

DATA PROCESSING METHOD AND DEVICE

TECHNICAL FIELD

[0001] This application relates to the field of Internet technologies, and in particular, to a data processing method and device.

5

BACKGROUND

[0002] With the rapid development of Internet technologies, people are increasingly inclined to enjoy various convenient services on the Internet. For service providers, how to provide users with accurate services in specific areas is an important factor in obtaining users' recognition. In order to meet the individual needs of the users and provide users with accurate services, it is necessary to collect users' personal attributes and store these attributes, so that users can be provided with accurate services according to the users' personal attributes.

10

[0003] The personal attributes include some of the user's behavioral habits, interest preferences, customization requirements, etc. For example, the personal attributes include a user's preferred payment method, the user's preferred payment account number, the category of the product that the user is interested in, the user's frequently visited business, and the user's own default login account number.

15

[0004] The user's personal attribute includes the attribute name of the personal attribute and the attribute value of the personal attribute. For example, the attribute name of a personal attribute of the user is "default payment method" and the attribute value is "online payment". For another example, the attribute name of another personal attribute of the user is "category of the product of interest", and the attribute value is "digital home appliance".

20

[0005] Currently, in the existing technology, user's personal attributes are generally stored in the method shown in Table 1 below. Each field in the leftmost column in Table 1 stores an ID () of each user, for example, user A to user M respectively; each field in the top row of Table 1 stores the attribute name of each personal attribute that is collected in advance by the technician, for example, attribute name 1 to attribute name

25

N are respectively stored. Table 1 includes a total of $(M+1)*(N+1)$ fields.

[0006] When it is necessary to store an attribute value of a personal attribute of a certain user in Table 1, Table 1 is searched for the row where the user ID of the user is located and for the column where the attribute name of the personal attribute is located.

5 Then, the field located in both the found row and the found column is determined, and the field is filled in with the attribute value of the personal attribute.

[0007] In Table 1, if the field in a row where a user ID is located and in a column where an attribute name is located is an empty field, it indicates that the user corresponding to the user ID does not have the personal attribute corresponding to the attribute name.

10

Table 1

	Attribute name 1	Attribute name 2	Attribute name 3	...	Attribute name N
User A	A1	A2			AN
User B		B2	B3		
...					
User M	M1		M3		

[0008] In a process of implementing this application, the inventors find that at least the following disadvantage exists in the solutions of related technologies:

15 **[0009]** There are various personal attributes collected by technicians in advance, and thus there will be a lot of fields included in the first row in Table 1, but sometimes the personal attributes of different users have large differences and low coincidence, and the personal attributes of each user account for only a small portion of all personal attributes collected in advance by the technicians. As a result, only a few fields in a row
20 corresponding to a user ID may be filled with attribute values, and other fields are all empty fields. As such, there will be a lot of empty fields in Table 1, and because the empty fields occupy a certain amount of storage space, a lot of empty fields in Table 1 will take a huge amount of storage space.

SUMMARY

[0010] To alleviate the problems in the related technologies, the present disclosure provides a data processing method and device.

[0011] According to the first aspect of the implementations of the present disclosure,
5 a data processing method is provided, and the method includes the following: obtaining a user identifier as well as an attribute identifier and an attribute value of at least one personal attribute corresponding to the user identifier; for each of the personal attributes, generating a target attribute correspondence of the personal attribute based on the attribute identifier and the attribute value of the personal attribute; determining whether
10 a target attribute storage field corresponding to the user identifier exists in stored first correspondences between user identifiers and attribute storage fields; and storing all the target attribute correspondences in the target attribute storage field when the target attribute storage field corresponding to the user identifier exists in the first correspondences; or when no target attribute storage field corresponding to the user
15 identifier exists in the first correspondences, creating an attribute storage field corresponding to the user identifier, and storing all the target attribute correspondences in the created attribute storage field.

[0012] The generating a target attribute correspondence of the personal attribute based on the attribute identifier and the attribute value of the personal attribute includes
20 the following: establishing a target attribute correspondence between the attribute identifier of the personal attribute and the attribute value of the personal attribute; or obtaining stored second correspondences between attribute identifiers and index identifiers of sequence indexes; searching the second correspondences for an index identifier corresponding to the attribute identifier; and establishing a target attribute
25 correspondence between the index identifier and the attribute value of the personal attribute.

[0013] The storing all the target attribute correspondences in the target attribute storage field includes the following: determining whether storage space occupied by all the target attribute correspondences is less than or equal to idle storage space of the
30 target attribute storage field; and storing all the target attribute correspondences in the target attribute storage field when the storage space occupied by all the target attribute correspondences is less than or equal to the idle storage space of the target attribute storage field; or creating at least one new attribute storage field corresponding to the

user identifier when the storage space occupied by all the target attribute correspondences is greater than the idle storage space of the target attribute storage field; and storing all the target attribute correspondences in the new attribute storage field or in the target attribute storage field and the new attribute storage field.

5 **[0014]** The storing all the target attribute correspondences in the target attribute storage field includes the following: determining a location where the target attribute correspondence is to be stored in the target attribute storage field based on the index identifier in the target attribute correspondence and index identifiers in all attribute correspondences stored in the target attribute storage field; and storing the target
10 attribute correspondence in the determined location.

[0015] According to the second aspect of the implementations of the present disclosure, a data processing method is provided, and the method includes the following: when obtaining an acquisition request including a user identifier and an attribute
15 identifier of a personal attribute, searching first correspondences between user identifiers and attribute storage fields to determine whether an attribute storage field corresponding to the user identifier exists in the first correspondences; when the attribute storage field corresponding to the user identifier exists, searching the attribute storage field to determine whether an attribute correspondence corresponding to the attribute identifier exists in the attribute storage field; and when the attribute
20 correspondence corresponding to the attribute identifier exists, obtaining an attribute value in the attribute correspondence.

[0016] The searching the attribute storage field to determine whether an attribute correspondence corresponding to the attribute identifier exists in the attribute storage field includes the following: searching the attribute storage field to determine whether
25 an attribute correspondence including the attribute identifier exists in the attribute storage field; and determining that the attribute correspondence corresponding to the attribute identifier exists in the attribute storage field when the attribute correspondence including the attribute identifier exists.

[0017] The searching the attribute storage field to determine whether an attribute
30 correspondence corresponding to the attribute identifier exists in the attribute storage field includes the following: obtaining second correspondences between attribute identifiers and index identifiers; searching the second correspondences for an index identifier corresponding to the attribute identifier; searching the attribute storage field to determine whether an attribute correspondence including the index identifier exists

in the attribute storage field; and determining that the attribute correspondence corresponding to the attribute identifier exists in the attribute storage field when the attribute correspondence including the index identifier exists.

