that corresponds to each of the plurality of dimensions.

[0074] The historical query information can be understood as a query record of the data source 1 in the historical period (for example, a previous month or quarter), including but not limited to an invoking amount, a validity amount, a quantity of queries, a quantity of successes (also referred to as a success quantity), charging information, a response time, etc.

[0075] The plurality of dimensions can be understood as different aspects of evaluating the query performance of the data source 1, for example, a success rate, a recall rate, a price, a response time, and a throughput.

[0076] The indicator values of the plurality of dimensions can be understood as specific values obtained after statistics about the historical query information is collected in the plurality of dimensions. For example, if the dimension is a success rate, the indicator value is a specific value of the success rate.

[0077] Specifically, the invoking amount is used as an example. The data query system can obtain an invoking record of the data source 1 in the past one month, to count a quantity of data queries in the data source 1. The success quantity is used as an example. The data query system can obtain an invoking status of the data source 1 in the past one month, to count a quantity of successes without an error in a query result returned by each invoking. The validity amount is used as an example. The data query system can obtain invoking information of the data source 1 in the past one month, to count a quantity of query results that include valid information and that are returned by each invoking.

[0078] As shown in Table 1, the data query system can obtain historical query information of each of the data source 1 and the data source 2 in the historical period, and obtain indicator values respectively corresponding to a success rate, a recall rate, and a price in Table 1 through statistics collection based on the historical query information.

Table 1

Data source	Invoking amount/day	Success rate	Recall rate	Unit price (yuan)
Data source 1	120	80%	95%	2
Data source 2	10	90%	100%	3

[0079] In this embodiment, the indicator value is obtained by performing statistical analysis

on the historical query information, so that the indicator value can be dynamic, the query performance of the plurality of data sources can be evaluated more accurately, and a data source with best performance is selected more efficiently for a query.

[0080] S402: Sort the plurality of data sources based on the indicator data of the plurality of data sources, to obtain a target order.

[0081] For example, sorting can be understood as sorting a query sequence of the plurality of data sources, to obtain a target order that represents the query sequence of the plurality of data sources. That is, the target order can represent the query sequence in which a data query is performed in the plurality of data sources.

[0082] For example, the plurality of data sources include a data source 1, a data source 2, and a data source 3. Indicator data of the data source 1 is indicator data 1, indicator data of the data source 2 is indicator data 2, and indicator data of the data source 3 is indicator data 3. S402 can be understood as follows: The data query system sorts query sequences of the data source 1, the data source 2, and the data source 3 based on the indicator data 1, the indicator data 2, and the indicator data 3, to obtain the target order. The target order represents a sequence in which the data source 1, the data source 2, and the data source 3 are selected when the data query system performs a data query.

[0083] With reference to the above-mentioned analysis, it can be learned that the indicator data dynamically changes. That is, the target order is determined based on the dynamically changing indicator data. Therefore, target orders in different data queries can also be different. That is, the target order also dynamically changes.

[0084] In some embodiments, S402 can include the following steps 1 and 2:

[0085] Step 1: Determine a target sorting policy, where the target sorting policy indicates a sorting basis of the plurality of data sources.

[0086] For example, the sorting basis is a basis, a reason, a standard, a rule, etc. for sorting. Therefore, the target sorting policy can be understood as a basis, a reason, etc. for sorting the plurality of data sources. For example, a sorting basis and algorithm are defined in the target sorting policy. An objective of the target sorting policy is to determine a priority sequence of the data source during a query, to obtain required data more efficiently. For example, the target sorting policy can be one of the following: "Sort by price", "Sort by success rate", "Sort by considering both a price and a success rate", "Sort by considering a price and then considering a success rate",

"Sort by success rate when a price meets a specific condition", etc.

[0087] The target sorting policy can be preconfigured. For example, the target sorting policy can be a fixed sorting basis preconfigured by the data query system. Alternatively, the target sorting policy can be dynamically determined. For example, the data query system can dynamically determine the target sorting policy based on a query scenario, a query requirement, etc.

[0088] In some embodiments, step 1 can include: obtaining preset configuration information, and obtaining the target sorting policy from the configuration information.

