

METHODS AND APPARATUSES FOR PUSHING A MESSAGE

TECHNICAL FIELD

[001] The present application relates to message pushing technologies, and more particularly, to message pushing methods and apparatuses.

BACKGROUND

[002] With continuous development of communication technologies, message pushing techniques may have been applied to numerous scenarios. For example, in a mobile client application, data may be actively pushed to a mobile device from the background. As another example, in a Java platform enterprise edition (Java EE) application, data may be actively pushed from one application system to another application system.

SUMMARY

[003] The present disclosure provides message pushing methods and apparatuses to solve deficiency in the related art.

[004] In some aspects, the present disclosure is directed to a method for pushing a message. The method may include extracting, when the message to be pushed is detected, user information and application information in the message. The method may also include acquiring link information corresponding to the user information and the application information. The method may further include determining registered devices of a user account and registered applications on each registered device in accordance with the user information, the application information, and the link information. In addition, the method may include adding, when a plurality of registered applications on a registered device exist, identification information about the plurality of registered applications to the message corresponding to the registered device. The identification information and message contents contained in the message may be extracted, and the message contents may be respectively

sent to each registered application corresponding to the identification information by the registered device after receiving the message.

[005] In some aspects, the present disclosure is directed to another method for pushing a message. The method may include receiving a pushed message. The method may also include extracting message contents and identification information contained in the pushed message. The method may further include determining a target application corresponding to each piece of the identification information, and sending the message contents to each corresponding target application.

[006] In some aspects, the present disclosure is directed to an apparatus for pushing a message. The apparatus may include an extraction unit configured to extract, when the message to be pushed is detected, user information and application information in the message. The apparatus may also include an acquisition unit configured to acquire link information corresponding to the user information and the application information. The apparatus may further include a determination unit configured to determine registered devices of a user account and registered applications on each registered device in accordance with the user information, the application information, and the link information. In addition, the apparatus may include an adding unit configured to add, when a plurality of registered applications on a registered device exist, identification information about the plurality of registered applications to the message corresponding to the registered device. The identification information and message contents contained in the message may be extracted, and the message contents may be respectively sent to each registered application corresponding to the identification information by the registered device after receiving the message.

[007] In some aspects, the present disclosure is directed to another apparatus for pushing a message. The apparatus may include a receiving unit configured to receive a

pushed message. The apparatus may also include an extraction unit configured to extract message contents and identification information contained in the pushed message. The apparatus may further include a determining and sending unit configured to determine a target application corresponding to each piece of the identification information, and send the message contents to each corresponding target application.

[008] Message pushing methods and apparatuses in the present disclosure may include acquiring, when a message to be pushed is detected, user information and application information in the message and the corresponding link information to learn real-time, accurate login conditions of a user account. By adding identification information about a plurality of registered applications corresponding to the registered device to the message, only one message may need to be sent to the same registered device. All the plurality of registered applications may receive their corresponding message contents. It may be helpful to decrease the data volume of messages to be pushed, and may reduce power consumption of the registered device.

BRIEF DESCRIPTION OF THE DRAWINGS

[009] **Fig. 1** illustrates a schematic structural diagram of an exemplary real-time message platform, according to some embodiments of the present disclosure.

[010] **Fig. 2** is a flow chart of an exemplary method for pushing a message, according to some embodiments of the present disclosure.

[011] **Fig. 3** is a flow chart of another exemplary method for pushing a message, according to some embodiments of the present disclosure.

[012] **Fig. 4** illustrates a processing flow chart of exemplary client registration, according to some embodiments of the present disclosure.

[013] **Fig. 5** shows a processing flow chart of exemplary client logout, according to some embodiments of the present disclosure.

[014] **Fig. 6** is a processing flow chart of an exemplary abnormal disconnection of a client, according to some embodiments of the present disclosure.

[015] **Fig. 7** is a flow chart of another exemplary method for pushing a message, according to some embodiments of the present disclosure.

[016] **Fig. 8** shows a schematic diagram of an exemplary merging process, according to some embodiments of the present disclosure.

[017] **Fig. 9** is a schematic diagram of an exemplary grouping process, according to some embodiments of the present disclosure.

[018] **Fig. 10** illustrates a schematic structural diagram of an exemplary electronic device, according to some embodiments of the present disclosure.

