TASK PROCESSING METHOD, APPARATUS, DEVICE AND SYSTEM, AND STORAGE MEDIUM

CROSS-REFERENCE TO RELATED APPLICATION

[0001] This patent application claims priority to Chinese patent application No. 202011160128.6 filed on October 27, 2020 and entitled "TASK PROCESSING METHOD, APPARATUS, DEVICE AND SYSTEM, AND STORAGE MEDIUM", the disclosure of which is hereby incorporated by reference in its entirety.

TECHNICAL FIELD

[0002] The disclosure relates to the field of Internet technologies, and in particular, to a task processing method, apparatus, device and system, and a storage medium.

BACKGROUND

[0003] As rapid development of the Internet techniques, on-line programming education is more and more popularized, for example, on-line artificial intelligence (AI) education. In related technologies, only programing source data resources such as some cases are generally provided, and a user needs to voluntarily configure a running environment (i.e., a running resource).

SUMMARY

[0004] The disclosure provides a task processing method, apparatus, device and system, and a storage medium. The technical solutions of the disclosure are as follows.

[0005] According to an aspect of embodiments of the disclosure, a task processing method is provided, including: receiving a resource schedule request of a target task sent by a terminal, the resource schedule request including source data configuration information and running configuration information of the target task; determining task source data of the target task according to the source data configuration information; creating and starting a target container in a target resource node according to the running configuration information; determining a route entry according to the target container and sending the task source data to the target resource node; and sending a resource schedule completion message to the terminal, so that the terminal calls the target container and the task source data to execute the target task through the route entry.

[0006] According to another aspect of the embodiments of the disclosure, a task processing system is further provided, including a terminal and a server.

[0007] The terminal is configured to send a resource schedule request of a target task to a server, the resource schedule request including source data configuration information and running configuration information of the target task, and call a target container and task source data to execute the target task through a route entry.

[0008] The server is configured to determine task source data of the target task according to the source data configuration information, create and start a target container in a target resource node according to the running configuration information, determine a route entry according to the target container and send the task source data to the target source node, and send a resource schedule completion message to the terminal.

[0009] According to another aspect of the embodiments of the disclosure, a task processing apparatus is further provided, including the following parts.

[0010] A resource schedule request receiving part is configured to receive a resource schedule request of a target task sent by a terminal, the resource schedule request including source data configuration information and running configuration information of the target task.

[0011] A task source data determining part is configured to determine task source data of the target task according to the source data configuration information.

[0012] A container creating and starting part is configured to create and start a target container in a target resource node according to the running configuration information.

[0013] A route entry determining part is configured to determine a route entry according to the target container.

[0014] A task source data sending part is configured to send the task source data to the target resource node.

[0015] A resource schedule completion message sending part is configured to send a resource schedule completion message to the terminal, so that the terminal calls the target container and the task source data to execute the target task through the route entry.

[0016] According to another aspect of the embodiments of the disclosure, a task processing apparatus is further provided, including the following parts.

[0017] A resource schedule request sending part is configured to send a resource schedule request of a target task to a server, the resource schedule request including source data configuration information and running configuration information of the target task, so that the server determines task source data of the target task according to the source data configuration information; create and start a target container in a target resource node according to the running configuration information; and determine a route entry according to the target container.

[0018] A resource schedule completion message receiving part, configured to receive a resource schedule completion message sent by the server.

[0019] A target task executing part, configured to call the target container and the task source data to execute the target task through the route entry.

[0020] According to another aspect of the embodiments of the disclosure, a task processing device is provided, including a processor and a memory configure to store processor-executable instructions. The processor is configured to execute the instructions to implement the task processing method above.

[0021] According to another aspect of the embodiments of the disclosure, a computer-readable storage medium is provided. When executed by a processor of a task processing device, instructions in the storage medium cause the task processing device to execute the task processing method of the embodiments of the disclosure above.

[0022] According to another aspect of the embodiments of the disclosure, a computer program is provided, including a computer-readable code. When running in a task processing device, the computer-readable code is executed by a processor in the task processing device to execute the task processing method above.

[0023] In the embodiments of the disclosure, a user can directly select source data configuration information and running configuration information through the terminal, to implement flexible and simple resource configurations, which does not need running environment construction by the user and does not need resource data configuration by the user, either, so as to greatly reduce the complexity and error rate of the resource

configuration of the on-line programming education, thereby further greatly improving flexibility and task processing speed in the task processing process.

[0024] It should be understood that the above general description and the following detailed description are only exemplary and explanatory, and cannot limit the disclosure.

BRIEF DESCRIPTION OF THE DRAWINGS

[0025] To describe the technical solutions in the embodiments of the disclosure more clearly, the accompanying drawings required for describing the embodiments are briefly described hereinafter. Apparently, the accompanying drawings in the following description show merely some embodiments of the disclosure, and a person of ordinary skill in the art may obtain other accompanying drawings from these accompanying drawings without creative efforts.

[0026] FIG. 1 is a schematic diagram I of an application environment according to an embodiment of the disclosure.

[0027] FIG. 2 is a schematic diagram II of an application environment according to an embodiment of the disclosure.

[0028] FIG. 3 is a flowchart I of a task processing method according to an embodiment of the disclosure.

[0029] FIG. 4 is a flowchart of creating and starting, by a resource arrangement server, a target container in a target resource node according to running configuration information according to an embodiment of the disclosure.

[0030] FIG. 5 is a flowchart of idle resource recovery according to an embodiment of the disclosure.

[0031] FIG. 6 is a flowchart of recommended running configuration information according to an embodiment of the disclosure.

[0032] FIG. 7 is a flowchart II of a task processing method according to an embodiment of the disclosure.

[0033] FIG. 8 is a schematic structural diagram of a resource schedule system according to an embodiment of the disclosure.

[0034] FIG. 9 is a schematic diagram of a data stream of a task processing method according to an embodiment of the disclosure.

[0035] FIG. 10 is a block diagram I of a task processing apparatus according to an embodiment of the disclosure.

[0036] FIG. 11 is a block diagram II of a task processing apparatus according to an embodiment of the disclosure.

[0037] FIG. 12 is a block diagram I of an electronic device for task processing according to an embodiment of the disclosure.

[0038] FIG. 13 is a block diagram II of an electronic device for task processing according to an embodiment of the disclosure.

DETAILED DESCRIPTION

[0039] To make persons having ordinary skill in the art understand the technical solutions of the disclosure better, the following describes the technical solutions in the embodiments of the disclosure with reference to the accompanying drawings more clearly and completely.

[0040] It is to be explained that the terms such as "first" and "second" in this specification, the claims, and the foregoing accompanying drawings of the disclosure are intended to distinguish similar objects rather than describing a particular sequence or a

precedence order. It is to be understood that data used in such a manner can be exchanged in a proper situation so as to facilitate the described embodiments of the disclosure to be implemented according to sequences in addition to those shown or described herein. The implementations described in the exemplary embodiments below do not represent all implementations consistent with the disclosure. Instead, the implementations are only examples of apparatuses and methods consistent with some aspects of the disclosure as detailed in the appended claims.

[0041] As rapid development of the Internet techniques, on-line programming education is more and more popularized, for example, on-line AI education. However, the AI development environment is more complicated to build, involving a graphics driver of a deep learning framework, a Compute Unified Device Architecture (CUDA), and a core of a main chip graphics card of a deep neural network, i.e., a Graphics Processing Unit (GPU) acceleration library CUDNN (CUDA Deep Neural Network) and other dependent libraries installation, the process is cumbersome and error-prone.

[0042] Based on the problems above, related technologies provide a development environment for AI learning, which can be applied to on-line programming education platforms, but generally has the following problems.

[0043] (1) Generally, only one type of deep learning framework is provided, and the user does not have much choice.

[0044] (2) There are some educational scenarios that do not provide hashrate part, but only provide cases, and the user configures the operating environment.

[0045] (3) There are only a few options for the provided hashrate part, which cannot be flexibly configured and lacks flexibility.

[0046] (4) The platform is integrated with user management and project management, and is not suitable for third-party integration.

[0047] In view of the above, on-line programming education platforms in related technologies have limited options for configuring operating environments or frameworks and other operating resources, and the hashrate configuration is not flexible enough to meet the current diversified on-line AI programming education needs.

[0048] FIG. 1 is a schematic diagram I of an application environment according to an embodiment of the disclosure. As shown in FIG. 1, the application environment may include a terminal 100 and a server 200.