[0089] For example, the configuration information can be a predefined rule or setting in the data query system, and is used to indicate an order of the plurality of data sources. For example, with reference to the above-mentioned examples, the data query system can output an interaction interface. An operator, etc. of the data query system can interact with the data query system in the interaction interface, to define a rule of sorting the plurality of data sources through the interaction. Correspondingly, the data query system can determine the configuration information through the interaction.

[0090] Different target order policies can be preconfigured for different application scenarios. That is, the configuration information can include a sorting policy applicable to a plurality of different scenarios. As shown in FIG. 5, the data query system can obtain the target sorting policy from the configuration information based on a specific requirement of a current scenario. The following provides descriptions with reference to several examples.

[0091] In some embodiments, requesters of different data queries may have different query requirements. Therefore, different sorting policies can be respectively configured for different requesters. For example, some requesters expect that costs (price) of the data query is relatively low, and a sorting policy corresponding to the requester can be "sort by price"; and some requesters expect that a query success rate is relatively high, and a sorting policy corresponding to the requester can be "sort by success rate".

[0092] In this case, the configuration information can include a sorting policy corresponding to each of different requesters. The data query system can determine a target requester corresponding to the data query request, and use a sorting policy corresponding to the target requester in the configuration information as the target sorting policy.

[0093] That is, in this embodiment, each requester has a sorting policy corresponding to the requester. The data query system selects the sorting policy corresponding to the target requester in

the configuration information as the target sorting policy based on a target requester corresponding to the data query request. Different sorting policies can be used for different requesters, to improve flexibility and diversity of sorting the data sources.

[0094] In some embodiments, different development stages of the same requester can correspond to different development policies. With reference to the above-mentioned example, the sorting policy of the target requester includes sorting policies respectively corresponding to the target requester in different development stages. Correspondingly, after determining the target requester, the data query system can further determine a current development stage of the target requester, and determine a sorting policy corresponding to the current development stage from sorting policies of the target requester as the target sorting policy.

[0095] For example, in an initial development stage of the target requester, the corresponding sorting policy can be "sort by price" in consideration of factors such as costs. With continuous development of the target requester, the initial development stage changes to a middle development stage. Correspondingly, the corresponding sorting policy can be "sort by success rate".

[0096] In some embodiments, different object types to be queried may have different query requirements. Therefore, different sorting policies can be respectively configured for different object types. For example, when the object type to be queried is logistics, the sorting policy is "sort by price"; and when the object type to be queried is a merchant, the sorting policy is "sort by success rate".

[0097] In this case, the data query request includes a to-be-queried target object, and the configuration information includes a sorting policy corresponding to each of different object types. The data query system can determine a type of the target object, and use a sorting policy corresponding to the type of the target object in the configuration information as the target sorting policy.

[0098] That is, in this embodiment, types of different query objects have respectively corresponding sorting policies, and the data query system selects a corresponding sorting policy in the configuration information based on a type of a current to-be-queried target object as the target sorting policy. Different sorting policies can be used for types of different query object, to improve flexibility and diversity of sorting the data sources.

[0099] Step 2: Sort the plurality of data sources based on the target sorting policy and the indicator data of the plurality of data sources, to obtain the target order.

[00100] For example, the data query system can sort the plurality of data sources based on the target sorting policy and the indicator data of the plurality of data sources, to obtain the target order.

[00101] With reference to the above-mentioned analysis of step 1 and step 2, it can be learned that in this embodiment, the query sequence of the plurality of data sources can be determined more properly by sorting the plurality of data sources based on the target sorting policy and the indicator data, to improve rationality and reliability of sorting.

[00102] In some embodiments, the indicator data includes the indicator values of the plurality of dimensions. For example, the plurality of dimensions can include a success rate, a recall rate, a price, an average response time, a throughput, etc. The target sorting policy indicates K target dimensions in the plurality of dimensions, and K is an integer greater than or equal to 1. The K target dimensions are some of the plurality of dimensions.

[00103] In this case, the data query system can obtain K indicator values corresponding to the K target dimensions from indicator data of each data source; and sort the plurality of data sources based on the K indicator values corresponding to each of the plurality of data sources, to obtain the target order.

[00104] For example, if the target sorting policy is "sort by price", the K target dimensions indicated by the target sorting policy include a price. In this case, the data query system can obtain an indicator value of "price" from the indicator data of each data source, and then sort the plurality of data sources based on indicator values of "price" of the plurality of data sources.