[019] **Fig. 11** is a block diagram of an exemplary apparatus for pushing a message, according to some embodiments of the present disclosure.

[020] **Fig. 12** shows a schematic structural diagram of another exemplary electronic device, according to some embodiments of the present disclosure.

[021] **Fig. 13** is a block diagram of another exemplary apparatus for pushing a message, according to some embodiments of the present disclosure.

DETAILED DESCRIPTION

[022] **Fig. 1** illustrates a schematic structural diagram of an exemplary real-time message platform, according to some embodiments of the present disclosure. As shown in **Fig. 1**, a service message from a service server may be processed by a real-time message platform and may be pushed to a client. **Fig. 2** is a flow chart of an exemplary method for pushing a message, according to some embodiments of the present disclosure. The method

may be applied to a message center (MSCenter) shown in **Fig. 1**. The method may include the following steps:

[023] Step 202: Extract, when the message to be pushed is detected, user information and application information in the message.

[024] Step 204: Acquire link information corresponding to the user information and the application information.

[025] In some embodiments, after a user logs in to an account on a device, the device may establish a link with a client access server (MSNode), and may initiate a registration request based on the link. The registration request may contain user information and application information. In some embodiments, the MSNode may also read link information from the link. The MSNode may send the link information, the application information, and the user information to a client link management center (CMCenter) for storing the correspondence between them.

[026] The user information may be information representing the user identity, such as user account information for login. The application information may be information representing a corresponding application program on the device, such as an application ID. The link information may correspond to information about a link that may be established with the MSNode and/or information about a device corresponding to the link, such as a link ID and/or a device ID. Each device may establish a plurality of links with the MSNode.

[027] Step 206: Determine registered devices of a user account and registered applications on each registered device in accordance with the user information, the application information, and the link information.

[028] In some embodiments, the MSCenter may acquire, from each message to be pushed, user information about one or more users and one or more pieces of application information corresponding to each piece of the user information to form one or more “user

information - application information” pairs. In addition, combining with the link information acquired at the CMCenter, the method may also include determining the real-time login state of the user account.

[029] In some embodiments, if the MSCenter acquires, according to a “user information - application information” pair, one corresponding piece of link information or a plurality of pieces of identical link information from the CMCenter, it may indicate that the user may log in by the user account on one application program on one device.

[030] In some embodiments, if the MSCenter acquires, according to a “user information - application information” pair, a corresponding plurality of pieces of different link information from the CMCenter, it may indicate that the user may log in by the same user account on the same application program on a plurality of devices.

[031] In some embodiments, if the MSCenter acquires, according to a plurality of “user information - application information” pairs corresponding to the same user account, the corresponding one piece of link information or a plurality of pieces of identical link information from the CMCenter, it may indicate that the user may log in by the same user account on a plurality of application programs on one device.

[032] Step 208: Add, when a plurality of registered applications on a registered device exist, identification information about the plurality of registered applications to the message corresponding to the registered device. The identification information and message contents contained in the message may be extracted and the message contents may be respectively sent to each registered application corresponding to the identification information by the registered device after receiving the message.

[033] In some embodiments, the identification information about the registered applications may be contained in the corresponding application information. For example, the identification information may be the application IDs or other types of identification to

distinguish the plurality of registered applications. By adding identification information about a plurality of registered applications to a message to be pushed, the method may need only one message to be pushed to a device. The plurality of registered applications may all receive the corresponding message contents. It may be helpful to decrease the data volume of messages to be pushed, and may reduce power consumption of the device.

[034] Corresponding to the processing procedure of the MSCenter described above, the same registered device above may process a received pushing message via a processing procedure shown in **Fig. 3** to ensure that a plurality of registered applications may all receive the message contents. **Fig. 3** is a flow chart of another exemplary method for pushing a message, according to some embodiments of the present disclosure. As shown in **Fig. 3**, the method may be applied to a terminal or a device. The method may include the following steps:

[035] Step 302: Receive a pushed message.

[036] In some embodiments, the pushed message in step 302 may be the message to be pushed shown in **Fig. 2**.

[037] Step 304: Extract message contents and identification information contained in the pushed message.

[038] Step 306: Determine a target application corresponding to each piece of the identification information, and send the message contents to each corresponding target application.

