

MODULE PROCESSING METHOD AND APPARATUS FOR FOREGROUND AND BACKGROUND SEPARATION SYSTEM, AND STORAGE MEDIUM

TECHNICAL FIELD

[001] The present application relates to the field of computer technologies, and in particular, to module processing methods and apparatuses for a foreground and background separation system and storage media.

BACKGROUND

[002] A foreground and background separation system refers to a system whose foreground and background are independent with each other. The foreground may refer to an interface layer, and the background may refer to a function layer.

[003] The foreground is used to provide an interactive interface on which a flow chart including a plurality of modules is displayed, and a user may trigger an operation on a certain module through the interactive interface.

[004] The background is used to perform a data logic function corresponding to this operation and provide a data storage function.

[005] After triggering an operation on a certain module on the foreground, the user usually encounters a scenario where it is expected to undo the operation, or is expected to perform the operation again after undoing it.

[006] In the above scenario, how to undo or redo the operation may involve a synchronization between foreground data and background data. Therefore, it is urgent to study a module processing method based on the foreground and background separation system, so that the user can achieve undoing or redoing an operation on a module in the foreground and background separation system.

SUMMARY

[007] The examples of the present application provide module processing methods and apparatuses for a foreground and background separation system and storage media, which can achieve undoing or redoing a user operation in the foreground and background separation system.

[008] In a first aspect, a module processing method for a foreground and background separation system is provided, wherein the foreground and background separation system includes a foreground and a background, the foreground displays an interactive interface that is

configured to display a plurality of modules, and the background is configured with a data center that is configured to store module data of each of the plurality of modules, wherein the module data includes a display manner of the module, function data of the module, and a logical relationship between the module and each of other modules;

[009] and the method includes:

[010] in response to a target operation on a target module, controlling, by the foreground and based on the target operation, the background to update data corresponding to the target module in the data center, and updating, by the foreground, the interactive interface in accordance with the updated data corresponding to the target module, wherein there are one or more target modules among the plurality of modules; and

[011] in a case where an undo instruction for the target operation is detected, controlling, by the foreground and based on the undo instruction, the background to restore the data corresponding to the target module in the data center, and updating, by the foreground, the interactive interface in accordance with the restored data corresponding to the target module.

[012] In a second aspect, a module processing apparatus for a foreground and background separation system is provided,

[013] wherein the foreground and background separation system includes a foreground and a background, the foreground displays an interactive interface that is configured to display a plurality of modules, and the background is configured with a data center that is configured to store module data of each of the plurality of modules, wherein the module data includes a display manner of the module, function data of the module, and a logical relationship between the module and each of other modules;

[014] and the apparatus includes:

[015] a first controlling module, configured to in response to a target operation on a target module, enable the foreground to control based on the target operation the background to update data corresponding to the target module in the data center, and enable the foreground to update the interactive interface in accordance with the updated data corresponding to the target module, wherein there are one or more target modules among the plurality of modules; and

[016] a second controlling module, configured to in a case where an undo instruction for the target operation is detected, enable the foreground to control based on the undo instruction the background to restore the data corresponding to the target module in the data center, and enable the foreground to update the interactive interface in accordance with the restored data corresponding to the target module.

[017] In a third aspect, a module processing apparatus for a foreground and background separation system is provided, and the apparatus includes:

[018] one or more processors; and

[019] one or more memories for storing instructions executable by the one or more processors;

[020] wherein the one or more processors are configured to perform any step of the module processing method for a foreground and background separation system provided in the above first aspect.

[021] In a fourth aspect, a computer-readable storage medium storing instructions is provided, wherein the instructions, when executed by one or more processors, implement any step of the module processing method for a foreground and background separation system provided in the above first aspect.

[022] In a fifth aspect, a computer program product including instructions is provided, wherein the instructions, when running on a computer, cause the computer to execute any step of the module processing method for a foreground and background separation system provided in the above first aspect.

[023] The beneficial effects of the technical solutions provided by the examples of the present application include at least the following.

[024] In the foreground and background separation system provided in the examples of the present application, the module data stored in the data center of the background includes the display manners of the modules, the function data of the modules and the logical relationships between the modules, that is, the background stores both the foreground data and the background data. As can be seen that the examples of the present application take the background as the data center for all the data. Therefore, the foreground can control the background to update both the foreground data and the background data when it detects a target operation on a target module, and also control the background to update both the foreground data and the background data when the target operation is undone, so as to ensure a consistency among the content displayed on the interactive interface of the foreground, the foreground data and the background data during performing or undoing the target operation.

BRIEF DESCRIPTION OF THE DRAWINGS

[025] The drawings related to the description of the examples will be briefly introduced to explain the technical solutions provided by examples of the present application more clearly. It is obvious that, the drawings in the following description illustrate only some examples of the

present application, and based on these drawings, those of ordinary skill in the art may obtain other drawings without creative work.

[026] FIG. 1 is a schematic architecture diagram of a foreground and background separation system provided in an example of the present application.

[027] FIG. 2 is a flowchart of a module processing method for a foreground and background separation system provided in an example of the present application.

[028] FIG. 3 is a schematic flowchart provided in an example of the present application.

[029] FIG. 4 is another schematic flowchart provided in an example of the present application.

[030] FIG. 5 is a schematic diagram of a deleting module provided in an example of the present application.

[031] FIG. 6 is a flowchart of another module processing method for a foreground and background separation system provided in an example of the present application.

[032] FIG. 7 is a schematic structure diagram of a module processing apparatus for a foreground and background separation system provided in an example of the present application.

[033] FIG. 8 is a schematic structure diagram of a terminal provided by an example of the present application.

[034] FIG. 9 is a schematic structure diagram of a server provided in an example of the present application.

DETAILED DESCRIPTION OF THE EMBODIMENTS

[035] In order to make the purpose, technical solutions and advantages of the present application clearer, the implementations of the present application will be further described in detail below in conjunction with the accompanying drawings.

[036] In a foreground and background separation system, the foreground is configured for software interface display and provides an interactive interface for a user's operation, while the background provides detailed function logics and records data. Typically, both a client/server (C/S) system and a browser/server (B/S) system belong to the foreground and background separation systems.

[037] At present, in a machine vision industry or other industries, some different function modules (which are also called function units) are extracted in advance and are combined to achieve a certain specific combined function. Process configuration means a process of arranging the function modules in a certain order and configuring the relationships between

them. A process configuration system means a software that provides a process configuration functionality. Considering that there generally is a graphical flowchart on the interactive interface provided by the foreground of the process configuration system, the process configuration system is also called a graphical process configuration system.

[038] The process configuration system is the typical foreground and background separation system. The module processing methods provided in the examples of the present application may be applied in the process configuration system. Optionally, the module processing methods provided in the examples of the present application may also be applied to other foreground and background separation systems, which is not limited in the examples of the present application.

[039] For the convenience of following descriptions, two concepts involved in the examples of the present application are briefly introduced here.

[040] Undo: a software state is rolled back to a state before the last operation according to user requirements when a user interacts with a software system, which is called an undo. The undo may be chained. That is, the undo may be performed multiple times to return to the state before multiple operations.

[041] Redo: as the opposite of the undo action, the software state is advanced to a state before the last undo according to user requirements when the software reaches a certain state due to the undo, which is called a redo. The redo may also be chained. That is, the redo may be performed multiple times to restore the state before multiple undoes.

[042] The purpose of the undo or the redo is to reset the state of the software to a state after any one operation between the last operation and previous N-th operation, that is, to give the user a chance to withdraw their actions.

[043] In the foreground and background separation system, the foreground provides an interactive interface on which the user performs operations, and the background provides data logical functions and storage. In such a system, both two technologies, how to achieve the undo or the redo while ensuring a synchronization between the foreground data and the background data and how to achieve the undo or the redo for a user-triggered operation when multiple function modules change as a result of the change of a single function module in the process configuration system, are currently hot topics of research. The module processing methods provided in the examples of the present application can implement the above two technologies, whose specific implementations are to be described in detail in subsequent examples.

[044] FIG. 1 is a schematic architecture diagram of a foreground and background separation

system provided in an example of the present application. As illustrated in FIG. 1, the foreground and background separation system includes a foreground and a background.

[045] An interactive interface provided by the foreground displays Module 1 and Module 2, as well as a connection relationship between Module 1 and Module 2. Meanwhile, Module 1 is displayed based on a specific display manner, and Module 2 is also displayed based on a specific display manner. FIG. 1 illustrates the display manner of Module 2, which specifically includes the position coordinate of upper left corner of Module 2 being (200, 300), the height of Module 2 being 30, the width of Module 2 being 90, and the color of Module 2 being orange.

[046] The display manners of Module 1 and Module 2 as above may be called foreground data. The foreground data is also called interface data.

[047] The background provides specific logical functions. As illustrated in FIG. 1, there is a piece of code stored in the background, as follows:

(Process 0:

Function data of Module 1

Function data of Module 2

Connection relationship: Module 1 points to Module 2).

[048] The meaning indicated by the piece of code is: what is displayed on the foreground is a flowchart that includes two function modules, namely Module 1 and Module 2, and the connection relationship is that Module 1 points to Module 2, i.e., the output of Module 1 is the input of Module 2.

[049] The function data of Module 1 in the above piece of code indicates an internal execution logic of Module 1. The internal execution logic indicates how to obtain the output of Module 1 based on the input of Module 1. The function data of Module 2 indicates an internal execution logic of Module 2. The internal execution logic indicates how to obtain the output of Module 2 based on the input of Module 2. The above piece of code is generally referred to as background data. That is, in the example of the present application, the background data indicates the function data of each of a plurality of modules displayed on the foreground interactive interface and the logical relationships between the plurality of modules. The background data is also called underlying data, and the background is also called an underlying layer.

[050] As illustrated in FIG. 1, two modules are placed on the foreground interactive interface, with the upper left corner being at coordinate (200, 300), the width being 90, the height being 30, and the color being orange for Module 2. The background stores the logical relationships

between the modules and the function data of each module. When Module 2 is deleted, it means for the background that the function data of Module 2 and one connection relationship are missing from the background data. For an undo operation, in addition to restoring the function data of Module 2 and the connection relationship between Module 2 and Module 1 in the background, it is necessary to restore all of the position, the length and width, and the color on the foreground interactive interface to completely restore to a state before the deletion operation.