[00105] For example, if the target sorting policy is "sort by success rate", the K target dimensions indicated by the target sorting policy include a success rate. In this case, the data query system can obtain an indicator value of "success rate" from the indicator data of each data source, and then sort the plurality of data sources based on the indicator value of "success rate" of the plurality of data sources.

[00106] For example, if the target sorting policy is "sort by considering both a price and a success rate", the K target dimensions indicated by the target sorting policy include a price and a success rate. In this case, the data query system can obtain the indicator value of "price" and the indicator value of "success rate" from the indicator data of each data source. Further, the data query system can perform weighted calculation on the two indicator values of each data source to obtain a comprehensive indicator value, and sort the plurality of data sources based on a comprehensive indicator value of the plurality of data sources.

[00107] With reference to the above-mentioned analysis, it can be learned that, in this embodiment, the target sorting policy can indicate one or more target dimensions used to sort the plurality of data sources. The data query system can sort the plurality of data sources based on the indicator values of the one or more target dimensions. In this way, the plurality of data sources can be sorted based on a plurality of sorting policies. That is, diversity and flexibility of sorting the data source can be implemented.

[00108] In some embodiments, when K is greater than 1, the target sorting policy can further indicate priorities of the K target dimensions. In this case, the data query system can sort the plurality of data sources based on the K indicator values corresponding to each of the plurality of data sources and the priorities of the K target dimensions, to obtain the target order.

[00109] The priorities of the K target dimensions can be understood as priorities corresponding to all of the K target dimensions. For example, if the target sorting policy is "Sort by considering a success rate and then considering a price", the target sorting policy indicates that the K target dimensions include "success rate" and "price", and further indicates priorities respectively corresponding to "success rate" and "price". For example, "success rate" takes precedence over "price"; or "success rate" has a first priority, and "price" has a second priority. The data query system can sort the plurality of data sources by considering success rates of the plurality of data sources and then considering prices of the plurality of data sources.

[00110] With reference to the above-mentioned examples and FIG. 5, values of success rates of the data source 1, the data source 2, and the data source 3 are a success rate 1, a success rate 2, and a success rate 3; values of prices respectively corresponding to the data source 1, the data source 2, and the data source 3 are a price 1, a price 2, and a price 3. The success rate has the first priority, and the price has the second priority.

[00111] If success rate 1>success rate 2=success rate 3, the data query system first sorts the data source 1, the data source 2, and the data source 3 based on the success rates, and then sorts the data source 2 and the data source 3 based on the prices. The obtained target order can be data source 1>data source 3>data source 2. The target order indicates that the data query system preferentially performs a data query by using the data source 1, then can perform a data query by using the data source 3, and finally performs a data query by using the data source 2.

[00112] With reference to the above-mentioned analysis, it can be learned that, in some embodiments, the data query system can classify data sources in the target order into a primary

data source and a secondary data source. The 1st data source in the target order can be referred to as a primary data source, and another data source can be referred to as a secondary data source. When there are a plurality of secondary data sources, the data query system can determine a query sequence relationship between two adjacent secondary data sources based on a sorting relationship between the two adjacent secondary data sources in the target order.

[00113] With reference to the above-mentioned examples and FIG. 6, if the 1st data source in the target order is the data source 1, the 2nd data source is the data source 3, and the 3rd data source is the data source 2, The primary data source is the data source 1. The secondary data source includes the data source 3 and the data source 2. The data source 1 is the first queried data source. A query sequence relationship between the data source 2 and the data source 3 is as follows: The data source 3 is queried before the data source 2 (if the data source 2 needs to be queried).

[00114] It should be noted that when the target sorting policy indicates the priorities of the K target dimensions, a plurality of expression forms can be used. This is not limited in this specification. The above-mentioned listed example of "Sort by considering a success rate and then considering a price" is only a possible example. In actual applications, another expression form can be used. For example, the target sorting policy is "Sort by success rate when a price meets a specific condition". The target sorting policy indicates two target dimensions: "price" and "success rate", and also indicates that a priority of "price" is higher than that of "success rate".

[00115] In this embodiment, when there are a plurality of target dimensions, the plurality of data sources are sorted with reference to the priorities of K target dimensions, so that priorities of different target dimensions can be considered, and sorting has both flexibility and diversity, and is targeted and personalized.

