

DATA PROCESSING METHOD, APPARATUS, SYSTEM AND DEVICE AND COMPUTER-READABLE STORAGE MEDIUM

TECHNICAL FIELD

[0001] The disclosure relates to a computer network technology, and particularly to a data processing method, apparatus, system and device and computer-readable storage medium.

BACKGROUND

[0002] At present, for the application scenario of recording the status of a game table through a service on a server, it is necessary to modify and update a configuration parameter of the service in time under the condition that table layout, participant information, game rules, network connection configuration and other parameters of the game table change. However, after the configuration parameter is modified in a present parameter configuration method, the service on the server needs to be restarted, and the modified configuration parameter is loaded by using an initialization process of the service. The parameters configured in such a parameter configuration process take effect slowly, and will lead to downtime when the server cannot provide services, thus reducing the parameter configuration efficiency.

SUMMARY

[0003] Embodiments of the disclosure provide a data processing method, apparatus, system and device and a computer-readable storage medium, which can improve the parameter configuration efficiency.

[0004] The technical solutions of the embodiments of the disclosure are implemented as follows.

[0005] The embodiments of the disclosure provide a data processing method, which may include the following operations.

[0006] A target parameter modification event in a first preset event registry is triggered according to a target parameter in a parameter modification request under the condition that the parameter modification request for a first service is received; a remote parameter corresponding to the target parameter in a remote storage space is updated through a unified calling interface, the first preset event registry representing a mapping relationship among a first preset parameter, a first preset parameter modification event and a first preset callback function in the first service, and the remote storage space being configured to store at least one remote parameter of at least one service.

[0007] A target callback function corresponding to the target parameter modification event is called by using the target parameter, and the target parameter is loaded to the first service.

[0008] The first service is run, and the game status of a game area is recorded based on the target parameter.

[0009] Under the condition that the parameter modification request is received, the corresponding target parameter modification event may be triggered according to the target parameter, and then as a response to the target parameter modification event, the corresponding target callback function may be called to load the target parameter to the first service in real time, so that the first service may record the game status based on a new target parameter value, the callback function of the parameter modification event may be configured as a parameter loading method in the event registry in advance, the parameter modification request may be responded by an event triggering mechanism, and the corresponding callback function may be called in real time to reach the effect that the parameter takes effect immediately after being changed. In addition, for the remote storage space composed of different types of storage devices in a real scenario, the corresponding remote parameters in the remote storage space may be managed and updated indistinguishably through the unified calling interface, thereby implementing the parameter synchronization of at least one service, thus greatly improving the parameter configuration efficiency.

- [0010] Before the operation that the target parameter modification event in the first preset event registry is triggered according to the target parameter in the parameter modification request, the above method may further include the following operations.
- [0011] At least one first preset parameter corresponding to the first service and a first preset parameter loading method corresponding to each of the at least one first preset parameter are acquired under the condition that the first service is initialized. The target parameter belongs to the at least one first preset parameter, and the first preset parameter loading method is a parameter loading method pre-configured in the first service.
- [0012] A first preset parameter modification event corresponding to each of the first preset parameters is generated for each of the first preset parameters. A first preset callback function corresponding to the first preset parameter modification event is the first preset parameter loading method corresponding to each of the first preset parameters. The target parameter modification event is a first preset parameter modification event corresponding to the target parameter in the first preset event registry. The target callback function is the first preset callback function corresponding to the target parameter modification event.
- [0013] The first preset parameter modification event is correspondingly registered on each first preset parameter to obtain the mapping relationship among the first preset parameter, the first preset parameter modification event and the first preset callback function as the first preset event registry.
- [0014] The mapping relationship among the at least one first preset parameter, the first preset parameter modification event and the first preset callback function is realized as the first preset event registry by generating the first preset parameter modification event corresponding to each of the first preset parameters and registering it on the corresponding first preset parameter, and setting the first preset callback function of the first preset parameter modification event as the first preset parameter loading method of the corresponding first preset parameter. As such, under the condition that the first preset parameter in the first preset event registry changes, the first preset callback function corresponding to the event may be called through a corresponding event trigger response mechanism, the first preset parameter loading method implemented in the first preset

callback function is executed, and the corresponding target parameter is loaded to the first service in real time, thereby improving the parameter configuration efficiency.

[0015] In the above method, the operation that the target parameter modification event in the first preset event registry is triggered according to the target parameter in the parameter modification request under the condition that the parameter modification request for the first service is received may include the following operations.

[0016] A preset event-driven method implemented in advance in a preset parameter modification interface is called and the first preset event registry is accessed according to the target parameter under the condition that the parameter modification request is received through the preset parameter modification interface.

[0017] The first preset parameter modification event corresponding to the target parameter is determined as the target parameter modification event in the first preset event registry, and the target parameter modification event is triggered through the preset event-driven method.

[0018] By implementing the preset event-driven method in the preset parameter modification interface in advance, the target parameter modification event corresponding to the target parameter may be triggered immediately under the condition that the parameter modification request is received, which improves the parameter configuration efficiency.

[0019] In the above method, the operation that the target callback function corresponding to the target parameter modification event is called by using the target parameter, and the target parameter is loaded to the first service may include the following operations.

[0020] The target callback function corresponding to the target parameter modification event is determined according to the first preset event registry.

[0021] The target callback function is called by using the target parameter through the preset event-driven method; and the target parameter is transferred into the target preset parameter loading method implemented in the target callback function, and the target

parameter is loaded to the first service through a target predicted parameter loading method. The target preset parameter loading method is the first preset parameter loading method corresponding to the target parameter in at least one first preset parameter loading method, and the at least one first preset parameter loading method is a preset parameter loading method corresponding to the at least one first preset parameter in the first service.

[0022] Due to the fact that the first preset callback function corresponding to each of the first preset parameter modification events is the first preset parameter loading method of the corresponding first preset parameter in the first preset event registry, under the condition that the target callback function corresponding to the target parameter modification event is obtained through the first preset event registry, the target parameter may be transferred into the target parameter loading method corresponding to the target parameter by calling the target callback function, and then the target parameter may be loaded into the first service in real time, so that the first service may load the latest modified parameter configuration in real time during the present operation, which improves the parameter configuration efficiency.

[0023] In the above method, the operation that the remote parameter corresponding to the target parameter in the remote storage space is updated through the unified calling interface may include the following operations.

[0024] The remote storage space is accessed through the unified calling interface, and a target remote parameter is determined from the at least one remote parameter according to a target parameter name of the target parameter. The remote storage space may contain a sub-storage space of at least one storage type. The unified calling interface is configured to implement read-write operation on the sub-storage space of the at least one storage type in a unified calling mode. The target remote parameter is a shared parameter of the first service and the second service. The second service is configured to synchronize service data of the first service.

[0025] The target remote parameter is updated by using the target parameter, and the target parameter is synchronously updated to the second service.

[0026] By updating the target remote parameter in the remote storage space with the target parameter value, the parameter values stored locally and remotely in the first service may be synchronized in real time, and further, a parameter change corresponding to the parameter modification request for the first service may be synchronized to a plurality of other services sharing parameters with the first service in time through the change of the target remote parameter, thereby improving the parameter configuration efficiency.

[0027] The above method may further include the following operations.

[0028] A first local target parameter corresponding to the target parameter in the first local storage is updated according to the target parameter. The first local storage is the storage space of a first host where the first service is located.

[0029] The parameter change specified by the parameter modification request is synchronously updated to a local cache corresponding to the first service, which may ensure the consistency between operation parameters and local cache parameters, thereby improving the stability of operation of the first service.

[0030] The embodiments of the disclosure provide a data processing method, which may include the following operations.

[0031] A second local parameter of a second service in second local storage is compared with a second remote parameter in a remote storage space through a unified calling interface according to a preset time interval to obtain a comparison result. The second local storage is the storage space on a second host where the second service is located. The second remote parameter may contain a shared parameter of a first service and the second service. The first service is configured to record the game status of a game area.

[0032] A target parameter is determined from the second remote parameter under the condition that the comparison result represents that the second local parameter is inconsistent with the second remote parameter. A target parameter modification event corresponding to a target parameter name in a second preset event registry is triggered according to the target parameter. The second preset event registry represents a mapping

relationship among a second preset parameter name, a second preset parameter modification event and a second preset callback function in the second service.

[0033] The target parameter is loaded to the second service by calling a target callback function corresponding to the target parameter modification event.

[0034] The second service is run and the first service is synchronized based on the target parameter.

[0035] A parameter change event is quickly found by regularly comparing the differences between a local parameter and a remote parameter, and by designing an event driven model in advance and defining an event to be executed after the parameter change in advance, the updated remote parameter is synchronized to the local for loading in time through an event trigger response mechanism, so that the changed configuration parameter may take effect in real time on the second service, thereby improving the parameter configuration efficiency.

[0036] Before the operation that the target parameter modification event corresponding to the target parameter name in the second preset event registry according to the target parameter, the above method may further include the following operations.

[0037] A second preset event registry is obtained based on at least one second preset parameter corresponding to the second service and at least one second preset parameter loading method corresponding to the at least one second preset parameter under the condition that the second service is initialized. In the second preset event registry, the at least one second preset parameter is in one-to-one correspondence with at least one second preset parameter modification event, and each of the at least one second preset parameter loading method is a callback function corresponding to the at least one second preset parameter modification event.

[0038] By generating the second preset event registry of the mapping relationship among each of the second preset parameters, the second preset parameter modification event and the second preset callback function, the target parameter related to the parameter change may be loaded to the second service in real time through the event trigger response mechanism, thereby improving the parameter configuration efficiency.

- [0039] After the operation that the target parameter is determined from the second remote parameter, the above method may further include the following operations.
- [0040] The target parameter is read from the remote storage space through the unified calling interface, and the remote storage space may contain a sub-storage space of at least one storage type.
- [0041] A second local target parameter corresponding to the target parameter in first local storage is updated according to the target parameter, and the second local storage is the storage space of a second host where the second service is located.
- [0042] By executing calling through the unified calling interface, unified abstraction of the read-write operation on different types of sub-storage spaces is implemented, so that the second service may manage the system configuration parameters of each storage location indistinguishably, thereby improving the efficiency and maintainability of parameter configuration.
- [0043] The embodiments of the disclosure provide a data processing system, which may include a first host and a second host. A first service runs on the first host, and a second service runs on the second host.
- [0044] The first host is configured to trigger a target parameter modification event in a first preset event registry according to a target parameter in a parameter modification request under the condition that the parameter modification request for the first service is received; update a remote parameter corresponding to the target parameter in a remote storage space through a unified calling interface, the first preset event registry representing a mapping relationship among a first preset parameter, a first preset parameter modification event and a first preset callback function in the first service; call a target callback function corresponding to the target parameter modification event by using the target parameter, and load the target parameter to the first service; and run the first service and record the game status of a game area based on the target parameter.
- [0045] The second host is configured to compare a second local parameter of the second service in second local storage with a second remote parameter in the remote storage space through the unified calling interface according to a preset time interval to

obtain a comparison result; determine the target parameter from the second remote parameter under the condition that the comparison result represents that the second local parameter is inconsistent with the second remote parameter; trigger a target parameter modification event corresponding to a target parameter name in a second preset event registry according to the target parameter, the target parameter being a shared parameter of the first service in the at least one service and the second service; load the target parameter to the second service by calling the target callback function corresponding to the target parameter modification event; and run the second service and synchronize the first service based on the target parameter.