[0018] According to the third aspect of the implementations of the present disclosure, a data processing device is provided, and the device includes the following: a first obtaining module, configured to obtain a user identifier as well as an attribute identifier and an attribute value of at least one personal attribute corresponding to the user identifier; a generating module, configured to generate, for each of the personal attributes, a target attribute correspondence of the personal attribute based on the attribute identifier and the attribute value of the personal attribute; a determining module, configured to determine whether a target attribute storage field corresponding to the user identifier exists in stored first correspondences between user identifiers and attribute storage fields; a first storing module, configured to store all the target attribute correspondences in the target attribute storage field when the target attribute storage field corresponding to the user identifier exists in the first correspondences; and a creating module, configured to create an attribute storage field corresponding to the user identifier when no target attribute storage field corresponding to the user identifier exists in the first correspondences; and a second storing module, configured to store all the target attribute correspondences in the created attribute storage field.

[0019] The generating module includes the following: a first establishment unit, configured to establish a target attribute correspondence between the attribute identifier of the personal attribute and the attribute value of the personal attribute; or a first obtaining unit, configured to obtain stored second correspondences between attribute identifiers and index identifiers of sequence indexes; a searching unit, configured to search the second correspondences for an index identifier corresponding to the attribute identifier; and a second establishment unit, configured to establish a target attribute correspondence between the index identifier and the attribute value of the personal attribute.

[0020] The first storing module includes the following: a determining unit, configured to determine whether storage space occupied by all the target attribute correspondences is less than or equal to idle storage space of the target attribute storage field; and a first storing unit, configured to store all the target attribute correspondences in the target attribute storage field when the storage space occupied by all the target attribute correspondences is less than or equal to the idle storage space of the target

attribute storage field; a creating unit, configured to create at least one new attribute storage field corresponding to the user identifier when the storage space occupied by all the target attribute correspondences is greater than the idle storage space of the target attribute storage field; and a second storing unit, configured to store all the target attribute correspondences in the new attribute storage field or in the target attribute storage field and the new attribute storage field.

[0021] The first storing module includes the following: a first determining unit, configured to determine a location where the target attribute correspondence is to be stored in the target attribute storage field based on the index identifier in the target attribute correspondence and index identifiers in all attribute correspondences stored in the target attribute storage field; and a third storing unit, configured to store the target attribute correspondence in the determined location.

[0022] According to the fourth aspect of the implementations of the present disclosure, a data processing device is provided, and the device includes the following: a first searching module, configured to search, when an acquisition request including a user identifier and an attribute identifier of a personal attribute is obtained, first correspondences between user identifiers and attribute storage fields to determine whether an attribute storage field corresponding to the user identifier exists in the first correspondences; a second searching module, configured to search, when the attribute storage field corresponding to the user identifier exists, the attribute storage field to determine whether an attribute correspondence corresponding to the attribute identifier exists in the attribute storage field; and a second obtaining module, configured to obtain an attribute value in the attribute correspondence when the attribute correspondence corresponding to the attribute identifier exists.

[0023] The second searching module includes the following: a first searching unit, configured to search the attribute storage field to determine whether an attribute correspondence including the attribute identifier exists in the attribute storage field; and a second determining unit, configured to determine that the attribute correspondence corresponding to the attribute identifier exists in the attribute storage field when the attribute correspondence including the attribute identifier exists.

[0024] The second searching module includes the following: a second obtaining unit, configured to obtain second correspondences between attribute identifiers and index identifiers; a second searching unit, configured to search the second correspondences for an index identifier corresponding to the attribute identifier; a third

searching unit, configured to search the attribute storage field to determine whether an attribute correspondence including the index identifier exists in the attribute storage field; and a third determining unit, configured to determine that the attribute correspondence corresponding to the attribute identifier exists in the attribute storage field when the attribute correspondence including the index identifier exists.

[0025] The technical solutions provided in the implementations of the present disclosure have the following beneficial effects:

[0026] In the implementations of the present disclosure, a user ID as well as an attribute identifier ID and an attribute value of at least one personal attribute corresponding to the user ID are obtained; for each of the personal attributes, a target attribute correspondence of the personal attribute is generated based on the attribute ID and the attribute value of the personal attribute; whether a target attribute storage field corresponding to the user ID exists in stored first correspondences between user IDs and attribute storage fields is determined; and all the target attribute correspondences are stored in the target attribute storage field when the target attribute storage field corresponding to the user ID exists in the first correspondences; alternatively an attribute storage field corresponding to the user ID is created when no target attribute storage field corresponding to the user ID exists in the first correspondences, and all the target attribute correspondences are stored in the created attribute storage field.

[0027] The storage structure in the existing technology is abandoned in the implementation of the present disclosure. First, there is no empty field, thereby avoiding the problem of wasting storage space for empty fields, and achieving the purpose of saving storage space. In addition, in the existing technology, only one attribute value is stored in each field. However, in the implementations of the present disclosure, a plurality of attribute correspondences are stored in one attribute storage field, and each attribute correspondence includes an attribute value. Thus, compared with the existing technology, in the implementations of the present disclosure, a plurality of attribute values are stored in one attribute storage field, and the storage space in an attribute storage field is fully utilized, thereby saving storage space.

[0028] Moreover, with the rapid development of service business, the user's personalized needs emerge in an inexhaustible variety. Therefore, it is often necessary to open up new personal attributes. In this case, a field needs to be added on the rightmost side of the first row in Table 1 in the existing technology, and the attribute name of the new personal attribute is added to the newly added field. Then a new field

needs to be added on the rightmost side of the row of each user ID in Table 1 so that when a user has the new personal attribute, the newly added field on the rightmost side of the row where the user ID is located is filled in with the attribute value of the personal attribute.

5 **[0029]** However, in this case, not all of the users corresponding to the user IDs in Table 1 have the new personal attribute, and thus will not fill the newly added fields on the rightmost side of the row where all the user IDs are located with attribute values, which leads to some empty fields and wasting storage space, more or less.

10 **[0030]** In addition, in the process of adding new fields to Table 1, it is necessary to lock Table 1, and Table 1 in the locked state cannot be used. When there are a great many user IDs stored in Table 1, that is, there is a large number of rows in Table 1, adding new fields to the rightmost side of the row where all the user IDs are located will consume plenty of time. As a result, Table 1 cannot be used for too much time, resulting in the inability to provide users with accurate services in a long period.

15 However, in the implementation of the present disclosure, even if a new personal attribute is developed, there is no need to allocate a field for the new personal attribute that has been developed, and there is no need to lock the stored personal attributes, so that the users can be provided with accurate services constantly. In addition, there is no need to allocate fields for the new personal attributes that have been developed, and
20 accordingly there is no empty field, thereby alleviating the problem of wasting storage space.

