(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
04 April 2024 (04.04.2024)

WIPO | PCT

(10) International Publication Number
WO 2024/065147 A1

(51) International Patent Classification:
G06F 15/16 (2006.01)

(21) International Application Number:
PCT/CN2022/121549

(22) International Filing Date:
27 September 2022 (27.09.2022)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **CITRIX SYSTEMS, INC.** [US/US]; 851 West Cypress Creek Road, Fort Lauderdale, Florida 33309 (US).

(72) Inventors: **ZHANG, Yuan**; c/o Citrix Systems Information Technology (Beijing) Ltd. Nanjing Branch, C3 Building, No. 19 Suyuan Avenue, Jiangning District, Nanjing, Jiangsu 211106 (CN). **QIAO, Zongpeng**; c/o Citrix Systems Information Technology (Beijing) Ltd. Nanjing Branch, C3 Building, No. 19 Suyuan Avenue, Jiangning District, Nanjing, Jiangsu 211106 (CN). **XU, Ke**; c/o Citrix Systems Information Technology (Beijing) Ltd. Nanjing Branch, C3 Building, No. 19 Suyuan Avenue, Jiangning District, Nanjing, Jiangsu 211106 (CN).

(74) Agent: **AFD CHINA INTELLECTUAL PROPERTY LAW OFFICE**; Tower B, 21st Fl., Suite 2108, 38 Xueqing Road, Haidian, Beijing 100083 (CN).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA,

(54) Title: GROUP MANAGEMENT

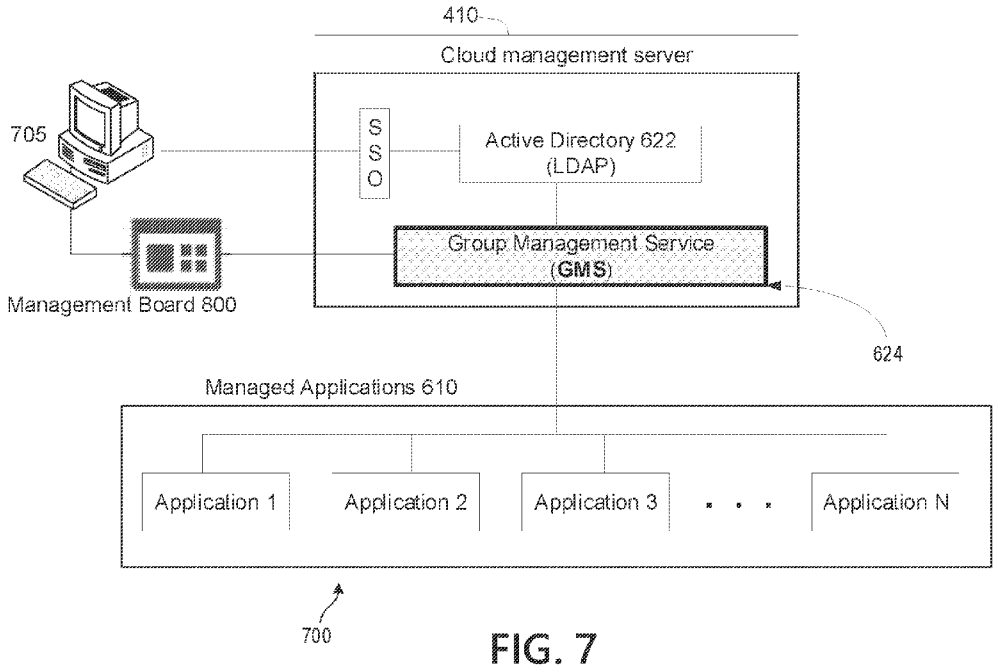


FIG. 7

(57) Abstract: Methods and apparatus for managing groups are described herein. A group management service may centrally manage a plurality of applications such that group information in the plurality of applications may stay up to date with changes in the group information. The group management service may monitor/receive different types of messages from different classes of users (e.g., a group member, a group creator/owner, a system administrator, etc.) and may synchronize, based on the messages, the group information across the plurality of applications. A group member may send, to the group management service and via an application, a message (e.g., quit membership from a group) for changes in the group information. A system administrator may send, to the group management service and via a graphical user interface, a message (e.g., cancel/add a member of a group, create a new group, etc.) for changes in the group information.