[0046] The embodiments of the disclosure provide a first data processing apparatus, which may include a first triggering part, a first calling part, and a recording part.

[0047] The first triggering part is configured to trigger a target parameter modification event in a first preset event registry according to a target parameter in a parameter modification request under the condition that the parameter modification request for a first service is received; and update a remote parameter corresponding to the target parameter in the remote storage space through a unified calling interface. The first preset event registry represents a mapping relationship among a first preset parameter, a first preset parameter modification event and a first preset callback function in the first service, and the remote storage space is configured to store at least one remote parameter of at least one service.

[0048] The first calling part is configured to call a target callback function corresponding to the target parameter modification event by using the target parameter, and load the target parameter to the first service.

[0049] The recording part is configured to run the first service and record the game status of a game area based on the target parameter.

[0050] The above first data processing apparatus may also include a first registration part. The first registration part is configured to acquire at least one first preset parameter corresponding to the first service and a first preset parameter loading method corresponding to each of the at least one first preset parameter when the first service is

initialized before the target parameter modification event in the first preset event registry is triggered according to the target parameter in the parameter modification request, the target parameter belonging to the at least one first preset parameter, and the first preset parameter loading method being a parameter loading method configured in the first service in advance; generate a first preset parameter modification event corresponding to each of the first preset parameters for each of the first preset parameters, the first preset callback function corresponding to the first preset parameter modification event being the first preset parameter loading method corresponding to each of the first preset parameters, the target parameter modification event being a first preset parameter modification event corresponding to the target parameter in the first preset event registry, the target callback function being the first preset callback function corresponding to the target parameter modification event; and correspondingly register the first preset parameter modification event on each of the first preset parameters to obtain the mapping relationship among the first preset parameter, the first preset parameter modification event and the first preset callback function as the first preset event registry.

[0051] In the above apparatus, the first triggering part is also configured to call a preset event-driven method implemented in advance in a preset parameter modification interface and access the first preset event registry according to the target parameter under the condition that the parameter modification request is received through the preset parameter modification interface; and determine the first preset parameter modification event corresponding to the target parameter as the target parameter modification event in the first preset event registry, and trigger the target parameter modification event through the preset event-driven method.

[0052] In the above apparatus, the first calling part is also configured to determine a target callback function corresponding to the target parameter modification event according to the first preset event registry; call the target callback function by using the target parameter through the preset event-driven method; and transfer the target parameter into the target preset parameter loading method implemented in the target callback function, and load the target parameter to the first service through a target predicted parameter loading method. The target preset parameter loading method is the first preset parameter loading method corresponding to the target parameter in at least

one first preset parameter loading method, and the at least one first preset parameter loading method is a preset parameter loading method corresponding to the at least one first preset parameter in the first service.

[0053] In the above apparatus, the first triggering part is also configured to access the remote storage space through the unified calling interface, and determine a target remote parameter from the at least one remote parameter according to a target parameter name of the target parameter, the remote storage space containing a sub-storage space of at least one storage type, the unified calling interface being configured to implement read-write operation on the sub-storage space of the at least one storage type in a unified calling mode, the target remote parameter being a shared parameter of the first service and the second service, the second service being configured to synchronize service data of the first service; and update the target remote parameter by using the target parameter, and synchronously update the target parameter to the second service.

[0054] In the above apparatus, the first data processing apparatus may further include a first updating part. The first updating part is configured to update the first local target parameter corresponding to the target parameter in the first local storage according to the target parameter. The first local storage is the storage space of a first host where the first service is located.

[0055] The embodiments of the disclosure provide a second data processing apparatus, which may include a comparison part, a second triggering part, a second calling part, and a synchronization part.

[0056] The comparison part is configured to compare a second local parameter of a second service in second local storage with a second remote parameter in a remote storage space through a unified calling interface according to a preset time interval to obtain a comparison result. The second local storage is the storage space of a second host where the second service is located. The second remote parameter may contain a shared parameter of a first service and the second service. The first service is configured to record the game status of a game area.

[0057] The second triggering part is configured to determine a target parameter from

the second remote parameter under the condition that the comparison result represents that the second local parameter is inconsistent with the second remote parameter; and trigger a target parameter modification event corresponding to a target parameter name in a second preset event registry according to the target parameter. The second preset event registry represents a mapping relationship among a second preset parameter name, a second preset parameter modification event and a second preset callback function in the second service.

[0058] The second calling part is configured to load the target parameter to the second service by calling a target callback function corresponding to the target parameter modification event.

[0059] The synchronization part is configured to run the second service and synchronize the first service based on the target parameter.

[0060] The above second data processing apparatus may further include a second registration part. The second registration part is configured to obtain a second preset event registry based on at least one second preset parameter corresponding to the second service and at least one second preset parameter loading method corresponding to the at least one second preset parameter under the condition that the second service is initialized before the target parameter modification event corresponding to the target parameter name in the second preset event registry is triggered according to the target parameter. In the second preset event registry, the at least one second preset parameter is in one-to-one correspondence with at least one second preset parameter modification event. Each of the at least one second preset parameter loading method is a callback function corresponding to the at least one second preset parameter modification event.

[0061] In the above apparatus, the second data processing apparatus may further include a second updating part. The second updating part is configured to read the target parameter from the remote storage space through the unified calling interface after the target parameter is determined from the second remote parameter, the remote storage space containing a sub-storage space of at least one storage type; and update the second local target parameter corresponding to the target parameter in the first local storage according to the target parameter, the second local storage being the storage space of a

second host where the second service is located.

[0062] The embodiments of the disclosure provide an electronic device, which may include a memory and a processor.

[0063] The memory is configured to store an executable instruction.

[0064] The processor is configured to implement the method according to any one of the embodiments of the disclosure when executing the executable instruction stored in the memory.

[0065] The embodiments of the disclosure provide a computer-readable storage medium, storing an executable instruction, configured to be executed by a processor to implement the method provided by any one of the embodiments of the disclosure.

[0066] The embodiments of the disclosure have the following beneficial effects.

[0067] Under the condition that the parameter modification request is received, the corresponding target parameter modification event may be triggered according to the target parameter, and then as a response to the target parameter modification event, the corresponding target callback function may be called to load the target parameter to the first service in real time, so that the first service may record the game status based on a new target parameter value, the callback function of the parameter modification event may be configured as a parameter loading method in the event registry in advance, the parameter modification request may be responded by an event triggering mechanism, and the corresponding callback function may be called in real time to reach the effect that the parameter takes effect immediately after being changed. In addition, the corresponding remote parameter in the remote storage space may be updated through the unified calling interface, thereby implementing the parameter synchronization of at least one service, thus greatly improving the parameter configuration efficiency.

BRIEF DESCRIPTION OF THE DRAWINGS

[0068] FIG. 1 is an optional architecture diagram of a data processing system

architecture according to an embodiment of the disclosure.

[0069] FIG. 2 is an optical architecture diagram of a first data processing apparatus according to an embodiment of the disclosure.

[0070] FIG. 3 is an optical architecture diagram of a second data processing apparatus according to an embodiment of the disclosure.

[0071] FIG. 4 is an optional flowchart of a data processing method on a first host according to an embodiment of the disclosure.

[0072] FIG. 5 is an optional flowchart of a data processing method on a first host according to an embodiment of the disclosure.

[0073] FIG. 6 is an optional flowchart of a data processing method on a first host according to an embodiment of the disclosure.

[0074] FIG. 7 is an optional flowchart of a data processing method on a first host according to an embodiment of the disclosure.

[0075] FIG. 8 is an optional flowchart of a data processing method on a first host according to an embodiment of the disclosure.

[0076] FIG. 9 is a schematic diagram of a calling relationship of a first service according to an embodiment of the disclosure.

[0077] FIG. 10 is an optional flowchart of a data processing method on a second host according to an embodiment of the disclosure.

[0078] FIG. 11 is an optional flowchart of a data processing method on a second host according to an embodiment of the disclosure.

[0079] FIG. 12 is a flowchart of a process of sharing a parameter by a first service and a second service according to an embodiment of the disclosure.

[0080] FIG. 13 is an exemplary application diagram in a practical application scenario according to an embodiment of the disclosure.

DETAILED DESCRIPTION

[0081] For making the objectives, technical solutions, and advantages of the disclosure clearer, the disclosure will further be described below in combination with the drawings in detail. The described embodiments should not be considered as limits to the disclosure. All other embodiments obtained by those of ordinary skill in the art without creative work shall fall within the scope of protection of the disclosure.

[0082] "Some embodiments" involved in the following descriptions describes a subset of all possible embodiments. However, it can be understood that "some embodiments" may be the same subset or different subsets of all the possible embodiments, and may be combined without conflicts.

[0083] Term "first/second/third" involved in the following descriptions is only for distinguishing similar objects and does not represent a specific sequence of the objects. It can be understood that "first/second/third" may be interchanged to specific sequences or orders if allowed to implement the embodiments of the disclosure described herein in sequences except the illustrated or described ones.

[0084] Unless otherwise defined, all technological and scientific terms used in the disclosure have meanings the same as those usually understood by those skilled in the art of the disclosure. The terms used in the disclosure are only adopted to describe the embodiments of the disclosure and not intended to limit the disclosure.

[0085] Before the embodiments of the disclosure are further described in detail, nouns and terms involved in the embodiments of the disclosure will be described. The nouns and terms involved in the embodiments of the disclosure are suitable for the following explanations.

[0086] 1) Distributed deployment is a technical solution that may not deploy a data processing business to different devices through massive data carried by one device or due to special requirements such as security problems. The essence of distribution is to distribute tasks originally completed by one machine to multiple machines so as to

improve efficiency.

[0087] 2) A distributed system is a software system built on a distributed deployment network. In a distributed database, a user does not feel that the data is distributed, that is, the user does not need to know whether the relationship is divided, whether there is a copy, which site the data is stored in and which site things are executed on. From the perspective of process, two programs run on the processes of two hosts, cooperate with each other and finally complete the same service, then the system composed of the two programs may be called "distributed system".