[0031] It should be understood that the foregoing general description and the following detailed description are merely examples, and impose no limitation on the present disclosure.

25 **BRIEF DESCRIPTION OF DRAWINGS**

[0032] The accompanying drawings, which are incorporated in and constitute a part of the present specification, illustrate implementations consistent with the present disclosure and, together with the description, serve to explain the principles of the present disclosure.

30 **[0033]** FIG. 1 is a flowchart illustrating a data processing method according to an implementation;

[0034] FIG. 2 is a flowchart illustrating a data processing method according to an

implementation;

[0035] FIG. 3 is a flowchart illustrating a data processing method according to an implementation;

5 [0036] FIG. 4 is a flowchart illustrating a data processing method according to an implementation;

[0037] FIG. 5 is a block diagram illustrating a data processing device according to an implementation; and

[0038] FIG. 6 is a block diagram illustrating a data processing device according to an implementation.

10 DESCRIPTION OF IMPLEMENTATIONS

[0039] The implementations are described in detail herein, and examples of the implementations 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.

15 Implementations described in the following do not represent all implementations consistent with the present disclosure. On the contrary, the implementations are only examples of devices and methods that are described in the appended claims in details and consistent with some aspects of the present disclosure.

20 [0040] FIG. 1 is a flowchart illustrating a data processing method according to an implementation. As shown in FIG. 1, the method includes the following steps.

[0041] Step S101: Obtain a user identifier (ID) as well as an attribute identifier (ID) and an attribute value of at least one personal attribute corresponding to the user ID.

25 [0042] The user ID may be a user's account number. For example, the user can register a user account in the server in advance, and the user ID can be the account number of the user account.

[0043] The attribute ID of the personal attribute can be the name of the personal attribute, for example, "default payment method" and "default payment account number", etc.

30 [0044] When the name of the personal attribute is "default payment method", the attribute value of the personal attribute can be "online payment" or "cash on delivery".

[0045] When the name of the personal attribute is "default payment account number", the attribute value of the personal attribute may be an account number, for

example, "273356214" or "178549624".

[0046] When the name of the personal attribute is "category of the product of interest", the attribute value of the personal attribute can be "digital home appliance", "food", or "shoe/bag", etc.

5 **[0047]** In the implementation of the present disclosure, when the server stores personal attributes of a certain user for the first time, the server can obtain the user ID of the user, and the attribute ID and the attribute value of each personal attribute of the user from the user account of the user.

[0048] Afterwards, after the user logs in to the server using the user's own user ID
10 on the user's own terminal device, if the user updates the attribute value corresponding to the attribute ID of one or more existing personal attributes in the user's own account, the terminal device obtains the user ID of the user, the attribute ID of the one or more updated personal attributes, and the attribute value corresponding to each attribute ID of the updated one or more personal attributes, and then sends the obtained user ID, the
15 attribute ID, and the attribute value to the server. The server receives the user ID, the attribute ID, and the attribute value sent by the terminal device.

[0049] After the user logs in to the server using the user's own user ID on the user's own terminal device, if the user adds an attribute ID of one or more new personal attributes and the attribute value corresponding to each new attribute ID to the user's
20 own account, the terminal device obtains the user ID of the user, the attribute ID of each new personal attribute, and the attribute value corresponding to the attribute ID of each new personal attribute, and then sends the obtained user ID, the attribute ID, and the attribute value to the server. The server receives the user ID, the attribute ID, and the attribute value sent by the terminal device.

25 **[0050]** Step S102. For each of the personal attributes, generate a target attribute correspondence of the personal attribute based on the attribute ID and the attribute value of the personal attribute.

[0051] In an implementation of the present disclosure, for each of the personal attributes, the target attribute correspondence between the attribute ID of the personal
30 attribute and the attribute value of the personal attribute can be established. The target attribute correspondence includes the attribute ID of the personal attribute and the attribute value of the personal attribute. The attribute ID of the personal attribute and the attribute value of the personal attribute can form a key-value pair. The operation is executed for each personal attribute.

[0052] In the key-value pair, "key" is used to represent the attribute ID of the personal attribute, "value" is used to represent the attribute value of the personal attribute, and the attribute ID and the attribute value are connected by using a first predetermined identifier. The first predetermined identifier may be "=", "-", or "+", etc., which is not limited in the present disclosure.

[0053] For example, assume that the attribute ID of the personal attribute is "default payment method" and the attribute value of the personal attribute is "online payment", the target attribute correspondence may be "default payment method=online payment".

[0054] In the foregoing implementation, the attribute ID of the personal attribute is usually the name of the personal attribute, for example, "default payment method", "default payment account number", and "category of the product of interest". However, the name of the personal attribute usually includes many characters, and thus the attribute ID of the personal attribute occupies a large amount of storage space. Since the attribute correspondence includes the attribute ID of the personal attribute, the attribute correspondence occupies a relatively large amount of storage space if the attribute ID occupies a large amount of storage space.

[0055] The present disclosure aims to store as many attribute correspondences as possible in one attribute storage field. However, the total storage space of one attribute storage field is limited. Therefore, in the case where the attribute correspondence occupies a relatively large storage space, the number of attribute correspondences that can be stored in one attribute storage field is smaller.

[0056] To store more attribute correspondences in one attribute storage field, in another implementation of the present disclosure, an index identifier (ID) can be allocated for each personal attribute in advance. Different index IDs are allocated for different personal attributes. The storage space occupied by the attribute ID of each personal attribute is smaller than the storage space occupied by the index ID allocated for the attribute ID of the personal attribute.

[0057] Then the attribute ID of each personal attribute and the index ID allocated for the personal attribute are stored in the second correspondences between attribute IDs and index IDs of sequence indexes.

[0058] There is a predetermined index order among all allocated index IDs. For example, the index IDs are numerical numbers, such as 1, 2, 3, 4 and 5, etc., and the numerical numbers are arranged in ascending order.

[0059] As such, in another implementation of the present disclosure, the generating

a target attribute correspondence of the personal attribute based on the attribute ID and the attribute value of the personal attribute may be as follows:

[0060] For any personal attribute, the stored second correspondences between attribute IDs and index IDs of sequence indexes are obtained, the second correspondences are searched for an index ID corresponding to an attribute ID of the personal attribute, and a target attribute correspondence between the index ID and the attribute value of the personal attribute is established so that the target attribute correspondence of the personal attribute can be generated based on the attribute ID and the attribute value of the personal attribute. The target attribute correspondence includes the index ID corresponding to the attribute ID of the personal attribute and the attribute value of the personal attribute. The index ID corresponding to the attribute ID of the personal attribute and the attribute value of the personal attribute can form a key-value pair. The operation is executed for each personal attribute.