[0049] In an embodiment, the terminal 100 can provide a task resource schedule configuration page, for the user to customize source data configuration information and running configuration information for the target task according to needs, and can send to the server 200 a resource schedule request of the target task reflecting user demands (the resource schedule request including the source data configuration information and the running configuration information).

[0050] The terminal 100 may include, but not limited to, a smart phone, a desktop computer, a tablet, a laptop, a smart speaker, a personal digital assistant, an augmented reality (AR)/virtual reality (VR) device, a smart wearable device, and other types of entity devices. It may also be software running on an entity device, for example, an application program, a browser, and the like. Optionally, the operating system running on the entity device may include, but not limited to, an Android system, an IOS Input Output System (IOS), Linux, windows, and the like.

[0051] In some embodiments of the disclosure, the server 200 can perform resource scheduling on a target task based on a resource schedule request on the target task reflecting user demands sent by a terminal. In the embodiments of the disclosure, the

scheduled resources may include running resources and source data resources. The source data resource may be pre-stored in the server, and accordingly, the server may schedule the corresponding source data resource in combination with the source data configuration information in the resource schedule request. The running resource can be provided by the server 200, and can be combined with the running configuration information in the resource schedule request here to schedule the corresponding running resource for the target task.

[0052] The server 200 may be an independent physical server, may be a server cluster or a distributed system constituted by a plurality of physical servers, and may further include a cloud server or a cloud server cluster.

[0053] In addition, it should be noted that FIG. 1 shows a possible application environment of the task processing method provided by the disclosure. In actual applications, other application environments may also be included. For example, running resources may be provided by a resource node cluster 300. Correspondingly, FIG. 2 is a schematic diagram II of an application environment according to an embodiment of the disclosure. As shown in FIG. 2, in an embodiment, the resource node cluster 300 may include a resource arrangement server 301 and a resource node cluster 302. The resource arrangement server 301 may allocate corresponding resources to the user from the resource node cluster 302 in accordance with user demands. The resource node cluster 302 may be configured to execute the target task.

[0054] Herein, the resource arrangement server 301 may include an independent physical server, may also include a server cluster or a distributed system constituted by a plurality of physical servers, and may further include a cloud server or a cloud server cluster. The server 302 may include a server cluster or a distributed system constituted by a plurality of physical servers, and may further include a cloud server cluster.

[0055] In the embodiment of the disclosure, the terminal 100, the server 200, and the resource node cluster 300 may be directly or indirectly connected in a wired or wireless communication mode, which is not limited in the disclosure.

[0056] The following describes a task processing method of the disclosure. FIG. 3 is a flowchart I of a task processing method according to an embodiment of the disclosure. This specification provides operation steps of the method described in the following embodiments or flowcharts. However, more or fewer operation steps may be included based on conventional or non-creative work. The sequence of steps listed in the embodiments may be one of many execution sequences of steps, and does not represent the unique execution sequence. When executed the actual systems or server products may be executed sequentially or concurrently according to the methods shown in the embodiments or the drawings (for example, a parallel processor or a multi-threaded processing environment). Specifically, as shown in FIG. 3, the task processing method may include the following steps.

[0057] S301: a terminal sends a resource schedule request of a target task to a server.

[0058] In the embodiments of the disclosure, the resource schedule request may include source data configuration information and running configuration information of the target task. Herein, the user can provide a task resource schedule configuration page on the terminal, customize the source data configuration information and running configuration information of the target task according to own demands, and trigger the terminal to send a resource schedule request including the source data configuration information and the running configuration information to the server through a Hypertext Transfer Protocol (HTTP). In the embodiments of the disclosure, the target task may be

any programming learning task that needs to be processed by using scheduled resources. In the embodiments of the disclosure, there may be one or more types of tasks in combination with actual application requirements. In a specific embodiment, the types of tasks may include, but are not limited to, deep learning types, test types, and the like.

[0059] Herein, the running configuration information may include the resource configuration for executing the target task. In an implementation of the embodiments of the disclosure, the running configuration information may include a container mirror image and hashrate information.

[0060] Herein, the container mirror image may include data required to create and run the target container (the target container is a container that performs task processing). In an implementation of the embodiments of the disclosure, the container mirror image may include, but is not limited to, files such as programs, libraries, resources, and configurations required by the container to run.

[0061] Herein, the hashrate information may include hashrate for executing the target task. In an implementation of the embodiments of the disclosure, the hashrate information may include, but is not limited to, performance parameters of a Central Processing Unit (CPU), an internal memory, and a Graphics Processing Unit (GPU).

[0062] The source data configuration information may include source data information needed for implementing the target task. In an embodiment, in a case that the target task may be a deep learning task (i.e., a task of a deep learning type), the source data configuration information herein may be the type of images in a training data set, the number of images in the training data set, and structure information of the neural network, and at the same time, the images in the training data set may be images with annotation information. In another embodiment, in a case that the target task is a test task, the source data configuration information herein may be identification information of test data and identification information of a test program.

[0063] In practical applications, different target tasks correspond to different running configuration information and source data configuration information, and the user can set same in the terminal according to actual needs.

[0064] S303: the server determines task source data of the target task according to source data configuration information.

[0065] In the embodiments of the disclosure, the server may pre-store source data for processing a variety of tasks. In a case that the target task is a deep learning task, task source data of the target task includes a training data set corresponding to the source data configuration information and a neural network corresponding to the source data configuration information.

[0066] In a case that the source data configuration information is the type of an image in the training data set, the number of images in the training data set, and the structure information of the neural network, the number of images of the above type can be selected according to the type of image and the number of images as the training data set. The neural network corresponding to the structure information is selected according to the structure information of the neural network. The selected training data set and the neural network are used as the task source data of the target task.

[0067] In a case that the source data configuration information may be the identification information of the test data and the identification information of the test program, the corresponding test data and test program can be selected according to the identification information of the test data and the identification information of the test

program respectively. The selected test data and the test program are used as the task source data of the target task.

[0068] S305: the server creates and starts a target container in a target resource node according to running configuration information.

[0069] In the embodiments of the disclosure, the creating and starting, by the server, a target container in the target resource node according to the running configuration information includes the following step. The server sends a resource allocation request to a resource arrangement server, the resource allocation request including the running configuration information. The resource arrangement server creates and starts the target container in the target resource node based on the running configuration information.

[0070] In an implementation of the embodiments of the disclosure, the sending a resource allocation request to a resource arrangement server includes the following steps.

[0071] A resource schedule task including running configuration information is sent to the resource scheduling server through a message queue.

[0072] The resource scheduling server sends the resource allocation request to the resource arrangement server, the resource allocation request including the running configuration information.

[0073] Herein, in the process of task resource scheduling management, two servers (the resource management server and the resource scheduling server) can be adopted to joint scheduling management. The resource management server receives resource schedule requests from a plurality of terminals and puts the plurality of resource schedule requests into the message queue in the form of resource schedule tasks. Then the resource scheduling server takes out the resource schedule tasks from the message queue in turn to schedule running resources, which can realize the decoupling of scheduling and requests, greatly improving the availability and stability of resource scheduling.

[0074] The server may be provided with a process generation module. Correspondingly, the process generation module may establish communication with the resource arrangement server, and send a resource allocation request including running configuration information to a resource arrangement server.

[0075] In an implementation of the embodiments of the disclosure, FIG. 4 is a flowchart of creating and starting, by a resource arrangement server, a target container in a target resource node based on the running configuration information according to an embodiment of the disclosure. As shown in FIG. 4, the creating and starting, by a resource arrangement server, a target container in a target resource node based on the running configuration information may include the following steps.

[0076] S3051: the resource arrangement server determines a target resource node from a resource node cluster according to hashrate information.

[0077] S3053: the resource arrangement server sends a container creating and starting instruction to the target resource node, the container creating and starting instruction including a container mirror image.

[0078] S3055: the target resource node uses the container mirror image to locally create and start the target container.

[0079] In the embodiments of the disclosure, the servers may also be a server cluster. The server cluster may include a main server and a resource service cluster. The creating and starting, by the server, the target container in the target resource node according to the running configuration information may include the following steps.

[0080] The main server can determine the target resource node from the resource service cluster according to the hashrate information. The main server sends the container

creating and starting instruction to the target resource node, the container creating and starting instruction including the container mirror image. The target resource node uses the container mirror image to locally create and start the target container.