[051] In this scenario, if the foreground data is stored in the foreground and the background data is stored in the background, undoing a deletion operation means restoring the foreground data in the foreground while restoring the function data and the logical relationships in the background independently, which can easily lead to inconsistency between the foreground data and the background data. Therefore, in the example of the present application, all of the data is configured to be stored in a unique center, i.e., the background. In this way, the foreground data is also stored in the background. For the undo, it only needs restore the data stored in the background, and then the foreground obtains the foreground data from the background. That is, the foreground is only configured to obtain the foreground data from the background and display it, without storing the data, so as to solve the problem of possible data inconsistency. Since in the example of the present application, both the foreground data and the background data are stored in the background, for the convenience of following descriptions, the data stored in the background is collectively referred to as module data. That is, the module data includes the foreground data and the background data. Specifically, the module data of any one module includes the display manner of the module, the function data of the module, and the logical relationship between the module and each of the other modules.

[052] It should be noted that the foreground and background separation system illustrated in FIG. 1 is merely an example and does not constitute a limitation on the foreground and background separation system involved in the examples of the present application.

[053] FIG. 2 is a flowchart of a module processing method for a foreground and background separation system provided in an example of the present application. The method is applied to the foreground and background separation system. The foreground and background separation system includes a foreground and a background. The foreground displays an interactive interface. The interactive interface is configured to display a plurality of modules. The background is configured with a data center. The data center is configured to store module data of each of the plurality of modules. The module data of any one module includes a display manner of the module, function data of the module, and a logical relationship between the

module and each of the other modules. Specifically, as illustrated in FIG. 2, the method includes the following steps.

[054] At step 201, in response to a target operation on a target module, the foreground controls, based on the target operation, the background to update data corresponding to the target module in the data center, and the foreground updates the interactive interface in accordance with the updated data corresponding to the target module. There are one or more target modules among the plurality of modules, and the target operation is any operation on the target module.

[055] The data corresponding to the target module includes the module data of the target module in the data center and the data related to the target module in the module data of the other modules. These contents are to be explained and described in detail in the following and will not be elaborated here.

[056] The above target operation may be a delete operation on the target module, an add operation on the target module, or a modification operation on one or more attributes of the target module such as a display color. The delete operation specifically refers to deleting the target module displayed on the interactive interface, and the add operation specifically refers to adding the target module on the interactive interface.

[057] As an example, in a case where the target operation is the delete operation on the target module, step 201 specifically means that the foreground controls the background based on the target operation to delete the module data of the target module from the data center and delete the data related to the target module from the module data of the other modules, and redisplay the interactive interface based on the data center after the data is deleted.

[058] As another example, in a case where the target operation is the add operation on the target module, step 201 specifically means that the foreground controls the background based on the target operation to add the module data of the target module into the data center and add the data related to the target module into the module data of the other modules, and redisplay the interactive interface based on the data center after the data is added.

[059] In a possible implementation, a module deleting control and a module adding control are displayed in the interactive interface. When the foreground detects a selection operation on the module deleting control or the module adding control from a user, it may be determined that the target operation on the target module is detected. In this case, the target operation may be performed through step 201.

[060] After detecting the target operation, the foreground controls the background based on

the target operation to update the data corresponding to the target module in the data center, and updates the interactive interface in accordance with the updated data corresponding to the target module in the data center, thereby achieving a consistency among the interactive interface, the foreground data and the background data.

[061] The data corresponding to the target module includes the module data of the target module and the data related to the target module in the module data of the other modules. The module data of the target module includes the function data of the target module, the display manner of the target module, and the logical relationship between the target module and each of the other modules. The function data of the target module indicates an internal execution logic of the target module. The internal execution logic indicates how to obtain the output of the target module based on the input of the target module. The display manner of the target module includes, for example, a display position, a display size, a display color, etc. of the target module. The data related to the target module in the module data of each of the other modules includes the logical relationship between said other module and the target module.

[062] In a possible implementation, the logical relationship between the target module and each of the other modules includes a connection relationship between the target module and said other module. The connection relationship indicates a connecting line relationship between the target module and said other module on the interactive interface. The connecting line relationship specifically indicates an execution order between the target module and said other module during executing the flowchart. For example, the target module is Module 1 illustrated in FIG. 1, and said other module is Module 2. The connection relationship between the target module and said other module may indicate that the internal execution logic of Module 1 is executed firstly, and the internal execution logic of Module 2 is executed subsequently.

[063] In a case where the foreground and background separation system is a process configuration system, FIG. 3 is a schematic flowchart provided in an example of the present application. The flowchart shown on the foreground is usually as illustrated in FIG. 3. The connection relationships between various modules in the flowchart illustrated in FIG. 3 only represent an execution order. There is data interaction between the modules. It requires that an output of the previous module is used as an input or a parameter of the subsequent module to participate in the calculation, otherwise the process is meaningless. If considering the input and output parameters and the parameters of the modules themselves, the practical flowchart is much more complicated than the flowchart illustrated in FIG. 3.

[064] FIG. 4 is another schematic flowchart provided in an example of the present

application. As illustrated in FIG. 4, in addition to the connection relationships between the modules, there are also complex internal subscription relationships between parameters, inputs, and outputs. For example, an internal parameter of a certain module may be an output of the previous module in the execution order and an output of the module may be an internal parameter of the next module in the execution order, or an output of a certain module may be an input of the next module in the execution order, and so on. That is, a certain module may subscribe to the inputs or outputs of a certain previous module. Thus, the flowchart is actually a very complex one-way network. When a certain module is deleted, it not only affects the module itself, but also affects the subscription relationships between all subsequent modules and both it and its previous modules on the flowchart.

[065] Therefore, in another possible implementation, the logical relationship between the target module and each of the other modules may include a subscription relationship between the target module and said other module in addition to the connection relationship between the target module and said other module. The subscription relationship indicates an association relationship between at least one of the internal parameter or the input of the target module and the output of said other module, and/or an association relationship between the output of the target module and at least one of the internal parameter or the input of said other module, and/or an association relationship between the input of the target module and the output of said other module.

[066] That is, in the example of the present application, for any one module, the logical relationship between the module and each of the other modules includes the connection relationship between the module and said other module, and the subscription relationship between the module and said other module. The connection relationship indicates a connecting line relationship between the module and said other module on the display interface. The subscription relationship indicates the association relationship between the internal parameter of the module and the output of said other module, and/or the association relationship between the output of the module and the internal parameter of said other module, and/or the association relationship between the output of the module and the input of said other module, and/or the association relationship between the input of the module and the output of said other module. That is, the subscription relationship indicates the association relationship between at least one of the internal parameter or the input of the module and the output of said other module, and/or an association relationship between the output of the module and at least one of the internal parameter or the input of said other module.

[067] It should be noted that each module may have one or more internal parameters. In addition, each module may have one or more inputs and one or more outputs. The aforementioned association relationship, as a superior association relationship concept, may include an association relationship between a specific parameter and another specific parameter, and may also include association relationships between multiple parameters and other parameters separately. For example, the association relationship between the internal parameter(s) of the target module and the output(s) of the other modules may mean that one output of a certain other module is one internal parameter of the target module, or two outputs of a certain other module are two internal parameters of the target module, respectively, or one output of each of two other modules is one of two internal parameters of the target module. The above examples are only used for illustration and do not constitute a limitation on the association relationship involved in the examples of the present application.

[068] At step 202, in a case where an undo instruction for the target operation is detected, the foreground control the background based on the undo instruction to restore the data corresponding to the target module in the data center, and updates the interactive interface in accordance with the restored data corresponding to the target module.

[069] In a possible implementation, an undo operation control is displayed in the interactive interface. When the foreground detects a selection operation on the undo operation control from the user, it may be determined that the undo instruction for the target operation is detected. In this case, an undo process for the target operation may be executed through step 202.

[070] In a case where the target operation is a delete operation on the target module, the above step 202 specifically means that the foreground controls the background based on the undo instruction to re-add the data corresponding to the target module into the data center.

[071] In a case where the target operation is an add operation on the target module, the above step 202 specifically means that the foreground controls the background based on the undo instruction to delete the data corresponding to the target module from the data center.

[072] In the example of the present application, the above steps 201 and 202 may be specifically performed by taking the foreground as the operation center, or taking the background as the operation center. Taking the foreground as the operation center means that the foreground records various specific operations of performing the target operation, while taking the background as the center means that the background records various specific operations of performing the target operation. That is, the difference between the two lies in whether the file recording the specific operations is saved in the foreground or the background.

Two implementations are distinguished and described respectively as follows.

[073] (1) Taking the foreground as the center

[074] In this case, the implementation process of the above step 201 may be that the foreground sends one or more first data update instructions to the background based on the target operation. Any one of the one or more first data update instructions instructs the background to update a part of the data corresponding to the target module. The foreground generates an operation record based on the sending of the one or more data update instructions.

[075] For example, in a scenario where the target operation is to delete a module or add a module, the foreground sends a first logical relationship update instruction and a first module update instruction to the background, and generates the operation record about sending the first logical relationship update instruction and the first module update instruction. The first logical relationship update instruction instructs the background to update, based on the target operation, the logical relationship between each of the other modules and the target module in the module data of said other module in the data center. The first module update instruction instructs the background to update the module data of the target module in the data center based on the target operation.

[076] It should be noted that the above operation record indicates an order in which various first data update instructions are sent, and the data change indicated by each instruction.

[077] In this case, the implementation process of the above step 202 may be that the foreground sends one or more second data update instructions to the background based on an inverse operation of the target operation indicated by the operation record. Any one of the one or more second data update instructions instructs the background to restore the logical relationship between each of the other modules and the target module in the module data of said other module to a state before the target operation is performed, or instructs to restore the module data of the target module to the state before the target operation is performed.

[078] For example, in a scenario where the target operation is to delete a module or add a module, the foreground sends a second logical relationship update instruction and a second module update instruction to the background based on the inverse operation of the target operation indicated by the operation record. The second logical relationship update instruction instructs the background to restore the logical relationship between the target module and each of the other modules in the module data of said other module in the data center to the state before the target operation is performed. The second module update instruction instructs the background to restore the module data of the target module in the data center to the state before

the target operation is performed, so as to help the foreground to subsequently obtain the updated data corresponding to the target module from the background, and update the interactive interface based on the updated data corresponding to the target module.