[039] Accordingly, the methods for pushing a message in the present disclosure may include acquiring, when a message to be pushed is detected, user information and application information in the message and the corresponding link information to learn real-time, accurate login conditions of a user account. By adding identification information about a plurality of registered applications corresponding to the registered device to the message,

only one message may need to be sent to the same registered device. All the plurality of registered applications may receive their corresponding message contents. It may be helpful to decrease the data volume of messages to be pushed, and may reduce power consumption of the registered device.

[040] In addition to the above methods for a single device, the present disclosure is also directed to methods for a plurality of devices. When a plurality of registered devices corresponding to the same user account exist, the message to be pushed may be respectively pushed to each of the registered devices.

[041] In some embodiments, the MSCenter may duplicate the message to obtain a plurality of messages that may be one-to-one corresponding to the plurality of registered devices. The MSCenter may also acquire a node ID of a client access server corresponding to each of the registered devices, and may add each node ID to each corresponding message to be pushed accordingly. The MSCenter may add these messages each containing the node ID to a message pushing queue. Finally, the MSNode may push these messages in the message pushing queue.

[042] In some embodiments, a user may log in by an account respectively via applications on a plurality of devices. The plurality of devices may all receive the corresponding pushing messages to ensure that the user may be able to view the pushed message via any one of the devices.

[043] **Figs. 4-6** illustrate several exemplary processes, including a client registration process, a corresponding logout process, and an abnormal disconnection process. These processes are further described below.

[044] 1. Registration Process

[045] **Fig. 4** illustrates a processing flow chart of exemplary client registration, according to some embodiments of the present disclosure. As shown in **Fig. 4**, a registration procedure may include the following steps:

[046] Step 402: Initiate, by a client, a registration request to a current MSNode node.

[047] Step 404: Parse, by the current MSNode node, a protocol of the registration request.

[048] Step 406: Perform, by the current MSNode node based on a protocol parsing result, identity authentication and authorization on a user account that initiates the registration request.

[049] Step 408: Control a number of connections on the current MSNode node that processes the registration request.

[050] In some embodiments, the MSNode may be a server cluster containing a plurality of nodes. Each MSNode node may be configured with a preset number of connections. If the current MSNode node detects that the number of established connections exceeds the preset number of connections, the current registration request may be allocated to another idle MSNode node.

[051] The current MSNode node may return a route instruction to the client to guide and re-connect the client to another idle MSNode node.

[052] Step 410: Acquire, by the current MSNode node, user information and/or application information from the registration request, and extract link information from an established link.

[053] Step 412A: Send, by the current MSNode node, the user information, the application information, and the link information to a CMCenter.

[054] Step 414A: Update data in accordance with the received user information, application information, and link information if client registration information about the user account exists in the CMCenter. Register as new client registration information if client registration information about the user account does not exist.

[055] Step 416A: Store, by the CMCenter, the client registration information in a link management database or a distributed cache system.

[056] Step 412B: Send, by the current MSNode node, the user information and the application information to the MSCenter.

[057] Step 414B: Acquire, by the current MSCenter node, the corresponding off-line data in accordance with the user information and the application information.

[058] Step 416B: Clean, by the current MSCenter node, the off-line data.

[059] Step 418: Generate, by the current MSCenter node, a message to be pushed from the processed off-line data, and add the message to a message pushing queue.

[060] Step 420: Acquire, by the current MSNode node, the message to be pushed from the message pushing queue, and push the message to the client.

[061] 2. Logout Process

[062] **Fig. 5** shows a processing flow chart of exemplary client logout, according to some embodiments of the present disclosure. As shown in **Fig. 5**, the client logout procedure may include the following steps:

[063] Step 502: Initiate, by a client, a logout request to a current MSNode node.

[064] Step 504: Parse, by the current MSNode node, a protocol of the received logout request.

[065] Step 506: Perform, by the current MSNode node based on a protocol parsing result, identity authentication and authorization on the client.

[066] Step 508: Acquire, by the current MSNode node, user information and application information contained in the logout request, and delete all links established between the client and the current MSNode from a link manager.

[067] Step 510: Send, by the current MSNode node, a link logout command to a CMCenter.

[068] Step 512: Delete, by the CMCenter, connection information established between the current MSNode and the client, including deleting the corresponding link information from a database or a distributed cache. The client may be a device on which the client program is installed.