WO 2024/065147 A1

[Continued on next page]

NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) **Designated States** *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report (Art. 21(3))*

GROUP MANAGEMENT

FIELD

[0001] Aspects described herein generally relate to computer networking, remote computer access, virtualization, enterprise mobility management, and hardware and software related thereto. More specifically, one or more aspects describe herein provide features to manage member information of a group associated with managed applications in virtual machine environment.

BACKGROUND

[0002] A group (e.g., an email group, a chat group, etc.) may be created/managed for facilitating communications/events among members of the group. A member of the group may, via one or more applications (e.g., Outlook, Teams, etc.), send group messages or schedule group events for other members of the group. Changes (e.g., canceling a member) in the group may occur and the one or more applications may have outdated/inconsistent group information with the changes.

SUMMARY

[0003] The following presents a simplified summary of various aspects described herein. This summary is not an extensive overview, and is not intended to identify required or critical elements or to delineate the scope of the claims. The following summary merely presents some concepts in a simplified form as an introductory prelude to the more detailed description provided below.

[0004] To overcome limitations in the prior art described above, and to overcome other limitations that will be apparent upon reading and understanding the present specification, aspects described herein are directed towards methods and apparatus for managing groups.

[0005] An apparatus may perform a method comprising multiple operations. The computing device may authenticate, by a primary group management service layer associated with the apparatus and based on user information. The apparatus may grant the computing device, based on the authentication, access to a plurality of applications managed by a secondary group management service layer associated with the apparatus. The apparatus may receive, from the computing device via a graphical management service user interface associated with the secondary group management service layer, a message indicating to update data associated with a user group. The apparatus may update, based on the message, data stored

in a centralized data storage managed by the secondary group management service layer. The apparatus may identify, based on the data stored in the centralized data storage, one or more applications, of the plurality of applications, associated with the user group. The apparatus may synchronize, by the secondary group management service layer associated with the apparatus, data stored in one or more distributed data storages with the data stored in the centralized data storage, each distributed data storage associated with respective one of the identified one or more applications of the plurality of applications. The apparatus may, based on the synchronized data and via one or more graphical application user interfaces, each associated with respective one of the identified one or more applications, send, to one or more members of the user group, a notification of updated data associated with the user group.

[0006] The apparatus may also perform one or more additional operations. The apparatus, based on a lightweight directory access protocol (LDAP) and via the secondary group management service layer, may access, by the primary group management service layer, the plurality of applications, wherein each of the plurality of applications comprises a distributed data storage that is blocked, by a boundary, from other distributed data storages of other applications of the plurality of applications. The apparatus may perform, by the primary group management service layer, a single sign-on (SSO) based authentication process with the computing device. The apparatus may, based on successful completion of the SSO based authentication process, bypass boundaries, set by the primary group management service layer, between the distributed data storages associated with the plurality of applications. The apparatus may establish, by the secondary group management service layer, the graphical management service user interface between the computing device and the plurality of applications. The apparatus may bypass the primary group management service layer and manage, via the graphical management service user interface, the distributed data storages without the boundaries.

[0007] The apparatus may update member information of the user group stored in the centralized data storage by adding one or more new members to the user group, deleting one or more members of the user group, changing member contact information of the user group, or changing a class of one or more members of the user group. The apparatus may determine, based on the synchronized data, that one or more members of the user group are changed. The apparatus may resend, based on the determination, a notification of a previously scheduled event to the one or more members of the user group. The apparatus may detect a member of the user group searching, via one or more of the identified one or more applications, for the

user group. The apparatus may, based on the detection, indicate, to the member, a changed status of one or more members of the user group. The apparatus may prompt, based on the synchronized data and via one or more of the identified one or more applications, a text message indicating one or more changed statuses of one or more members of the user group and one or more names of the user group. The apparatus may send a notification of updated data including a message, the message indicating a name of the user group and indicating that the one or more members no longer belongs to the user group. The apparatus may, via the secondary group management service layer, manage a centralized data storage, wherein the centralized data storage comprises a data structure storing a name of the user group, a creator of the user group, a type of the user group, member information of the user group, a creation time of the user group, and a last update time of the user group.