[079] Specifically, in a case where the target operation is the delete operation on the target module, when the target operation is performed, the foreground sends the first logical relationship update instruction firstly and sends the first module update instruction subsequently to the background. In this way, the background can be controlled to delete the logical relationship between the target module and each of the other modules in the module data of said other module firstly, delete the module data of the target module subsequently, and then generate an operation record about sending the first logical relationship update instruction firstly and sending the first module update instruction subsequently. When the target operation is undone, based on the inverse operation of the target operation indicated by the aforementioned operation record, the foreground sends the second module update instruction firstly and sends the second logical relationship update instruction subsequently to the background. In this way, the background can be controlled to restore the module data of the target module in the data center firstly, and restore the logical relationship between the target module and each of the other modules in the module data of said other module.

[080] Similarly, in a case where the target operation is the add operation on the target module, when the target operation is performed, the foreground sends the first module update instruction firstly and sends the first logical relationship update instruction subsequently to the background. In this way, the background can be controlled to add the module data of the target module into the data center firstly and add the logical relationship between the target module and each of the other modules into the module data of said other module in the data center subsequently, and generate the operation record about sending the first module update instruction firstly and sending the first logical relationship update instruction subsequently. When the target operation is undone, based on the inverse operation of the target operation indicated by the aforementioned operation record, the foreground sends the second logical relationship update instruction firstly and sends the second module update instruction subsequently to the background. In this way, the background can be controlled to delete the logical relationship between the target module and each of the other modules from the module data of said other module in the data center firstly, and delete the module data of the target module from the data center subsequently.

[081] The purpose of sending instructions in the above order is that, when the foreground

undoes the target operation, it can be implemented completely based on the inverse operation of the target operation to avoid being unable to roll back to the state before the target operation is performed.

[082] For example, when the target operation is the delete operation on Module 2 illustrated in FIG. 1 and if the foreground is taken as the center, the specific processes of step 201 and step 202 is as follows.

[083] When performing the target operation, the following processes are included.

[084] 1) The foreground sends a module data obtaining command to the background, and receives the data corresponding to the target module sent by the background. The data corresponding to the target module includes the module data of the target module and the data related to the target module in the module data of each of the other modules.

[085] 2) The foreground analyzes the obtained data corresponding to the target module, and determines that for deleting the target module, it is required to delete the logical relationship between the target module and each of the other modules from the module data of said other module firstly and delete the module data of the target module subsequently. Based on the analysis result, the foreground sends a connection line deleting command to the background, that is, sends the above-mentioned first logical relationship update instruction to instruct the background to delete the logical relationship between the target module and said other module from the module data of said other module in the data center, and generates an operation record corresponding to sending the first logical relationship update instruction. For the convenience of subsequent descriptions, this operation record is referred to as a first operation record. When deleting the logical relationship, the background backs up the connection data related to the target module to another storage location firstly, that is, the backup indicates the logical relationship between the target module and each of the other modules, and deletes the connection data related to the target module from the background data subsequently, so as to facilitate a subsequent rapid recovery of the deleted data.

[086] 3) The foreground sends a module deleting command to the background, that is, sends the above-mentioned first module update instruction, so as to instruct the background to delete the module data of the target module and generate an operation record corresponding to sending the first logical relationship update instruction. For the convenience of subsequent description, this operation record is referred to as a second operation record. When deleting the module data of the target module, the background backs up the module data of the target module to another storage location, and then deletes the module data of the target module from the background

data, so as to facilitate a subsequent rapid recovery of the deleted data.

[087] When undoing the target operation, the following processes are included.

[088] 1) Since the first operation record is generated before the second operation record, in order to achieve the inverse operation of the above target operation, the foreground sends a module importing command firstly to the background based on the second operation record, that is, sends the above second module update instruction, so as to instruct the background to add the module data of the target module into the data center.

[089] 2) Then, the foreground sends a command for establishing a connection between Module 1 and Module 2 to the background based on the first operation record, that is, sends the above second logical relationship update instruction, so as to instruct the background to add the logical relationship between the target module and each of the other modules into the module data of said other module in the data center.

[090] 3) The foreground obtains module interface data from the background, where the module interface data is the display manner of the target module in the module data of the target module, and redisplay the target module on the interactive interface based on the obtained display manner.

[091] The above entire procedure is driven by the foreground, with the background responding passively and providing some basic interfaces for obtaining and importing the data.

[092] At present, it achieves sending instructions by the foreground to the background by calling the interfaces of background, that is, the background provides a series of interfaces. When needing to send an instruction, the foreground calls a certain interface of the background to send the instruction, so as to control the background to perform certain specific operations. For example, the background provides a logical relationship deleting interface and a module deleting interface. In this scenario, if the target operation is to delete a module, the first logical relationship update instruction sent by the foreground to the background may be implemented by calling the logical relationship deleting interface, and the first module update instruction sent by the foreground to the background may be implemented by calling the module deleting interface.

[093] In the above implementations with the foreground as the center, the foreground knows what steps included in the specific operation and an affected scope best. The background only provides the interfaces and is not expected to pay attention to a specific business. Therefore, the system is more likely to keep stable during the undo process.

[094] (2) Taking the background as the center

[095] At present, the implementation process of the above step 201 may be that the foreground updates the interactive interface based on the target operation, and then the foreground sends an update instruction to the background. The update instruction instructs the background to update based on the target operation the logical relationship between each of the other modules and the target module in the module data of said other module in the data center and update the module data of the target module in the data center. Meanwhile, the background generates the operation record corresponding to each of the above operations.

[096] Correspondingly, the implementation process of the above step 202 may be: the foreground sends a recovery instruction to the background. The recovery instruction instructs the background to restore, according to the inverse operation of the target operation indicated by the operation record locally stored, both the logical relationship between each of the other modules and the target module in the module data of said other module in the data center and the module data of the target module in the data center to the state before the target operation is performed. The foreground obtains the display manner in the updated module data of the target module from the background, and updates the interactive interface based on the updated display manner.

[097] For example, when the target operation is the delete operation on Module 2 illustrated in FIG. 1 and if the background is taken as the center, the specific processes of step 201 and step 202 are as follows.

[098] When performing the target operation, the following processes are included.

[099] 1) After detecting a module cleaning instruction, the foreground deletes the target module on the interactive interface firstly and then sends the module deleting instruction to the background. The module deleting instruction carries an identification of the target module.

[0100] 2) After receiving the module deleting instruction, the background backs up the connection data related to the target module in the module data of each of the other modules to another storage location firstly, then deletes the connection data related to the target module from the module data of each of the other modules in the data center, and generates an operation record corresponding to the operation on deleting the connection data, which is marked as a third operation record. The connection data related to the target module in the module data of each of the other modules indicates the logical relationship between the target module and said other module.

[0101] 3) The background backs up the module data of the target module to another storage location, deletes the module data of the target module from the data center, and generates an

operation record corresponding to the operation on deleting the module data, which is marked as a fourth operation record.

[0102] When undoing the target operation, the following various processes are included.

[0103] 1) The foreground sends an undo command to the background.

[0104] 2) The background queries the operation records and finds, based on the above fourth operation record, that the last operation is deleting the module data of a module, and performs the inverse operation of deleting the module data of the module, that is, importing the module data of the target module into the data center from the backup data.

[0105] 3) The background queries the operation records and finds, based on the above third operation record, that the previous operation of deleting the module data is deleting the connection data of the module, and performs the inverse operation of deleting the connection data of the module, that is, adding the connection data related to the target module into the data center from the module data of each of the other modules in the backup data.

[0106] 4) The background notifies the foreground that the undo is complete, and the foreground obtains the newest display manner of the target module from the background and updates the interactive interface.

[0107] The entire procedure is driven by the foreground through sending the instructions, and the background provides a full set of data records and the reverse operations to achieve deleting and restoring the data. The foreground performs the data querying and updating. It should be noted that in the above implementations with taking the foreground as the center, the foreground knows what steps included in the specific operation and an affected scope best, which helps developers to learn and control the actual processing status of the target operations at any time. Therefore, when applying the examples of the present application, the implementation with taking the foreground as the center may be considered to be adopted.

[0108] Furthermore, there may be a plurality of actual implementations for deleting the data corresponding to the target module as described above. As illustrated in FIG. 4, the actual flowchart is very complex. In this case, if the target operation is a delete operation on the target module, in order to deal with the undo and the redo under the complex relationships, there are two strategies for a more complex scenario where a deletion is performed and then is undone.

[0109] 1) A false deletion strategy. When the target module is deleted based on the target operation, only the module data of the target module is set with a deletion mark, without actually being deleted. Then, the logical relationship between each of the other modules and the deleted target module is cleaned up. When the target operation is undone, the deletion mark is removed

and the logical relationship between each of the other modules and the target module is restored in the data center.

[0110] That is, in the false deletion strategy, the specific implementation of the foreground sending one or more first data update instructions to the background in sequence based on the target operation as described above may be that: after the foreground sends the first logical relationship update instruction firstly and sends the first module update instruction subsequently to the background, the background deletes the logical relationship between each of the other modules and the target module from the module data of said other module when receiving the first logical relationship update instruction, and adds a deletion mark for the module data of the target module in the data center when receiving the first module update instruction.

[0111] Correspondingly, the specific implementation of the foreground sending one or more second data update instructions to the background in sequence based on the inverse operation of the target operation indicated by the operation record may be that: the foreground sends, based on the inverse operation of the target operation indicated by the operation record, the second module update instruction firstly and the second logical relationship update instruction subsequently to the background, and the background deletes the deletion mark for the module data of the target module when receiving the second module update instruction and adds the logical relationship between each of the other modules and the target module into the module data of said other module when receiving the second logical relationship update instruction.

[0112] 2) An export and import strategy. When the target module is deleted based on the target operation, the data corresponding to the target module is exported, and then the data corresponding to the target module is cleared. When it is undone, the module data of the target module is imported, and the connection and subscription relationship between each of the other modules and the target module is restored based on the previously exported data. That is, during deleting data, it exports the related data firstly and then directly delete the related data in the data center.

[0113] That is, in the export and import strategy, the specific implementation of the foreground sending one or more first data update instructions to the background in sequence based on the target operation may be that: the foreground sends the first logical relationship update instruction firstly and sends the first module update instruction subsequently to the background, so as to control the background to delete the logical relationship between the target module and each of the other modules from the module data of said other module in the data center firstly and delete the module data of the target module subsequently.