[069] 3. Abnormal Disconnection Process

[070] **Fig. 6** is a processing flow chart of an exemplary abnormal disconnection of a client, according to some embodiments of the present disclosure. As shown in **Fig. 6**, the abnormal disconnection procedure may include the following steps:

[071] Step 602: A client being abnormally disconnected from an MSNode node.

[072] Step 604: Capture, by the current MSNode node, abnormal disconnection information, and send a client link disconnection event.

[073] Step 606: Acquire, in response to the client link disconnection event, link information registered by the client with the current MSNode from a link manager of the current MSNode node, and delete the link information.

[074] Step 608: Send, by the current MSNode node, a link logout command to a CMCenter.

[075] Step 610: Delete, by the CMCenter, connection information established between the current MSNode and the client, including deleting the corresponding link information from a database or a distributed cache.

[076] By these procedures in **Figs. 4-6**, the CMCenter may record the user information, the application information, and the link information as client registration information.

[077] The MSCenter may accurately push a message based on the recorded client registration information. Referring to **Fig. 7**, a message pushing process may be described in detail below. **Fig. 7** is a flow chart of another exemplary method for pushing a message, according to some embodiments of the present disclosure. As shown in **Fig. 7**, when the real-time message platform shown in **Fig. 1** may be used for pushing a message, the method may include the following steps:

[078] Step 702: Receive, by a message agent center (MSAgent), a service application message delivered by a service server, and process, re-combine, and fill in the message in accordance with message header information in the message.

[079] Step 704: Generate, by the MSAgent, a message to be scheduled based on the processed, re-combined, and filled-in message, and add the message to a message scheduling queue.

[080] Step 706: Monitor, by a message center (MSCenter), whether a consumable message exists in the scheduling queue.

[081] Step 708: Acquire, by the MSCenter, the consumable message if the consumable message exists in the scheduling queue.

[082] Step 710: Clean, by the MSCenter, the message.

[083] Step 712, Acquire, by the MSCenter, user information and application information from a message header of the message.

[084] Step 714: Acquire, by the MSCenter, the corresponding link information from a client link management center (CMCenter) in accordance with the acquired user information and application information.

[085] In some embodiments, the link information, as shown in **Figs. 4-6**, may be stored in client registration information in a database or a distributed cache by the CMCenter.

[086] Step 716: Merge or group, by the MSCenter, the messages in accordance with the user information, the application information, and the link information.

[087] In some embodiments, the merging process and the grouping process above may be applied to different scenarios. The two processes are described below.

[088] 1) Merging Process

[089] In a scenario where a user may log in by the same user account respectively on a plurality of applications on the same device, the merging process may be performed on the messages so that only one message may need to be pushed to the device. The plurality of applications may acquire the corresponding message contents accordingly. It may decrease the data volume of pushed messages, and may reduce power consumption of the device. Especially for a mobile terminal device, this method may be helpful to reduce the data traffic consumption and prolong the standby time.

[090] **Fig. 8** shows a schematic diagram of an exemplary merging process, according to some embodiments of the present disclosure. As shown in **Fig. 8**, a user may use the same user account, and may respectively log in on an application A, an application B, and an application C on a device 1. After detecting a message to be pushed in a message scheduling queue, the MSCenter may be configured to extract user information, e.g., user ID = 111, and application information, e.g., one of application ID1 = A, application ID2 = B, or application ID3 = C, in a message header of the message. The MSCenter may also be configured to acquire the corresponding link information, e.g., one of device ID = 1 or device ID1 = device ID2 = ... = 1, from a CMCenter. The MSCenter may further configured to determine that the same user account corresponds to device 1 and application A,

application B, and application C on device 1. This is the scenario where a plurality of applications correspond to the same device.

[091] Accordingly, the MSCenter may be configured to extract IDs of the application A, the application B, and the application C from link metadata information, such as application ID1, application ID2, and application ID3. The MSCenter may also be configured to add all the ID information to the same message, and add the message to a message pushing queue. After acquiring the message from the message pushing queue, the MSNode may be configured to directly send the message to device 1.

[092] Correspondingly, after receiving the pushed message, device 1 may be configured to acquire the ID information, i.e., application ID1, application ID2, and application ID3, and message contents contained in the received message by parsing the received message. Device 1 may also be configured to send the message contents to application A, application B, and application C corresponding to application ID1, application ID2, and application ID3, respectively.