[0061] In the key-value pair, "key" is used to represent the index ID corresponding to the attribute ID of the personal attribute, "value" is used to represent the attribute value of the personal attribute, and the index ID and the attribute value are connected by using a first predetermined identifier. The first predetermined identifier may be "=", "-", or "+", etc., which is not limited in the present disclosure.

[0062] For example, assume that the attribute ID of the personal attribute is "default payment method", the attribute value of the personal attribute is "online payment", and the index ID corresponding to the attribute ID of the personal attribute is 1, the target attribute correspondence may be "1=online payment".

[0063] Step S103: Determine whether a target attribute storage field corresponding to the user ID exists in stored first correspondences between user IDs and attribute storage fields.

[0064] For any user who registers an account in the server, when the server stores the attribute correspondence of the user for the first time, an attribute storage field corresponding to the user ID is established in the first correspondences between user IDs and attribute storage fields. Then the attribute correspondence of the user is stored in the attribute storage field. In the implementation of the present disclosure, if a new personal attribute of the user is added later, after a new attribute correspondence of the new personal attribute is generated based on the attribute ID and the attribute value of the new personal attribute, the new attribute correspondence continues to be stored in the attribute storage field. The same applies to every other user who registers an account

with the server.

[0065] After the target attribute correspondence is generated, it needs to be determined whether the target attribute storage field corresponding to the user ID exists in the stored first correspondences between user IDs and attribute storage fields.

5 **[0066]** If the target attribute storage field corresponding to the user ID exists, the target attribute correspondence can be directly stored in the target attribute storage field.

[0067] If the target attribute storage field corresponding to the user ID does not exist, it indicates that the attribute correspondence of the user has not been stored before, that is, no attribute storage field corresponding to the user ID has been stored. In this
10 case, the attribute storage field corresponding to the user ID needs to be created in the first correspondences, and then the target attribute correspondence can be stored in the created attribute storage field.

[0068] Step S104: Store all the target attribute correspondences in the target attribute storage field when the target attribute storage field corresponding to the user
15 ID exists in the first correspondences.

[0069] Step S105: Create an attribute storage field corresponding to the user ID when no target attribute storage field corresponding to the user ID exists in the first correspondences, and execute step S106.

[0070] Step S106: Store all the target attribute correspondences in the created
20 attribute storage field.

[0071] In the implementation shown in FIG. 1 of the present disclosure, a user ID as well as an attribute ID and an attribute value of at least one personal attribute corresponding to the user ID are obtained; for each of the personal attributes, a target attribute correspondence of the personal attribute is generated based on the attribute ID
25 and the attribute value of the personal attribute; whether a target attribute storage field corresponding to the user ID exists in stored first correspondences between user IDs and attribute storage fields is determined; and all the target attribute correspondences are stored in the target attribute storage field when the target attribute storage field corresponding to the user ID exists in the first correspondences; alternatively an
30 attribute storage field corresponding to the user ID is created when no target attribute storage field corresponding to the user ID exists in the first correspondences, and all the target attribute correspondences are stored in the created attribute storage field.

[0072] The storage structure in the existing technology is abandoned in the implementation of the present disclosure. First, there is no empty field, thereby avoiding

the problem of wasting storage space for empty fields, and achieving the purpose of saving storage space. In addition, in the existing technology, only one attribute value is stored in each field. However, in the implementations of the present disclosure, a plurality of attribute correspondences are stored in one attribute storage field, and each attribute correspondence includes an attribute value. Thus, compared with the existing technology, in the implementations of the present disclosure, a plurality of attribute values are stored in one attribute storage field, and the storage space in an attribute storage field is fully utilized, thereby saving storage space.

[0073] Moreover, with the rapid development of service business, the user's personalized needs emerge in an inexhaustible variety. Therefore, it is often necessary to open up new personal attributes. In this case, a field needs to be added on the rightmost side of the first row in Table 1 in the existing technology, and the attribute name of the new personal attribute is added to the newly added field. Then a new field needs to be added on the rightmost side of the row of each user ID in Table 1 so that when a user has the new personal attribute, the newly added field on the rightmost side of the row where the user ID is located is filled in with the attribute value of the personal attribute.

[0074] However, in this case, not all of the users corresponding to the user IDs in Table 1 have the new personal attribute, and thus will not fill the newly added fields on the rightmost side of the row where all the user IDs are located with attribute values, which leads to some empty fields and wasting storage space, more or less.

[0075] In addition, in the process of adding new fields to Table 1, it is necessary to lock Table 1, and Table 1 in the locked state cannot be used. When there are a great many user IDs stored in Table 1, that is, there is a large number of rows in Table 1, adding new fields to the rightmost side of the row where all the user IDs are located will consume plenty of time. As a result, Table 1 cannot be used for too much time, resulting in the inability to provide users with accurate services in a long period. However, in the implementation of the present disclosure, even if a new personal attribute is developed, there is no need to allocate a field for the new personal attribute that has been developed, and there is no need to lock the stored personal attributes, so that the users can be provided with accurate services constantly. In addition, there is no need to allocate fields for the new personal attributes that have been developed, and accordingly there is no empty field, thereby alleviating the problem of wasting storage space.

[0076] In another implementation of the present disclosure, referring to FIG. 2, the step S104 includes the following:

[0077] Step S201: Determine whether storage space occupied by all the target attribute correspondences is less than or equal to idle storage space of the target attribute storage field.

[0078] In the implementation of the present disclosure, each character occupies one unit of storage space, and each attribute correspondence includes a plurality of characters. Therefore, each attribute correspondence occupies a certain amount of storage space. Since the available storage space of an attribute storage field is limited, when an attribute correspondence needs to be stored in an attribute storage field, an objective condition must be satisfied: storage space occupied by the attribute correspondence is less than or equal to available storage space of the attribute storage field. Only when the condition is satisfied, the attribute correspondence can be successfully stored in the attribute storage field. Otherwise, if the storage space occupied by the attribute correspondence is greater than the available storage space of the attribute storage field, the attribute correspondence cannot be successfully stored in the attribute storage field.

[0079] Similarly, when a plurality of attribute correspondences need to be stored in an attribute storage field, it is also required to determine whether the storage space occupied by the plurality of attribute correspondences is less than or equal to the idle storage space of the attribute storage field.

[0080] Step S202: Store all the target attribute correspondences in the target attribute storage field when the storage space occupied by all the target attribute correspondences is less than or equal to the idle storage space of the target attribute storage field.