[0081] Herein, in conjunction with the hashrate information in the running configuration information, a resource node with idle hashrate meeting the hashrate required for processing the target task can be selected as the target resource node. The target resource node combines with the container mirror image to locally create and start the target container in the target resource node to realize the configuration of the running resources of the target task. In conjunction with the container mirror image and hashrate information, a container that runs the target task separately is created and started, which can achieve the balance of resource usage requirements among different target tasks.

[0082] S307: the server determines a route entry according to the target container.

[0083] In an implementation of the embodiments of the disclosure, taking the application scenario where the server includes a main server and a resource service cluster as an example, the main server can obtain the route entry of the target container and add the route entry locally.

[0084] The route entry of the target container may be path information pointing to the network address of the target container. Herein, the route entry may include the destination Internet Protocol (IP), that is, the IP address of the target container, the network mask (identifying the address of a network segment where the target container is located together with the destination IP), the next-hop address (the next-hop IP address of the route entry) and the source IP (that is, the IP address of the terminal).

[0085] In an implementation of the embodiments of the disclosure, taking the above application scenario combining the resource arrangement server and the resource node cluster as an example, determining, by the server, the route entry according to the target container may include the following steps. The resource arrangement server obtains the route entry of the target container. The resource arrangement server sends a container startup success message to the server, the container startup success message including a route entry. The server sends the route entry to a route management server. The route management server adds the route entry locally.

[0086] In the embodiments of the disclosure, after the target container is created and started, the resource arrangement server may obtain the route entry of the target container, and feed back the container startup success message including the route entry to the server.

[0087] Herein, the server can be provided with a route proxy module. Accordingly, the route entry of the target container can be obtained through the route proxy module, and the route entry of the target container can be sent to the route management server, so that the route management server can add the route entry of the target container.

[0088] In an implementation of the embodiments of the disclosure, the route entry of the target container is managed by the route management server, and the subsequent terminal can directly control the target container to perform task running through the route management server, so as to realize the decoupling between the terminal and the server. User management and resource scheduling management are independent of each other, even the user can also control the task running through a third-party account, which greatly improves the flexibility and task processing speed in the task processing process.

[0089] S309: the server sends the task source data to the target resource node.

[0090] In an implementation of the embodiments of the disclosure, after the target container is created and started, the server can obtain address information of the target resource node, and send task source data to the target resource node according to the address information.

[0091] In an implementation of the embodiments of the disclosure, after the target container is created and started, the resource arrangement server can also obtain the address information of the target resource node. Accordingly, the container startup success message sent by the resource arrangement server may also carry the address information of the target resource node. The server sends the task source data to the target resource node according to the address information.

[0092] The address information of the target resource node may be a network address, which is a logical address of a node on the Internet in the network. The addressing of the node can be realized based on the network address.

[0093] In the embodiments of the disclosure, the server establishes communication with the target resource node according to the address information of the target resource node, and then synchronizes the task source data to the target resource node to realize the configuration of the source data resource of the task. In practical applications, the target resource node may be provided with an operation and maintenance service module. Correspondingly, the operation and maintenance service module may communicate with the server to realize synchronization of the task source data and provide source data support for the subsequent target container to execute the target task.

[0094] S311: the server sends a resource schedule completion message to the terminal.

[0095] In the embodiment of the disclosure, after the running resource and the source data resource are scheduled, and the route configuration of the target container is realized, a resource schedule completion message can be sent to the terminal, so that the user can trigger a task running request.

[0096] In one embodiment, the server can send resource scheduling progress synchronization information of the target task to the terminal in real time. Correspondingly, the terminal can display a resource scheduling progress interface, and update the resource scheduling progress synchronization information in the resource scheduling progress interface according to the resource scheduling progress synchronization information fed back by the server. In an embodiment, the resource scheduling progress synchronization information may include, but is not limited to, container startup progress information, data synchronization progress information, and the like.

[0097] Herein, after the terminal sends a resource schedule request to the server, the resource scheduling progress synchronization information in the resource scheduling progress interface displayed by the terminal is in resource scheduling. Herein, after receiving the container startup success message, the server can feed back the container startup success information to the terminal. Correspondingly, the resource scheduling progress synchronization information in the resource scheduling progress interface displayed by the terminal can be in container startup success and data packet synchronization. Herein, after the server sends a resource schedule completion message to the terminal, the resource scheduling progress synchronization information in the resource scheduling progress interface displayed by the terminal may be completion of container startup success and data packet synchronization.

[0098] In an implementation of the embodiments of the disclosure, by sending the resource scheduling progress synchronization information of the target task to the

terminal in real time, the user can be informed about the resource scheduling progress timely, effectively improving the user experience.

[0099] S313: the terminal calls the target container and the task source data to execute the target task through the route entry.

[00100] In the embodiments of the disclosure, the calling, by the terminal, the target container and the task source data to execute the target task through the route entry may include the following steps. The terminal sends a task running request of the target task to a route management server. The route management server sends the task running request to the target container based on the route entry. The target container runs the target task based on the task source data in response to the task running request.

[00101] Herein, after the terminal receives the resource schedule completion message, the task running acknowledgment page may be displayed. The task running acknowledgment page may be provided with a corresponding trigger area, such as a trigger button. Correspondingly, the user can trigger the terminal to send the task running request of the target task to the route management server through the trigger button.

[00102] In practical applications, the task running request can carry the identification information of the target task. Correspondingly, the route entry of the target container allocated to each target task can also correspond to the identification information of the target task. Correspondingly, the route management server can determine the corresponding route entry by combining the identification information of the target task based on the received task running request of a target task, and then forward the task running request to the corresponding target container based on the route entry.

[00103] In the embodiments of the disclosure, the target container can call corresponding task source data from the target resource node where it is located in response to the task running request, and then run the target task.

[00104] In the embodiments of the disclosure, the route entry of the target container is added to the route management server. In the task running process, the terminal can directly control the target container to perform task running through the route management server, so as to decouple the terminal from the server, so that user management and resource scheduling management are independent of each other, even the user can also control the task running through a third-party account to realize third-party integration, which greatly improves the flexibility and task processing speed in the task processing process.

[00105] As can be seen from the technical solution provided by the embodiments of the disclosure, the user in the embodiments of the disclosure can directly select source data configuration information and running configuration information through the terminal, to implement flexible and simple resource configurations, which does not need running environment construction by the user and does not need resource data configuration by the user, either, so as to greatly reduce the complexity and error rate of the resource configuration of the on-line programming education, thereby further greatly improving flexibility and task processing speed in the task processing process. Moreover, in the running resource configuration process, the route entry of the target container is added to the route management server. In the task running process, the terminal can directly control the target container to perform task running through the route management server, to decouple the terminal from the server, so that user management and resource scheduling management are independent of each other, even the user can also control the task running through a third-party account to realize third-party integration, which greatly improves the flexibility and task processing speed in the task processing process.

[00106] In an implementation of the embodiments of the disclosure, the calling, by the terminal, the target container and the task source data to execute the target task through the route entry may include the following steps. The terminal sends a task running request of the target task to a route management server. The server sends the task running request to the target container based on the route entry. The target container runs the target task based on the task source data in response to the task running request.

[00107] For the specific detailed steps of the foregoing embodiment that combines the server forwarding the task running request to implement the target task execution, reference is made to the specific detailed steps of the foregoing embodiment that combines the routing management server forwarding the task running request to implement the target task execution, and the details are not repeated here.

[00108] As can be seen from the technical solution provided by the embodiments of the disclosure, the user in the embodiments of the disclosure can directly select source data configuration information and running configuration information through the terminal, to implement flexible and simple resource configurations, which does not need running environment construction by the user and does not need resource data configuration by the user, either, so as to greatly reduce the complexity and error rate of the resource configuration of the on-line programming education, thereby further greatly improving flexibility and task processing speed in the task processing process.

[00109] In practical applications, there is a problem that the container does not run tasks after it is started, resulting in a waste of running resources. In some embodiments, a solution for providing idle resource recovery can also be provided. FIG. 5 is a flowchart of idle resource recovery according to an embodiment of the disclosure, which specifically includes the following steps.

[00110] S501: the server monitors a running state of the target container.

[00111] S503: in a case of monitoring that the running state of the target container is an idle state and duration of the idle state is greater than or equal to a preset idle duration threshold, the server sends a resource destruction instruction to the resource arrangement server.

[00112] S505: the resource arrangement server sends an idle recovery instruction to the target resource node.

[00113] S507: the target resource node deletes the target container.

[00114] In the embodiments of the disclosure, the idle state may be that the target container starts but does not run tasks. The preset idle duration threshold can be set according to actual application requirements.