[093] By the above merging process, although application A, application B, and application C on the device 1 all have the user account logged in, the method may need only one message to be pushed. All the applications may obtain the corresponding message contents accordingly. It may effectively decrease the data volume of messages to be pushed, especially when the device 1 may be a mobile device. As a result, the data traffic consumption may be reduced accordingly. Moreover, message pushing may be performed directly via only one link with the MSNode. It may reduce power consumption of the device 1, and may also save electric quantity of power supply.

[094] 2) Grouping Process

[095] In a scenario where a user may log in by the same user account on the same application on a plurality of devices, the grouping process on messages may be performed so

that each of the devices may receive the same pushed message. The user may be able to read the corresponding pushed message via any one of the devices. It may be helpful to avoid missing the message.

[096] **Fig. 9** is a schematic diagram of an exemplary grouping process, according to some embodiments of the present disclosure. As shown in **Fig. 9**, a user may use the same user account to log in on the same application A on a device 1, a device 2 and a device 3, respectively. After detecting a message to be pushed from a message scheduling queue, the MSCenter may be configured to extract user information, e.g., user ID = 111, and application information, e.g., application ID = A, in a message header of the message, and acquire the corresponding link information, e.g., device ID1 = 1, device ID2 = 2, or device ID3 = 3, from a CMCenter. The MSCenter may also be configured to determine that the same user account corresponds to the application A on the device 1, the device 2 and the device 3. This is the scenario where a plurality of devices may correspond to the same application.

[097] Accordingly, the MSCenter may take each device as a group, may divide one message to be pushed into a plurality of messages that are one-to-one corresponding to the plurality of devices, and may push the plurality of messages via an MSNode. The MSCenter may determine an MSNode node corresponding to each of the devices in accordance with link information. For example, as shown in **Fig. 9**, the device 1 may correspond to an MSNode node 1, the device 2 may correspond to an MSNode node 2, and the device 3 may correspond to an MSNode node 3. An ID of the MSNode node 1, an ID of the MSNode node 2, and an ID of the MSNode node 3 may be respectively added to their corresponding messages to be pushed. The messages to be pushed may be added to a message pushing queue. Each of the MSNode nodes may select the corresponding message to be pushed from the message pushing queue in accordance with the ID thereof and may perform a pushing

operation, such as pushing the corresponding message to the device 2 via the MSNode node 2 or pushing the corresponding message to the device 3 via the MSNode node 3.

[098] Referring back to **Fig. 7**, after the message merging/grouping of step 716, the following steps can be performed:

[099] Step 718: Add, by the MSCenter, a message selector path into the message, e.g., a node ID of a client access server (MSNode) for pushing the message, in accordance with the link information.

[0100] Step 720: Add, by the MSCenter, the message to a pushing queue.

[0101] Step 722: Monitor, by the MSNode, whether a consumable message exists in the pushing queue. If a consumable message exists in the pushing queue, extract the corresponding message from the pushing queue. The MSNode may generally contain a plurality of nodes, and each node may have a corresponding unique ID. By detecting a node ID contained in each message in the pushing queue, the message may be extracted and processed by the corresponding node.

[0102] Step 724: Acquire, by the MSNode, the link information from the extracted message, and read a link connection instance corresponding to the link information from a local link manager.

[0103] Step 726: Encapsulate, by the MSNode, the link connection instance and the message content in the message into a pushing event, and broadcast it.

[0104] Step 728: Push, by a local pushing event listener of the MSNode, the message content to the client using the corresponding link connection instance after receiving the broadcast pushing event.

[0105] Step 730: Return, by the client, an acknowledgment (ACK) after receiving the pushed message content.

[0106] Step 732: Perform, by the MSNode, a certain process in accordance with a receiving condition of the ACK.

[0107] In some embodiments, if the MSNode has not received the ACK from the client within a specified period of time, a re-sending operation may be performed in step 732. In some embodiments, when the re-sending operation may be performed for a certain number of times, the message may be sent to a failure processing queue. A failure message processor may perform a grading process on messages in accordance with a type of the failure queue. For example, for a Front type queue, the message re-sending may be attempted. Messages in a Backend queue may be recovered manually to the Front queue, or may be re-sent directly by a manual trigger. Messages in a dead letter queue (DLQ) may only be re-sent by a manual trigger.