[0114] Correspondingly, the specific implementation of the foreground sending one or more second data update instructions to the background in sequence based on the inverse operation of the target operation indicated by the operation record may be that: the foreground sends, based on the inverse operation of the target operation indicated by the operation record, the second module update instruction firstly and the second logical relationship update instruction to the background, so as to control the background to restore the module data of the target module in the data center firstly and restore the logical relationship between the target module and each of the other modules in the module data of said other module.

[0115] Furthermore, due to the synchronization between the foreground data and the background data in the above false deletion strategy, if the foreground and the background only implement the deletion in a form of mark after a module is to be deleted, the module deleted in the form of mark requires to be specially processed during executing the entire flowchart, so as to ignore it to avoid triggering its execution. However, the increasing number of such special data will affect a maintainability of the foreground and background data, thereby causing instability. Therefore, when applying the examples of the present application, the import and export strategy may be considered to be adopted for deleting the target module.

[0116] Furthermore, in the above export and import strategy, in order to facilitate the subsequent rapid recovery of the deleted data, the foreground may control the background to export and back up the data corresponding to the target module in the data center to a reference storage location before sending the first logical relationship update instruction firstly and sending the first module update instruction subsequently to the background. Then, after the foreground sends the second module update instruction firstly and sends the second logical relationship update instruction subsequently to the background, the background imports the module data of the target module from the reference storage location into the data center in response to the received second module update instruction, and imports the logical relationship between the target module and each of the other modules from the reference storage location into the data center in response to the received second logical relationship update instruction.

[0117] Furthermore, in a case where the target operation is the module deleting operation in an example of the present application, the same target operation may be performed on a batch of modules at the same time, that is, there is a scenario of batch delete operations. When performing the batch delete operations, there are multiple target modules in step 201 and step 202.

[0118] In such a scenario, there are also two strategies to achieve performing and undoing the

above target operation. One is taking operation as the center and the other is taking data as the center.

[0119] The first strategy: taking operation as the center.

[0120] Taking operation as the center means that, when performing a target operation on multiple target modules, the target operation is divided into a plurality of sub-operations on various single modules, that is, each sub-operation is only for one module. Then, the plurality of sub-operations are performed in sequence according to a performance order of the plurality of sub-operations, thereby achieving the target operation on the multiple target modules. During performing the target operation on the multiple target modules, for each sub-operation that is performed, an operation record for the sub-operation is generated and stored. When undoing the target operation on the multiple target modules, the undo action is completed according to the inverse operation of the sub-operations indicated by the stored operation records.

[0121] In the strategy with taking operation as the center, the specific implementation of the foreground sending the first logical relationship update instruction to the background as described above may be that: the foreground sends a subscription relationship deleting instruction A to the background, where the subscription relationship deleting instruction A instructs the background to delete, for each of the multiple target modules, subscription relationships of a post-executed module to one or more pre-executed modules and the target module separately, the post-executed module is a module whose execution occurs behind the target module in a flowchart displayed on the foreground, and the pre-executed module is a module whose execution occurs before the target module in the flowchart; the foreground sends a subscription relationship deleting instruction B to the background, where the subscription relationship deleting instruction B instructs the background to delete subscription relationships between the multiple target modules; the foreground sends a subscription relationship deleting instruction C to the background, where the subscription relationship deleting instruction C instructs the background to delete subscription relationships between each of the target modules and the one or more pre-executed modules; the foreground sends a connection relationship deleting instruction A to the background, where the connection relationship deleting instruction A instructs the background to delete a connection relationship between the post-executed module and each of the target modules; the foreground sends a connection relationship deleting instruction B to the background, where the connection relationship deleting instruction B instructs the background to delete connection relationships between the multiple target modules; and the foreground sends a connection relationship deleting instruction C to the background,

where the connection relationship deleting instruction C instructs the background to delete connection relationships between each of the target modules and the one or more pre-executed modules.

[0122] After sending the first logical relationship update instruction through the above implementations, the foreground sends the first module update instruction to the background, so as to control the background to delete the remaining data in the module data of each of the multiple target modules.

[0123] That is, in the strategy with taking operation as the center, the foreground analyzes the subscription relationships and the connecting line relationships between the multiple target modules and between each of the multiple target modules and the other modules firstly, deletes these subscription relationships and these connecting line relationships through the plurality of first logical relationship update instructions, and then deletes the module data of each target module.

[0124] The second strategy: taking data as the center.

[0125] Taking data as the center means that when performing a target operation on the multiple target modules, the multiple target modules are regarded as a whole. In this case, if the multiple target modules need to be deleted, the module data of the multiple target modules are deleted as a whole, and then the logical relationships between one or more edge target modules of the multiple target modules and the other modules are updated. The edge target module is of the multiple target modules and has a logical relationship with other modules. The operation records are also generated at the time of updating the module data and updating the logical relationships. When undoing the target operation on the multiple target modules, the undo action is completed according to the inverse operation of the target operation indicated by the stored operation records.

[0126] In the strategy with taking data as the center, the specific implementation of the foreground sending the first logical relationship update instruction firstly and sending the first module update instruction subsequently to the background may be that: the foreground sends a first subscription relationship deleting instruction to the background, where the first subscription relationship deleting instruction instructs the background to delete, from the module data of each of the other modules, subscription relationships between said other module and one or more edge target modules; the foreground sends a first connection relationship deleting instruction to the background, where the first connection relationship deleting instruction instructs the background to delete, from the module data of each of the other

modules, connection relationships between said other module and the one or more edge target modules; and the foreground sends a first module deleting instruction to the background, where the first module deleting instruction instructs the background to delete the module data of the multiple target modules.

[0127] That is, in the strategy with taking data as the center, the foreground only analyzes the subscription relationships and the connecting line relationships between the one or more edge target modules of the multiple target modules and the other modules, deletes these subscription relationships and these connecting line relationships through the plurality of first logical relationship update instructions, and then deletes the module data of each target module, which, instead of deleting the subscription relationships or the connecting line relationships among the various target modules one by one, saves the time required for the target operation.

[0128] FIG. 5 is a schematic diagram of a deleting module provided in an example of the present application, which is applicable to the delete operation, undoing the add operation, redoing the delete operation, etc. As illustrated in FIG. 5, there are two strategies for deleting Modules 2 to 5 in the flowchart.

[0129] 1. Taking operation as the center and decomposing batch operations into a combination of sub-operations for single modules. The various sub-operations are as follows.

[0130] 1) Deleting the subscription relationships between Modules 6-10 and Modules 1-5 in sequence, that is, deleting the subscription relationship between each of Modules 6-10 and each of Modules 1-5, respectively.

[0131] 2) Deleting the subscription relationship between Modules 3-5 and Modules 1-2 in sequence, that is, deleting the subscription relationship between each of Modules 3-5 and each of Modules 1-2, respectively.

[0132] 3) Deleting the subscription relationship between Module 2 and Module 1.

[0133] 4) Deleting the connecting line between Module 6 and Module 3, the connecting line between Module 7 and Module 4, and the connecting line between Module 8 and Module 5 in sequence, that is, deleting the connection relationship between Module 6 and Module 3, the connection relationship between Module 7 and Module 4, and the connection relationship between Module 8 and Module 5, respectively.

[0134] 5) Deleting the connecting lines between Modules 3-5 and Module 2 in sequence, that is, deleting the connection relationship between each of Modules 3-5 and Module 2, respectively.

[0135] 6) Deleting the connecting line between Module 2 and Module 1, that is, deleting the

connection relationship between Module 2 and Module 1.

[0136] 7) Deleting Modules 3-5 in sequence, that is, deleting the module data of each of Modules 3-5.

[0137] 8) Deleting Module 2, that is, deleting the module data of Module 2.

[0138] For the undo, by being performed in a reverse order from 8) to 1), the module data, the connection relationships, and the subscription relationships are added back one by one, so as to achieve undoing the delete operation on Module 2 to Module 5.

[0139] 2. A batch import and export strategy with taking data as the center.

[0140] 1) The module data of Modules 2-5 separately in the data center is exported as a whole and backed up to a reference location.

[0141] 2) Deleting the subscription relationship with Modules 1-5 from the module data of Modules 6-10 and generating an operation record. That is, considering Modules 6-10 as a whole and Modules 1-5 as a whole, the subscription relationship with Modules 1-5 as a whole is deleted from the module data of Modules 6-10 as a whole. The subscription relationship is specifically subscription relationship between the edge modules 3/4/5 of Modules 2-5 and the other modules 6-10.

[0142] 3) Deleting the connection relationships with Modules 2-5 from the module data of the other modules, deleting the individual module data of Modules 2-5, and generating an operation record.

[0143] For the undo:

[0144] 1) Importing the individual module data of Modules 2-5 as a whole.

[0145] 2) Restoring the connecting lines between the other modules and Modules 2-5 by utilizing the operation records. Specifically, the connecting lines include the connection relationship with Modules 2-5 in the module data of Module 1, and the connection relationship with Modules 2-5 in the module data of each of Modules 6-10.

[0146] 3) Restoring the subscription relationships between Modules 6-10 and Modules 1-5 by utilizing the operation records. Specifically, the subscription relationships include the subscription relationship with Modules 2-5 in the module data of Module 1, and the subscription relationship with Modules 2-5 in the module data of each of Modules 6-10.

[0147] The problem in the first strategy of converting batch operations into many single operations is that since an undoing order depends on a processing order, it will cause an error once an error occurs in the order and thus the module has not been restored when the connection relationship is restored. Even if all are restored, the flowchart cannot be completely restored to

its previous state because a certain connection relationship between some modules may be missing. Only the relationships between the exported whole and its surroundings need to be recorded according to the batch import and export strategy of the above second strategy. The internal relationships of the exported whole are already included in the module data of the exported module. When performing the undo or redo, it only needs to import the module data of the modules firstly and then restore the peripheral relationships between various other modules and the whole.

[0148] Furthermore, in an example of the present application, the first strategy with taking operation as the center is suitable for such a scenario of undoing and redoing a data change of a single module. For example, after modifying the location or color of a certain module, the most economical and effective approach is to perform the inverse operation of the location or color change.

[0149] Therefore, in a possible implementation in a case where the number of target modules is one, the foreground may divide the target operation into one or more sub-operations before sending one or more first data update instructions to the background based on the target operation. In this scenario, the above one or more first data update instructions correspond respectively to one or more sub-operations, and the background completes the target operation based on the one or more sub-operations. That is, the target operation is divided into multiple sub-operations, and then the multiple sub-operations are performed step by step, and the operation records are generated, so as to facilitate undoing the target operation subsequently based on the operation records of the plurality of sub-operations.