[00116] S403: Perform a data query in at least some of the plurality of data sources based on the target order, to obtain a data query result.

[00117] With reference to the above-mentioned examples, FIG. 5, and FIG. 6, the target order represents a sequence in which the data source 1, the data source 2, and the data source 3 are selected when the data query system performs a data query. The data query system can perform a data query in at least some of the data source 1, the data source 2, and the data source 3 based on the sequence. For example, if the target order represents that the data source 1 is ranked the first, the data query system can perform a data query in at least the data source 1.

[00118] With reference to the above-mentioned analysis of S401 to S403, it can be learned that

the data query system can obtain the dynamically changing indicator data of the plurality of data sources, to determine the target order by sorting the plurality of data sources based on the dynamically changing indicator data, and further perform a data query in at least some of the plurality of data sources based on the target order. According to the above-mentioned solutions, there is relatively high flexibility and diversity in terms of a query sequence of the plurality of data sources, to help meet query requirements of different scenarios, and improve data query experience of the user. In addition, the data sources are sorted based on historical performance of the data sources, and a data source with better performance is preferentially queried, to shorten a data query time, and help efficiently find required data in a plurality of scenarios.

[00119] In some embodiments, S403 can include step 1 to step 3:

[00120] Step 1: Perform a data query in the 1st data source based on the target order, to obtain a sub-query result of the 1st data source.

[00121] For example, step 1 can be understood as follows: The data query system selects, from the plurality of data sources for a data query, a data source that is ranked in front in the target order, namely, a data source with the highest query priority.

[00122] With reference to the above-mentioned example and FIG. 6, it can be learned that the data source 1 is a primary data source and has the highest query priority. The data query system preferentially performs a data query in the data source 1, to obtain a sub-query result 1 of the data source 1.

[00123] Step 2: Determine the sub-query result of the 1st data source as the data query result if the sub-query result of the 1st data source meets a preset condition.

[00124] The preset condition can be understood as a limitation on the sub-query result. For example, integrity of the sub-query result is limited, or a confidence of the sub-query result is limited. The preset condition can be determined by the data query system based on a requirement, a historical record, a test, etc. This is not limited in this embodiment.

[00125] For example, the preset condition is a limitation on the integrity of the sub-query result. With reference to the above-mentioned example, when obtaining the sub-query result 1, the data query system can determine whether the sub-query result 1 has integrity. Step 2 can be understood as follows: If a determining result is yes (that is, the sub-query result 1 has integrity), the data query system determines the sub-query result 1 as the data query result.

[00126] With reference to the above-mentioned example and FIG. 6, if the sub-query result 1

obtained based on the primary data source meets the preset condition, for the query result, the sub-query result 1 is determined and output as the data query result, and the data query ends.

[00127] Step 3: If the sub-query result of the 1st data source does not meet a preset condition, sequentially perform a data query in other data sources based on the target order, until sub-query results of the first i data sources meet the preset condition, or until a query in the plurality of data sources is completed, and generate the data query result based on the sub-query results of the first i data sources.

[00128] It can be understood that when the data query system performs a data query based on the data source, there may be insufficient coverage due to invoking of a data source or the data query system may have a return failure or return a query result of invalid information.

[00129] Correspondingly, with reference to the above-mentioned example and FIG. 6, a determining result of the data query system for the sub-query result 1 may alternatively be no. If no, the data query system performs a data query in the data source 3, to obtain a sub-query result 3. The data query system determines whether the sub-query result 1 and the sub-query result 3 has integrity.

[00130] If yes, the data query system generates the data query result based on the sub-query result 1 and the sub-query result 3. If no, the data query system continues to perform a data query in the data source 2, to obtain the sub-query result 2. Similar operations are performed, until a data query result satisfying integrity is obtained or queries in all data sources are completed.

[00131] That is, in this embodiment, if one or more sub-query results from one or more data sources cannot meet the preset condition, the data query system can fuse sub-query results of more data sources to determine a final data query result, instead of replacing the data query by using different data sources. In this way, a fused query between different data sources is implemented, good information of different data sources is combined, and it is ensured that quality of the data query result exceeds a simple combination effect, that is, a 1+1>2 effect is achieved. In other words, quality of a result obtained by the data query system by performing a query in the plurality of data sources in the target order is better than quality obtained by simply mechanically combining query results of all data sources.