[0081] When the storage space occupied by all the target attribute correspondences is less than or equal to the idle storage space of the target attribute storage field, it indicates that the target attribute storage field can accommodate all the target attribute correspondences, and therefore all the target attribute correspondences can be directly stored in the target attribute storage field.

[0082] Step S203: Create at least one new attribute storage field corresponding to the user ID when the storage space occupied by all the target attribute correspondences is greater than the idle storage space of the target attribute storage field, and then execute step S204.

[0083] When the storage space occupied by all the target attribute correspondences is greater than the idle storage space of the target attribute storage field, it indicates that the target attribute storage field cannot accommodate all the target attribute correspondences, and therefore at least one new attribute storage field corresponding to the user ID needs to be created.

[0084] The number of new attribute storage fields to be created can be determined based on the available storage space of one attribute storage field, the storage space occupied by all the target attribute correspondences, or the available storage space of the target attribute field. A specific determining method is not limited in the present disclosure.

[0085] Step S204: Store all the target attribute correspondences in the new attribute storage field or in the target attribute storage field and the new attribute storage field.

[0086] If the target attribute storage field has been fully occupied, the target attribute correspondences are stored in the new attribute storage field. If the attribute storage field has not been fully occupied, some target attribute correspondences can be first stored in the target attribute storage field, and after the target attribute storage field is fully occupied, the remaining target attribute correspondences are stored in the new attribute storage field.

[0087] When all the target attribute correspondences are being stored, the predetermined second separator can be used to separate adjacent attribute correspondences so as to prevent different attribute correspondences from being confused with each other. The predetermined second separator can be "|", "*", or "~", etc., which is not limited in the present disclosure.

[0088] For example, assume that an attribute correspondence is "default payment method=online payment", another attribute correspondence is "category of the product of interest=digital home appliance", and still another attribute correspondence is "default payment account number=123456789". When the three attribute correspondences are stored in the attribute storage field, the predetermined second separator "|" can be used to separate the three attribute correspondences: "default payment method=online payment | category of the product of interest=digital home appliance | default payment account number=123456789".

[0089] In the previous implementation, the attribute correspondence includes the index ID corresponding to the attribute ID of the personal attribute and the attribute value of the personal attribute, and there is a predetermined index order among the

index IDs of all the personal attributes. Therefore, in the implementation of the present disclosure, the target attribute correspondences can be stored based on the index IDs included in the target attribute correspondences and the predetermined index order among the index IDs in the stored attribute correspondences, so as to improve the efficiency of querying an attribute value of a personal attribute subsequently.

[0090] Thus, in another implementation of the present disclosure, referring to FIG. 3, the step S104 includes the following:

[0091] Step S301: Determine a location where the target attribute correspondence is to be stored in the target attribute storage field, based on the index ID in the target attribute correspondence and index IDs in all attribute correspondences stored in the target attribute storage field.

[0092] The locations of the attribute correspondences in the attribute storage field in the implementation of the present disclosure are determined after being sorted based on the predetermined index order among the index IDs included in the attribute correspondences.

[0093] For example, assume that the index ID is a serial number, for example, 1, 2, 3, 4, and 5, etc. For any attribute storage field, in two attribute storage fields adjacent to the attribute storage field, the serial number of one attribute storage field is greater than that of the attribute storage field, and the serial number of the other attribute storage field is less than that of the attribute storage field. The same applies to every other attribute storage field.

[0094] In the present step, the target attribute correspondence can be sorted in all the stored attribute correspondences based on the index ID in the target attribute correspondence and the index IDs in the stored attribute correspondences, and two attribute correspondences in the target attribute storage field between which the target attribute correspondence is located are determined. In the two attribute correspondences, the serial number of one attribute correspondence is greater than the serial number of the target attribute correspondence, and the serial number of the other attribute correspondence is less than the serial number of the target attribute correspondence.

The location between the two attribute correspondences can be determined as the location where the target attribute correspondence is to be stored.

[0095] Step S302: Store the target attribute correspondence in the determined location.

[0096] When it is necessary to provide the user with precise services, the server

needs to query the attribute value of one or more personal attributes of the user. Referring to FIG. 4, the query can be performed according to the following method. The method includes the following steps:

[0097] Step S401: When an acquisition request including a user ID and an attribute ID of a personal attribute is obtained, search first correspondences between user IDs and attribute storage fields to determine whether an attribute storage field corresponding to the user ID exists in the first correspondences.

[0098] In the implementation of the present disclosure, when it is necessary to provide a precise service to the user, the attribute value of one or more personal attributes of the user needs to be obtained. First, an acquisition request is generated based on the user ID of the user and the attribute ID of the personal attribute that needs to be obtained. Afterwards, the first correspondences between user IDs and attribute storage fields can be obtained, and the first correspondences are searched to determine whether the attribute storage field corresponding to the user ID exists in the first correspondences.

[0099] Step S402: When the attribute storage field corresponding to the user ID exists, search the attribute storage field to determine whether an attribute correspondence corresponding to the attribute ID exists in the attribute storage field.

[0100] In an implementation of the present disclosure, if the attribute correspondence is a correspondence between the attribute ID and the attribute value, the attribute storage field can be searched in the present step to determine whether an attribute correspondence including the attribute ID exists in the attribute storage field; and it can be determined that the attribute correspondence corresponding to the attribute ID exists in the attribute storage field when the attribute correspondence including the attribute ID exists.

[0101] Specifically, the first attribute correspondence is obtained in the attribute storage field, an attribute ID in the first attribute correspondence is obtained, and whether the attribute ID in the first attribute correspondence is the same as the attribute ID is determined. When the two are the same, the first attribute correspondence is determined as the attribute correspondence corresponding to the attribute ID. Alternatively, when the two are different, the second attribute correspondence is obtained, an attribute ID in the second attribute correspondence is obtained, and whether the attribute ID in the second attribute correspondence is the same as the attribute ID is determined. When the two are the same, the second attribute

correspondence is determined as the attribute correspondence corresponding to the attribute ID. Alternatively, when the two are different, a next attribute correspondence is obtained, and the foregoing procedure is executed until an attribute ID in an attribute correspondence is the same as the attribute ID, and the obtained attribute

5 correspondence is determined as the attribute correspondence corresponding to the attribute ID. If the attribute IDs in all the attribute correspondences in the attribute storage field are different from the attribute ID, it can be determined that there is no attribute correspondence corresponding to the attribute ID in the attribute storage field.

[0102] When there is no attribute storage field corresponding to the attribute ID,

10 the procedure ends.

[0103] Generally, the attribute storage field stores a plurality of attribute correspondences. If the attribute correspondence corresponding to the attribute ID is stored in a later position in the attribute storage field, most of the attribute correspondences will be traversed according to the foregoing method, which takes a

15 long time and has low search efficiency.