[0150] Optionally, after the undo operation, the user may also withdraw the undo operation, that is, re-perform the operation. In this case, the above redo process may be implemented through step 203 illustrated in FIG. 6.

[0151] At step 203, in a case where a redo instruction for the target operation is detected, the foreground controls based on the target instruction the background to update the data corresponding to the target module in the data center, and updates the interactive interface in accordance with the updated data corresponding to the target module.

[0152] The specific implementation of step 203 is substantially the same as that of step 201 and is not repeated herein.

[0153] In view of the above, in the foreground and background separation system provided in the examples of the present application, the module data stored in the data center of the background includes the display manners of the modules, the function data of the modules and

the logical relationships between the modules. That is, the background stores both the foreground data and the background data at the same time. As can be seen that the examples of the present application take the background as the data center for all the data. Therefore, the foreground can control the background to update both the foreground data and the background data when it detects a target operation on a target module, and also control the background to update both the foreground data and the background data when the target operation is undone, so as to ensure a consistency among the content displayed on the interactive interface of the foreground, the foreground data and the background data during performing or undoing the target operation.

[0154] The following is a summary of the various technical effects provided by the examples of the present application.

[0155] The innovations of the examples of the present application are to propose several practical and feasible implementation strategies for the undo or redo in the foreground and background separation system.

[0156] 1) In a case where there is data in both the foreground and the background, a single data center strategy may be adopted to store all data in the background.

[0157] 2) In a case where the foreground and the background are separated, the operation records may be centered on the foreground. For the undo or redo, the entire undo procedure is driven by the foreground.

[0158] 3) For complex flowcharts with many relationships, the way of importing and exporting data instead of the false deletion strategy may be adopted in a deleting or undoing scenario.

[0159] 4) For complex flowcharts with many relationships, the way of importing and exporting data instead of the way of combining operations may be adopted when deleting and undoing batch data.

[0160] 5) For mere data changes in the flowchart, the strategy with taking operation as the center may be adopted.

[0161] By the above strategies, the undo or redo in the foreground and background separation system may be better implemented, with balancing implementation complexity, memory usage, and system stability.

[0162] All the above optional technical solutions can be combined in any way to form optional examples of the present application, which are not repeated one by one in the examples of the present application.

[0163] FIG. 7 is a schematic structure diagram of a module processing apparatus for a foreground and background separation system provided in an example of the present application. The apparatus may be implemented by software, hardware, or a combination of both. The apparatus is applied in the foreground and background separation system. The foreground and background separation system includes a foreground and a background. The foreground displays an interactive interface that is configured to display a plurality of modules, and the background is configured with a data center that is configured to store module data of each of the plurality of modules. For any one module, its module data includes a display manner of the module, function data of the module, and a logical relationship between the module and each of the other modules.

[0164] As illustrated in FIG. 7, the apparatus 700 includes the following modules.

[0165] A first controlling module 701 that is configured to in response to a target operation on a target module, enable the foreground to control, based on the target operation, the background to update data corresponding to the target module in the data center, and enable the foreground to update the interactive interface in accordance with the updated data corresponding to the target module. There are one or more target modules among the plurality of modules and the target operation is any one operation on the target module.

[0166] A second controlling module 702 that is configured to in a case where an undo instruction for the target operation is detected, enable the foreground and to control, based on the undo instruction, the background to restore the data corresponding to the target module in the data center, and enable the foreground to update the interactive interface in accordance with the restored data corresponding to the target module.

[0167] In a possible implementation based on the above apparatus, as illustrated in FIG. 7, the apparatus further includes:

[0168] a third controlling module 703 that is configured to in a case where a redo instruction for the target operation is detected, enable the foreground to control, based on the target operation, the background to update the data corresponding to the target module in the data center, and enable the foreground to update the interactive interface in accordance with the updated data corresponding to the target module.

[0169] In a possible implementation based on the above apparatus, the first controlling module is configured to:

[0170] enable the foreground to send one or more first data update instructions to the background sequentially based on the target operation, where any one of the one or more first

data update instructions instructs the background to update a part of the data corresponding to the target module; and

[0171] enable the foreground to generate an operation record based on the sending of the one or more data update instructions.

[0172] Accordingly, the second controlling module is configured to:

[0173] enable the foreground to send one or more second data update instructions to the background sequentially based on an inverse operation of the target operation indicated by the operation record, where any one of the one or more second data update instructions instructs the background to restore the part of the data corresponding to the target module in the data center to a state before the target operation is performed.

[0174] In a possible implementation based on the above apparatus, in a case where the target operation is an add operation for the target module, the first controlling module is configured to:

[0175] enable the foreground to send a first module update instruction firstly and a first logical relationship update instruction subsequently to the background, so as to control the background to add the module data of the target module into the data center firstly and add a logical relationship between the target module and each of the other modules into the module data of said other module subsequently.

[0176] Accordingly, the second controlling module is configured to:

[0177] enable the foreground to send, based on the inverse operation of the target operation indicated by the operation record, a second logical relationship update instruction firstly and a second module update instruction subsequently to the background, so as to control the background to delete, from the module data of each of the other modules in the data center, the logical relationship between the target module and said other module firstly and delete the module data of the target module from the data center subsequently.

[0178] In a possible implementation based on the above apparatus, in a case where the target operation is a delete operation for the target module, the first controlling module is configured to:

[0179] enable the foreground to send a first logical relationship update instruction firstly and a first module update instruction subsequently to the background, so as to control the background to delete, from the module data of each of the other modules in the data center, a logical relationship between the target module and said other module firstly and delete the module data of the target module from the data center subsequently.

[0180] Accordingly, the second controlling module is configured to:

[0181] enable the foreground to send, based on the inverse operation of the target operation indicated by the operation record, a second module update instruction firstly and a second logical relationship update instruction subsequently to the background, so as to control the background to restore the module data of the target module in the data center firstly and restore the logical relationship between the target module and each of the other modules in the module data of said other module subsequently.

[0182] In a possible implementation based on the above apparatus, in a case where the target operation is a delete operation for the target module, the first controlling module is configured to:

[0183] enable the foreground to send a first logical relationship update instruction firstly and a first module update instruction subsequently to the background, so as to control the background to delete, when receiving the first logical relationship update instruction, a logical relationship between each of the other modules and the target module from the module data of said other module in the data center, and control the background to add, when receiving the first module update instruction, a deletion mark for the module data of the target module in the data center.

[0184] Accordingly, the second controlling module is configured to:

[0185] enable the foreground to send, based on an inverse operation of the target operation indicated by the operation record, a second module update instruction firstly and a second logical relationship update instruction subsequently to the background, so as to control the background to delete, when receiving the second module update instruction, the deletion mark for the module data of the target module in the data center, and control the background to add, when receiving the second logical relationship update instruction, the logical relationship between each of the other modules and the target module into the module data of said other module in the data center.

[0186] In a possible implementation based on the above apparatus, the apparatus further includes:

[0187] an exporting module that is configured to before the foreground sends the first logical relationship update instruction firstly and the first module update instruction subsequently to the background, enable the foreground to control the background to export and backup the data corresponding to the target module in the data center to a reference storage location, where the data corresponding to the target module includes the module data of the target module and the

logical relationship between the target module and each of the other modules in the module data of said other module; and

[0188] an importing module that is configured to

[0189] after the foreground sends the second module update instruction firstly and the second logical relationship update instruction subsequently, enable the background to import the module data of the target module from the reference storage location into the data center in response to the received second module update instruction, and enable the background to import the logical relationship between the target module and each of the other modules from the reference storage location into the data center in response to the received second logical relationship update instruction.

[0190] In a possible implementation based on the above apparatus, the logical relationship between any one of the plurality of modules and each of the other modules includes a connection relationship between the any one module and each of the other modules, and a subscription relationship between the any one module and each of the other modules. The connection relationship indicates a connecting line relationship between the any one module and each of the other modules on the interactive interface. The subscription relationship indicates an association relationship between at least one of an internal parameter or an input of the any one module and an output of each of the other modules, and/or indicates an association relationship between an output of the any one module and at least one of an internal parameter or an input of each of the other modules.

[0191] In a possible implementation based on the above apparatus, in a case where there are multiple target modules, the first controlling module is configured to:

[0192] enable the foreground to send a first subscription relationship deleting instruction to the background, where the first subscription relationship deleting instruction instructs the background to delete subscription relationships between each of the other modules and one or more edge target modules from the module data of said other module, and where the edge target module is of the multiple target modules and has a logical relationship with said other module;

[0193] enable the foreground to send a first connection relationship deleting instruction to the background, where the first connection relationship deleting instruction instructs the background to delete connection relationships between each of the other modules and one or more edge target modules from the module data of said other module; and

[0194] enable the foreground to send a first module deleting instruction to the background, where the first module deleting instruction instructs the background to delete the module data

of the multiple target modules.

[0195] In a possible implementation based on the above apparatus, in a case where there are multiple target modules, the first controlling module is configured to:

[0196] enable the foreground to send a subscription relationship deleting instruction A to the background, where the subscription relationship deleting instruction A instructs the background to delete, for each of the multiple target module, subscription relationships of a post-executed module to one or more pre-executed modules and the target module respectively, where the post-executed module is a module whose execution occurs behind the target module in a flowchart displayed on the foreground, and the pre-executed module is a module whose execution occurs before the target module in the flowchart;

[0197] enable the foreground to send a subscription relationship deleting instruction B to the background, where the subscription relationship deleting instruction B instructs the background to delete subscription relationships between the multiple target modules;

[0198] enable the foreground to send a subscription relationship deleting instruction C to the background, where the subscription relationship deleting instruction C instructs the background to delete subscription relationships between each of the target modules and the one or more pre-executed modules;

[0199] enable the foreground to send a connection relationship deleting instruction A to the background, where the connection relationship deleting instruction A instructs the background to delete a connection relationship between the post-executed module and each of the target modules;

[0200] enable the foreground to send a connection relationship deleting instruction B to the background, where the connection relationship deleting instruction B instructs the background to delete connection relationships between the multiple target modules;

[0201] enable the foreground to send a connection relationship deleting instruction C to the background, where the connection relationship deleting instruction C instructs the background to delete connection relationships between each of the target modules and the one or more pre-executed modules; and

[0202] enable the foreground to send a first module deleting instruction to the background, where the first module deleting instruction instructs to delete remaining data of the module data of the multiple target modules.