[00132] With reference to the above-mentioned analysis, it can be learned that the indicator data dynamically changes with the data query. Therefore, in some embodiments, as shown in FIG. 5, after completing a current data query request, the data query system can update the historical query

information based on a current data query status, to use the updated historical query information as historical query information of the next data query.

[00133] Correspondingly, because the historical query information of the next data query changes, indicator data determined based on the changed historical query information can also change in comparison with current indicator data. Therefore, the plurality of data sources are dynamically sorted based on the dynamic indicator data of the plurality of data sources, to perform a data query based on the sorted data sources. That is, the plurality of data sources can be dynamically selected for a data query.

[00134] For example, as shown in FIG. 7, in the i^{th} data query, the target order is an order shown in FIG. 6, that is, data source 1>data source 3>data source 2. It is assumed that the target order is determined based on a success rate. For example, before the i^{th} data query, success rates of the data source 1, the data source 2, and the data source 3 are respectively 90%, 80%, and 85%. After the i^{th} data query, indicator data of the three data sources is updated based on a data query status of the i^{th} data query. After the indicator data is updated, the success rates of the data source 1, the data source 2, and the data source 3 are respectively 85%, 80%, and 90%. In this case, when the data query system performs the $(i+1)^{th}$ data query, a target order of the data source 1, the data source 2, and the data source 3 is data source 3>data source 1>data source 2.

[00135] In some embodiments, a query request of the target object includes a query parameter. With reference to the above-mentioned analysis, it can be learned that a logistics parameter can be the target object in the above-mentioned example or at least partial information in the target object. For example, in a logistics scenario, the query parameter can be a logistics tracking number.

[00136] In some embodiments, the data query system can first perform an input preparation operation on the query parameter when obtaining the query parameter to perform a data query in a data source based on the query parameter. An input preparation is used to convert or encapsulate the query parameter into a format corresponding to the data source.

[00137] For example, the data query system needs to encapsulate the query parameter into a format supported by the data source 1, and then invoke the data source 1 to input the query parameter to the data source 1 for a data query.

[00138] Correspondingly, output results of different data sources may have different formats. Therefore, after obtaining the query result based on the data source, the data query system can perform a standard operation on the query result, so that different query results are output in the

same standard format.

[00139] In this embodiment, the input preparation and the standard operation are performed, to avoid a difference when the data query system invokes different data sources. This implements validity and reliability of invoking different data sources.

[00140] In some embodiments, the data query method in this specification further includes: generating and outputting a visual interface, where at least one of the following information is displayed in the visual interface: the indicator data of the plurality of data sources, the target order corresponding to the plurality of data sources, and the data query result.

[00141] For example, the data query system can generate and output a visual interface shown in FIG. 8, so that an order, a data query result, etc. is interpretable. With reference to the above-mentioned examples and FIG. 8, three data sources, three pieces of indicator data, and the target order are shown as examples in the visual interface.

[00142] In this embodiment, the visual interface is generated and output, to show the indicator data, the target order, and the data query in the at least some data sources in the plurality of data sources. In this way, an entire data query procedure is traceable, verifiable, and detectable. In addition, visual management of the plurality of data sources can also be implemented.

[00143] It is worthwhile to note that, the above-mentioned examples are merely used to describe examples of a possible implementation of the data query method in this specification, and cannot be understood as a limitation on the implementation of the data query method in this specification. For example, based on the above-mentioned technical concept, some of the above-mentioned technical features can be combined to obtain a new embodiment, a new technical feature can be added based on the above-mentioned example to obtain a new embodiment, some technical features can be reduced based on the above-mentioned example to obtain a new embodiment, some technical features in the above-mentioned example may be replaced with other technical features, or some technical features and orders in the above-mentioned example can be adjusted to obtain a new embodiment. Examples are not listed one by one here.

[00144] According to the above-mentioned technical concept, this specification further provides a computer-readable non-transitory storage medium. The computer-readable non-transitory storage medium stores at least one instruction set, and when the at least one instruction set is executed by a processor, steps of the data query method described in this specification are implemented.