[0088] 3) The so-called cluster is a way to continuously extend the same application horizontally on multiple devices, such as servers, in order to improve service capabilities.

[0089] 4) Cluster storage aggregates the storage spaces in multiple storage devices into one storage pool and provides a unified access interface and management interface for application servers. Applications may access and utilize disks on all storage devices transparently through the access interface, and give full play to the performance and disk utilization of the storage devices. Data in the cluster storage will be stored and read from multiple storage devices according to certain rules, so as to obtain higher concurrent access performance.

[0090] At present, a complete business application on the server often needs to adapt to various scenarios and requirements, such as database address switching between test environment and production environment, log level change or specific rule adaptation, etc. Usually, the system configuration parameters of a business application are modified to adapt to different application scenario parameters. For business applications that have been running online, online environment is needed to modify system configuration parameters, which can be applied immediately. However, the present scheme generally needs to restart the business application after modifying the system configuration, so that it may be initialized with new system configuration parameters, thus causing the business application to be unable to provide services for a period of time, which is obviously unreasonable for high availability scenarios.

[0091] Additionally, for some requirements of multi-type storage adaptation, such as

multi-vendor interface adaptation, a business application needs to adapt to many different storage sources, such as local files, databases, Object Storage Service (OSS), etc. The system configuration parameters of different modules in the business application may be maintained by different storage sources and managed and maintained by the business application. If the configuration changes every time, the business application needs to be restarted and initialized, which will greatly reduce the usability of the business application.

[0092] Furthermore, when the business application needs distributed deployment, the status between multiple services of the business application needs to be synchronized, and the modification of parameter configuration needs to be synchronized to other services quickly. However, based on the present parameter configuration scheme, the rapid synchronization of the modification of business application parameters cannot be realized in the distributed deployment scenario.

[0093] The embodiments of the disclosure provide a data processing method, apparatus, system and device, and a computer-readable storage medium, which can improve the parameter configuration efficiency. An exemplary application of the electronic device provided in the embodiments of the disclosure will be described below. The electronic device provided in the embodiments of the disclosure may be implemented as various types of user terminals such as a notebook computer, a tablet computer, a desktop computer, a set-top box, and a mobile device (for example, a mobile phone, a portable music player, a personal digital assistant, a dedicated messaging device, and a portable game device), and may also be implemented as a server. An exemplary application when the device is implemented as the server will be described as follows.

[0094] Referring to FIG. 1, FIG. 1 is an optional architecture diagram of a data processing system 100 according to an embodiment of the disclosure. In order to realize a business application supporting a cluster deployment, a client device 400 is connected to a server 200 through a network 300. Herein, the server 200 may be a cluster including at least one host (a host 200-1 to a host 200-n are exemplarily shown), and the network 300 may be a wide area network or a local area network, or a combination of the two. At least

one service (a first service, a second service to a nth service are exemplarily shown) is respectively deployed on the host 200-1, the host 200-2 to the host 200-n in a distributed deployment manner. At least one service may store its corresponding parameter as at least one remote parameter in a remote storage space such as a storage device 500, and each remote parameter corresponds to a data record in the storage device 500. In the initialization stage of the data processing system 100, the parameter value of at least one remote parameter in the storage device 500 may be initialized first, and at least one service may be read from the storage device 500 through a unified call interface 510 to its corresponding local storage space, such as a cache 1, a cache 2 to a cache n, for its own service operation. As such, at least one service may share parameters through the storage device 500. At least one service of distributed deployment needs to keep the parameters loaded in present operation, local parameters in a local storage space and corresponding remote parameters in a remote storage space consistent during operation. That is, for the at least one service of distributed deployment, when the parameters of a certain service change, the parameter change needs to be synchronized to the loaded parameters, local parameters and remote parameters corresponding to each service in time.

[0095] In some embodiments, the first service on the host 200-1 may be a main service that is currently providing a business service to the outside world, such as a service that records the game status of a game area. The second service to the nth service on the hosts 200-2 to 200-n may be duplicate services consistent with the function of the first service, and the second to nth services do not provide business services to the outside world and are configured for keeping real-time synchronization with business data of the first service, such as business status and business progress, during the running process of the first service, so that they may be used as business backup of the first service. Therefore, when the first service fails, the data processing system 100 may switch to other services in the same business status or progress at any time, such as any one of the second service to the nth service, thereby reducing the downtime of the server and serving the high availability scenario. In such a scenario, the parameters of the first service, the second service to the nth service are the same parameters, which are maintained by the first service, the second service to the nth service in their respective

local storage spaces and stored as shared parameters in the remote storage space, namely, the storage device 500. As such, when the parameter modification request modifies the parameters on the first service, the modified parameters need to be synchronized to the second service to the nth service in time.

[0096] In some embodiments, the second service to the nth service may also be applications sharing the same parameters with the first service, run together with the first service in an application scenario configured with the same shared parameters, and jointly provide business services to the outside world. Exemplarily, for the scenario where the first service, the second service to the nth service may respectively record the game statuses of different game areas, and heat maps corresponding to the different game areas may be the same. Thus, the same heat map may be uniformly stored in the storage device 500 as a shared parameter, respectively read into respective local storage spaces via the parameters corresponding to the first service and the second service to the nth service, and loaded at runtime to normally realize the recording function of the game areas. For such a scenario, when the parameter corresponding to any one of the first service, the second service and the nth service is modified to change, the changed parameter needs to be synchronously updated to the corresponding shared parameter in the storage device 500 and other related services, so as to realize rapid unification of parameter configuration.

[0097] In FIG. 1, the storage device 500 (a storage device 500-1 and a storage device 500-2 are exemplarily shown) is configured to provide a sub-storage space of at least one storage type. Exemplarily, the at least one storage type may include file storage, database storage, object storage type, etc. The sub-storage space of the at least one storage type may be uniformly and indistinguishably managed by a first application, a second application to a nth application through the unified calling interface 510, such as a read-write operation.

[0098] In FIG. 1, a client device 400 belongs to the administrator of the data processing system 100, and is configured to initiate the modification operation on a specified configuration parameter through operation and maintenance client software 410, and take the specified configuration parameter as a target parameter to generate a

parameter modification request corresponding to the modification operation, and send the parameter modification request to the server 200 through the network 300. Generally, the parameter modification request will be sent to the first service on the host 200-1 that is currently providing the service. The parameters of the first service are respectively stored in the local cache 1 of the host 200-1 and the remote storage device 500. The host 200-1 is configured to trigger a target parameter modification event in a first preset event registry according to a target parameter in a parameter modification request under the condition that the parameter modification request for the first service is received; update the remote parameter corresponding to the target parameter in the remote storage space, i.e., the storage device 500, through a unified call interface, the first preset event registry representing a mapping relationship among a first preset parameter, a preset parameter modification event and a callback function in the first service; call a target callback function corresponding to the target parameter modification event by using the target parameter, and load the target parameter to the first service; and run the first service and record the game status of a game area based on the target parameter.

[0099] The host 200-2 is configured to compare a second local parameter of the second service in second local storage, i.e., the cache 2, with a second remote parameter in the remote storage space through the unified calling interface according to a preset time interval to obtain a comparison result; determine the target parameter from the second remote parameter under the condition that the comparison result represents that the second local parameter is inconsistent with the second remote parameter; trigger a target parameter modification event corresponding to a target parameter name in a second preset event registry according to the target parameter, and load the target parameter to the second service by calling a target callback function corresponding to the target parameter modification event, the second preset event registry representing the mapping relationship among a parameter name, a preset parameter modification event and a callback function in the second service; and run the second service and synchronize the first service based on the target parameter.

[00100] It is to be noted that, here, for other services that share parameters with the first service, such as a third service on a host 200-3 to the nth service on the host 200-n, the changed shared parameters in the storage device 500 may be updated by the same data

processing method as the second service, so as to realize the global real-time synchronization of parameter changes by a distributed deployment service, which will not be repeated here again.

[00101] It is to be noted that, based on a network structure of the data processing system 100, the second service or the nth service on the host 200-2 or the host 200-n may also be used as the service that currently provides a business function, the parameter modification request is received through the second service or the nth service, and the real-time loading of the target parameter and the update of the shared parameter in the storage device 500 may be implemented through the data processing method in the embodiment of the disclosure. Synchronization and real-time loading of the updated shared parameters may be implemented through the first service. That is, the business roles of the host 200-1, the host 200-2 to the host 200-n may be interchanged in the data processing system 100. A specific choice is made according to actual conditions, which is not limited in the embodiments of the disclosure.

[00102] In some embodiments, the server 200 may be an independent physical server, or a server cluster consisting of multiple physical servers or a distributed system, or may be a cloud server that provides basic cloud computing service such as cloud service, a cloud database, cloud computing, a cloud function, cloud storage, network service, cloud communication, middleware service, domain name service, security service, a Content Delivery Network (CDN), and a big data and artificial intelligence platform. The terminal 400 may be a smart phone, a tablet computer, a notebook computer, a desktop computer, a smart speaker, a smart watch, etc. However, the terminal is not limited thereto. The terminal and the server may be connected directly or indirectly through wired or wireless communication, which is not limited in the embodiments of the disclosure.

[00103] Referring to FIG. 2, FIG. 2 is an architecture diagram of a host 200-1 in a server 200 according to an embodiment of the disclosure. The host 200-1 shown in FIG. 2 may include at least one first processor 210-1, a first memory 250-1, at least one first network interface 220-1 and a first user interface 230-1. The various components in the host 200-1 are coupled together through a first bus system 240-1. It is to be understood

that the first bus system 240-1 is configured to implement the connection communication between these components. The first bus system 240-1 may further include a power bus, a control bus and a status signal bus in addition to a data bus. However, for clarity, the various buses are designated as the first bus system 240-1 in FIG. 2.

[00104] The first processor 210-1 may be an integrated circuit chip with signal processing capability, such as a general-purpose processor, a Digital Signal Processor (DSP), or other programmable logic devices, discrete gate or transistor logic devices, discrete hardware components, etc. The general-purpose processor may be a microprocessor or any conventional processor, etc.

[00105] The first user interface 230-1 may include one or more first output devices 231-1 that enable presentation of media content, including one or more speakers and/or one or more visual display screens. The first user interface 230-1 may also include one or more first input devices 232-1, including a user interface component that facilitates user input, such as a keyboard, a mouse, a microphone, a touch screen display, a camera, other input buttons and controls.

[00106] The first memory 250-1 may be removable, non-removable or a combination thereof. An exemplary hardware device may include a solid-state memory, a hard disk drive, an optical disk drive, etc. The first memory 250-1 may optionally include one or more storage devices that are physically remote from the first processor 210-1.