[0203] In view of the above, in the foreground and background separation system provided in the examples of the present application, the module data stored in the data center of the

background includes the display manners of the modules, the function data of the modules and the logical relationships between the modules. That is, the background stores both the foreground data and the background data. As can be seen that the examples of the present application take the background as the data center for all the data. Therefore, the foreground can control the background to update both the foreground data and the background data when it detects a target operation on a target module, and also control the background to update both the foreground data and the background data when the target operation is undone, so as to ensure a consistency among the content displayed on the interactive interface of the foreground, the foreground data and the background data during performing or undoing the target operation.

[0204] It should be noted that the division of the above functional modules is only used as an example for illustrating the operations on the modules performed by the module processing apparatus for a foreground and background separation system. In practical applications, the above functions may be assigned to different functional modules for implementation as required. That is, the internal structure of the apparatus may be divided into different functional modules to implement all or a part of the functions described above. In addition, the module processing apparatuses for the foreground and background separation system provided in the above examples and the module processing method examples for the foreground and background separation system are under the same concept. The specific implementation processes of the module processing apparatuses for the foreground and background separation system refer to the method examples in detail and are not repeated here.

[0205] FIG. 8 is a schematic structure diagram of a terminal 800 provided by an example of the present application. The foreground involved in the aforementioned examples may all be implemented by the structure illustrated in FIG. 8. The terminal 800 may be a smart phone, a tablet computer, a moving picture experts group audio layer III (MP3) player, a moving picture experts group audio layer IV (MP4) player, a laptop computer or a desktop computer. The terminal 800 may also be referred to as user equipment, portable terminal, laptop terminal, desktop terminal, or other names.

[0206] Typically, the terminal 800 includes a processor 801 and a memory 802.

[0207] The processor 801 may include one or more processing cores, such as a quad-core processor, an octa-core processor, etc. The processor 801 may be implemented in at least one hardware form of a digital signal processing (DSP), a field-programmable gate array (FPGA), or a programmable logic array. The processor 801 may further include a main processor and one or more co-processors. The main processor is configured to process data in a awake state,

also called a central processing unit (CPU), and the co-processor is a low-power processor for processing the data in a standby state. In some examples, the processor 801 may be integrated with a graphics processing unit (GPU) that is responsible for rendering and drawing the contents to be displayed on a display screen. In some examples, the processor 801 may further include an artificial intelligence (AI) processor that is configured to process computing operations related to machine learning.

[0208] The memory 802 may include one or more computer-readable storage media that may be non-transitory. The memory 802 may further include one or more high-speed random access memories, and one or more non-volatile memories, such as one or more disk storage devices and flash memory storage devices. In some examples, the non-transitory computer-readable storage medium in the memory 802 is configured to store at least one instruction. The at least one instruction is configured to be executed by the processor 801 to implement the module processing method for the foreground and background separation system provided in the method examples of the present application.

[0209] In some examples, optionally, the terminal 800 may further include a peripheral device interface 803 and at least one peripheral device. The processor 801, the memory 802 and the peripheral device interface 803 may be connected via a bus or a signal line. Each peripheral device may be connected to the peripheral device interface 803 via a bus, a signal line or a circuit board. Specifically, the peripheral device includes at least one of a radio frequency circuit 804, a display screen 805, a camera component 806, an audio circuit 807, a positioning component 808 or a power supply 809.

[0210] The peripheral device interface 803 may be configured to connect at least one peripheral device related to input/output (I/O) to the processor 801 and the memory 802. In some examples, the processor 801, the memory 802 and the peripheral device interface 803 are integrated on the same circuit module or circuit board. In some other examples, any one or two of the processor 801, the memory 802 and the peripheral device interface 803 may be implemented on a separate circuit module or circuit board, which is not limited in this example.

[0211] The radio frequency circuit 804 is configured to receive and transmit radio frequency (RF) signals, also called electromagnetic signals. The radio frequency circuit 804 communicates with a communication network and other communication devices through the electromagnetic signals. The radio frequency circuit 804 converts electrical signals into electromagnetic signals for transmission, or converts received electromagnetic signals into electrical signals. Optionally, the radio frequency circuit 804 includes an antenna system, an

RF transceiver, one or more amplifiers, a tuner, an oscillator, a digital signal processor, a codec circuit module group, a user identity module card, and the like. The radio frequency circuit 804 may communicate with other terminals through at least one wireless communication protocol. The wireless communication protocol includes, but not limited to a metropolitan area network, various generations of mobile communication networks (2G, 3G, 4G and 5G), a wireless local area network and/or a wireless fidelity (WiFi) network. In some examples, the radio frequency circuit 804 may further include a near field communication (NFC) related circuit, which is not limited in the present application.

[0212] The display screen 805 is configured to display a user interface (UI). The UI may include graphics, texts, icons, videos, and any combination thereof. The display screen 805, when being a touch display screen, further has an ability of collecting touch signals on the surface or above the surface of the display screen 805. The touch signal may be inputted as a control signal to the processor 801 for processing. In this case, the display screen 805 may further be configured to provide virtual buttons and/or virtual keyboards, also called soft buttons and/or soft keyboards. In some examples, there may be one display screen 805 that is disposed on a front panel of the terminal 800. In other examples, there may be at least two display screens 805 that are disposed on different surfaces of the terminal 800, respectively or in a folding design. In still other examples, the display screen 805 may be a flexible display screen that is disposed on a curved surface or a folding surface of the terminal 800. Furthermore, the display screen 805 may be configured to be in a non-rectangular irregular shape, that is, a special-shaped screen. The display screen 805 may be made of materials such as liquid crystal display (LCD) and organic light-emitting diode (OLED).

[0213] The camera component 806 is configured to capture images or videos. Optionally, the camera component 806 includes a front camera and a rear camera. Typically, the front camera is disposed on the front panel of the terminal, and the rear camera is disposed on the back of the terminal. In some examples, there are at least two rear cameras, which are any one of a main camera, a depth of field camera, a wide-angle camera and a telephoto camera, respectively. Thus, a fusion of the main camera and the depth of field camera achieves a background blur function, and a fusion of the main camera and the wide-angle camera achieves a panoramic shooting function and a virtual reality (VR) shooting function or other fusion shooting functions. In some examples, the camera component 806 may further include a flashbulb. The flashbulb may be a single-color temperature flashbulb or a dual-color temperature flashbulb. The dual-color temperature flashbulb refers to a combination of warm flashbulb and cold flashbulb,

which may be configured to compensate for light under different color temperatures.

[0214] The audio circuit 807 may include a microphone and a speaker. The microphone is configured to collect sound waves from a user and the environment, and convert the sound waves into electrical signals and input them into the processor 801 for processing, or input them into the RF circuit 804 to achieve voice communication. For the purpose of stereo sound collection or noise reduction, there may be multiple microphones, which are separately arranged at different locations of the terminal 800. The microphone may further be an array microphone or an omnidirectional collection microphone. The speaker is configured to convert the electrical signals from the processor 801 or the RF circuit 804 into sound waves. The speaker may be a conventional film speaker or a piezoelectric ceramic speaker. The speaker, when being a piezoelectric ceramic speaker, may not only convert the electrical signals into sound waves audible to humans, but also convert electrical signals into sound waves inaudible to humans for the purposes such as distance measurement. In some examples, the audio circuit 807 may further include a headphone jack.

[0215] The positioning component 808 is configured to locate a current geographical location of the terminal 800 to implement a navigation or a location based service (LBS).

[0216] The power supply 809 is configured to supply power to various components in the terminal 800. The power supply 809 may be AC power, DC power, a disposable battery, or rechargeable battery. When the power supply 809 includes the rechargeable battery, the rechargeable battery may support wired charging or wireless charging. The rechargeable battery may also be configured to support a fast charging technology.

[0217] In some examples, the terminal 800 further includes one or more sensors 810. The one or more sensors 810 include, but not limited to, an acceleration sensor 811, a gyroscope sensor 812, a pressure sensor 813, a fingerprint sensor 814, an optical sensor 815, and a proximity sensor 816.

[0218] The acceleration sensor 811 may detect the magnitude of acceleration on three coordinate axes of a coordinate system established by the terminal 800. For example, the acceleration sensor 811 may be configured to detect the components of gravity acceleration on three coordinate axes. The processor 801 may control the display screen 805 to display the user interface in a horizontal view or a vertical view according to the gravity acceleration signals collected by the acceleration sensor 811. The acceleration sensor 811 may further be configured to collect game or user motion data.

[0219] The gyroscope sensor 812 may detect a body direction and a rotation angle of the

terminal 800. The gyroscope sensor 812 may cooperate with the acceleration sensor 811 to collect user's 3D actions on the terminal 800. The processor 801 may implement the following functions based on the data collected by the gyroscope sensor 812: motion sensing (such as changing the UI according to a user's tilt operation), image stabilization during shooting, game control, and inertial navigation.

[0220] The pressure sensor 813 may be disposed on a side frame of the terminal 800 and/or at a layer under the display screen 805. The pressure sensor 813, when being disposed on the side frame of the terminal 800, may detect user's holding signals of the terminal 800. The processor 801 performs left/right hand recognition or shortcut operations according to the holding signals collected by the pressure sensor 813. When the pressure sensor 813 is disposed at the layer under the display screen 805, the processor 801 achieves controlling one or more operable controls on the UI interface according to the pressure operations of the user on the display screen 805. The one or more operable controls include at least one of a button control, a scroll bar control, an icon control, or a menu control.

[0221] The fingerprint sensor 814 is configured to collect a fingerprint of the user. The processor 801 recognizes the identity of the user according to the fingerprint collected by the fingerprint sensor 814, or the fingerprint sensor 814 recognizes the identity of the user according to the collected fingerprint. When the user's identity is identified as a trusted identity, the processor 801 authorizes the user to perform relevant sensitive operations, including unlocking the screen, viewing encrypted information, downloading a software, making a payment, and changing settings. The fingerprint sensor 814 may be disposed on the front, back, or side of the terminal 800. When a physical button or a manufacturer's logo is provided on the terminal 800, the fingerprint sensor 814 may be integrated with the physical button or the manufacturer's logo.