[0008] These and additional aspects will be appreciated with the benefit of the disclosures discussed in further detail below.

BRIEF DESCRIPTION OF THE DRAWINGS

[0009] A more complete understanding of aspects described herein and the advantages thereof may be acquired by referring to the following description in consideration of the accompanying drawings, in which like reference numbers indicate like features, and wherein:

[0010] Figure 1 depicts an illustrative computer system architecture that may be used in accordance with one or more illustrative aspects described herein.

[0011] Figure 2 depicts an illustrative remote-access system architecture that may be used in accordance with one or more illustrative aspects described herein.

[0012] Figure 3 depicts an illustrative virtualized system architecture that may be used in accordance with one or more illustrative aspects described herein.

[0013] Figure 4 depicts an illustrative cloud-based system architecture that may be used in accordance with one or more illustrative aspects described herein.

[0014] Figure 5 depicts an illustrative enterprise mobility management system.

[0015] Figure 6 depicts another illustrative enterprise mobility management system.

[0016] Figure 7 depicts an illustrative group management system.

[0017] Figure 8 depicts an illustrative graphical user interface (GUI).

[0018] Figure 9 depicts illustrative actions performed via application(s).

[0019] Figure 10 depicts illustrative interactions between application(s) and the GUI .

[0020] Figure 11 depicts illustrative message sequences for synchronizing applications initiated by an administrator.

[0021] Figure 12 depicts illustrative message sequences for synchronizing applications initiated by a group member.

[0022] Figure 13 depicts illustrative alternative message sequences for synchronizing applications initiated by a group member.

[0023] Figure 14 depicts an illustrative flowchart performed by group management system.

DETAILED DESCRIPTION

[0024] In the following description of the various embodiments, reference is made to the accompanying drawings identified above and which form a part hereof, and in which is shown by way of illustration various embodiments in which aspects described herein may be practiced. It is to be understood that other embodiments may be utilized and structural and functional modifications may be made without departing from the scope described herein. Various aspects are capable of other embodiments and of being practiced or being carried out in various different ways.

[0025] As a general introduction to the subject matter described in more detail below, aspects described herein are directed towards managing groups associated with a plurality of applications. A group management service may centrally manage the applications such that group information across the applications may stay up to date/consistent with changes in the group information. The group management service may monitor/receive different types of messages from different classes of users (e.g., a group member, a group creator/owner, a system administrator, etc.). The group management service may receive, from the group member and via one (e.g., Slack) of the applications, a message (e.g., quit membership from a group) causing/initiating changes in the group information. Alternatively, the group management service may receive, from the system administrator and via a graphical user interface (GUI) (e.g., management board), a message (e.g., cancel/add a member of a group, create a new group, etc.) causing changes in the group information. The group management service may synchronize, based on the message from the group member or the message from the system administrator, the group information across the applications and may confirm the changes made in the group information. The users may advantageously utilize the group management service for centrally managing/synchronizing the group information across the applications (e.g.,

Outlook, Slack, Teams, JIRA, Wrike, ShareFile, or any SaaS, etc.). The users may avoid inconveniences of accessing every one of the applications to update/synchronize the group information in the applications each time changes occur in the group information.

[0026] It is to be understood that the phraseology and terminology used herein are for the purpose of description and should not be regarded as limiting. Rather, the phrases and terms used herein are to be given their broadest interpretation and meaning. The use of "including" and "comprising" and variations thereof is meant to encompass the items listed thereafter and equivalents thereof as well as additional items and equivalents thereof. The use of the terms "mounted," "connected," "coupled," "positioned," "engaged" and similar terms, is meant to include both direct and indirect mounting, connecting, coupling, positioning and engaging.