[00107] The first memory 250-1 may include a volatile memory or a nonvolatile memory, and may also include both the volatile memory and the nonvolatile memory. The nonvolatile memory may be Read Only Memory (ROM), and the volatile memory may be Random Access Memory (RAM). The first memory 250-1 described in the embodiment of the disclosure is intended to include any suitable type of memory.

[00108] In some embodiments, the first memory 250-1 can store data to support various operations, examples of which may include programs, modules and data structures or subsets or supersets thereof, as exemplified below.

[00109] A first operating system 251-1 may include a system program, configured to process various basic system services and execute hardware-related tasks, such as a

framework layer, a core library layer, a driver layer, etc., and is configured to implement various basic services and process the hardware-based tasks.

[00110] A network communication module 252-1 is configured to reach other computing devices via one or more (wired or wireless) first network interfaces 220-1. An exemplary first network interface 220 may include Bluetooth, WiFi, Universal Serial Bus (USB), etc.

[00111] A first presentation module 253-1 is configured to enable presentation of information (e.g., a user interface configured to operate peripheral devices and display content and information) via one or more first output apparatuses 231-1 (e.g., a display screen, a speaker, etc.) associated with the first user interface 230-1.

[00112] A first input processing module 254-1 is configured to detect one or more user inputs or interactions from one or more first input apparatuses 232-1 and translate the detected inputs or interactions.

[00113] In some embodiments, the apparatus provided by the embodiments of the disclosure may be implemented in a form of software. FIG. 2 shows a first data processing apparatus 255-1 stored in a first memory 250-1, which may be software in the form of programs and plug-ins, and may include the following software parts of a first triggering part 2551, a first calling part 2552 and a recording part 2553 to realize the functions of the host 200-1. These software parts are logical, so they can be arbitrarily combined or further split according to the realized functions.

[00114] Referring to FIG. 3, FIG. 3 is an architecture diagram of a host 200-2 in a server 200 according to an embodiment of the disclosure. The host 200-2 shown in FIG. 3 may include at least one second processor 210-2, a second memory 250-2, at least one second network interface 220-2 and a second user interface 230-2. The functions of each component in the host 200-2 are similar to those of each component with corresponding names in the host 200-1, which will not be repeated here. FIG. 3 shows a second data processing apparatus 255-2 stored in a second memory 250-2, which may be software in the form of programs and plug-ins, and may include the following software parts of a comparison part 2551, a second triggering part 2552, a second calling part 2553 and a

synchronization part 2554 to realize the functions of the host 200-2. These software parts are logical, so they can be arbitrarily combined or further split according to the realized functions.

[00115] The function of each software part in FIG. 2 and FIG. 3 will be explained below.

[00116] In other embodiments, the apparatus according to the embodiments of the disclosure may be implemented in a form of hardware. As an example, the apparatus according to the embodiments of the disclosure may be a processor in the form of a hardware decoding processor, which is programmed to execute the data processing method provided by the embodiments of the disclosure. For example, the processor in the form of the hardware decoding processor may adopt one or more Application Specific Integrated Circuits (ASIC), Digital Signal Processors (DSP), Programmable Logic Devices (PLD), Complex Programmable Logic Devices (CPLD), Field-Programmable Gate Arrays (FPGA) or other electronic components.

[00117] The data processing method according to the embodiment of the disclosure will be explained with reference to the exemplary application and implementation of the server provided by the embodiment of the disclosure.

[00118] Referring to FIG. 4, FIG. 4 is an optional flowchart of a data processing method on a first host according to an embodiment of the disclosure, which will be described with reference to the steps shown in FIG. 4.

[00119] In S101, a target parameter modification event in a first preset event registry is triggered according to a target parameter in a parameter modification request under the condition that the parameter modification request for a first service is received; and a remote parameter corresponding to the target parameter in a remote storage space is updated through a unified calling interface. The first preset event registry represents a mapping relationship among a first preset parameter, a first preset parameter modification event and a first preset callback function in the first service, and the remote storage space is configured to store at least one remote parameter of at least one service.

[00120] In the embodiment of the disclosure, a user may log in to the data processing

system through a client device, initiate a parameter modification request for the target parameter, and send the parameter modification request to a server cluster containing a plurality of hosts. Generally, the parameter modification request will be received by a host that is currently providing a business service in the server cluster, namely a first host. The first service on the first host is an application service that is providing the business service. In some embodiments, the first service may be configured to detect the game status of a game area.

[00121] In the embodiment of the disclosure, the first service correspondingly maintains a first preset event registry, which represents the mapping relationship among the first preset parameter, the preset parameter modification event and the callback function in the first service. The first preset parameter is a configuration parameter corresponding to the first service, and the first preset event registry contains at least one first preset parameter modification event, which may be registered on at least one first preset parameter corresponding to the first service in one-to-one correspondence at the initialization stage of the first service as an event handler for the at least one first preset parameter. As such, when the status of the first preset parameter changes, such as a parameter modification request to modify a specified first preset parameter into a target parameter, the first service will call the corresponding first preset parameter modification event, namely, the target parameter modification event, according to the object that triggered the event.

[00122] In the embodiment of the disclosure, each of the first preset parameter modification events is correspondingly bound with a first preset callback function, and each of the first preset callback functions is configured to be called by the first service as an event-triggered response mode when triggered in the corresponding first preset parameter modification event. The internal implementation of the first preset callback function corresponding to each of the first preset parameter modification events is the first preset parameter loading method of the first preset parameter corresponding to the first preset parameter modification event.

[00123] In the embodiment of the disclosure, each of the hosts of the data processing system may be configured with its own preset event registry, for example, for the second

service on the second host, a second preset event registry may be correspondingly configured. As such, when the service on each of the hosts monitors the target parameter needing to be configured, for example, when the first service receives the target parameter in the parameter modification request, and when the second service monitors that the target parameter in the remote storage space is updated, the service on each of the hosts may call the preset loading method corresponding to the target parameter by using an event triggering response mechanism according to its corresponding preset event registry, so as to load the target parameter in real time.

[00124] In some embodiments, the first preset parameter may include camera angle, game participant information, a heat map, game rules, and network configuration parameters of the server itself, such as network address, maximum number of connections, connected client address, port number, and so on. The camera is configured to record game scenario images generated by a game process in the game area from different angles. The heat map is various pre-divided game function areas on the game area. The areas corresponding to the heat map in the game scenario image may be recognized and analyzed by an image recognition method, and the present status or progress of the game process may be intelligently recognized.

[00125] In some embodiments, the first host may set an event driver at a functional interface receiving the parameter modification request, so that when the parameter modification request is received through a parameter modification interface, the event driver may be called to trigger the corresponding target parameter modification event in the first preset event registry according to the target parameter.

[00126] In the embodiment of the disclosure, when the first host receives the parameter modification request for the first service, it may parse a network data packet corresponding to the parameter modification request to obtain the target parameter contained therein. The target parameter may include a target parameter name and a target parameter value. As thus, the process that the first host triggers the target parameter modification event in the first preset event registry according to the target parameter in the parameter modification request may be implemented by an operation that the corresponding first preset parameter modification event is determined from the first

preset event registry as the target parameter modification event according to the target parameter name of the target parameter, and an operation that the target parameter is used to trigger the target parameter modification event, and the target parameter value is transferred into the internal handler of the target parameter modification event, so that the target callback function bound in advance in the target parameter modification event may obtain the target parameter value and load the target parameter value in real time.

[00127] In the embodiment of the disclosure, in order to keep the synchronization between the local parameter and the remote parameter, the first host may also update the remote parameter corresponding to the target parameter in the remote storage space through the unified call interface. The remote storage space is configured to store at least one remote parameter of at least one service, and the at least one remote parameter may include the shared parameter of the first service and the second service, or more services.

[00128] In some embodiments, when the parameter modification request for the first service is received, the first host may use different processes to respectively execute two procedures that the target parameter modification event in the first preset event registry is triggered according to the target parameter in the parameter modification request and that the remote parameter corresponding to the target parameter in the remote storage space is updated through the unified call interface.

[00129] In S102, a target callback function corresponding to the target parameter modification event is called by using the target parameter, and the target parameter is loaded to the first service.

[00130] In the embodiment of the disclosure, as a response to the triggering of the target parameter modification event, the first host may call up the target callback function corresponding to the target parameter modification event by using the target parameter and transfer the target parameter value of the target parameter into the target callback function, and then execute the target callback function and load the target parameter value to the first service, so that the target parameter may be configured to take effect, and the first service may run and provide a business service based on the target parameter value.

[00131] In some embodiments, the first service may store temporary data generated during the running process and the present or commonly-used first preset parameter in the local storage space of the first host, such as a cache, so as to quickly call related parameters or data, therefore, the first host may also update a first local reference value corresponding to the target parameter name in the first local storage according to the target parameter name in the target parameter, so as to synchronously update the corresponding parameters of the first service in a local cache. The first local storage is the storage space of the first host where the first service is located.

[00132] In S103, the first service is run, and the game status of the game area is recorded based on the target parameter.

[00133] In the embodiment of the disclosure, the first service may apply the newly configured parameter, i.e., the target parameter, to the running process, and the first host may continue to record the game status of the game area in real time based on the target parameter by running the first service, without having to restart to make the configured target parameter take effect.

[00134] In some embodiments, the game area may be any type of game scenario area such as a game table, a game field, etc., which is specifically selected according to the actual situation, which is not limited in the embodiments of the disclosure.

[00135] It is to be understood that in the embodiment of the disclosure, under the condition that the parameter modification request is received, the corresponding target parameter modification event may be triggered according to the target parameter, and then as a response to the target parameter modification event, the corresponding target callback function may be called to load the target parameter to the first service in real time, so that the first service may record the game status based on a new target parameter value, the callback function of the parameter modification event may be configured as a parameter loading method in the event registry in advance, the parameter modification request may be responded by an event triggering mechanism, and the corresponding callback function may be called in real time to reach the effect that the parameter takes effect immediately after being changed. In addition, for the remote storage space composed of different types of storage devices in a real scenario, the corresponding

remote parameters in the remote storage space may be managed and updated indistinguishably through the unified calling interface, thereby implementing the parameter synchronization of at least one service and greatly improving the parameter configuration efficiency.

[00136] Referring to FIG. 5, FIG. 5 is an optional flowchart of a data processing method on a first host according to an embodiment of the disclosure. Based on FIG. 4, S001-S003 may also be included before S101, which will be described with reference to the steps shown in FIG. 5.

[00137] In S001, at least one first preset parameter corresponding to the first service and a first preset parameter loading method corresponding to each of the at least one first preset parameter are acquired under the condition that the first service is initialized, where the target parameter belongs to at least one first preset parameter, and the first preset parameter loading method is a parameter loading method configured in advance in the first service.