[00115] In some embodiments of the disclosure, the resource arrangement server may periodically report the running state of the target container to the server, so that the server can grasp the running state of the target container in time. The server can be provided with an idle recovery module. Correspondingly, the idle recovery module can be configured to monitor the running state of the target container. In a case of that the idle recovery module monitors that the running state of the target container is an idle state and a duration of the idle state is greater than or equal to a preset idle duration threshold, the idle recovery module may send a resource destruction instruction to the resource arrangement server through the corresponding application program interface.

[00116] In some embodiments of the disclosure, the idle resource recovery may include the following steps. The server monitors the running state of the target container. In a case of monitoring that the running state of the target container is an idle state and duration of the idle state is greater than or equal to a preset idle duration threshold, the

server sends an idle recovery instruction to the target resource node. The target resource node deletes the target container.

[00117] In the embodiments of the disclosure, by monitoring the running state of the target container, idle resources can be recovered in time, which effectively avoids resource waste, and further improves the availability of the resource cluster.

[00118] In some embodiments of the disclosure, before the terminal sends the resource schedule request of the target task to the server, the running configuration information can be recommended in combination with the source data configuration information inputted by the user. FIG. 6 is a flowchart of recommended running configuration information according to an embodiment of the disclosure, which specifically includes the following steps.

[00119] S601: in a case of monitoring, by the terminal, that source data configuration information is inputted in a task resource schedule configuration page, a resource configuration recommendation request is sent to the server, the resource configuration recommendation request including the inputted source data configuration information.

[00120] S603: the server determines recommended running configuration information according to the inputted source data configuration information.

[00121] S605: the server sends the recommended running configuration information to the terminal.

[00122] S607: the terminal displays a resource configuration acknowledgment message on the task resource schedule configuration page, the resource configuration acknowledgment message including the recommended running configuration information.

[00123] Correspondingly, the sending, by the terminal, the resource schedule request of the target task to the server may include the following step. The terminal sends the resource schedule request of the target task to the server in response to an acknowledgment instruction for adopting the recommended running configuration information.

[00124] In some embodiments of the disclosure, the user may not use the recommended running configuration information. Correspondingly, the resource configuration acknowledgment message may include a trigger acknowledgment area that represents the use of the recommended running configuration information and a trigger cancel area that represents not use the recommended running configuration information. Herein, the user can click the trigger acknowledgment area to trigger the acknowledgment instruction using the recommended running configuration information. Herein, the user can also click the trigger cancel area to trigger the acknowledgment instruction not using the recommended running configuration information.

[00125] In the embodiments of the disclosure, by combining the source data configuration information inputted by the user, the server recommends the running configuration information, which can greatly improve the accuracy of running resource configuration and avoid resource waste caused by excessively high running resources requested by the user, or problems such as failure or low efficiency of subsequent task processing caused by excessively low resources requested by the user.

[00126] In some embodiments of the disclosure, after sending, by the server, the task source data to the target resource node according to the address information, the method may further include the following step. The target resource node sends resource usage information of the target container to the server.

[00127] Correspondingly, the resource schedule completion message may further include the resource usage information of the target container. The method may further include the following step. The terminal displays a task running acknowledgment page including the resource usage information.

[00128] In the embodiments of the disclosure, the resource usage information may be the proportion of the used running resources in the requested running resources. Here, after the resource scheduling is completed, the resource usage information of the target container can be fed back to the terminal, so that the user can grasp the usage of running resources.

[00129] In some embodiments of the disclosure, after sending, by the target resource node, the resource usage information of the target container to the server, the method may further include the following steps. The server performs rationality analysis on the running resources of the target task according to the resource usage information. If an analysis result is unreasonable, the server sends restart task schedule information to the terminal.

[00130] Correspondingly, the task running acknowledgment page may further include restart task schedule information.

[00131] Correspondingly, the method may further include the following steps. The terminal displays a task resource schedule configuration page in response to a restart instruction based on the restart task schedule information.

[00132] The terminal sends a resource recovery instruction to the server.

[00133] The server executes a resource recovery operation in response to the resource recovery instruction.

[00134] In some embodiments of the disclosure, the performing, by the server, a recovery operation may include the following step. A resource destruction instruction is sent to the resource arrangement server, so that the resource arrangement server sends an idle recovery instruction to the target resource node, and accordingly, the target resource node deletes the target container.

[00135] In some embodiments of the disclosure, the performing, by the server, a recovery operation may include the following step. The server sends an idle recovery instruction to the target resource node, and accordingly, the target resource node deletes the target container.

[00136] In the embodiment of the disclosure, the restart task schedule information may include acknowledgment information about whether to restart the task scheduling.

[00137] In some embodiments of the disclosure, a running resource proportion threshold can be determined by taking completed resource scheduling as a node and combining the proportions of the running resources that are used by the node to the requested running resources with many tasks. For example, the average value of the proportions of the running resources that are used by the node to the requested running resources with a large number of tasks is the running resource ratio threshold, in a case that an absolute value of the difference between the ratio of the used running resources of the target task to the requested running resources (resource usage information) and the running resource proportion threshold is greater than a certain threshold, it is determined to be unreasonable, and the server can send the restart task schedule information to the terminal. Otherwise, it can be determined to be reasonable, and the server can only send resource usage information to the terminal.

[00138] In the embodiments of the disclosure, if the user triggers an instruction to the restart task scheduling, the terminal can display the task resource schedule configuration

page. If the user triggers an instruction to cancel the restart task scheduling, the terminal can send a task running request to the route management server.

[00139] In the embodiments of the disclosure, rationality analysis is performed on the scheduled running resources according to the resource usage information after the resource scheduling is completed, which can greatly improve the accuracy of running resource configuration and avoid resource waste caused by excessively high running resources requested by the user, or problems such as failure or low efficiency of subsequent task processing caused by excessively low resources requested by the user.

[00140] The following introduces a task processing method provided by an embodiment of the disclosure with the server as an executive subject, which specifically includes the following steps.

[00141] A resource schedule request of a target task sent by a terminal is received, the resource schedule request including source data configuration information and running configuration information of the target task.

[00142] Task source data of the target task is determined according to the source data configuration information.

[00143] A target container is created and started in a target resource node according to the running configuration information.

[00144] A route entry is determined according to the target container and the task source data is sent to the target resource node.

[00145] A resource schedule completion message is sent to the terminal, so that the terminal calls the target container and the task source data to execute the target task through the route entry.

[00146] In some embodiments of the disclosure, the creating and starting the target container in the target resource node according to the running configuration information includes the following step.

[00147] A resource allocation request is sent to a resource arrangement server, the resource allocation request including the running configuration information, so that the resource arrangement server creates and starts the target container in the target resource node based on the running configuration information.

[00148] In some embodiments of the disclosure, the sending a resource allocation request to a resource arrangement server includes the following step.

[00149] A resource schedule task including the running configuration information is sent to a resource scheduling server through a message queue, so that the resource scheduling server sends the resource allocation request to the resource arrangement server. The resource allocation request includes the running configuration information.

[00150] In some embodiments of the disclosure, the determining a route entry according to the target container includes the following steps.

[00151] A container startup success message sent by the resource arrangement server is received. The container startup success message includes the route entry of the target container.

[00152] The route entry is sent to a route management server, so that the route management server adds the route entry locally.

[00153] In some embodiments of the disclosure, the container startup success message further includes address information of the target resource node. The sending the task source data to the target resource node includes the following step.

[00154] The task source data is sent to the target resource node according to the address information.

[00155] In some embodiments of the disclosure, before receiving the resource schedule request of the target task sent by the terminal, the method further includes the following steps.

[00156] In a case of monitoring that source data configuration information is inputted in a task resource schedule configuration page, a resource configuration recommendation request sent by the terminal is received. The resource configuration recommendation request includes the inputted source data configuration information.

[00157] Recommended running configuration information is determined according to the inputted source data configuration information.

[00158] The recommended running configuration information is sent to the terminal, so that the terminal displays a resource configuration acknowledgment message on the task resource schedule configuration page. The resource configuration acknowledgment message includes the recommended running configuration information.

[00159] Correspondingly, the receiving the resource schedule request of the target task sent by the terminal includes the following step.

[00160] A resource schedule request of the target task sent by the terminal in response to an acknowledgment instruction for adopting the recommended running configuration information is received.

[00161] In some embodiments of the disclosure, after receiving a container startup success message sent by the resource arrangement server, the method further includes the following steps.

[00162] A running state of the target container is monitored.