[0027] **COMPUTING ARCHITECTURE**

[0028] Computer software, hardware, and networks may be utilized in a variety of different system environments, including standalone, networked, remote-access (also known as remote desktop), virtualized, and/or cloud-based environments, among others. FIG. 1 illustrates one example of a system architecture and data processing device that may be used to implement one or more illustrative aspects described herein in a standalone and/or networked environment. Various network nodes 103, 105, 107, and 109 may be interconnected via a wide area network (WAN) 101, such as the Internet. Other networks may also or alternatively be used, including private intranets, corporate networks, local area networks (LAN), metropolitan area networks (MAN), wireless networks, personal networks (PAN), and the like. Network 101 is for illustration purposes and may be replaced with fewer or additional computer networks. A local area network 133 may have one or more of any known LAN topology and may use one or more of a variety of different protocols, such as Ethernet. Devices 103, 105, 107, and 109 and other devices (not shown) may be connected to one or more of the networks via twisted pair wires, coaxial cable, fiber optics, radio waves, or other communication media.

[0029] The term "network" as used herein and depicted in the drawings refers not only to systems in which remote storage devices are coupled together via one or more communication paths, but also to stand-alone devices that may be coupled, from time to time, to such systems that have storage capability. Consequently, the term "network" includes not only a "physical network" but also a "content network," which is comprised of the data—attributable to a single entity—which resides across all physical networks.

[0030] The components may include data server 103, web server 105, and client computers 107, 109. Data server 103 provides overall access, control and administration of databases and control software for performing one or more illustrative aspects describe herein. Data server 103 may be connected to web server 105 through which users interact with and obtain data as requested. Alternatively, data server 103 may act as a web server itself and be directly connected to the Internet. Data server 103 may be connected to web server 105 through the local area network 133, the wide area network 101 (e.g., the Internet), via direct or indirect connection, or via some other network. Users may interact with the data server 103 using remote computers 107, 109, e.g., using a web browser to connect to the data server 103 via one or more externally exposed web sites hosted by web server 105. Client computers 107, 109 may be used in concert with data server 103 to access data stored therein, or may be used for other purposes. For example, from client device 107 a user may access web server 105 using an Internet browser, as is known in the art, or by executing a software application that communicates with web server 105 and/or data server 103 over a computer network (such as the Internet).

[0031] Servers and applications may be combined on the same physical machines, and retain separate virtual or logical addresses, or may reside on separate physical machines. FIG. 1 illustrates just one example of a network architecture that may be used, and those of skill in the art will appreciate that the specific network architecture and data processing devices used may vary, and are secondary to the functionality that they provide, as further described herein. For example, services provided by web server 105 and data server 103 may be combined on a single server.

[0032] Each component 103, 105, 107, 109 may be any type of known computer, server, or data processing device. Data server 103, e.g., may include a processor 111 controlling overall operation of the data server 103. Data server 103 may further include random access memory (RAM) 113, read only memory (ROM) 115, network interface 117, input/output interfaces 119 (e.g., keyboard, mouse, display, printer, etc.), and memory 121. Input/output (I/O) 119 may include a variety of interface units and drives for reading, writing, displaying, and/or printing data or files. Memory 121 may further store operating system software 123 for controlling overall operation of the data processing device 103, control logic 125 for instructing data server 103 to perform aspects described herein, and other application software 127 providing secondary, support, and/or other functionality which may or might not be used in conjunction with aspects described herein. The control logic 125 may also be referred to herein

as the data server software 125. Functionality of the data server software 125 may refer to operations or decisions made automatically based on rules coded into the control logic 125, made manually by a user providing input into the system, and/or a combination of automatic processing based on user input (e.g., queries, data updates, etc.).