[00138] In the embodiment of the disclosure, the first host may acquire at least one first preset parameter corresponding to the first service when initializing the first service, and acquire a loading method for each of the first preset parameters configured in advance in the first service as a first preset parameter loading method corresponding to each of the first preset parameters. The target parameter is one belonging to the at least one first preset parameter.

[00139] In S002, a first preset parameter modification event corresponding to each of the first preset parameters is generated for each first preset parameter, where a first preset callback function corresponding to the first preset parameter modification event is the first preset parameter loading method corresponding to each first preset parameter, the target parameter modification event is a first preset parameter modification event corresponding to the target parameter in the first preset event registry, the target callback function is the first preset callback function corresponding to the target parameter modification event.

[00140] In the embodiment of the disclosure, the first host may take each first preset

parameter as an event object in an event mechanism and generate a corresponding first preset parameter modification event for each of the first preset parameters. Furthermore, the first preset parameter loading method corresponding to each of the first preset parameters acquired in S001 is taken as the first preset callback function of the first preset parameter modification event corresponding to each of the first preset parameters, so as to realize the binding of the first preset parameter, the first preset parameter modification event and the first preset parameter loading method.

[00141] In the embodiment of the disclosure, the target parameter modification event is the first preset parameter modification event corresponding to the target parameter in the first preset event registry. The target callback function is the first preset callback function corresponding to the target parameter modification event.

[00142] In S003, the first preset parameter modification event is correspondingly registered on each first preset parameter to obtain the mapping relationship among the first preset parameter, the first preset parameter modification event and the first preset callback function as the first preset event registry.

[00143] In the embodiment of the disclosure, for each of the first preset parameters, the first host may register the first preset parameter modification event on the corresponding first preset parameter in a mounting mode, so as to obtain the mapping relationship among at least one first preset parameter, the first preset parameter modification event and the first preset callback function as the first preset event registry corresponding to the first service on the first host.

[00144] In the embodiment of the disclosure, for each of the first preset parameters corresponding to the first service, a change management program corresponding to each of the first preset parameters may be pre-deployed and implemented on the first host, which is configured to process a change event when the corresponding first preset parameter changes. The first host may mount the first preset parameter modification event on the change management program of the corresponding first preset parameter, so that when the first preset parameter changes, such as receiving a corresponding parameter modification request, the first preset parameter modification event corresponding to the first preset parameter may be triggered by a preset event-driven method, and then the first

preset callback function corresponding to the first preset parameter modification event may be called.

[00145] In some embodiments, after S001, the first host may also load each of the first preset parameters to the first service by a first preset parameter loading method corresponding to each of the first preset parameters, so as to initialize the first service by using each of the first preset parameters including an initial first preset parameter value.

[00146] It is to be understood that, in the embodiment of the disclosure, the mapping relationship among the at least one first preset parameter, the first preset parameter modification event and the first preset callback function is realized as the first preset event registry by generating the first preset parameter modification event corresponding to each of the first preset parameters and registering it on the corresponding first preset parameter, and setting the first preset callback function of the first preset parameter modification event as the first preset parameter loading method of the corresponding first preset parameter by the first host. As such, under the condition that the first preset parameter in the first preset event registry changes, the first preset callback function corresponding to the event may be called through a corresponding event trigger response mechanism, the first preset parameter loading method implemented in the first preset callback function is executed, and the corresponding target parameter is loaded to the first service in real time, thereby improving the parameter configuration efficiency.

[00147] Referring to FIG. 6, FIG. 6 is an optional flowchart of a data processing method on a first host according to an embodiment of the disclosure. Based on FIG. 4 or FIG. 5, S101 may be implemented by executing S101-1 that when the parameter modification request for the first service is received, the target parameter modification event in the first preset event registry is triggered according to the target parameter in the parameter modification request and S101-2 that the remote parameter corresponding to the target parameter in the remote storage space is updated through the unified call interface. S101-1 may be implemented by executing S101-11 to S101-12, which will be described with reference to each step.

[00148] In S101-11, a preset event-driven method implemented in advance in a preset parameter modification interface is called and the first preset event registry is accessed

according to the target parameter under the condition that the parameter modification request is received through the preset parameter modification interface.

[00149] In the embodiment of the disclosure, the first host may implement the preset event-driven method in advance in the preset parameter modification interface configured to interact with a client, so that when the parameter modification request is received, the preset event-driven method implemented in advance in the preset parameter modification interface is called, the first preset event registry is accessed according to the target parameter, and it is determined whether the target parameter has the corresponding first preset parameter modification event in the first preset event registry.

[00150] In some embodiments, the preset event-driven method may be implemented by a function in the form of `event_edit_parameter(p-a)`, where p-a represents the target parameter.

[00151] In S101-12, the first preset parameter modification event corresponding to the target parameter is determined as the target parameter modification event in the first preset event registry, and the target parameter modification event is triggered through the preset event-driven method.

[00152] In the embodiment of the disclosure, the first host may find the first preset parameter to which the target parameter belongs in the first preset event registry, and take the first preset parameter modification event registered on the first preset parameter to which the target parameter belongs as the target parameter modification event according to the first preset event registry.

[00153] In the embodiment of the disclosure, under the condition that the target parameter modification event is determined, the first host triggers the target parameter modification event by the preset event-driven method.

[00154] It is to be understood that in the embodiment of the disclosure, by implementing the preset event-driven method in the preset parameter modification interface in advance, the target parameter modification event corresponding to the target parameter may be triggered immediately under the condition that the parameter modification request is received, which improves the parameter configuration efficiency.

[00155] Referring to FIG. 7, FIG. 7 is an optional flowchart of a data processing method on a first host according to an embodiment of the disclosure. Based on FIG. 6, S102 may be implemented by executing S1021-S1022, which will be described with reference to each step.

[00156] In S1021, the target callback function corresponding to the target parameter modification event is determined according to the first preset event registry.

[00157] In the embodiment of the disclosure, under the condition that the target parameter modification event is triggered, the first host takes the first preset callback function corresponding to the target parameter modification event in the first preset event registry as the target callback function to respond to the target parameter modification event by calling the target callback function.

[00158] In S1022, the target callback function is called by using the target parameter through the preset event-driven method, the target parameter is transferred into the target preset parameter loading method implemented in the target callback function, and the target parameter is loaded to the first service through the target predicted parameter loading method, the target preset parameter loading method being the first preset parameter loading method corresponding to the target parameter in the at least one first preset parameter loading method, and the at least one first preset parameter loading method being a preset parameter loading method corresponding to the at least one first preset parameter in the first service.

[00159] In the embodiment of the disclosure, when the target callback function is determined, the first host may take the target parameter as an incoming parameter of the target callback function, call the target callback function through the preset event-driven method, transfer the target parameter to the target callback function, and load the target parameter to the first service through the target preset parameter loading method correspondingly implemented in the target callback function during the execution of the target callback function.

[00160] In the embodiment of the disclosure, the target preset parameter loading method is the first preset parameter loading method corresponding to the target parameter

in the at least one first preset parameter loading method. The at least one first preset parameter loading method is a preset parameter loading method corresponding to the at least one first preset parameter of the first service.

[00161] It is to be understood that in the embodiment of the disclosure, due to the fact that the first preset callback function corresponding to each of the first preset parameter modification events is the first preset parameter loading method of the corresponding first preset parameter in the first preset event registry, under the condition that the target callback function corresponding to the target parameter modification event is obtained through the first preset event registry, the first host may transfer the target parameter into the target parameter loading method corresponding to the target parameter by calling the target callback function, and then the target parameter may be loaded into the first service in real time, so that the first service may load the latest modified parameter configuration in real time during the present operation, thereby improving the parameter configuration efficiency.

[00162] In some embodiments, under the condition that the parameter modification request for the first service is received, the first host may also update a first local target parameter corresponding to the target parameter in the first local storage according to the target parameter. The first local storage is the storage space of the first host where the first service is located. In some embodiments, the first local storage may be a cache space used by the first service on the first host. As such, the parameter change specified by the parameter modification request is synchronously updated to a local cache corresponding to the first service, which may ensure the consistency between an operation parameter and a local cache parameter, which improves the stability of operation of the first service.

[00163] Referring to FIG. 8, FIG. 8 is an optional flowchart of a data processing method on a first host according to an embodiment of the disclosure. Based on FIG. 4 or FIG. 5, S101-2 in S101 may be implemented by executing S101-21 to S101-22, which will be described with reference to each step.

[00164] In S101-21, the remote storage space is accessed through the unified calling interface, and the target remote parameter is determined from the at least one remote parameter according to a target parameter name of the target parameter. The remote

storage space may contain a sub-storage space of at least one storage type. The unified calling interface is configured to implement read-write operation on the sub-storage space of the at least one storage type in a unified calling mode. The target remote parameter is a shared parameter of the first service and the second service. The second service is configured to synchronize service data of the first service.

[00165] In the embodiment of the disclosure, for the process of updating the remote storage space by using the target parameter, the first host may access the remote storage space through the unified call interface. The remote storage space may contain the sub-storage space of at least one storage type. The first host may use the unified calling interface to implement the read-write operation on the sub-storage space of the at least one storage type in a unified calling mode.

[00166] In the embodiment of the disclosure, the first host may determine the target remote parameter corresponding to the target parameter name from the at least one remote parameter in the remote storage space according to the target parameter name of the target parameter. The parameter name of the target remote parameter is consistent with the target parameter name, and the parameter value of the target remote parameter is an original parameter value corresponding to the target parameter name before receiving the parameter modification request.

[00167] In S101-22, the target remote parameter is updated by using the target parameter name of the target parameter, so that the second service synchronously updates the target remote parameter.

[00168] In the embodiment of the disclosure, the first host may update the parameter value of the target remote parameter by using the target parameter value of the target parameter. Here, since the target remote parameter is the shared parameter between the first service and the second service, after the target remote parameter is updated, the latest target remote parameter may be acquired by the second service for synchronous update, so that the parameter change corresponding to the parameter modification request for the first service may be synchronized to a plurality of other services sharing parameters with it.

[00169] It is to be understood that the target remote parameter in the remote storage space is updated with the target parameter value, so that the parameter values stored locally and remotely in the first service may be synchronized in real time, and further, the parameter change corresponding to the parameter modification request for the first service may be synchronized to the plurality of other services sharing parameters with it in time through the change of the target remote parameter, thereby improving the parameter configuration efficiency.