[00145] In some possible implementations, aspects of this specification can further be implemented in a form of a program product, including program code. When the program product runs on the data query system 300, the program code is used to enable the data query system 300 to perform the steps of the data query method described in this specification. The program product used to implement the above-mentioned method can be a portable compact disk read-only memory (CD-ROM) including program code, and can run on the data query system 300. However, the program product in this specification is not limited thereto. In this specification, a readable storage medium can be any tangible medium that includes or stores a program, and the program can be used by or in combination with an instruction execution system. The program product can be any combination of one or more readable media. The readable medium can be a readable signal medium or a readable storage medium. The readable storage medium can be, for example, an electrical system, apparatus, or device, a magnetic system, apparatus, or device, an optical system, apparatus, or device, an electromagnetic system, apparatus, or device, an infrared system, apparatus, or device, or a semiconductor system, apparatus, or device, or any combination thereof. More specific examples of the readable storage medium include: an electrical connection including one or more conducting wires, a portable disk, a hard disk, a random access memory (RAM), a read-only memory (ROM), an erasable programmable read-only memory (EPROM or flash memory), an optical fiber, a portable compact disc read-only memory (CD-ROM), an optical storage component, a magnetic storage component, or any proper combination thereof. The computer-readable storage medium can include a data signal propagated in a baseband or as part of a carrier, and carries readable program code. The propagated data signal can be in a plurality of forms, and includes but is not limited to an electromagnetic signal, an optical signal, or any proper combination thereof. The readable storage medium can further be any readable medium other than the readable storage medium, and the readable medium can be used to send, propagate, or transmit a program used by or in combination with an instruction execution system, apparatus, or component. The program code included in the readable storage medium can be transmitted in any proper medium, including but not limited to a wireless medium, a wired medium, optical cable, an RF medium, or any proper combination thereof. Program code for performing an operation of this specification can be written in any combination of one or more program design languages. The program design language includes an object-oriented program design language, for example, Java or C++, and further includes a conventional procedural program design language, for example,

"C" language, or a similar program design language. The program code can be completely executed on the data query system 300, partially executed on the data query system 300, executed as an independent software package, partially executed on the data query system 300 and partially executed on a remote query system, or completely executed on a remote data query system 300.

[00146] It should be noted that, collection, storage, use, processing, transmission, provision, disclosure, etc. of related information (for example, business information or a logistics tracking number) of a user in the technical solutions in this specification all comply with related laws and regulations, and do not violate public order and good morals.

[00147] Specific embodiments of this specification are described above. Other embodiments fall within the scope of the appended claims. In some cases, the actions or steps described in the claims can be performed in an order different from that in the embodiments, and the desired results can still be achieved. In addition, the process depicted in the accompanying drawings does not necessarily need a particular order or consecutive order to achieve the desired results. In some implementations, multitasking and parallel processing are possible or can be advantageous.

[00148] In conclusion, after reading the detailed disclosure, a person skilled in the art can understand that the detailed disclosure can be presented only in an example manner, and can impose no limitation. Although it is not explicitly stated here, a person skilled in the art can understand that the requirements of this specification include various proper changes, improvements, and modifications to the embodiments. These changes, improvements, and modifications are intended to be presented in this specification, and fall within the spirit and scope of the example embodiments of this specification.

[00149] In addition, some terms in this specification are used to describe the embodiments of this specification. For example, "one embodiment", "an embodiment", and/or "some embodiments" mean that particular features, structures, or characteristics described with reference to the embodiments can be included in at least one embodiment of this specification. Therefore, it can be emphasized and understood that two or more of "an embodiment", "one embodiment", or "alternative embodiments" referred in parts of this specification do not necessarily indicate the same embodiment. In addition, particular features, structures, or characteristics can be properly combined in one or more embodiments of this specification.

[00150] It should be understood that in the above-mentioned descriptions of the embodiments of this specification, to help understand a feature, for the purpose of simplifying this specification,

various features are combined in a single embodiment, accompanying drawing, or description thereof in this specification. However, this does not mean that the combination of these features is necessary. It is entirely possible for a person skilled in the art to mark some devices and understand the devices as separate embodiments when reading this specification. In other words, the embodiments in this specification can also be understood as an integration of a plurality of secondary embodiments. Content of each secondary embodiment is also true when features are less than all features in one of the above-mentioned disclosed embodiments.