[0033] Memory 121 may also store data used in performance of one or more aspects described herein, including a first database 129 and a second database 131. In some embodiments, the first database 129 may include the second database 131 (e.g., as a separate table, report, etc.). That is, the information can be stored in a single database, or separated into different logical, virtual, or physical databases, depending on system design. Devices 105, 107, and 109 may have similar or different architecture as described with respect to device 103. Those of skill in the art will appreciate that the functionality of data processing device 103 (or device 105, 107, or 109) as described herein may be spread across multiple data processing devices, for example, to distribute processing load across multiple computers, to segregate transactions based on geographic location, user access level, quality of service (QoS), etc.

[0034] One or more aspects may be embodied in computer-usable or readable data and/or computer-executable instructions, such as in one or more program modules, executed by one or more computers or other devices as described herein. Generally, program modules include routines, programs, objects, components, data structures, etc. that perform particular tasks or implement particular abstract data types when executed by a processor in a computer or other device. The modules may be written in a source code programming language that is subsequently compiled for execution, or may be written in a scripting language such as (but not limited to) HyperText Markup Language (HTML) or Extensible Markup Language (XML). The computer executable instructions may be stored on a computer readable medium such as a nonvolatile storage device. Any suitable computer readable storage media may be utilized, including hard disks, CD-ROMs, optical storage devices, magnetic storage devices, solid state storage devices, and/or any combination thereof. In addition, various transmission (non-storage) media representing data or events as described herein may be transferred between a source and a destination in the form of electromagnetic waves traveling through signal-conducting media such as metal wires, optical fibers, and/or wireless transmission media (e.g., air and/or space). Various aspects described herein may be embodied as a method, a data processing system, or a computer program product. Therefore, various functionalities may be embodied in whole or in part in software, firmware, and/or hardware or hardware equivalents such as integrated circuits, field programmable gate arrays (FPGA), and the like. Particular

data structures may be used to more effectively implement one or more aspects described herein, and such data structures are contemplated within the scope of computer executable instructions and computer-usable data described herein.

[0035] With further reference to FIG. 2, one or more aspects described herein may be implemented in a remote-access environment. FIG. 2 depicts an example system architecture including a computing device 201 in an illustrative computing environment 200 that may be used according to one or more illustrative aspects described herein. Computing device 201 may be used as a server 206a in a single-server or multi-server desktop virtualization system (e.g., a remote access or cloud system) and can be configured to provide virtual machines for client access devices. The computing device 201 may have a processor 203 for controlling overall operation of the device 201 and its associated components, including RAM 205, ROM 207, Input/Output (I/O) module 209, and memory 215.

[0036] I/O module 209 may include a mouse, keypad, touch screen, scanner, optical reader, and/or stylus (or other input device(s)) through which a user of computing device 201 may provide input, and may also include one or more of a speaker for providing audio output and one or more of a video display device for providing textual, audiovisual, and/or graphical output. Software may be stored within memory 215 and/or other storage to provide instructions to processor 203 for configuring computing device 201 into a special purpose computing device in order to perform various functions as described herein. For example, memory 215 may store software used by the computing device 201, such as an operating system 217, application programs 219, and an associated database 221.

[0037] Computing device 201 may operate in a networked environment supporting connections to one or more remote computers, such as terminals 240 (also referred to as client devices and/or client machines). The terminals 240 may be personal computers, mobile devices, laptop computers, tablets, or servers that include many or all of the elements described above with respect to the computing device 103 or 201. The network connections depicted in FIG. 2 include a local area network (LAN) 225 and a wide area network (WAN) 229, but may also include other networks. When used in a LAN networking environment, computing device 201 may be connected to the LAN 225 through a network interface or adapter 223. When used in a WAN networking environment, computing device 201 may include a modem or other wide area network interface 227 for establishing communications over the WAN 229, such as computer network 230 (e.g., the Internet). It will be appreciated that the network connections shown are illustrative and other means of establishing a communications link between the

computers may be used. Computing device 201 and/or terminals 240 may also be mobile terminals (e.g., mobile phones, smartphones, personal digital assistants (PDAs), notebooks, etc.) including various other components, such as a battery, speaker, and antennas (not shown).