[0104] In another implementation of the present disclosure, if the attribute correspondence is the correspondence between the index ID corresponding to the attribute ID and the attribute value, second correspondences between attribute IDs and index IDs can be obtained in this step. The second correspondences are searched for the

20 index ID corresponding to the attribute ID; the attribute storage field is searched to determine whether the attribute correspondence including the index ID exists in the attribute storage field; and when the attribute correspondence including the index ID exists, it can be determined that the attribute correspondence corresponding to the attribute ID exists in the attribute storage field.

[0105] The locations of the attribute correspondences in the attribute storage field in the implementation of the present disclosure are determined after being sorted based on the predetermined index order among the index IDs included in the attribute correspondences. Therefore, to improve the search efficiency, when the attribute storage field is searched to determine whether the attribute correspondence including

25 the index ID exists in the attribute storage field, first the index ID can be compared with index IDs in attribute correspondences at one or more predetermined locations in the attribute storage field; the approximate location of the attribute correspondence including the index ID in the attribute storage field is determined based on the comparison result; and then the index ID is compared with the index ID in one or more

30

attribute correspondence at the approximate location so as to determine whether the attribute correspondence including the index ID exists.

[0106] Step S403: Obtain an attribute value in the attribute correspondence when the attribute correspondence corresponding to the attribute ID exists.

5 **[0107]** When the attribute correspondence corresponding to the attribute ID does not exist, it indicates that the user corresponding to the user ID does not have the personal attribute. Moreover, it can be prompted to the user that the user corresponding to the user ID does not have the personal attribute.

[0108] FIG. 5 is a block diagram illustrating a data processing device according to an implementation. Referring to FIG. 5, the device includes the following: a first obtaining module 11, configured to obtain a user ID as well as an attribute ID and an attribute value of at least one personal attribute corresponding to the user ID; a generating module 12, configured to generate, for each of the personal attributes, a target attribute correspondence of the personal attribute based on the attribute ID and the attribute value of the personal attribute; a determining module 13, configured to determine whether a target attribute storage field corresponding to the user ID exists in stored first correspondences between user IDs and attribute storage fields; a first storing module 14, configured to store all the target attribute correspondences in the target attribute storage field when the target attribute storage field corresponding to the user ID exists in the first correspondences; and a creating module 15, configured to create an attribute storage field corresponding to the user ID when no target attribute storage field corresponding to the user ID exists in the first correspondences; and a second storing module 16, configured to store all the target attribute correspondences in the created attribute storage field.

25 **[0109]** The generating module 12 includes the following: a first establishment unit, configured to establish a target attribute correspondence between the attribute ID of the personal attribute and the attribute value of the personal attribute; or a first obtaining unit, configured to obtain second correspondences between stored attribute IDs and index IDs of sequence indexes; a searching unit, configured to search the second correspondences for an index ID corresponding to the attribute ID; and a second establishment unit, configured to establish a target attribute correspondence between the index ID and the attribute value of the personal attribute.

30 **[0110]** The first storing module 14 includes the following: a determining unit, configured to determine whether storage space occupied by all the target attribute

correspondences is less than or equal to idle storage space of the target attribute storage field; and a first storing unit, configured to store all the target attribute correspondences in the target attribute storage field when the storage space occupied by all the target attribute correspondences is less than or equal to the idle storage space of the target attribute storage field; a creating unit, configured to create at least one new attribute storage field corresponding to the user ID when the storage space occupied by all the target attribute correspondences is greater than the idle storage space of the target attribute storage field; and a second storing unit, configured to store all the target attribute correspondences in the new attribute storage field or in the target attribute storage field and the new attribute storage field.

[0111] The first storing module 14 includes the following: a first determining unit, configured to determine a location where the target attribute correspondence is to be stored in the target attribute storage field based on the index ID in the target attribute correspondence and index IDs in all attribute correspondences stored in the target attribute storage field; and a third storing unit, configured to store the target attribute correspondence in the determined location.

[0112] In the implementation shown in FIG. 5 of the present disclosure, a user ID as well as an attribute ID and an attribute value of at least one personal attribute corresponding to the user ID are obtained. For each of the personal attributes, a target attribute correspondence of the personal attribute is generated based on the attribute ID and the attribute value of the personal attribute. Whether a target attribute storage field corresponding to the user ID exists in stored first correspondences between user IDs and attribute storage fields is determined; and all the target attribute correspondences are stored in the target attribute storage field when the target attribute storage field corresponding to the user ID exists in the first correspondences. Alternatively an attribute storage field corresponding to the user ID is created when no target attribute storage field corresponding to the user ID exists in the first correspondences, and all the target attribute correspondences are stored in the created attribute storage field.

[0113] The storage structure in the existing technology is abandoned in the implementation of the present disclosure. First, there is no empty field, thereby avoiding the problem of wasting storage space for empty fields, and achieving the purpose of saving storage space. In addition, in the existing technology, only one attribute value is stored in each field. However, in the implementations of the present disclosure, a plurality of attribute correspondences are stored in one attribute storage field, and each

attribute correspondence includes an attribute value. Thus, compared with the existing technology, in the implementations of the present disclosure, a plurality of attribute values are stored in one attribute storage field, and the storage space in an attribute storage field is fully utilized, thereby saving storage space.

5 [0114] Moreover, with the rapid development of service business, the user's personalized needs emerge in an inexhaustible variety. Therefore, it is often necessary to open up new personal attributes. In this case, a field needs to be added on the rightmost side of the first row in Table 1 in the existing technology, and the attribute name of the new personal attribute is added to the newly added field. Then a new field
10 needs to be added on the rightmost side of the row of each user ID in Table 1 so that when a user has the new personal attribute, the newly added field on the rightmost side of the row where the user ID is located is filled in with the attribute value of the personal attribute.

[0115] However, not all of the users corresponding to the user IDs in Table 1 have
15 the new personal attribute, and thus will not fill the newly added fields on the rightmost side of the row where all the user IDs are located with attribute values, which leads to some empty fields and wasting storage space, more or less.

[0116] In addition, in the process of adding new fields to Table 1, it is necessary to lock Table 1, and Table 1 in the locked state cannot be used. When there are a great
20 many user IDs stored in Table 1, that is, there is a large number of rows in Table 1, adding new fields to the rightmost side of the row where all the user IDs are located will consume plenty of time. As a result, Table 1 cannot be used for too much time, resulting in the inability to provide users with accurate services in a long period. However, in the implementation of the present disclosure, even if a new personal
25 attribute is developed, there is no need to allocate a field for the new personal attribute that has been developed, and there is no need to lock the stored personal attributes, so that the users can be provided with accurate services constantly. In addition, there is no need to allocate fields for the new personal attributes that have been developed, and accordingly there is no empty field, thereby alleviating the problem of wasting storage
30 space.