[00163] In a case of monitoring that the running state of the target container is an idle state and a duration of the idle state is greater than or equal to a preset idle duration threshold, a resource destruction instruction is sent to the resource arrangement server, so that the resource arrangement server controls the target resource node to delete the target container.

[00164] In some embodiments of the disclosure, after sending task source data to a target resource node, the method further includes the following step. Resource usage information of the target container sent by the target resource node is received.

[00165] Correspondingly, the resource schedule completion message further includes the resource usage information of the target container, so that the terminal displays a task running acknowledgment page including the resource usage information.

[00166] In some embodiments of the disclosure, after receiving resource usage information of the target container sent by the target resource node, the method further includes the following steps.

[00167] Rationality analysis is performed on a running resource of the target task according to the resource usage information.

[00168] In a case that an analysis result is unreasonable, restart task schedule information is sent to the terminal, so that the terminal displays the task running acknowledgment page further including the restart task schedule information.

[00169] In some embodiments of the disclosure, the target task includes a deep learning task. Task source data of the target task includes: a training data set corresponding to the source data configuration information and a neural network corresponding to the source data configuration information.

[00170] For the detailed solution of the task processing method with the server as the executive subject, reference is made to the embodiment of the task processing method written from the interactive perspective, and the details are not repeated here.

[00171] The following introduces a task processing method provided by an embodiment of the disclosure with the terminal as an executive subject, which specifically includes the following steps.

[00172] A resource schedule request of a target task is sent to server. The resource schedule request includes source data configuration information and running configuration information of the target task, so that the server determines task source data of the target task according to the source data configuration information. A target container is created and started in a target resource node according to the running configuration information. A route entry is determined according to the target container.

[00173] A resource schedule completion message sent by the server is received.

[00174] The target container and the task source data are called to execute the target task through the route entry.

[00175] In some embodiments of the disclosure, the calling target container and the task source data to execute the target task through the route entry includes the following step. A task running request of the target task is sent to a route management server, so that the route management server sends the task running request to the target container based on the route entry, and the target container runs the target task based on the task source data in response to the task running request.

[00176] In some embodiments of the disclosure, before sending the resource schedule request of the target task to the server, the method further includes the following steps. In a case of monitoring that source data configuration information is inputted in a task resource schedule configuration page, a resource configuration recommendation request is sent to the server, the resource configuration recommendation request including the inputted source data configuration information, so that the server determines recommended running configuration information according to the inputted source data configuration information. A resource configuration acknowledgment message is displayed on the task resource schedule configuration page. The resource configuration acknowledgment message includes the recommended running configuration information.

[00177] Correspondingly, the sending the resource schedule request of the target task to the server may include the following step. The resource schedule request of the target task is sent to the server in response to an acknowledgment instruction for adopting the recommended running configuration information.

[00178] In some embodiments of the disclosure, the resource schedule completion message further includes the resource usage information of the target container. The method further includes the following step. A task running acknowledgment page including the resource usage information is displayed.

[00179] In some embodiments of the disclosure, the task running acknowledgment page further includes restart task schedule information. The method further includes the following steps. A task resource schedule configuration page is displayed in response to a restart instruction based on the restart task schedule information. A resource recovery instruction is sent to the server, so that the server executes a resource recovery operation.

[00180] For the detailed solution of the task processing method with the terminal as the executive subject, reference is made to the embodiment of the task processing method written from the interactive perspective, and the details are not repeated here.

[00181] According to another aspect of the embodiments of the disclosure, a task processing system is further provided, including a terminal and a server.

[00182] The server is configured to send a resource schedule request of a target task to a server, the resource schedule request including source data configuration information

and running configuration information of the target task, and call a target container and task source data to execute the target task through a route entry.

[00183] The server is configured to determine task source data of the target task according to the source data configuration information, create and start a target container in a target resource node according to the running configuration information, determine a route entry according to the target container and send the task source data to the target source node, and send a resource schedule completion message to the terminal.

[00184] In some embodiments of the disclosure, the server may include a resource management server, a resource scheduling server, a resource arrangement server, and a route management server. Optionally, FIG. 7 is a flowchart II of a task processing method according to an embodiment of the disclosure, which specifically includes the following steps.

[00185] S701: a terminal sends a resource schedule request of a target task to a resource management server. The resource schedule request includes source data configuration information and running configuration information.

[00186] S703: the resource management server determines task source data of the target task according to the source data configuration information.

[00187] S705: the resource management server sends a resource schedule task to a resource scheduling server through a message queue. The resource schedule task includes the running configuration information.

[00188] S707: the resource scheduling server sends a resource allocation request to a resource arrangement server. The resource allocation request includes the running configuration information.

[00189] S709: the resource arrangement server creates and starts a target container in a target resource node according to running configuration information.

[00190] S711: the resource arrangement server obtains the address information of the target resource node and the route entry of the target container.

[00191] S713: the resource arrangement server sends a container startup success message to the resource scheduling server. The container startup success message includes the route entry and the address information.

[00192] S715: the resource scheduling server sends the route entry to a route management server.

[00193] S717: the resource scheduling server sends the address information to a resource management server through the message queue.

[00194] S719: the resource management server sends the task source data to the target resource node according to the address information.

[00195] S721: the resource management server sends a resource schedule completion message to a terminal.

[00196] S723: the terminal sends a task running request to the resource arrangement server.

[00197] S725: the route management server sends the task running request to the target container based on the route entry.

[00198] S727: the target container runs the target task based on the task source data in response to the task running request.

[00199] In some embodiments, sending, by the resource management server, the resource schedule task to the resource scheduling server through the message queue may include the following steps. The resource management server starts the resource schedule task. The resource schedule task includes running configuration information. The

resource management server stores the resource schedule task in the message queue. The resource scheduling server pulls the resource schedule task from the message queue.

[00200] In the embodiments of the disclosure, in the process of task resource scheduling management, the resource management server and the resource scheduling server can be adopted to joint scheduling management. The resource management server receives resource schedule requests from a plurality of terminals and puts the plurality of resource schedule requests into the message queue in the form of resource schedule tasks. Then the resource scheduling server takes out the resource schedule tasks from the message queue in turn to schedule running resources, which can realize the decoupling of scheduling and requests, greatly improving the availability and stability of resource scheduling.

[00201] Hereinafter, an exemplary description of the task processing method is given in conjunction with a Kubernetes-based resource scheduling system.

[00202] FIG. 8 is a schematic structural diagram of a resource scheduling system according to an embodiment of the disclosure. As shown in FIG. 8, the resource scheduling system mainly includes three parts: a Web front end, a project management and scheduling center, and a Kubernetes-based container management platform. The container management platform includes a multi-layer architecture, the bottom layer is a Linux system that deploys some resources, followed by a resource node cluster k8s&Docker layer including running resources (a container engine), and a basic resource layer, including the data transport protocol rabbitmq, a file storage system Ceph Fs, a database Mysql, and a load balancing tool Traefik for dynamic route management, and finally, a container management layer based on server Service, including container management Hub, mainly through proxy, a container destruction module cull_idle, a program module spawner, and an interface server api-server for providing an interface. In addition, the container management platform also includes an operation and maintenance service module, such as a log. The Web front end is configured to send a schedule request to the project management and scheduling center based on an http request. The project management and scheduling center is configured to send schedule request to the container management platform through a Message Queue (MQ). The container management platform is mainly responsible for container start and stop, monitoring and file storage mounting, routing management, etc., as well as the operation and maintenance service module.

[00203] Based on the resource scheduling system shown in FIG. 8, FIG. 9 is a schematic diagram of a data flow of a task processing method according to an embodiment of the disclosure. As shown in FIG. 9, after the user triggers the start or stop of the container through the Web front end, the front end is in communication with the project management and scheduling center through the http request, and sends the resource schedule request of the target task to the project management and scheduling center. The schedule request carries source data configuration information and running configuration information. After the project management and scheduling center determines the task source data of the target task according to the source data configuration information, the scheduling center initiates a schedule task. That is, the scheduling center sends the schedule task to the container management center Service through the MQ. Herein, the schedule task includes the running configuration information.

[00204] Then, the container management center is in communication with the k8s&Docker cluster through the spawner module according to the running configuration

information in the schedule task, and sends a resource allocation request including the running configuration information to the resource node cluster k8s&Docker. The resource node cluster creates and starts a container engine using the control node Master and the running node Node based on the running configuration information. The startup message includes the mirror image information of the container, i.e., the address information of the target resource node, and the hashrate information, i.e., the route entry of the target container, etc.