[00151] Each patent, patent application, publication of a patent application, and other materials, for example, articles, books, instructions, publications, documents, and products, that are cited in this specification and that are other than any document that is inconsistent or conflict with this document or that imposes limiting impact on the widest scope of the claims can be incorporated herein by reference and used for all purposes now or later associated with this document. In addition, the term in this document shall prevail when there is any inconsistency or conflict between the description, definition and/or use of a related term in any material and the description, definition and/or use of a related term in this document.

[00152] Finally, it should be understood that the principles of the implementation solutions of this specification are described in the implementation solutions of the application disclosed in this specification. Other modified embodiments also fall within the scope of this specification. Therefore, the embodiments disclosed in this specification are merely examples and not limitations. A person skilled in the art can use an alternative configuration according to the embodiments of this specification to implement the application in this specification. Therefore, the embodiments of this specification are not limited to the embodiments accurately described in the application.

CLAIMS

What is claimed is:

1. A data query method, comprising:

obtaining indicator data of a plurality of data sources in response to a data query request, wherein indicator data of one data source represents query performance of the data source in a historical period;

sorting the plurality of data sources based on the indicator data of the plurality of data sources, to obtain a target order; and

performing a data query in at least some of the plurality of data sources based on the target order, to obtain a data query result.

2. The method according to claim 1, wherein the sorting the plurality of data sources based on the indicator data of the plurality of data sources, to obtain a target order comprises:

determining a target sorting policy, wherein the target sorting policy indicates a sorting basis of the plurality of data sources; and

sorting the plurality of data sources based on the target sorting policy and the indicator data of the plurality of data sources, to obtain the target order.

3. The method according to claim 2, wherein the indicator data comprises indicator values of a plurality of dimensions, the target sorting policy indicates K target dimensions in the plurality of dimensions, and K is an integer greater than or equal to 1; and the sorting the plurality of data sources based on the target sorting policy and the indicator data of the plurality of data sources, to obtain the target order comprises:

obtaining K indicator values corresponding to the K target dimensions from indicator data of each data source; and

sorting the plurality of data sources based on the K indicator values corresponding to each of the plurality of data sources, to obtain the target order.

4. The method according to claim 3, wherein K is greater than 1, and the target sorting policy further indicates priorities of the K target dimensions; and the sorting the plurality of data sources based on the K indicator values corresponding to each of the plurality of data sources, to obtain the target order comprises:

sorting the plurality of data sources based on the K indicator values corresponding to each of

the plurality of data sources and the priorities of the K target dimensions, to obtain the target order.

5. The method according to claim 2, wherein the determining a target sorting policy comprises:

obtaining preset configuration information; and

obtaining the target sorting policy from the configuration information.

6. The method according to claim 5, wherein the configuration information comprises a sorting policy corresponding to each of different requesters; and the obtaining the target sorting policy from the configuration information comprises:

determining a target requester corresponding to the data query request; and

using a sorting policy corresponding to the target requester in the configuration information as the target sorting policy.

7. The method according to claim 5, wherein the data query request comprises a to-be-queried target object, and the configuration information comprises a sorting policy corresponding to each of different object types; and the obtaining the target sorting policy from the configuration information comprises:

determining a type of the target object; and

using a sorting policy corresponding to the type of the target object in the configuration information as the target sorting policy.

8. The method according to claim 1, wherein the indicator data of one data source comprises indicator values of a plurality of dimensions; and the obtaining indicator data of a plurality of data sources comprises:

for any data source in the plurality of data sources, obtaining historical query information of the any data source in the historical period; and

collecting statistics in the plurality of dimensions based on the historical query information, to obtain the indicator values of the plurality of dimensions.

9. The method according to claim 1, wherein the performing a data query in at least some of the plurality of data sources based on the target order, to obtain a data query result comprises:

performing a data query in the 1st data source based on the target order, to obtain a sub-query result of the 1st data source; and

determining the sub-query result of the 1st data source as the data query result if the sub-query result of the 1st data source meets a preset condition; or

if the sub-query result of the 1st data source does not meet a preset condition, sequentially

performing a data query in other data sources based on the target order, until sub-query results of the first i data sources meet the preset condition, or until a query in the plurality of data sources is completed, and generating the data query result based on the sub-query results of the first i data sources.