[0117] FIG. 6 is a block diagram illustrating a data processing device according to an implementation. Referring to FIG. 6, the device includes the following: a first searching module 21, configured to search, when an acquisition request including a user ID and an attribute ID of a personal attribute is obtained, first correspondences between

user IDs and attribute storage fields to determine whether an attribute storage field corresponding to the user ID exists in the first correspondences; a second searching module 22, configured to search, when the attribute storage field corresponding to the user ID exists, the attribute storage field to determine whether an attribute correspondence corresponding to the attribute ID exists in the attribute storage field; and a second obtaining module 23, configured to obtain an attribute value in the attribute correspondence when the attribute correspondence corresponding to the attribute ID exists.

[0118] The second searching module 22 includes the following: a first searching unit, configured to search the attribute storage field to determine whether an attribute correspondence including the attribute ID exists in the attribute storage field; and a second determining unit, configured to determine that the attribute correspondence corresponding to the attribute ID exists in the attribute storage field when the attribute correspondence including the attribute ID exists.

[0119] The second searching module 22 includes the following: a second obtaining unit, configured to obtain second correspondences between attribute IDs and index IDs; a second searching unit, configured to search the second correspondences for an index ID corresponding to the attribute ID; a third searching unit, configured to search the attribute storage field to determine whether an attribute correspondence including the index ID exists in the attribute storage field; and a third determining unit, configured to determine that the attribute correspondence corresponding to the attribute ID exists in the attribute storage field when the attribute correspondence including the index ID exists.

[0120] In the implementation shown in FIG. 6 of the present disclosure, a user ID as well as an attribute ID and an attribute value of at least one personal attribute corresponding to the user ID are obtained. For each of the personal attributes, a target attribute correspondence of the personal attribute is generated based on the attribute ID and the attribute value of the personal attribute. Whether a target attribute storage field corresponding to the user ID exists in stored first correspondences between user IDs and attribute storage fields is determined; and all the target attribute correspondences are stored in the target attribute storage field when the target attribute storage field corresponding to the user ID exists in the first correspondences. Alternatively an attribute storage field corresponding to the user ID is created when no target attribute storage field corresponding to the user ID exists in the first correspondences, and all

the target attribute correspondences are stored in the created attribute storage field.

[0121] The storage structure in the existing technology is abandoned in the implementation shown in FIG. 1 of the present disclosure. First, there is no empty field, thereby avoiding the problem of wasting storage space for empty fields, and achieving the purpose of saving storage space. In addition, in the existing technology, only one attribute value is stored in each field. However, in the implementation shown in FIG. 6 of the present disclosure, a plurality of attribute correspondences are stored in one attribute storage field, and each attribute correspondence includes an attribute value. Thus, compared with the existing technology, in the implementation shown in FIG. 6 of the present disclosure, a plurality of attribute values are stored in one attribute storage field, and the storage space in an attribute storage field is fully utilized, thereby saving storage space.

[0122] Moreover, with the rapid development of service business, the user's personalized needs emerge in an inexhaustible variety. Therefore, it is often necessary to open up new personal attributes. In this case, a field needs to be added on the rightmost side of the first row in Table 1 in the existing technology, and the attribute name of the new personal attribute is added to the newly added field. Then a new field needs to be added on the rightmost side of the row of each user ID in Table 1 so that when a user has the new personal attribute, the newly added field on the rightmost side of the row where the user ID is located is filled in with the attribute value of the personal attribute.

[0123] However, in this case, not all of the users corresponding to the user IDs in Table 1 have the new personal attribute, and thus will not fill the newly added fields on the rightmost side of the row where all the user IDs are located with attribute values, which leads to some empty fields and wasting storage space, more or less.

[0124] In addition, in the process of adding new fields to Table 1, it is necessary to lock Table 1, and Table 1 in the locked state cannot be used. When there are a great many user IDs stored in Table 1, that is, there is a large number of rows in Table 1, adding new fields to the rightmost side of the row where all the user IDs are located will consume plenty of time. As a result, Table 1 cannot be used for too much time, resulting in the inability to provide users with accurate services in a long period. However, in the implementation shown in FIG. 6 of the present disclosure, even if a new personal attribute is developed, there is no need to allocate a field for the new personal attribute that has been developed, and there is no need to lock the stored

personal attributes, so that the users can be provided with accurate services constantly. In addition, there is no need to allocate fields for the new personal attributes that have been developed, and accordingly there is no empty field, thereby alleviating the problem of wasting storage space.

5 **[0125]** With regard to the device in the above implementations, the specific methods for performing operations by the modules have been described in detail in the implementations relating to the method, and details are omitted here for simplicity

10 **[0126]** A person of ordinary skill in the art can easily figure out another implementation of the present disclosure after thinking over the specification and practicing the present disclosure here. The present application is intended to cover any variations, uses, or adaptations of the present disclosure, and these variations, uses, or adaptations follow the general principles of the present disclosure and include common knowledge or conventional techniques that are not disclosed in the technical field of the present disclosure. The specification and implementations are considered as
15 examples only, with the true scope and spirit of the present disclosure being indicated by the appended claims.

20 **[0127]** It should be understood that the present disclosure is not limited to the precise structures that have been described above and shown in the drawings, and various modifications and changes can be made without departing from the scope of the present disclosure. The scope of the present disclosure is limited by the appended claims only.

CLAIMS

1. A data processing method, wherein the method comprises:

obtaining a user identifier as well as an attribute identifier and an attribute value of at least one personal attribute corresponding to the user identifier;

5 for each of the personal attributes, generating a target attribute correspondence of the personal attribute based on the attribute identifier and the attribute value of the personal attribute;

10 determining whether a target attribute storage field corresponding to the user identifier exists in stored first correspondences between user identifiers and attribute storage fields; and

storing all the target attribute correspondences in the target attribute storage field when the target attribute storage field corresponding to the user identifier exists in the first correspondences; or

15 when no target attribute storage field corresponding to the user identifier exists in the first correspondences, creating an attribute storage field corresponding to the user identifier, and storing all the target attribute correspondences in the created attribute storage field.

2. The method according to claim 1, wherein the generating a target attribute correspondence of the personal attribute based on the attribute identifier and the attribute value of the personal attribute comprises:

20 establishing a target attribute correspondence between the attribute identifier of the personal attribute and the attribute value of the personal attribute;

or

25 obtaining stored second correspondences between attribute identifiers and index identifiers of sequence indexes; searching the second correspondences for an index identifier corresponding to the attribute identifier; and establishing a target attribute correspondence between the index identifier and the attribute value of the personal attribute.