[00205] After the container is started successfully, the cluster k8s&Docker sends a container startup success message containing the route entry and address information to the container management center. The container management center sends the route entry to the route manager traefik-proxy, and sends the address information to the project management and scheduling center through the MQ. The project management and scheduling center sends the task source data to the resource node cluster k8s&Docker according to the address information, and then sends the resource schedule completion message to the front end. On the other hand, the project management and scheduling center can also communicate with the operation and maintenance service through the message queue after obtaining the address information, and issue a synchronization instruction of the data set for file synchronization, resource information, etc. Furthermore, after the scheduling process is completed, the user can directly communicate with the started target container through the traefik-proxy based on the obtained route entry through the browser front end to run the project flexibly.

[00206] For specific detailed solutions of related steps in the foregoing embodiment, reference may be made to the foregoing related embodiment of a single server corresponding to FIG. 3, and the details are not repeated here.

[00207] According to another aspect of the embodiments of the disclosure, a task processing apparatus is further provided. FIG. 10 is a block diagram I of a task processing apparatus according to an embodiment of the disclosure. As shown in FIG. 10, the task processing apparatus may include a resource schedule request receiving part 810, a task source data determining part 820, a container creating and starting part 830, a route entry determining part 840, a task source data sending part 850, and a resource schedule completion message sending part 860.

[00208] The resource schedule request receiving part 810 is configured to receive a resource schedule request of a target task sent by a terminal. The resource schedule request includes source data configuration information and running configuration information of the target task.

[00209] The task source data determining part 820 is configured to determine task source data of the target task according to the source data configuration information.

[00210] The container creating and starting part 830 is configured to create and start a target container in a target resource node according to the running configuration information.

[00211] The route entry determining part 840 is configured to determine a route entry according to the target container.

[00212] The task source data sending part 850 is configured to send the task source data to the target resource node.

[00213] The resource schedule completion message sending part 860 is configured to send a resource schedule completion message to the terminal, so that the terminal calls the target container and the task source data to execute the target task through the route entry.

[00214] In some embodiments, the container creating and starting part includes a resource allocation request sending part, configured to send a resource allocation request to a resource arrangement server, the resource allocation request including the running configuration information, so that the resource arrangement server creates and starts the target container in the target resource node based on the running configuration information.

[00215] In some embodiment, the resource allocation request sending part is specifically configured to send a resource schedule task including the running configuration information to a resource scheduling server through a message queue, so that the resource scheduling server sends the resource allocation request to the resource arrangement server. The resource allocation request includes the running configuration information.

[00216] In some embodiments, the route entry determining part includes a container startup success message receiving part and a route entry sending part. The container startup success message receiving part is configured to receive a container startup success message sent by the resource arrangement server. The container startup success message includes the route entry of the target container.

[00217] The route entry sending part is configured to send the route entry to a route management server, so that the route management server adds the route entry locally.

[00218] In some embodiments, the container startup success message further includes address information of the target resource node. The task source data sending part is specifically configured to send the task source data to the target resource node according to the address information.

[00219] In some embodiments, the apparatus further includes a resource configuration recommendation request receiving part, a recommended running configuration information determining part, and a recommended running configuration information sending part. The resource configuration recommendation request receiving part is configured to receive a resource configuration recommendation request sent by the terminal in a case of monitoring that source data configuration information is inputted in a task resource schedule configuration page. The resource configuration recommendation request includes the inputted source data configuration information. The recommended running configuration information determining part is configured to determine recommended running configuration information according to the inputted source data configuration information. The recommended running configuration information sending part is configured to send the recommended running configuration information to the terminal, so that the terminal displays a resource configuration acknowledgment message on the task resource schedule configuration page. The resource configuration acknowledgment message includes the recommended running configuration information. Correspondingly, the resource schedule request receiving part is further configured to receive a resource schedule request of the target task sent by the terminal in response to an acknowledgment message for adopting the recommended running configuration information.

[00220] In some embodiments, the apparatus further includes an idle recovery part and a resource destruction instruction sending part. The idle recovery part is configured to monitor a running state of the target container. The resource destruction instruction sending part is configured to send, in a case of monitoring that the running state of the target container is an idle state and a duration of the idle state is greater than or equal to a preset idle duration threshold, a resource destruction instruction to the resource

arrangement server, so that the resource arrangement server controls the target resource node to delete the target container.

[00221] In some embodiments, after sending the task source data to the target resource node according to the address information, the apparatus further includes a resource usage information receiving part, configured to receive resource usage information of the target container sent by the target resource node. Correspondingly, the resource schedule completion message further includes the resource usage information of the target container, so that the terminal displays a task running acknowledgment page including the resource usage information.

[00222] In some embodiments, the apparatus further includes a rationality analysis part and a restart task schedule information sending part. The rationality analysis part is configured to perform rationality analysis on a running resource of the target task according to the resource usage information. The restart task schedule information sending part is configured to send, in a case that an analysis result is unreasonable, restart task schedule information to the terminal, so that the terminal displays the task running acknowledgment page further including the restart task schedule information.

[00223] In some embodiments, the target task includes a deep learning task. Task source data of the target task includes a training data set corresponding to the source data configuration information and a neural network corresponding to the source data configuration information.

[00224] According to another aspect of the embodiments of the disclosure, a task processing apparatus is further provided. FIG. 11 is a block diagram II of a task processing apparatus according to an embodiment of the disclosure. As shown in FIG. 11, the task processing apparatus may include a resource schedule request sending part 910, a resource schedule completion message receiving part 920, and a target task executing part 930.

[00225] The resource schedule request sending part 910 is configured to send a resource schedule request of a target task to a server, the resource schedule request including source data configuration information and running configuration information of the target task, so that the server determines task source data of the target task according to the source data configuration information, create and start a target container in a target resource node according to the running configuration information, and determine a route entry according to the target container.

[00226] The resource schedule completion message receiving part 920 is configured to receive a resource schedule completion message sent by the server.

[00227] The target task executing part 930 is configured to call the target container and the task source data to execute the target task through the route entry.

[00228] In some embodiments, the target task executing part includes a task running request sending part, configured to send a task running request of the target task to a route management server, so that the route management server sends the task running request to the target container based on the route entry, and the target container runs the target task based on the task source data in response to the task running request.

[00229] In some embodiments, the apparatus further includes a resource configuration recommendation request sending part, a recommended running configuration information receiving part, and a resource configuration acknowledgment message displaying part. The resource configuration recommendation request sending part is configured to send a resource configuration recommendation request to the server in a case of monitoring that source data configuration information is inputted in a task resource schedule

configuration page, the resource configuration recommendation request including the inputted source data configuration information, so that the server determines recommended running configuration information according to the inputted source data configuration information. The recommended running configuration information receiving part is configured to receive the recommended running configuration information sent by the server. The resource configuration acknowledgment message displaying part is configured to display a resource configuration acknowledgment message on the task resource schedule configuration page, the resource configuration acknowledgment message including the recommended running configuration information. Correspondingly, the resource schedule request sending part is configured to send the resource schedule request of the target task to the server in response to an acknowledgment instruction for adopting the recommended running configuration information.

[00230] In some embodiments, the resource schedule completion message further includes resource usage information of the target container. The apparatus further includes a task running acknowledgment page displaying part, configured to display a task running acknowledgment page including the resource usage information.

[00231] In some embodiments, the task running acknowledgment page further includes restart task schedule information. The apparatus further includes a task resource schedule configuration page displaying part and a resource recovery instruction sending part. The task resource schedule configuration page displaying part is configured to display a task resource schedule configuration page in response to a restart instruction based on the restart task schedule information. The resource recovery instruction sending part is configured to send a resource recovery instruction to the server, so that the server executes a resource recovery operation.

[00232] Regarding the apparatus in the foregoing embodiments, the specific manners for performing operations of each part thereof are described in detail in the embodiments related to the method, and detailed description is not given here.

[00233] FIG. 12 is a block diagram I of an electronic device for task processing according to an embodiment of the disclosure. The electronic device may be a terminal, and an internal structural diagram thereof may be shown in FIG. 12. The electronic device includes a processor, a memory, a network interface, a display screen, and an input apparatus that are connected through a system bus. The processor of the electronic device is configured to provide computing and controlling capabilities. The memory of the electronic device includes a non-volatile storage medium and an internal memory. The non-volatile storage medium stores an operating system and a computer program. The internal memory provides an environment for operating the operating system and the computer program in the non-volatile storage medium. The network interface of the electronic device is communicatively connected to an external terminal through a network. When executed by the processor, the computer program implements a task processing method. The display screen of the electronic device may be a liquid crystal display screen or an electronic ink display screen. The input apparatus of the electronic device may be a touch layer covering the display screen, may also be a key, a trackball, or a touchpad disposed on an outer casing of the electronic device, and may further be an externally connected keyboard, touchpad, mouse, or the like.