[0222] The optical sensor 815 is configured to collect the ambient light intensity. In an example, the processor 801 may control a display brightness of the display screen 805 according to the ambient light intensity collected by the optical sensor 815. Specifically, the display brightness of the display screen 805 is increased when the ambient light intensity is high and is decreased when the ambient light intensity is low. In another example, the processor 801 may further dynamically adjust shooting parameters of the camera component 806 according to the ambient light intensity collected by the optical sensor 815.

[0223] The proximity sensor 816, also called a distance sensor, is usually disposed on the front panel of the terminal 800. The proximity sensor 816 is configured to collect a distance between

the user and the front of the terminal 800. In an example, when the proximity sensor 816 detects that the distance between the user and the front of the terminal 800 is gradually decreasing, the processor 801 controls the display screen 805 to switch from a screen-on state to a screen-off state; when the proximity sensor 816 detects that the distance between the user and the front of the terminal 800 is gradually increasing, the processor 801 controls the display screen 805 to switch from the screen-off state to the screen-on state.

[0224] Those skilled in the art will understand that the structure illustrated in FIG. 8 does not limit the terminal 800, which may include more or fewer components than illustrated in the drawings, or combine certain components, or adopt a different component arrangement.

[0225] Examples of the present application also provide a non-temporary computer-readable storage medium. Instructions in the storage medium, when executed by one or more processors of the terminal, enable the terminal to perform the module processing methods for the foreground and background separation system provided in the above examples.

[0226] Examples of the present application also provide a computer program product including instructions, which, when executed on a terminal, enables the terminal to perform the module processing methods for a foreground and background separation system provided in the above examples.

[0227] FIG. 9 is a schematic structure diagram of a server provided in an example of the present application. The background involved in the aforementioned examples may all be implemented by the structure illustrated in FIG. 9. The server may be a server in a background server cluster. Specifically,

[0228] the server 900 includes a central processing unit (CPU) 901, a system memory 904 including a random access memory (RAM) 902 and a read-only memory (ROM) 903, and a system bus 905 connecting the system memory 904 and the central processing unit 901. The server 900 further includes a basic input/output system (I/O system) 906 that helps transfer information between various components within the computer, and a mass storage device 907 for storing an operating system 913, application programs 914, and other program modules 915.

[0229] The basic input/output system 906 includes a displayer 908 for displaying information and an input device 909 such as a mouse and a keyboard for the user to input information. The displayer 908 and the input device 909 are both connected to the central processing unit 901 via an input/output controller 910 connected to the system bus 905. The basic input/output system 906 may further include an input/output controller 910 for receiving and processing the inputs from a variety of other devices such as the keyboard, the mouse, or an electronic

touchscreen pen. Similarly, the input/output controller 910 also provides its output to a display screen, a printer, or another type of output device.

[0230] The mass storage device 907 is connected to the central processing unit 901 through a mass storage controller (not shown) connected to the system bus 905. The mass storage device 907 and its associated computer-readable media provide a non-volatile storage for the server 900. That is, the mass storage device 907 may include a computer-readable medium (not shown) such as a hard disk or a CD-ROM drive.

[0231] Without loss of generality, the computer readable media may include computer storage media and communication media. The computer storage media includes volatile and nonvolatile, removable and non-removable media implemented in any method or technology for storing information such as computer readable instructions, data structures, program modules or other data. The computer storage media include RAM, ROM, EPROM, EEPROM, flash memory or other solid-state storage technology, CD-ROM, DVD or other optical storage, magnetic cassette, magnetic tape, disk storage or other magnetic storage devices. Of course, those skilled in the art will understand that the computer storage media are not limited to the above ones. The system memory 904 and the mass storage device 907 described above may be collectively referred to as a memory.

[0232] According to various examples of the present application, the server 900 may further be connected to and run on a remote computer on a network such as Internet. That is, the server 900 may be connected to the network 912 via a network interface unit 911 connected with the system bus 905, or be connected to other types of networks or remote computer systems (not shown) by adopting the network interface unit 911.

[0233] The above memory further includes one or more programs. The one or more programs are stored in the memory and configured to be executed by the CPU. The one or more programs include the module processing methods for the foreground and background separation system provided in the examples of the present application.

[0234] Examples of the present application also provide a non-temporary computer-readable storage medium. Instructions in the storage medium, when executed by one or more processors of the server, enable the server to perform the module processing methods for the foreground and background separation system provided in the above examples.

[0235] Examples of the present application also provide a computer program product including instructions, which, when executed on a server, enables the server to perform the module processing methods for a foreground and background separation system provided in

the above examples.

[0236] Persons of ordinary skill in the art may understand that the implementation of all or part of the steps in the above examples may be accomplished by hardware or by instructing related hardware through a program. The program may be stored in a computer-readable storage medium. The above-mentioned storage medium may be a read-only memory, a disk or an optical disk, etc.

[0237] The above are only some preferable examples of the present application, and are not intended to limit the present application. Any modifications, equivalent replacements, improvements, etc. made within the spirit and principles of the present application should be included in the protection scope of the present application.

CLAIMS

1. A module processing method for a foreground and background separation system, wherein the foreground and background separation system comprises a foreground and a background, the foreground displays an interactive interface that is configured to display a plurality of modules, and the background is configured with a data center that is configured to store module data of each of the plurality of modules, wherein the module data comprises a display manner of the module, function data of the module, and a logical relationship between the module and each of other modules;

comprising:

in response to a target operation on a target module, controlling, by the foreground and based on the target operation, the background to update data corresponding to the target module in the data center, and updating, by the foreground, the interactive interface in accordance with the updated data corresponding to the target module, wherein there are one or more target modules among the plurality of modules; and

in a case where an undo instruction for the target operation is detected, controlling, by the foreground and based on the undo instruction, the background to restore the data corresponding to the target module in the data center, and updating, by the foreground, the interactive interface in accordance with the restored data corresponding to the target module.

2. The method according to claim 1, wherein after controlling, by the foreground and based on the undo instruction, the background to restore the data corresponding to the target module in the data center, the method further comprises:

in a case where a redo instruction for the target operation is detected, controlling, by the foreground and based on the target operation, the background to update the data corresponding to the target module in the data center, and updating, by the foreground, the interactive interface in accordance with the updated data corresponding to the target module.

3. The method according to claim 1 or 2,

wherein controlling, by the foreground and based on the target operation, the background to update the data corresponding to the target module in the data center comprises:

sending, by the foreground, one or more first data update instructions to the background based on the target operation, wherein any one of the one or more first data update

instructions instructs the background to update a part of the data corresponding to the target module; and

generating, by the foreground, an operation record based on the sending of the one or more data update instructions; and

wherein controlling, by the foreground and based on the undo instruction, the background to restore the data corresponding to the target module in the data center comprises:

sending, by the foreground, one or more second data update instructions to the background based on an inverse operation of the target operation indicated by the operation record, wherein any one of the one or more second data update instructions instructs the background to restore the part of the data corresponding to the target module in the data center to a state before the target operation is performed.

4. The method according to claim 3, in a case where the target operation is an add operation on the target module,

wherein sending, by the foreground, the one or more first data update instructions to the background based on the target operation comprises:

sending, by the foreground, a first module update instruction firstly and a first logical relationship update instruction subsequently to the background, so as to control the background to add the module data of the target module into the data center firstly and add a logical relationship between the target module and each of the other modules into the module data of said other module subsequently; and

wherein sending, by the foreground, the one or more second data update instructions to the background based on the inverse operation of the target operation indicated by the operation record comprises:

sending, by the foreground, a second logical relationship update instruction firstly and a second module update instruction subsequently to the background, so as to control the background to delete, from the module data of each of the other modules in the data center, the logical relationship between the target module and said other module firstly and delete the module data of the target module from the data center subsequently.

5. The method according to claim 3, in a case where the target operation is a delete operation on the target module,

wherein sending, by the foreground, the one or more first data update instructions to the background based on the target operation comprises:

sending, by the foreground, a first logical relationship update instruction firstly and a first module update instruction subsequently to the background, so as to control the background to delete, from the module data of each of the other modules in the data center, a logical relationship between the target module and said other module firstly and delete the module data of the target module from the data center subsequently; and

wherein sending, by the foreground, the one or more second data update instructions to the background based on the inverse operation of the target operation indicated by the operation record comprises:

sending, by the foreground, a second module update instruction firstly and a second logical relationship update instruction subsequently to the background, so as to control the background to restore the module data of the target module in the data center firstly and restore the logical relationship between the target module and each of the other modules in the module data of said other module subsequently.

6. The method according to claim 5,

wherein before sending, by the foreground, the first logical relationship update instruction firstly and the first module update instruction subsequently to the background, the method further comprises:

controlling, by the foreground, the background to export and backup the data corresponding to the target module in the data center to a reference storage location, wherein the data corresponding to the target module comprises the module data of the target module and the logical relationship between the target module and each of the other modules in the module data of said other module; and

wherein after sending, by the foreground, the second module update instruction firstly and the second logical relationship update instruction subsequently to the background, the method further comprises:

importing, by the background, the module data of the target module from the reference storage location into the data center in response to the received second module update instruction, and importing, by the background, the logical relationship between the target module and each of the other modules from the reference storage location into the data center in response to the received second logical relationship update instruction.

7. The method according to claim 3, in a case where the target operation is a delete operation on the target module,

wherein sending, by the foreground, the one or more first data update instructions to the background based on the target operation comprises:

sending, by the foreground, a first logical relationship update instruction firstly and a first module update instruction subsequently to the background, so as to control the background to delete, when receiving the first logical relationship update instruction, a logical relationship between each of the other modules and the target module from the module data of said other module in the data center, and control the background to add, when receiving the first module update instruction, a deletion mark for the module data of the target module in the data center; and

wherein sending, by the foreground, the one or more second data update instructions to the background based on the inverse operation of the target operation indicated by the operation record comprises:

sending, by the foreground, a second module update instruction firstly and a second logical relationship update instruction subsequently to the background, so as to control the background to delete, when receiving the second module update instruction, the deletion mark for the module data of the target module in the data center, and control the background to add, when receiving the second logical relationship update instruction, the logical relationship between each of the other modules and the target module into the module data of said other module in the data center,.

8. The method according to any one of claims 1 to 3, wherein the logical relationship between the module and each of the other modules comprises a connection relationship between the module and each of the other modules, and a subscription relationship between the module and each of the other modules;

wherein the connection relationship indicates a connecting line relationship between the module and each of the other modules on the interactive interface, and

the subscription relationship indicates an association relationship between at least one of an internal parameter or an input of the module and an output of each of the other modules, and/or indicates an association relationship between an output of the module and at least one of an internal parameter or an input of each of the other modules.