[0108] **Fig. 10** illustrates a schematic structural diagram of an exemplary electronic device corresponding to an MSCenter, according to some embodiments of the present disclosure. Referring to **Fig. 10**, at the hardware level, the electronic device may include a processor, an internal bus, a network interface, a memory, and a non-volatile memory. In some embodiments, the electronic device may also include additional hardware required by other services. The processor may be configured to read the corresponding computer program from the non-volatile memory into the memory for execution. The processor at the logical level may be configured to be an apparatus for pushing a message. In addition to software implementation, the present disclosure does not exclude any other implementations, such as a logical device or a method combining hardware and software. The execution entities of the following procedures may not be limited to various logical units, and may also be hardware or a logical device.

[0109] **Fig. 11** is a block diagram of an exemplary apparatus for pushing a message, according to some embodiments of the present disclosure. In some embodiments, an

apparatus for pushing a message may include an extraction unit, an acquisition unit, a determination unit and a pushing unit.

[0110] The extraction unit may be configured to extract, when the message to be pushed is detected, user information and application information in the message.

[0111] The acquisition unit may be configured to acquire link information corresponding to the user information and the application information.

[0112] The determination unit may be configured to determine registered devices of a user account and registered applications on each registered device in accordance with the user information, the application information, and the link information.

[0113] The adding unit may be configured to add, when a plurality of registered applications on a registered device exist, identification information about the plurality of registered applications to the message corresponding to the registered device. The identification information and message contents contained in the message may be extracted and the message contents may be respectively sent to each registered application corresponding to the identification information by the registered device after receiving the message.

[0114] In some embodiments, the apparatus for pushing the message may further include a pushing unit configured to push, when a plurality of registered devices corresponding to the user account exist, the message to each registered device.

[0115] In some embodiments, the pushing unit may be configured to duplicate the message to obtain a plurality of messages corresponding to the plurality of registered devices. The pushing unit may also be configured to acquire a node identity (ID) of a client access server corresponding to each registered device, and add each node ID to each corresponding message. The pushing unit may further be configured to add the plurality of messages each containing the node ID to a message pushing queue.

[0116] In some embodiments, the acquisition unit may be configured to acquire, from a client link management center, link information corresponding to the user information and the application information. The link information may be acquired from a link established with a device when the client access server registers the device.

[0117] **Fig. 12** shows a schematic structural diagram of another exemplary electronic device, according to some embodiments of the present disclosure. At the hardware level, the electronic device may include a processor, an internal bus, a network interface, a memory and a non-volatile memory. In some embodiments, the electronic device may also include additional hardware required by other services. The processor may read the corresponding computer program from the non-volatile memory into the memory for execution. In some embodiments, the processor at the logical level may be configured to form an apparatus for pushing a message. In addition to software implementation, the present disclosure does not exclude any other implementations, such as a logical device or a method combining hardware and software. The execution entities of the following processing procedure may not be limited to various logical units, and may also be hardware or a logical device.

[0118] **Fig. 13** is a block diagram of another exemplary apparatus for pushing a message, according to some embodiments of the present disclosure. In some embodiments, an apparatus for pushing a message may include a receiving unit, an extraction unit, and a sending unit.

[0119] The receiving unit may be configured to receive a pushed message.

[0120] The extraction unit may be configured to extract message contents and identification information contained in the pushed message.

[0121] The determining and sending unit may be configured to determine a target application corresponding to each piece of the identification information, and send the message contents to each corresponding target application.

[0122] In a typical configuration, a computing device may include one or more processors (CPUs), an input/output interface, a network interface and a memory.

[0123] The memory may include a non-permanent memory, a random access memory (RAM) and/or a non-volatile memory, such as a read-only memory (ROM) or flash memory (flash RAM), in computer-readable media. The memory may be an example of the computer-readable media.

[0124] The computer-readable media may include permanent and non-permanent, removable and non-removable media and may realize information storage with any method or technology. The information may be a computer-readable instruction, data structure, program module or other data. The examples of computer storage media may include but are not limited to a phase change memory (PRAM), static random access memory (SRAM), dynamic random access memory (DRAM), other types of random access memories (RAMs), read-only memory (ROM), electrically erasable programmable read-only memory (EEPROM), flash memory or other memory technologies, compact disk read-only memory (CD-ROM), digital versatile disc (DVD) or other optical storage, magnetic cassette tape, magnetic tape and magnetic disk storage or other magnetic storage devices or any other non-transmission media, which can be used to store information that can be accessed by a computing device. According to the definition herein, the computer-readable media do not include transitory media, such as modulated data signals and carriers.