[0038] Aspects described herein may also be operational with numerous other general purpose or special purpose computing system environments or configurations. Examples of other computing systems, environments, and/or configurations that may be suitable for use with aspects described herein include, but are not limited to, personal computers, server computers, hand-held or laptop devices, multiprocessor systems, microprocessor-based systems, set top boxes, programmable consumer electronics, network personal computers (PCs), minicomputers, mainframe computers, distributed computing environments that include any of the above systems or devices, and the like.

[0039] As shown in FIG. 2, one or more client devices 240 may be in communication with one or more servers 206a-206n (generally referred to herein as "server(s) 206"). In one embodiment, the computing environment 200 may include a network appliance installed between the server(s) 206 and client machine(s) 240. The network appliance may manage client/server connections, and in some cases can load balance client connections amongst a plurality of backend servers 206.

[0040] The client machine(s) 240 may in some embodiments be referred to as a single client machine 240 or a single group of client machines 240, while server(s) 206 may be referred to as a single server 206 or a single group of servers 206. In one embodiment a single client machine 240 communicates with more than one server 206, while in another embodiment a single server 206 communicates with more than one client machine 240. In yet another embodiment, a single client machine 240 communicates with a single server 206.

[0041] A client machine 240 can, in some embodiments, be referenced by any one of the following non-exhaustive terms: client machine(s); client(s); client computer(s); client device(s); client computing device(s); local machine; remote machine; client node(s); endpoint(s); or endpoint node(s). The server 206, in some embodiments, may be referenced by any one of the following non-exhaustive terms: server(s), local machine; remote machine; server farm(s), or host computing device(s).

[0042] In one embodiment, the client machine 240 may be a virtual machine. The virtual machine may be any virtual machine, while in some embodiments the virtual machine may be any virtual machine managed by a Type 1 or Type 2 hypervisor, for example, a hypervisor

developed by Citrix Systems, IBM, VMware, or any other hypervisor. In some aspects, the virtual machine may be managed by a hypervisor, while in other aspects the virtual machine may be managed by a hypervisor executing on a server 206 or a hypervisor executing on a client 240.

[0043] Some embodiments include a client device 240 that displays application output generated by an application remotely executing on a server 206 or other remotely located machine. In these embodiments, the client device 240 may execute a virtual machine receiver program or application to display the output in an application window, a browser, or other output window. In one example, the application is a desktop, while in other examples the application is an application that generates or presents a desktop. A desktop may include a graphical shell providing a user interface for an instance of an operating system in which local and/or remote applications can be integrated. Applications, as used herein, are programs that execute after an instance of an operating system (and, optionally, also the desktop) has been loaded.

[0044] The server 206, in some embodiments, uses a remote presentation protocol or other program to send data to a thin-client or remote-display application executing on the client to present display output generated by an application executing on the server 206. The thin-client or remote-display protocol can be any one of the following non-exhaustive list of protocols: the Independent Computing Architecture (ICA) protocol developed by Citrix Systems, Inc. of Ft. Lauderdale, Florida; or the Remote Desktop Protocol (RDP) manufactured by the Microsoft Corporation of Redmond, Washington.

[0045] A remote computing environment may include more than one server 206a-206n such that the servers 206a-206n are logically grouped together into a server farm 206, for example, in a cloud computing environment. The server farm 206 may include servers 206 that are geographically dispersed while logically grouped together, or servers 206 that are located proximate to each other while logically grouped together. Geographically dispersed servers 206a-206n within a server farm 206 can, in some embodiments, communicate using a WAN (wide), MAN (metropolitan), or LAN (local), where different geographic regions can be characterized as: different continents; different regions of a continent; different countries; different states; different cities; different campuses; different rooms; or any combination of the preceding geographical locations. In some embodiments the server farm 206 may be administered as a single entity, while in other embodiments the server farm 206 can include multiple server farms.

[0046] In some embodiments, a server farm may include servers 206 that execute a substantially similar type of operating system platform (e.g., WINDOWS, UNIX, LINUX, iOS, ANDROID, etc.) In other embodiments, server farm 206 may include a first group of one or more servers that execute a first type of operating system platform, and a second group of one or more servers that execute a second type of operating system platform.