9. The method according to claim 8, wherein in a case where there are multiple target modules, the foreground sends a plurality of first data update instructions to the background based on the target operation, which comprises:

sending, by the foreground, a first subscription relationship deleting instruction to the background, wherein the first subscription relationship deleting instruction instructs the background to delete subscription relationships between each of the other modules and one or more edge target modules from the module data of said other module, and wherein the edge target module is of the multiple target modules and has a logical relationship with said other module;

sending, by the foreground, a first connection relationship deleting instruction to the background, wherein the first connection relationship deleting instruction instructs the background to delete connection relationships between each of the other modules and the one or more edge target modules from the module data of said other module; and

sending, by the foreground, a first module deleting instruction to the background, wherein the first module deleting instruction instructs the background to delete the module data of the multiple target modules.

10. The method according to claim 8, wherein in a case where there are multiple target modules, the foreground sends a plurality of first data update instructions to the background based on the target operation, which comprises:

sending, by the foreground, a subscription relationship deleting instruction A to the background, wherein the subscription relationship deleting instruction A instructs the background to delete, for each of the multiple target modules, subscription relationships of a post-executed module to one or more pre-executed modules and the target module respectively, wherein the post-executed module is a module whose execution occurs behind the target module in a flowchart displayed on the foreground, and the pre-executed module is a module whose execution occurs before the target module in the flowchart;

sending, by the foreground, a subscription relationship deleting instruction B to the background, wherein the subscription relationship deleting instruction B instructs the background to delete subscription relationships between the multiple target modules;

sending, by the foreground, a subscription relationship deleting instruction C to the background, wherein the subscription relationship deleting instruction C instructs the

background to delete subscription relationships between each of the target modules and the one or more pre-executed modules;

sending, by the foreground, a connection relationship deleting instruction A to the background, wherein the connection relationship deleting instruction A instructs the background to delete a connection relationship between the post-executed module and each of the target modules;

sending, by the foreground, a connection relationship deleting instruction B to the background, wherein the connection relationship deleting instruction B instructs the background to delete connection relationships between the multiple target modules;

sending, by the foreground, a connection relationship deleting instruction C to the background, wherein the connection relationship deleting instruction C instructs the background to delete connection relationships between each of the target modules and the one or more pre-executed modules; and

sending, by the foreground, a first module deleting instruction to the background, wherein the first module deleting instruction instructs to delete remaining data of the module data of the multiple target modules.

11. A module processing apparatus for a foreground and background separation system, wherein the foreground and background separation system comprises a foreground and a background, the foreground displays an interactive interface that is configured to display a plurality of modules, and the background is configured with a data center that is configured to store module data of each of the plurality of modules, wherein the module data comprises a display manner of the module, function data of the module, and a logical relationship between the module and each of other modules;

comprising:

a first controlling module, configured to in response to a target operation on a target module, enable the foreground to control, based on the target operation, the background to update data corresponding to the target module in the data center, and enable the foreground to update the interactive interface in accordance with the updated data corresponding to the target module, wherein there are one or more target modules among the plurality of modules; and

a second controlling module, configured to in a case where an undo instruction for the target operation is detected, enable the foreground and to control, based on the undo instruction, the background to restore the data corresponding to the target module in the data center, and

enable the foreground to update the interactive interface in accordance with the restored data corresponding to the target module.

12. The apparatus according to claim 11, further comprising:

a third controlling module, configured to in a case where a redo instruction for the target operation is detected, enable the foreground to control, based on the target operation, the background to update the data corresponding to the target module in the data center, and enable the foreground to update the interactive interface in accordance with the updated data corresponding to the target module.

13. The apparatus according to claim 11 or 12,

wherein the first controlling module is configured to:

enable the foreground to send one or more first data update instructions to the background based on the target operation, wherein any one of the one or more first data update instructions instructs the background to update a part of the data corresponding to the target module; and

enable the foreground to generate an operation record based on the sending of the one or more data update instructions; and

wherein the second controlling module is configured to:

enable the foreground to send one or more second data update instructions to the background based on an inverse operation of the target operation indicated by the operation record, wherein any one of the one or more second data update instructions instructs the background to restore the part of the data corresponding to the target module in the data center to a state before the target operation is performed.

14. The apparatus according to claim 13, in a case where the target operation is an add operation on the target module,

wherein the first controlling module is configured to:

enable the foreground to send a first module update instruction firstly and a first logical relationship update instruction subsequently to the background, so as to control the background to add the module data of the target module into the data center firstly and add a logical relationship between the target module and each of the other modules into the module data of said other module subsequently; and

wherein the second controlling module is configured to:

enable the foreground to send a second logical relationship update instruction firstly and a second module update instruction subsequently to the background, so as to control the background to delete, from the module data of each of the other modules in the data center, the logical relationship between the target module and said other module firstly and delete the module data of the target module from the data center subsequently.

15. The apparatus according to claim 13, in a case where the target operation is a delete operation on the target module,

wherein the first controlling module is configured to:

enable the foreground to send a first logical relationship update instruction firstly and a first module update instruction subsequently to the background, so as to control the background to delete, from the module data of each of the other modules in the data center, a logical relationship between the target module and said other module firstly and delete the module data of the target module from the data center subsequently; and

wherein the second controlling module is configured to:

enable the foreground to send a second module update instruction firstly and a second logical relationship update instruction subsequently to the background, so as to control the background to restore the module data of the target module in the data center firstly and restore the logical relationship between the target module and each of the other modules in the module data of said other module subsequently.

16. The apparatus according to claim 13, further comprising:

an exporting module, configured to:

before the foreground sends the first logical relationship update instruction firstly and the first module update instruction subsequently to the background, enable the foreground to control the background to export and backup the data corresponding to the target module in the data center to a reference storage location, wherein the data corresponding to the target module comprises the module data of the target module and the logical relationship between the target module and each of the other modules in the module data of said other module; and

an importing module that is configured to

after the foreground sends the second module update instruction firstly and the second logical relationship update instruction subsequently, enable the background to import the

module data of the target module from the reference storage location into the data center in response to the received second module update instruction, and enable the background to import the logical relationship between the target module and each of the other modules from the reference storage location into the data center in response to the received second logical relationship update instruction.

17. The apparatus according to claim 13, in a case where the target operation is a delete operation on the target module,

wherein the first controlling module is configured to:

enable the foreground to send a first logical relationship update instruction firstly and a first module update instruction subsequently to the background, so as to control the background to delete, when receiving the first logical relationship update instruction, a logical relationship between each of the other modules and the target module from the module data of said other module in the data center, and control the background to add, when receiving the first module update instruction, a deletion mark for the module data of the target module in the data center; and

wherein the second controlling module is configured to:

enable the foreground to send a second module update instruction firstly and a second logical relationship update instruction subsequently to the background, so as to control the background to delete, when receiving the second module update instruction, the deletion mark for the module data of the target module in the data center, and control the background to add, when receiving the second logical relationship update instruction, the logical relationship between each of the other modules and the target module into the module data of said other module in the data center.

18. The apparatus according to any one of claims 11 to 13, wherein the logical relationship between the module and each of other modules comprises a connection relationship between the module and each of the other modules, and a subscription relationship between the module and each of the other modules;

wherein the connection relationship indicates a connecting line relationship between the module and each of the other modules on the interactive interface, and

the subscription relationship indicates an association relationship between at least one of an internal parameter or an input of the module and an output of each of the other modules,

and/or indicates an association relationship between an output of the module and at least one of an internal parameter or an input of each of the other modules.

19. The apparatus according to claim 18, wherein in a case where there are multiple target modules, the first controlling module is configured to:

enable the foreground to send a first subscription relationship deleting instruction to the background, wherein the first subscription relationship deleting instruction instructs the background to delete subscription relationships between each of the other modules and one or more edge target modules from the module data of said other module, and wherein the edge target module is of the multiple target modules and has a logical relationship with said other module;

enable the foreground to send a first connection relationship deleting instruction to the background, wherein the first connection relationship deleting instruction instructs the background to delete connection relationships between each of the other modules and one or more edge target modules from the module data of said other module; and

enable the foreground to send a first module deleting instruction to the background, wherein the first module deleting instruction instructs the background to delete the module data of the multiple target modules.

20. The apparatus according to claim 18, wherein in a case where there are multiple target modules, the first controlling module is configured to:

enable the foreground to send a subscription relationship deleting instruction A to the background, wherein the subscription relationship deleting instruction A instructs the background to delete, for each of the multiple target module, subscription relationships of a post-executed module to one or more pre-executed modules and the target module respectively, wherein the post-executed module is a module whose execution occurs behind the target module in a flowchart displayed on the foreground, and the pre-executed module is a module whose execution occurs before the target module in the flowchart;

enable the foreground to send a subscription relationship deleting instruction B to the background, where the subscription relationship deleting instruction B instructs the background to delete subscription relationships between the multiple target modules;

enable the foreground to send a subscription relationship deleting instruction C to the background, where the subscription relationship deleting instruction C instructs the background

to delete subscription relationships between each of the target modules and the one or more pre-executed modules;

enable the foreground to send a connection relationship deleting instruction A to the background, wherein the connection relationship deleting instruction A instructs the background to delete a connection relationship between the post-executed module and each of the target modules;

enable the foreground to send a connection relationship deleting instruction B to the background, where the connection relationship deleting instruction B instructs the background to delete connection relationships between the multiple target modules;

enable the foreground to send a connection relationship deleting instruction C to the background, where the connection relationship deleting instruction C instructs the background to delete connection relationships between each of the target modules and the one or more pre-executed modules; and

enable the foreground to send a first module deleting instruction to the background, wherein the first module deleting instruction instructs to delete remaining data of the module data of the multiple target modules.

21. A module processing apparatus for a foreground and background separation system, comprising:

one or more processors; and

one or more memories for storing instructions executable by the one or more processors;

wherein the one or more processors are configured to perform the steps of the method according to any one of claims 1 to 10.

22. A non-volatile computer-readable storage medium storing instructions, wherein the instructions, when executed by one or more processors, implement the steps of the method according to any one of claims 1 to 10.

23. A computer program product comprising instructions, wherein the instructions, when running on a computer, cause the computer to execute the steps of the method according to any one of claims 1 to 10.