[00170] In some embodiments, referring to FIG. 9, FIG. 9 is a schematic diagram of a calling relationship of a first service according to an embodiment of the disclosure. As shown in FIG. 9, the first service on the first host itself may maintain the first preset event registry. In the initialization stage, the first service may read the required system configuration parameter from the remote storage space as the first preset parameter through the unified calling interface, such as p-a, where p-a correspondingly contains an initial preset parameter value. The first service calls an init_callback (p-a) method to load the preset parameter value of p-a, adds a first preset parameter modification event E-p-a in the first preset event registry according to p-a, and sets the callback function callback-p-a of E-p-a as a function that makes the new p-a parameter take effect, namely init_callback (p-a), so that the first preset event registry as shown in FIG. 9 may be generated according to at least one system configuration parameter corresponding to the first service. Under the condition that the first host receives the parameter modification request containing the target parameter p-a, the parameter value in the p-a is a new parameter value, the event driver event_edit_parameter () preset in the parameter modification interface may be called according to the p-a, and event_edit_parameter (p-a) is acquired to access the first preset event registry. The event E-p-a is modified from the target parameter corresponding to the first preset event registry p-a and triggered, and then as a response after the E-p-a event is triggered, a callback function callback-p-a corresponding to the E-p-a event is determined according to the first preset event registry, and the parameter loading method init_callback (p-a) set in the callback-p-a is called, and the p-a containing the new parameter value is loaded to the first service. In addition, the p-a containing the new parameter value is written into the remote storage space through the unified calling interface, and the corresponding remote parameter in

the remote storage space is updated.

[00171] Referring to FIG. 10, FIG. 10 is an optional flowchart of a data processing method on a second host according to an embodiment of the disclosure, which will be described with reference to the steps shown in FIG. 10.

[00172] In S201, a second local parameter of the second service in second local storage is compared with a second remote parameter corresponding to the second service in a remote storage space through a unified calling interface according to a preset time interval to obtain a comparison result, the remote storage space being configured to store at least one remote parameter of at least one service, the at least one remote parameter containing a shared parameter of a first service and the second service, the second local storage being a storage space on a second host where the second service is located, the first service being configured to record the game status of a game area.

[00173] In the embodiment of the disclosure, the second service runs on the second host, and at least one second preset parameter corresponding to the second service is stored in the second local storage and the remote storage space respectively. The second preset parameter stored in the second local storage is a second local parameter, and the second preset parameter stored in the remote storage space is a second remote parameter. The second remote parameter may include a shared parameter of the first service and the second service.

[00174] In the embodiment of the disclosure, the second host may compare the second local parameter with the second remote parameter according to the preset time interval through the unified call interface to obtain the comparison result.

[00175] In S202, a target parameter is determined from the second remote parameter under the condition that the comparison result represents that the second local parameter is inconsistent with the second remote parameter, and a target parameter modification event corresponding to a target parameter name in a second preset event registry is triggered according to the target parameter, the second preset event registry representing a mapping relationship among a second preset parameter name, a second preset parameter modification event and a second preset callback function in the second service.

[00176] In the embodiment of the disclosure, the comparison result represents whether the second local parameter is consistent with the second remote parameter, and when the second local parameter is inconsistent with the second remote parameter, it indicates that the second remote parameter has changed. Due to the fact that the second remote parameter contains the shared parameter of the first service and the second service, a certain shared parameter may have been updated by the first service, so the second service needs to synchronize the updated shared parameter to the second host in time.

[00177] In the embodiment of the disclosure, the second service on the second host correspondingly maintains the second preset event registry, and the second host determines the target parameter from the second remote parameter when the comparison result represents that the second local parameter is inconsistent with the second remote parameter, and triggers the target parameter modification event corresponding to the target parameter name in the second preset event registry according to the target parameter.

[00178] In the embodiment of the disclosure, the second preset event registry represents the mapping relationship between a second preset parameter name, a second preset parameter modification event and a second preset callback function in the second service. It is to be understood that the second preset parameter name here may include the parameter name of the shared parameter with the first service. As such, the second host may use the target parameter to determine the target parameter modification event corresponding to the target parameter and trigger the target parameter modification event in the mapping relationship of the second preset event registry, and its process is the same as the trigger process on the first host, which is not repeated here.

[00179] In some embodiments, the second host may take the second remote parameter which is inconsistent with the second local parameter as the target parameter to update the second local parameter of the second service differentially, or directly take all the second remote parameters as the target parameter to update the second local parameter globally, which is specifically selected according to the actual situation and is not limited in the embodiment of the disclosure

[00180] In S203, the target parameter is loaded to the second service by calling a target

callback function corresponding to the target parameter modification event

[00181] In the embodiment of the disclosure, the second service loads the target parameter to the second service by calling the target callback function corresponding to the target parameter modification event, which is the same as the corresponding process of calling the target callback function and loading the target parameter to the first service on the first host, which is not repeated here.

[00182] In S204, the second service is run, and the first service is synchronized based on the target parameter.

[00183] In the embodiment of the disclosure, the second service may be a backup service of the first service. When the target parameter is loaded into the second service, the second host may run the second service, and synchronize the present business status and business data of the first service into the second service based on the target parameter to implement the synchronization of the first service.

[00184] It is to be understood that in the embodiment of the disclosure, the second service may quickly find a parameter change event by regularly comparing the differences between a local parameter and a remote parameter, and by designing an event-driven model in advance and defining an event to be executed after the parameter change in advance, the updated remote parameter is synchronized to the local for loading in time through an event trigger response mechanism, so that the changed configuration parameter may take effect in real time on the second service, thereby improving the parameter configuration efficiency.

[00185] In some embodiments, before the operation that the target parameter modification event corresponding to the target parameter name in the second preset event registry is triggered according to the target parameter in S202, the following steps may also be executed to generate the second preset event registry. The second preset event registry is acquired based on the at least one second preset parameter corresponding to the second service and at least one second preset parameter loading method corresponding to the at least one second preset parameter under the condition of initializing the second service. In the second preset event registry, the at least one second

preset parameter is in one-to-one correspondence with at least one second preset parameter modification event, and each of the at least one second preset parameter loading method is a callback function corresponding to the at least one second preset parameter modification event.

[00186] The above process of generating the second preset event registry on the second host is the same as that described in the same processes S001-S003 on the first host, which is not repeated here.

[00187] Referring to FIG. 11, FIG. 11 is an optional flowchart of a data processing method on a second host according to an embodiment of the disclosure. Based on FIG. 10, S202 may be implemented by executing S202-1 that the target parameter is determined from the second remote parameter under the condition that the comparison result represents that the second local parameter is inconsistent with the second remote parameter and S202-2 that the target parameter modification event corresponding to the target parameter name in the second preset event registry is triggered according to the target parameter. After S202-1, S205-S206 may also be executed, which will be described with reference to the steps shown in FIG. 11.

[00188] In S205, the target parameter is read from the remote storage space through the unified calling interface, and the remote storage space containing a sub-storage space of at least one storage type.

[00189] In the embodiment of the disclosure, the second host may use the unified calling interface to shield different read-write methods corresponding to the sub-storage space of at least one storage type, and read the target parameter in the remote storage space to the local according to the unified read-write operation implemented in the unified calling interface.

[00190] In S206, a second local parameter value corresponding to the target parameter name in the second local storage is updated according to the target parameter name of the target parameter.

[00191] In the embodiment of the disclosure, the second host determines the corresponding second local target parameter in the second local storage according to the

target parameter name of the target parameter, and updates the parameter value of the second local target parameter by using the target parameter value of the target parameter, thereby implementing timely synchronization of parameter change in the remote storage space and the local storage space.

[00192] It is to be understood that unified abstraction of the read-write operation on different types of sub-storage spaces is implemented by executing calling through the unified calling interface, so that the second service may manage the system configuration parameters of each storage location indistinguishably, thereby improving the efficiency and maintainability of parameter configuration.

[00193] Referring to FIG. 12, FIG. 12 is a flowchart of a process of sharing a parameter by a first service and a second service according to an embodiment of the disclosure, which will be described with reference to the steps shown in FIG. 12.

[00194] In the embodiment of the disclosure, multiple services may use the same storage space at the same time in a distributed deployment scenario. For example, the first service and the second service share parameters and use the remote storage space jointly. Due to the fact that the parameter modification request of the client may only be sent to the first service that is currently providing services, in order to prevent the inconsistency between the cached parameter values of the multiple services and the inconsistency between the services, the second service periodically compares the remote storage space with the second local storage space through a parameter management program, for example, whether the parameter values in the second local cache change or not. When it is found through comparison that the remote storage space is inconsistent with the data corresponding to the configuration parameters of the second local cache, the second service sends a request to the remote storage space, and the remote storage space responds to the request, quickly synchronizes the target parameter containing a new parameter value to the parameter management program, updates the target parameter in the second local cache through the parameter management program, and triggers the target parameter modification event, namely the second parameter modification event, to load the target parameter to the second service, so that the new parameter value in the target parameter takes effect in real time.

[00195] Hereinafter, an exemplary application of the embodiment of the disclosure in an actual application scenario will be described in conjunction with FIG. 13.

[00196] As shown in FIG. 13, the remote storage space in an actual data processing system may be a storage space set containing different types of storage devices, such as a MySQL database, an object storage service and a local configuration file. By abstracting a unified calling interface as a unified configuration management interface, the data processing system processes IO operations of different storage types through the unified calling interface, so that multiple business services in the data processing system can manage the system configuration parameters of each storage location indistinguishably. In the initialization stage, the business service in the data processing system may read at least one of the first preset parameters from the remote storage space and load it to itself through the unified calling interface to complete initialization and start normally, and generate its corresponding preset event registry based on the at least one first preset parameter. As such, when a business service receives a parameter modification request for itself or finds that its corresponding local parameters are inconsistent with those in the remote storage space, it may execute a writing process corresponding to parameter change, trigger the target parameter modification event corresponding to the target parameter in the preset event registry, and call the target callback function corresponding to the target parameter modification event to load the target parameter to itself, thus implementing the status synchronization between the services during distributed deployment.

[00197] The following continues to explain the exemplary structure of the first data processing apparatus 255-1 implemented as a software part according to the embodiment of the disclosure. In some embodiments, as shown in FIG. 2, the software part stored in the first data processing apparatus 255-1 of a first memory 250-1 may include a first triggering part 2551, a first calling part 2552 and a recording part 2553.

[00198] The first triggering part 2551 is configured to trigger the target parameter modification event in the first preset event registry according to the target parameter in the parameter modification request under the condition of receiving the parameter modification request for the first service, and update a remote parameter corresponding to

the target parameter in the remote storage space through the unified calling interface. The first preset event registry represents the mapping relationship among a first preset parameter, a first preset parameter modification event and a first preset callback function in the first service, and the remote storage space is configured to store at least one remote parameter of at least one service.