[00234] FIG. 13 is a block diagram of an electronic device for task processing according to an embodiment of the disclosure. The electronic device may be a server, and an internal structural diagram thereof may be shown in FIG. 13. The electronic device

includes a processor, a memory, and a network interface that are connected through a system bus. The processor of the electronic device is configured to provide computing and controlling capabilities. The memory of the electronic device includes a non-volatile storage medium and an internal memory. The non-volatile storage medium stores an operating system and a computer program. The internal memory provides an environment for operating the operating system and the computer program in the non-volatile storage medium. The network interface of the electronic device is communicatively connected to an external terminal through a network. When executed by the processor, the computer program implements a task processing method.

[00235] A person skilled in the art may understand that, the structures shown in FIG. 12 and FIG. 13 are only block diagrams of a partial structure related to a solution of the disclosure and do not limit the electronic device to which the solution of the disclosure is applied. Specifically, the electronic device may include more or fewer components than those shown in the drawings, or include a combination of some components, or include different component layouts.

[00236] In the embodiments of the disclosure, a task processing device is further provided, including a processor and a memory configure to store processor-executable instructions. The processor is configured to execute the instructions to implement the task processing method in the embodiments of the disclosure.

[00237] In the embodiments of the disclosure, a computer-readable storage medium is further provided. When executed by a processor of a task processing device, instructions in the storage medium causes the task processing device to execute the task processing method in the embodiments of the disclosure.

[00238] In the embodiments of the disclosure, a computer program is further provided, including a computer-readable code. In a case that the computer-readable code runs in a task processing device, a processor in the task processing device is configured to execute the task processing method above.

[00239] A person of ordinary skill in the art may understand that all or some of the procedures of the methods of the foregoing embodiments may be implemented by computer programs instructing relevant hardware. The computer programs may be stored in a non-volatile computer-readable storage medium. When the computer programs are executed, the procedures of the embodiments of the foregoing methods may be included. References to the memory, the storage, the database, or other media used in the embodiments provided in the disclosure may all include a non-volatile memory or a volatile memory. The non-volatile memory may include a Read-Only Memory (ROM), a programmable ROM (PROM), an electrically programmable ROM (EPROM), an electrically erasable programmable ROM (EEPROM), or a flash memory. The volatile memory may include a Random Access Memory (RAM) or an external cache. By way of description rather than limitation, the RAM may be obtained in a plurality of forms, such as a static RAM (SRAM), a dynamic RAM (DRAM), a synchronous DRAM (SDRAM), a double data rate SDRAM (DDRSDRAM), an enhanced SDRAM (ESDRAM), a Synchlink DRAM (SLDRAM), a Rambus direct RAM (RDRAM), a direct Rambus dynamic RAM (DRDRAM), and a Rambus dynamic RAM (RDRAM).

[00240] After considering the description and practicing the invention disclosed herein, a person skilled in the art would easily conceive of other implementations of the disclosure. The disclosure aims at covering any variation, usage, or adaptive change of the disclosure. These variations, usages, or adaptive changes follow a general principle of the disclosure and include the common knowledge or common technical means in this

technical field that are not disclosed in the disclosure. The description and embodiments are merely considered as exemplary, and the true scope and spirits of the disclosure are pointed out in the following claims.

[00241] It should be understood that the disclosure is not limited to the accurate structures described and shown in the accompanying drawings above, and can be subjected to various amendments and changes without departing from its scope. Therefore, the scope of the disclosure shall only be limited by the appended claims.

INDUSTRIAL APPLICABILITY

[00242] In the embodiments of the disclosure, a resource schedule request of a target task sent by a terminal is received, the resource schedule request including source data configuration information and running configuration information of the target task, task source data of the target task is determined according to the source data configuration information, a target container is created and started in a target resource node according to the running configuration information, a route entry is determined according to the target container and the task source data is sent to the target resource node, and a resource schedule completion message is sent to the terminal, so that the terminal calls the target container and the task source data to execute the target task through the route entry. The complexity and error rate of the resource configuration of the on-line programming education is greatly reduced, thereby greatly improving flexibility and task processing speed in the task processing process.

CLAIMS

1. A task processing method, comprising:

receiving a resource schedule request of a target task sent by a terminal, the resource schedule request comprising source data configuration information and running configuration information of the target task;

determining task source data of the target task according to the source data configuration information;

creating and starting a target container in a target resource node according to the running configuration information;

determining a route entry according to the target container and sending the task source data to the target resource node; and

sending a resource schedule completion message to the terminal, so that the terminal calls the target container and the task source data to execute the target task through the route entry.

2. The task processing method of claim 1, wherein the creating and starting a target container in a target resource node according to the running configuration information comprises:

sending a resource allocation request to a resource arrangement server, the resource allocation request comprising the running configuration information, so that the resource arrangement server creates and starts the target container in the target resource node based on the running configuration information.

3. The task processing method of claim 2, wherein the sending a resource allocation request to a resource arrangement server comprises:

sending a resource schedule task comprising the running configuration information to a resource scheduling server through a message queue, so that the resource scheduling server sends the resource allocation request to the resource arrangement server, the resource allocation request comprising the running configuration information.

4. The task processing method of claim 2, wherein the determining a route entry according to the target container comprises:

receiving a container startup success message sent by the resource arrangement server, the container startup success message comprising the route entry of the target container; and

sending the route entry to a route management server, so that the route management server adds the route entry locally.

5. The task processing method of claim 4, wherein the container startup success message further comprises address information of the target resource node, and the sending the task source data to the target resource node comprises:

sending the task source data to the target resource node according to the address information.

6. The task processing method of claim 1, wherein before receiving a resource schedule request of a target task sent by a terminal, the method further comprises:

in a case of monitoring that source data configuration information is inputted in a task resource schedule configuration page, receiving a resource configuration recommendation

request sent by the terminal, the resource configuration recommendation request comprising the inputted source data configuration information;

determining recommended running configuration information according to the inputted source data configuration information; and

sending the recommended running configuration information to the terminal, so that the terminal displays a resource configuration acknowledgment message on the task resource schedule configuration page, the resource configuration acknowledgment message comprising the recommended running configuration information; and

the receiving a resource schedule request of a target task sent by a terminal comprises:

receiving a resource schedule request of the target task sent by the terminal in response to an acknowledgment instruction for adopting the recommended running configuration information.

7. The task processing method of claim 4, wherein after receiving a container startup success message sent by the resource arrangement server, the method further comprises:

monitoring a running state of the target container; and

in a case of monitoring that the running state of the target container is an idle state and a duration of the idle state is greater than or equal to a preset idle duration threshold, sending a resource destruction instruction to the resource arrangement server, so that the resource arrangement server controls the target resource node to delete the target container.

8. The task processing method of claim 1, wherein after sending the task source data to the target resource node, the method further comprises:

receiving resource usage information of the target container sent by the target resource node; and

the resource schedule completion message further comprising the resource usage information of the target container, so that the terminal displays a task running acknowledgment page comprising the resource usage information.

9. The task processing method of claim 8, wherein after receiving resource usage information of the target container sent by the target resource node, the method further comprises:

performing rationality analysis on a running resource of the target task according to the resource usage information; and

in a case that an analysis result is unreasonable, sending restart task schedule information to the terminal, so that the terminal displays the task running acknowledgment page further comprising the restart task schedule information.

10. The task processing method of any one of claims 1 to 9, wherein the target task comprises a deep learning task, and task source data of the target task comprises: a training data set corresponding to the source data configuration information and a neural network corresponding to the source data configuration information.

11. A task processing method, comprising:

sending a resource schedule request of a target task to a server, the resource schedule request comprising source data configuration information and running configuration information of the target task, so that the server determines task source data of the target task according to the source data configuration information; creating and starting a target container

in a target resource node according to the running configuration information; and determining a route entry according to the target container;

receiving a resource schedule completion message sent by the server; and

calling the target container and the task source data to execute the target task through the route entry.

12. The task processing method of claim 11, wherein the calling the target container and the task source data to execute the target task through the route entry comprises:

sending a task running request of the target task to a route management server, so that the route management server sends the task running request to the target container based on the route entry, and the target container runs the target task based on the task source data in response to the task running request.