[0125] It also needs to be noted that the terms “include,” “comprise,” or any other variations thereof are intended to cover a non-exclusive inclusion. Therefore, in the context of a process, method, merchant, or device that includes a series of elements, the process, method, merchant, or device may not only include such elements, but may also include other elements not specified explicitly, or may also include inherent elements of the process, method, merchant, or device. If there are no more limitations, elements defined by

“comprising a” do not exclude the situation where other identical elements exist in the process, method, merchant or device including the elements.

[0126] The descriptions presented above are only preferred embodiments of the present disclosure, and are not intended to limit the present disclosure. Any modification, equivalent replacement, improvement, etc. within the spirit and principle of the present application shall all fall within the scope of protection of the present disclosure.

WHAT IS CLAIMED IS:

1. A method for pushing a message, the method comprising:
 - extracting, when the message to be pushed is detected, user information and application information in the message to be pushed;
 - acquiring link information corresponding to the user information and the application information;
 - determining corresponding registered devices of a same user account and corresponding registered applications on each registered device in accordance with the user information, the application information, and the link information; and
 - adding, when a plurality of registered applications on a registered device exist, identification information about the plurality of registered applications to a message to be pushed corresponding to the registered device,wherein the registered device extracts, after receiving the message to be pushed, the identification information and message contents contained in the message, and sends the message contents respectively to each registered application corresponding to the identification information.
2. The method according to claim 1, further comprising:
 - pushing, when a plurality of registered devices corresponding to the same user account exist, the message to be pushed respectively to each registered device.

3. The method according to claim 2, wherein pushing, when the plurality of registered devices corresponding to the same user account exist, the message to be pushed respectively to each registered device includes:
 - duplicating the message to be pushed to obtain a plurality of messages to be pushed respectively corresponding to the plurality of registered devices;
 - respectively acquiring a node identity (ID) of a client access server corresponding to each registered device, and adding the node ID to the corresponding message to be pushed; and
 - adding the message to be pushed containing the node ID to a message pushing queue.
4. The method according to claim 1, wherein acquiring link information corresponding to the user information and the application information includes:
 - acquiring, from a client link management center, link information corresponding to the user information and the application information, wherein the link information is acquired from a link established with a device when the client access server registers the device.
5. A method for pushing a message, comprising:
 - receiving a pushed message;
 - extracting message contents and all identification information contained in the pushed message; and

respectively determining a target application corresponding to each piece of the identification information, and sending the message contents to the corresponding target application.

6. An apparatus for pushing a message, the apparatus comprising:

an extraction unit configured to extract, when the message to be pushed is detected, user information and application information in the message to be pushed;

an acquisition unit configured to acquire link information corresponding to the user information and the application information;

a determination unit configured to determine corresponding registered devices of a same user account and corresponding registered applications on each registered device in accordance with the user information, the application information, and the link information;

and

an adding unit configured to add, when a plurality of registered applications on a registered device exist, identification information about the plurality of registered applications to a message to be pushed corresponding to the registered device,

wherein the registered device extracts, after receiving the message to be pushed, the identification information and message contents contained in the message, and sends the message contents respectively to each registered application corresponding to the identification information.

7. The apparatus according to claim 6, further comprising:

a pushing unit configured to push, when a plurality of registered devices corresponding to the same user account exist, the message to be pushed respectively to each registered device.

8. The apparatus according to claim 7, wherein the pushing unit is configured to:
duplicate the message to be pushed to obtain a plurality of messages to be pushed respectively corresponding to the plurality of registered devices;
respectively acquire a node identity (ID) of a client access server corresponding to each registered device, and add the node ID to the corresponding message to be pushed; and
add the message to be pushed containing the node ID to a message pushing queue.
9. The apparatus according to claim 6, wherein the acquisition unit is configured to:
acquire, from a client link management center, link information corresponding to the user information and the application information, wherein the link information is acquired from a link established with a device when the client access server registers the device.
10. An apparatus for pushing a message, the apparatus comprising:
a receiving unit configured to receive a pushed message;
an extraction unit configured to extract message contents and all identification information contained in the pushed message; and

a sending unit configured to respectively determine a target application corresponding to each piece of the identification information, and send the message content to the corresponding target application.