10. The method according to claim 1, wherein the method further comprises:

generating and outputting a visual interface, wherein at least one of the following information is displayed in the visual interface: the indicator data of the plurality of data sources, the target order corresponding to the plurality of data sources, and the data query result.

11. A data query system, comprising:

at least one storage medium, storing at least one instruction set to perform a data query; and

at least one processor, communicatively connected to the at least one storage medium, wherein when running, the at least one processor reads the at least one instruction set, and performs the following operations based on an indication of the at least one instruction set:

obtaining indicator data of a plurality of data sources in response to a data query request, wherein indicator data of one data source represents query performance of the data source in a historical period;

sorting the plurality of data sources based on the indicator data of the plurality of data sources, to obtain a target order; and

performing a data query in at least some of the plurality of data sources based on the target order, to obtain a data query result.

12. The system according to claim 11, wherein to sort the plurality of data sources based on the indicator data of the plurality of data sources, to obtain the target order, the at least one processor performs the following operations:

determining a target sorting policy, wherein the target sorting policy indicates a sorting basis of the plurality of data sources; and

sorting the plurality of data sources based on the target sorting policy and the indicator data of the plurality of data sources, to obtain the target order.

13. The system according to claim 12, wherein the indicator data comprises indicator values of a plurality of dimensions, the target sorting policy indicates K target dimensions in the plurality of dimensions, and K is an integer greater than or equal to 1; and to sort the plurality of data sources based on the target sorting policy and the indicator data of the plurality of data sources, to obtain

the target order, the at least one processor performs the following operations:

obtaining K indicator values corresponding to the K target dimensions from indicator data of each data source; and

sorting the plurality of data sources based on the K indicator values corresponding to each of the plurality of data sources, to obtain the target order.

14. The system according to claim 13, wherein K is greater than 1, and the target sorting policy further indicates priorities of the K target dimensions; and to sort the plurality of data sources based on the K indicator values corresponding to each of the plurality of data sources, to obtain the target order, the at least one processor performs the following operation:

sorting the plurality of data sources based on the K indicator values corresponding to each of the plurality of data sources and the priorities of the K target dimensions, to obtain the target order.

15. The system according to claim 12, wherein to determine the target sorting policy, the at least one processor performs the following operations:

obtaining preset configuration information; and

obtaining the target sorting policy from the configuration information.

16. The system according to claim 15, wherein the configuration information comprises a sorting policy corresponding to each of different requesters; and to obtain the target sorting policy from the configuration information, the at least one processor performs the following operations:

determining a target requester corresponding to the data query request; and

using a sorting policy corresponding to the target requester in the configuration information as the target sorting policy.

17. The system according to claim 15, wherein the data query request comprises a to-be-queried target object, and the configuration information comprises a sorting policy corresponding to each of different object types; and to obtain the target sorting policy from the configuration information, the at least one processor performs the following operations:

determining a type of the target object; and

using a sorting policy corresponding to the type of the target object in the configuration information as the target sorting policy.

18. The system according to claim 11, wherein the indicator data of one data source comprises indicator values of a plurality of dimensions; and to obtain the indicator data of the plurality of data sources, the at least one processor performs the following operations:

for any data source in the plurality of data sources, obtaining historical query information of the any data source in the historical period; and

collecting statistics in the plurality of dimensions based on the historical query information, to obtain the indicator values of the plurality of dimensions.

19. The system according to claim 11, wherein to perform the data query in the at least some of the plurality of data sources based on the target order, to obtain the data query result, the at least one processor performs the following operations:

performing a data query in the 1st data source based on the target order, to obtain a sub-query result of the 1st data source; and

determining the sub-query result of the 1st data source as the data query result if the sub-query result of the 1st data source meets a preset condition; or

if the sub-query result of the 1st data source does not meet a preset condition, sequentially performing a data query in other data sources based on the target order, until sub-query results of the first i data sources meet the preset condition, or until a query in the plurality of data sources is completed, and generating the data query result based on the sub-query results of the first i data sources.

20. A computer-readable non-transitory storage medium, wherein the computer-readable non-transitory storage medium stores at least one instruction set, and when the at least one instruction set is executed by at least one processor, the method according to any one of claims 1 to 10 is implemented.