3. The method according to claim 1 or 2, wherein the storing all the target attribute correspondences in the target attribute storage field comprises:

30 determining whether storage space occupied by all the target attribute correspondences is less than or equal to idle storage space of the target attribute storage

field; and

storing all the target attribute correspondences in the target attribute storage field when the storage space occupied by all the target attribute correspondences is less than or equal to the idle storage space of the target attribute storage field; or

5 creating at least one new attribute storage field corresponding to the user identifier when the storage space occupied by all the target attribute correspondences is greater than the idle storage space of the target attribute storage field; and storing all the target attribute correspondences in the new attribute storage field or in the target attribute storage field and the new attribute storage field.

10 4. The method according to claim 2, wherein the storing all the target attribute correspondences in the target attribute storage field comprises:

determining a location where the target attribute correspondence is to be stored in the target attribute storage field based on the index identifier in the target attribute correspondence and index identifiers in all attribute correspondences stored in the target

15 attribute storage field; and

storing the target attribute correspondence in the determined location.

5. A data processing method, wherein the method comprises:

when obtaining an acquisition request comprising a user identifier and an attribute identifier of a personal attribute, searching first correspondences between user
20 identifiers and attribute storage fields to determine whether an attribute storage field corresponding to the user identifier exists in the first correspondences;

when the attribute storage field corresponding to the user identifier exists, searching the attribute storage field to determine whether an attribute correspondence corresponding to the attribute identifier exists in the attribute storage field; and

25 when the attribute correspondence corresponding to the attribute identifier exists, obtaining an attribute value in the attribute correspondence.

6. The method according to claim 5, wherein the searching the attribute storage field to determine whether an attribute correspondence corresponding to the attribute identifier exists in the attribute storage field comprises:

30 searching the attribute storage field to determine whether an attribute correspondence comprising the attribute identifier exists in the attribute storage field; and

determining that the attribute correspondence corresponding to the attribute identifier exists in the attribute storage field when the attribute correspondence

comprising the attribute identifier exists.

7. The method according to claim 5, wherein the searching the attribute storage field to determine whether an attribute correspondence corresponding to the attribute identifier exists in the attribute storage field comprises:

5 obtaining second correspondences between attribute identifiers and index identifiers;

 searching the second correspondences for an index identifier corresponding to the attribute identifier;

 searching the attribute storage field to determine whether an attribute
10 correspondence comprising the index identifier exists in the attribute storage field; and

 determining that the attribute correspondence corresponding to the attribute identifier exists in the attribute storage field when the attribute correspondence comprising the index identifier exists.

8. A data processing device, wherein the device comprises:

15 a first obtaining module, configured to obtain a user identifier as well as an attribute identifier and an attribute value of at least one personal attribute corresponding to the user identifier;

 a generating module, configured to generate, for each of the personal attributes, a target attribute correspondence of the personal attribute based on the attribute identifier
20 and the attribute value of the personal attribute;

 a determining module, configured to determine whether a target attribute storage field corresponding to the user identifier exists in stored first correspondences between user identifiers and attribute storage fields;

 a first storing module, configured to store all the target attribute correspondences
25 in the target attribute storage field when the target attribute storage field corresponding to the user identifier exists in the first correspondences; and

 a creating module, configured to create an attribute storage field corresponding to the user identifier when no target attribute storage field corresponding to the user identifier exists in the first correspondences; and a second storing module, configured
30 to store all the target attribute correspondences in the created attribute storage field.

9. The device according to claim 8, wherein the generating module comprises:

 a first establishment unit, configured to establish a target attribute correspondence between the attribute identifier of the personal attribute and the attribute value of the personal attribute;

or

a first obtaining unit, configured to obtain stored second correspondences between attribute identifiers and index identifiers of sequence indexes; a searching unit, configured to search the second correspondences for an index identifier corresponding to the attribute identifier; and a second establishment unit, configured to establish a target attribute correspondence between the index identifier and the attribute value of the personal attribute.

10. The device according to claim 8 or 9, wherein the first storing module comprises:

10 a determining unit, configured to determine whether storage space occupied by all the target attribute correspondences is less than or equal to idle storage space of the target attribute storage field; and

a first storing unit, configured to store all the target attribute correspondences in the target attribute storage field when the storage space occupied by all the target attribute correspondences is less than or equal to the idle storage space of the target attribute storage field; or

a creating unit, configured to create at least one new attribute storage field corresponding to the user identifier when the storage space occupied by all the target attribute correspondences is greater than the idle storage space of the target attribute storage field; and a second storing unit, configured to store all the target attribute correspondences in the new attribute storage field or in the target attribute storage field and the new attribute storage field.

11. The device according to claim 9, wherein the first storing module comprises:

a first determining unit, configured to determine a location where the target attribute correspondence is to be stored in the target attribute storage field based on the index identifier in the target attribute correspondence and index identifiers in all attribute correspondences stored in the target attribute storage field; and

a third storing unit, configured to store the target attribute correspondence in the determined location.

12. A data processing device, wherein the device comprises:

a first searching module, configured to search, when an acquisition request comprising a user identifier and an attribute identifier of a personal attribute is obtained, first correspondences between user identifiers and attribute storage fields to determine whether an attribute storage field corresponding to the user identifier exists in the first

correspondences;

a second searching module, configured to search, when the attribute storage field corresponding to the user identifier exists, the attribute storage field to determine whether an attribute correspondence corresponding to the attribute identifier exists in

5 the attribute storage field; and

a second obtaining module, configured to obtain an attribute value in the attribute correspondence when the attribute correspondence corresponding to the attribute identifier exists.

13. The device according to claim 12, wherein the second searching module
10 comprises:

a first searching unit, configured to search the attribute storage field to determine whether an attribute correspondence comprising the attribute identifier exists in the attribute storage field; and

a second determining unit, configured to determine that the attribute
15 correspondence corresponding to the attribute identifier exists in the attribute storage field when the attribute correspondence comprising the attribute identifier exists.

14. The device according to claim 12, wherein the second searching module comprises:

a second obtaining unit, configured to obtain second correspondences between
20 attribute identifiers and index identifiers;

a second searching unit, configured to search the second correspondences for an index identifier corresponding to the attribute identifier;

a third searching unit, configured to search the attribute storage field to determine whether an attribute correspondence comprising the index identifier exists in the
25 attribute storage field; and

a third determining unit, configured to determine that the attribute correspondence corresponding to the attribute identifier exists in the attribute storage field when the attribute correspondence comprising the index identifier exists.