[00199] The first calling part 2552 is configured to call the target callback function corresponding to the target parameter modification event by using the target parameter, and load the target parameter to the first service.

[00200] The recording part 2553 is configured to run the first service and record the game status of the game area based on the target parameter.

[00201] In some embodiments, the first data processing apparatus 255-1 may also include a first registration part. The first registration part is configured to obtain at least one first preset parameter corresponding to the first service and a first preset parameter loading method corresponding to each of the at least one first preset parameter when the first service is initialized before the target parameter modification event in the first preset event registry is triggered according to the target parameter in the parameter modification request, the target parameter belonging to the at least one first preset parameter, and the first preset parameter loading method being a parameter loading method configured in the first service in advance; generate a first preset parameter modification event corresponding to each of the first preset parameters for each of the first preset parameters, the first preset callback function corresponding to the first preset parameter modification event being the first preset parameter loading method corresponding to each of the first preset parameters, the target parameter modification event being a first preset parameter modification event corresponding to the target parameter in the first preset event registry, the target callback function being the first preset callback function corresponding to the target parameter modification event; and correspondingly register the first preset parameter modification event on each of the first preset parameters to obtain the mapping relationship among the first preset parameter, the first preset parameter modification event and the first preset callback function as the first preset event registry.

[00202] In some embodiment, the first trigger part 2551 is further configured to call the preset event-driven method implemented in advance in the preset parameter modification interface and access the first preset event registry according to the target parameter when the parameter modification request is received through the preset parameter modification interface; determine a first preset parameter modification event corresponding to the target parameter as the target parameter modification event in the first preset event registry, and trigger the target parameter modification event through the preset event-driven method.

[00203] In some embodiments, the first calling part 2552 is also configured to determine a target callback function corresponding to the target parameter modification event according to the first preset event registry; call the target callback function by using the target parameter through the preset event-driven method; and transfer the target parameter into the target preset parameter loading method implemented in the target callback function, and load the target parameter to the first service through a target predicted parameter loading method. The target preset parameter loading method is the first preset parameter loading method corresponding to the target parameter in at least one first preset parameter loading method, and the at least one first preset parameter loading method is a preset parameter loading method corresponding to the at least one first preset parameter in the first service.

[00204] In some embodiments, the first triggering part 2551 is also configured to access the remote storage space through the unified calling interface, and determine a target remote parameter from the at least one remote parameter according to a target parameter name of the target parameter, the remote storage space containing a sub-storage space of at least one storage type, the unified calling interface being configured to implement a read-write operation on the sub-storage space of at least one storage type in a unified calling mode, the target remote parameter being a shared parameter of the first service and the second service, the second service being configured to synchronize service data of the first service; and update the target remote parameter by using the target parameter, and synchronously update the target parameter to the second service.

[00205] In some embodiments, the first data processing apparatus 255-1 may further

include a first updating part. The first updating part is configured to update the first local target parameter corresponding to the target parameter in the first local storage according to the target parameter. The first local storage is the storage space of the first host where the first service is located.

[00206] The following continues to explain the exemplary structure of the first data processing apparatus 255-2 implemented as a software part according to the embodiment of the disclosure. In some embodiments, as shown in FIG. 3, the software part stored in the second data processing apparatus 255-2 of a second memory 250-2 may include a comparison part 2551, a second triggering part 2552, a second triggering part 2553 and a synchronization part 2554.

[00207] The comparison part 2551 is configured to compare a second local parameter of the second service in second local storage with a second remote parameter in the remote storage space through a unified calling interface according to a preset time interval to obtain a comparison result. The second local storage is a storage space on a second host where the second service is located. The second remote parameter may contain a shared parameter of the first service and the second service. The first service is configured to record the game status of a game area.

[00208] The second triggering part 2552 is configured to determine the target parameter from the second remote parameter under the condition that the comparison result represents that the second local parameter is inconsistent with the second remote parameter; and trigger a target parameter modification event corresponding to a target parameter name in a second preset event registry according to the target parameter. The second preset event registry represents a mapping relationship among a second preset parameter name, a second preset parameter modification event and a second preset callback function in the second service.

[00209] The second calling part 2553 is configured to load the target parameter to the second service by calling a target callback function corresponding to the target parameter modification event.

[00210] The synchronization part 2554 is configured to run the second service and

synchronize the first service based on the target parameter.

[00211] In some embodiments, the second data processing apparatus 255-2 may further include a second registration part. The second registration part is configured to obtain a second preset event registry based on at least one second preset parameter corresponding to the second service and at least one second preset parameter loading method corresponding to the at least one second preset parameter under the condition that the second service is initialized before the target parameter modification event corresponding to the target parameter name in the second preset event registry is triggered according to the target parameter. In the second preset event registry, the at least one second preset parameter is in one-to-one correspondence with at least one second preset parameter modification event. Each of the at least one second preset parameter loading method is a callback function corresponding to the at least one second preset parameter modification event.

[00212] In some embodiments, the second data processing apparatus 255-2 may further include a second updating part. The second updating part is configured to read the target parameter from the remote storage space through the unified calling interface after the target parameter is determined from the second remote parameter, the remote storage space containing a sub-storage space of at least one storage type; and update the second local target parameter corresponding to the target parameter in the first local storage according to the target parameter, the second local storage being the storage space of a second host where the second service is located.

[00213] It is to be noted that the description of the above device embodiment is similar to the description of the above method embodiment, and has similar beneficial effects as the method embodiment. Technical details undisclosed in the device embodiment of the disclosure are understood with reference to the descriptions about the method embodiment of the disclosure.

[00214] The embodiments of the disclosure provide a computer program product or a computer program, which includes a computer instruction stored in a computer-readable storage medium. A processor of a computer device reads the computer instruction from the computer-readable storage medium. The processor executes the computer instruction

to enable the computer device to execute the data processing method of the embodiments of the disclosure.

[00215] The embodiments of the disclosure provide a computer-readable storage medium storing an executable instruction. The executable instruction is stored therein, and when executed by a processor, the executable instruction will cause the processor to execute the methods provided by the embodiments of the disclosure, for example, the methods shown in FIGS. 4-8, and FIGS. 10 and 11.

[00216] In some embodiments, the computer-readable storage medium may be FRAM, ROM, PROM, EPROM, EEPROM, flash memory, magnetic surface memory, optical disk, CD-ROM and other memories, and may also be various devices including one or any combination of the above memories.

[00217] In some embodiments, the executable instruction may be compiled according to a programming language of any form (including a compiling or interpretive language, or a declarative or procedural language) in form of a program, software, a software module, a script, or a code, and may be deployed according to any form, including deployed as an independent program or deployed as a module, a component, a subroutine or another unit suitable to be used in a computing environment.

[00218] As an example, the executable instruction may but not always correspond to a file in a file system, and may be stored in a part of a file that stores another program or data, for example, stored in one or more scripts in a Hyper Text Markup Language (HTML) document, stored in a single file dedicated to a discussed program, or stored in multiple collaborative files (for example, files storing one or more modules, subprograms or code parts).

[00219] As an example, the executable instruction may be deployed in a computing device for execution, or executed in multiple computing devices at the same place, or executed in multiple computing devices that are interconnected through a communication network at multiple places.

[00220] In conclusion, through the embodiments of the disclosure, under the condition that the parameter modification request is received, the corresponding target parameter

modification event may be triggered according to the target parameter, and then as a response to the target parameter modification event, the corresponding target callback function may be called to load the target parameter to the first service in real time, so that the first service may record the game status based on a new target parameter value, the callback function of the parameter modification event may be configured as a parameter loading method in the event registry in advance, the parameter modification request may be responded by an event triggering mechanism, and the corresponding callback function may be called in real time to reach the effect that the parameter takes effect immediately after being changed. In addition, for the remote storage space composed of different types of storage devices in a real scenario, the corresponding remote parameters in the remote storage space may be managed and updated indistinguishably through the unified calling interface, thereby implementing the parameter synchronization of at least one service and greatly improving the parameter configuration efficiency. Moreover, the second service may quickly find a parameter change event by regularly comparing the differences between a local parameter and a remote parameter, and synchronize the updated remote parameter to the local for loading in time through an event trigger response mechanism by designing an event-driven model in advance and defining an event to be executed after the parameter change in advance, so that the changed remote parameter may take effect in real time in the remote storage space, thereby improving the parameter configuration efficiency. Calling is executed through the unified calling interface, and unified abstraction of the read-write operation on different types of sub-storage spaces is implemented, so that the services may manage the system configuration parameters of each storage location indistinguishably, thereby improving the efficiency and maintainability of parameter configuration.

[00221] The above is only the embodiment of the disclosure and not intended to limit the scope of protection of the disclosure. Any modifications, equivalent replacements, improvements and the like made within the spirit and scope of the disclosure fall within the scope of protection of the disclosure.

CLAIMS

1. A data processing method, comprising:

triggering a target parameter modification event in a first preset event registry according to a target parameter in a parameter modification request in a case of receiving the parameter modification request for a first service; updating a remote parameter corresponding to the target parameter in a remote storage space through a unified calling interface, the first preset event registry representing a mapping relationship among a first preset parameter, a first preset parameter modification event and a first preset callback function in the first service, and the remote storage space being configured to store at least one remote parameter of at least one service;

calling a target callback function corresponding to the target parameter modification event by using the target parameter, and loading the target parameter to the first service; and

running the first service, and recording a game status of a game area based on the target parameter.

2. The method of claim 1, before triggering the target parameter modification event in the first preset event registry according to the target parameter in the parameter modification request, further comprising:

acquiring at least one first preset parameter corresponding to the first service and a first preset parameter loading method corresponding to each first preset parameter of at least one first preset parameter in a case of initializing the first service, wherein the target parameter belongs to the at least one first preset parameter, and the first preset parameter loading method is a parameter loading method pre-configured in the first service;

for the each first preset parameter, generating a first preset parameter modification event corresponding to the each first preset parameter, wherein a first preset callback function corresponding to the first preset parameter modification event is the first preset parameter loading method corresponding to the each first preset parameter, the target parameter

modification event is a first preset parameter modification event corresponding to the target parameter in the first preset event registry, and the target callback function is the first preset callback function corresponding to the target parameter modification event; and

correspondingly registering the first preset parameter modification event on the each first preset parameter to obtain the mapping relationship among the first preset parameter, the first preset parameter modification event and the first preset callback function as the first preset event registry.

3. The method of claim 1 or 2, wherein triggering the target parameter modification event in the first preset event registry according to the target parameter in the parameter modification request in the case of receiving the parameter modification request for the first service comprises:

calling a preset event-driven method implemented in advance in a preset parameter modification interface and accessing the first preset event registry according to the target parameter in a case of receiving the parameter modification request through the preset parameter modification interface; and

determining the first preset parameter modification event corresponding to the target parameter as the target parameter modification event in the first preset event registry, and triggering the target parameter modification event through the preset event-driven method.