13. The task processing method of claim 11, wherein before sending a resource schedule request of a target task to server, the method further comprises:

in a case of monitoring that source data configuration information is inputted in a task resource schedule configuration page, sending a resource configuration recommendation request to the server, the resource configuration recommendation request comprising the inputted source data configuration information, so that the server determines the recommended running configuration information according to the inputted source data configuration information;

receiving the recommended running configuration information sent by the server; and

displaying a resource configuration acknowledgment message on the task resource schedule configuration page, the resource configuration acknowledgment message comprising the recommended running configuration information; and

the sending a resource schedule request of a target task to server comprises:

sending the resource schedule request of the target task to the server in response to an acknowledgment instruction for adopting the recommended running configuration information.

14. The task processing method of claim 11, wherein the resource schedule completion message further comprises resource usage information of the target container; and the method further comprises:

displaying a task running acknowledgment page comprising the resource usage information.

15. The task processing method of claim 14, wherein the task running acknowledgment page further comprises restart task schedule information; and the method further comprises:

displaying a task resource schedule configuration page in response to a restart instruction based on the restart task schedule information; and

sending a resource recovery instruction to the server, so that the server executes a resource recovery operation.

16. A task processing system, comprising a terminal and a server, wherein

the terminal is configured to send a resource schedule request of a target task to a server, the resource schedule request comprising source data configuration information and running configuration information of the target task, and call a target container and task source data to execute the target task through a route entry; and

the server is configured to determine task source data of the target task according to the source data configuration information, create and start a target container in a target resource node according to the running configuration information, determine a route entry according to the target container and send the task source data to the target source node, and send a resource schedule completion message to the terminal.

17. A task processing apparatus, comprising:

a resource schedule request receiving part, configured to receive a resource schedule request of a target task sent by a terminal, the resource schedule request comprising source data configuration information and running configuration information of the target task;

a task source data determining part, configured to determine task source data of the target task according to the source data configuration information;

a container creating and starting part, configured to create and start a target container in a target resource node according to the running configuration information;

a route entry determining part, configured to determine a route entry according to the target container;

a task source data sending part, configured to send the task source data to the target resource node; and

a resource schedule completion message sending part, configured to send a resource schedule completion message to the terminal, so that the terminal calls the target container and the task source data to execute the target task through the route entry.

18. The task processing apparatus of claim 17, wherein the container creating and starting part comprises:

a resource allocation request sending part, configured to send a resource allocation request to a resource arrangement server, the resource allocation request comprising the running configuration information, so that the resource arrangement server creates and starts the target container in the target resource node based on the running configuration information.

19. The task processing apparatus of claim 18, wherein

the resource allocation request sending part is specifically configured to send a resource schedule task comprising the running configuration information to a resource scheduling server through a message queue, so that the resource scheduling server sends the resource allocation request to the resource arrangement server, the resource allocation request comprising the running configuration information.

20. The task processing apparatus of claim 18, wherein the route entry determining part comprises:

a container startup success message receiving part, configured to receive a container startup success message sent by the resource arrangement server, the container startup success message comprising the route entry of the target container; and

a route entry sending part, configured to send the route entry to a route management server, so that the route management server adds the route entry locally.

21. The task processing apparatus of claim 20, wherein the container startup success message further comprises address information of the target resource node, and the task source data

sending part is specifically configured to send the task source data to the target resource node according to the address information.

22. The task processing apparatus of claim 17, further comprising:

a resource configuration recommendation request receiving part, configured to receive, before receiving a resource schedule request of a target task sent by a terminal, a resource configuration recommendation request sent by the terminal in a case of monitoring that source data configuration information is inputted in a task resource schedule configuration page, the resource configuration recommendation request comprising the inputted source data configuration information;

a recommended running configuration information determining part, configured to determine recommended running configuration information according to the inputted source data configuration information; and

a recommended running configuration information sending part, configured to send the recommended running configuration information to the terminal, so that the terminal displays a resource configuration acknowledgment message on the task resource schedule configuration page, the resource configuration acknowledgment message comprising the recommended running configuration information; and

correspondingly, the resource schedule request receiving part is further configured to receive a resource schedule request of the target task sent by the terminal in response to an acknowledgment message for adopting the recommended running configuration information.

23. The task processing apparatus of claim 20, further comprising:

an idle recovery part, configured to monitor, after receiving a container startup success message sent by the resource arrangement server, a running state of the target container; and

a resource destruction instruction sending part, configured to send, in a case of monitoring that the running state of the target container is an idle state and a duration of the idle state is greater than or equal to a preset idle duration threshold, a resource destruction instruction to the resource arrangement server, so that the resource arrangement server controls the target resource node to delete the target container.

24. The task processing apparatus of claim 17, further comprising:

a resource usage information receiving part, configured to, after sending task source data to a target resource node according to address information, receive resource usage information of the target container sent by the target resource node; and

correspondingly, the resource schedule completion message further comprising the resource usage information of the target container, so that the terminal displays a task running acknowledgment page comprising the resource usage information.

25. The task processing apparatus of claim 24, further comprising:

a rationality analysis part, configured to perform, after receiving resource usage information of the target container sent by the target resource node, rationality analysis on a running resource of the target task according to the resource usage information; and

a restart task schedule information sending part, configured to send, in a case that an analysis result is unreasonable, restart task schedule information to the terminal, so that the terminal displays the task running acknowledgment page further comprising the restart task schedule information.

26. The task processing apparatus of claims 17 to 25, wherein the target task comprises a deep learning task, and task source data of the target task comprises: a training data set corresponding to the source data configuration information and a neural network corresponding to the source data configuration information.

27. A task processing apparatus, comprising:

a resource schedule request sending part, configured to send a resource schedule request of a target task to a server, the resource schedule request comprising source data configuration information and running configuration information of the target task, so that the server determines task source data of the target task according to the source data configuration information; create and start a target container in a target resource node according to the running configuration information; and determine a route entry according to the target container;

a resource schedule completion message receiving part, configured to receive a resource schedule completion message sent by the server; and

a target task executing part, configured to call the target container and the task source data to execute the target task through the route entry.

28. The task processing apparatus of claim 27, wherein the target task executing part comprises:

a task running request sending part, configured to send a task running request of the target task to a route management server, so that the route management server sends the task running request to the target container based on the route entry, and the target container runs the target task based on the task source data in response to the task running request.

29. The task processing apparatus of claim 27, further comprising:

a resource configuration recommendation request sending part, configured to send, before sending a resource schedule request of a target task to a server, a resource configuration recommendation request to the server in a case of monitoring that source data configuration information is inputted in a task resource schedule configuration page, the resource configuration recommendation request comprising the inputted source data configuration information, so that the server determines recommended running configuration information according to the inputted source data configuration information;

a recommended running configuration information receiving part, configured to receive the recommended running configuration information sent by the server; and

a resource configuration acknowledgment message displaying part, configured to display a resource configuration acknowledgment message on the task resource schedule configuration page, the resource configuration acknowledgment message comprising the recommended running configuration information; and

correspondingly, the resource schedule request sending part is further configured to send the resource schedule request of the target task to the server in response to an acknowledgment instruction for adopting the recommended running configuration information.

30. The task processing apparatus of claim 27, wherein the resource schedule completion message further comprises resource usage information of the target container, and the task processing apparatus further comprises:

a task running acknowledgment page displaying part, configured to display a task running acknowledgment page comprising the resource usage information.

31. The task processing apparatus of claim 30, wherein the task running acknowledgment page further comprises restart task schedule information; and the task processing apparatus further comprises:

a task resource schedule configuration page displaying part, configured to display a task resource schedule configuration page in response to a restart instruction based on the restart task schedule information; and

a resource recovery instruction sending part, configured to send a resource recovery instruction to the server, so that the server executes a resource recovery operation.

32. A task processing device, comprising:

a processor; and

a memory, configured to store processor-executable instructions;

wherein the processor is configured to execute the instructions to implement the task processing method of any one of claims 1 to 10 or any one of claims 11 to 15.

33. A computer-readable storage medium, wherein when instructions in the storage medium are executed by a processor of a task processing device, the task processing device is capable of executing the task processing method of any one of claims 1 to 10 or any one of claims 11 to 15.

34. A computer program, comprising a computer-readable code, wherein when running in a task processing device, the computer-readable code is executed by a processor in the task processing device to implement the task processing method of any one of claims 1 to 10 or any one of claims 11 to 15.