4. The method of claim 3, wherein calling the target callback function corresponding to the target parameter modification event by using the target parameter and loading the target parameter to the first service comprises:

determining the target callback function corresponding to the target parameter modification event according to the first preset event registry; and

calling the target callback function by using the target parameter through the preset event-driven method, transferring the target parameter into the target preset parameter loading

method implemented in the target callback function, and loading the target parameter to the first service through a target predicted parameter loading method, the target preset parameter loading method being the first preset parameter loading method corresponding to the target parameter in at least one first preset parameter loading method, and the at least one first preset parameter loading method being a preset parameter loading method corresponding to the at least one first preset parameter in the first service.

5. The method of claim 1 or 2, wherein updating the remote parameter corresponding to the target parameter in the remote storage space through the unified calling interface comprises:

accessing the remote storage space through the unified calling interface, and determining a target remote parameter from at least one remote parameter according to a target parameter name of the target parameter, the remote storage space containing a sub-storage space of at least one storage type, the unified calling interface being configured to implement read-write operation on the sub-storage space of the at least one storage type in a unified calling mode, the target remote parameter being a shared parameter of the first service and a second service, and the second service being configured to synchronize service data of the first service; and

updating the target remote parameter by using the target parameter, and synchronously updating the target parameter to the second service.

6. The method of claim 1 or 2, further comprising:

updating a first local target parameter corresponding to the target parameter in first local storage according to the target parameter, the first local storage being the storage space of a first host where the first service is located.

7. A data processing method, comprising:

comparing a second local parameter of a second service in second local storage with a second remote parameter in a remote storage space through a unified calling interface according to a preset time interval to obtain a comparison result, the second local storage being a storage space of a second host where the second service is located, the second remote parameter containing a shared parameter of a first service and the second service, and the first service being configured to record a game status of a game area;

determining a target parameter from the second remote parameter in a case where the comparison result represents that the second local parameter is inconsistent with the second remote parameter; triggering a target parameter modification event corresponding to a target parameter name in a second preset event registry according to the target parameter, the second preset event registry representing a mapping relationship among a second preset parameter name, a second preset parameter modification event and a second preset callback function in the second service;

loading the target parameter to the second service by calling a target callback function corresponding to the target parameter modification event; and

running the second service, and synchronizing the first service based on the target parameter.

8. The method of claim 7, before triggering the target parameter modification event corresponding to the target parameter name in the second preset event registry according to the target parameter, further comprising:

obtaining a second preset event registry based on at least one second preset parameter corresponding to the second service and at least one second preset parameter loading method corresponding to the at least one second preset parameter in a case of initializing the second service; in the second preset event registry, the at least one second preset parameter being in one-to-one correspondence with at least one second preset parameter modification event, and each second preset parameter of the at least one second preset parameter loading method being a callback function corresponding to the at least one second preset parameter modification event.

9. The method of claim 7 or 8, after determining the target parameter from the second remote parameter, further comprising:

reading the target parameter from the remote storage space through the unified calling interface, the remote storage space containing a sub-storage space of at least one storage type; and

updating a second local target parameter corresponding to the target parameter in first local storage according to the target parameter, the first local storage being the storage space of a second host where the second service is located.

10. A first host, comprising:

a first memory, configured to store an executable instruction; and

a first processor, configured to execute the executable instruction stored in the first memory to:

trigger a target parameter modification event in a first preset event registry according to a target parameter in a parameter modification request in a case of receiving the parameter modification request for a first service; update a remote parameter corresponding to the target parameter in a remote storage space through a unified calling interface, the first preset event registry representing a mapping relationship among a first preset parameter, a first preset parameter modification event and a first preset callback function in the first service, and the remote storage space being configured to store at least one remote parameter of at least one service;

call a target callback function corresponding to the target parameter modification event by using the target parameter, and load the target parameter to the first service; and

run the first service, and record a game status of a game area based on the target parameter.

11. The first host of claim 10, wherein before triggering the target parameter modification event in the first preset event registry according to the target parameter in the parameter modification request, the first processor is configured to:

acquire at least one first preset parameter corresponding to the first service and a first preset parameter loading method corresponding to each first preset parameter of at least one first preset parameter in a case of initializing the first service, wherein the target parameter belongs to the at least one first preset parameter, and the first preset parameter loading method is a parameter loading method pre-configured in the first service;

for the each first preset parameter, generate a first preset parameter modification event corresponding to the each first preset parameter, wherein a first preset callback function corresponding to the first preset parameter modification event is the first preset parameter loading method corresponding to the each first preset parameter, the target parameter modification event is a first preset parameter modification event corresponding to the target parameter in the first preset event registry, and the target callback function is the first preset callback function corresponding to the target parameter modification event; and

correspondingly register the first preset parameter modification event on the each first preset parameter to obtain the mapping relationship among the first preset parameter, the first preset parameter modification event and the first preset callback function as the first preset event registry.

12. The first host of claim 10 or 11, wherein when triggering the target parameter modification event in the first preset event registry according to the target parameter in the parameter modification request in the case of receiving the parameter modification request for the first service, the first processor is configured to:

call a preset event-driven method implemented in advance in a preset parameter modification interface and accessing the first preset event registry according to the target parameter in a case of receiving the parameter modification request through the preset

parameter modification interface; and

determine the first preset parameter modification event corresponding to the target parameter as the target parameter modification event in the first preset event registry, and trigger the target parameter modification event through the preset event-driven method.

13. The first host of claim 12, wherein when calling the target callback function corresponding to the target parameter modification event by using the target parameter and loading the target parameter to the first service, the first processor is configured to:

determine the target callback function corresponding to the target parameter modification event according to the first preset event registry; and

call the target callback function by using the target parameter through the preset event-driven method, transferring the target parameter into the target preset parameter loading method implemented in the target callback function, and load the target parameter to the first service through a target predicted parameter loading method, the target preset parameter loading method being the first preset parameter loading method corresponding to the target parameter in at least one first preset parameter loading method, and the at least one first preset parameter loading method being a preset parameter loading method corresponding to the at least one first preset parameter in the first service.

14. The first host of claim 10 or 11, wherein when updating the remote parameter corresponding to the target parameter in the remote storage space through the unified calling interface, the first processor is configured to:

access the remote storage space through the unified calling interface, and determining a target remote parameter from at least one remote parameter according to a target parameter name of the target parameter, the remote storage space containing a sub-storage space of at least one storage type, the unified calling interface being configured to implement read-write operation on the sub-storage space of the at least one storage type in a unified calling mode,

the target remote parameter being a shared parameter of the first service and a second service, and the second service being configured to synchronize service data of the first service; and

update the target remote parameter by using the target parameter, and synchronously update the target parameter to the second service.

15. The first host of claim 10 or 11, wherein the first processor is further configured to:

update a first local target parameter corresponding to the target parameter in first local storage according to the target parameter, the first local storage being the storage space of a first host where the first service is located.

16. A second host, comprising:

a second memory, configured to store an executable instruction; and

a second processor, configured to execute the executable instruction stored in second memory to:

compare a second local parameter of a second service in second local storage with a second remote parameter in a remote storage space through a unified calling interface according to a preset time interval to obtain a comparison result, the second local storage being a storage space of a second host where the second service is located, the second remote parameter containing a shared parameter of a first service and the second service, and the first service being configured to record a game status of a game area;

determine a target parameter from the second remote parameter in a case where the comparison result represents that the second local parameter is inconsistent with the second remote parameter; trigger a target parameter modification event corresponding to a target parameter name in a second preset event registry according to the target parameter, the second preset event registry representing a mapping relationship among a second preset parameter name, a second preset parameter modification event and a second preset callback function in

the second service;

load the target parameter to the second service by calling a target callback function corresponding to the target parameter modification event; and

run the second service, and synchronize the first service based on the target parameter.

17. The second host of claim 16, wherein before triggering the target parameter modification event corresponding to the target parameter name in the second preset event registry according to the target parameter, the second processor is further configured to:

obtain a second preset event registry based on at least one second preset parameter corresponding to the second service and at least one second preset parameter loading method corresponding to the at least one second preset parameter in a case of initializing the second service; in the second preset event registry, the at least one second preset parameter being in one-to-one correspondence with at least one second preset parameter modification event, and each second preset parameter of the at least one second preset parameter loading method being a callback function corresponding to the at least one second preset parameter modification event.

18. The second host of claim 16 or 17, wherein after determining the target parameter from the second remote parameter, the second processor is further configured to:

read the target parameter from the remote storage space through the unified calling interface, the remote storage space containing a sub-storage space of at least one storage type; and

update a second local target parameter corresponding to the target parameter in first local storage according to the target parameter, the first local storage being the storage space of a second host where the second service is located.

19. A computer-readable storage medium, storing an executable instruction, configured to be executed by a first processor to:

acquire at least one first preset parameter corresponding to the first service and a first preset parameter loading method corresponding to each first preset parameter of at least one first preset parameter in a case of initializing the first service, wherein the target parameter belongs to the at least one first preset parameter, and the first preset parameter loading method is a parameter loading method pre-configured in the first service;

for the each first preset parameter, generate a first preset parameter modification event corresponding to the each first preset parameter, wherein a first preset callback function corresponding to the first preset parameter modification event is the first preset parameter loading method corresponding to the each first preset parameter, the target parameter modification event is a first preset parameter modification event corresponding to the target parameter in the first preset event registry, and the target callback function is the first preset callback function corresponding to the target parameter modification event; and

correspondingly register the first preset parameter modification event on the each first preset parameter to obtain the mapping relationship among the first preset parameter, the first preset parameter modification event and the first preset callback function as the first preset event registry.

20. A computer-readable storage medium, storing an executable instruction, configured to be executed by a second processor to:

compare a second local parameter of a second service in second local storage with a second remote parameter in a remote storage space through a unified calling interface according to a preset time interval to obtain a comparison result, the second local storage being a storage space of a second host where the second service is located, the second remote parameter containing a shared parameter of a first service and the second service, and the first service being configured to record a game status of a game area;

determine a target parameter from the second remote parameter in a case where the

comparison result represents that the second local parameter is inconsistent with the second remote parameter; trigger a target parameter modification event corresponding to a target parameter name in a second preset event registry according to the target parameter, the second preset event registry representing a mapping relationship among a second preset parameter name, a second preset parameter modification event and a second preset callback function in the second service;

load the target parameter to the second service by calling a target callback function corresponding to the target parameter modification event; and

run the second service, and synchronize the first service based on the target parameter.