[0047] Server 206 may be configured as any type of server, as needed, e.g., a file server, an application server, a web server, a proxy server, an appliance, a network appliance, a gateway, an application gateway, a gateway server, a virtualization server, a deployment server, a Secure Sockets Layer (SSL) VPN server, a firewall, a web server, an application server or as a master application server, a server executing an active directory, or a server executing an application acceleration program that provides firewall functionality, application functionality, or load balancing functionality. Other server types may also be used.

[0048] Some embodiments include a first server 206a that receives requests from a client machine 240, forwards the request to a second server 206b (not shown), and responds to the request generated by the client machine 240 with a response from the second server 206b (not shown.) First server 206a may acquire an enumeration of applications available to the client machine 240 as well as address information associated with an application server 206 hosting an application identified within the enumeration of applications. First server 206a can then present a response to the client's request using a web interface, and communicate directly with the client 240 to provide the client 240 with access to an identified application. One or more clients 240 and/or one or more servers 206 may transmit data over network 230, e.g., network 101.

[0049] FIG. 3 shows a high-level architecture of an illustrative desktop virtualization system. As shown, the desktop virtualization system may be single-server or multi-server system, or cloud system, including at least one virtualization server 301 configured to provide virtual desktops and/or virtual applications to one or more client access devices 240. As used herein, a desktop refers to a graphical environment or space in which one or more applications may be hosted and/or executed. A desktop may include a graphical shell providing a user interface for an instance of an operating system in which local and/or remote applications can be integrated. Applications may include programs that execute after an instance of an operating system (and, optionally, also the desktop) has been loaded. Each instance of the operating system may be physical (e.g., one operating system per device) or virtual (e.g., many instances

of an OS running on a single device). Each application may be executed on a local device, or executed on a remotely located device (e.g., remoted).

[0050] A computer device 301 may be configured as a virtualization server in a virtualization environment, for example, a single-server, multi-server, or cloud computing environment. Virtualization server 301 illustrated in FIG. 3 can be deployed as and/or implemented by one or more embodiments of the server 206 illustrated in FIG. 2 or by other known computing devices. Included in virtualization server 301 is a hardware layer that can include one or more physical disks 304, one or more physical devices 306, one or more physical processors 308, and one or more physical memories 316. In some embodiments, firmware 312 can be stored within a memory element in the physical memory 316 and can be executed by one or more of the physical processors 308. Virtualization server 301 may further include an operating system 314 that may be stored in a memory element in the physical memory 316 and executed by one or more of the physical processors 308. Still further, a hypervisor 302 may be stored in a memory element in the physical memory 316 and can be executed by one or more of the physical processors 308.

[0051] Executing on one or more of the physical processors 308 may be one or more virtual machines 332A-C (generally 332). Each virtual machine 332 may have a virtual disk 326A-C and a virtual processor 328A-C. In some embodiments, a first virtual machine 332A may execute, using a virtual processor 328A, a control program 320 that includes a tools stack 324. Control program 320 may be referred to as a control virtual machine, Dom0, Domain 0, or other virtual machine used for system administration and/or control. In some embodiments, one or more virtual machines 332B-C can execute, using a virtual processor 328B-C, a guest operating system 330A-B.

[0052] Virtualization server 301 may include a hardware layer 310 with one or more pieces of hardware that communicate with the virtualization server 301. In some embodiments, the hardware layer 310 can include one or more physical disks 304, one or more physical devices 306, one or more physical processors 308, and one or more physical memory 316. Physical components 304, 306, 308, and 316 may include, for example, any of the components described above. Physical devices 306 may include, for example, a network interface card, a video card, a keyboard, a mouse, an input device, a monitor, a display device, speakers, an optical drive, a storage device, a universal serial bus connection, a printer, a scanner, a network element (e.g., router, firewall, network address translator, load balancer, virtual private network (VPN) gateway, Dynamic Host Configuration Protocol (DHCP) router, etc.), or any device connected

to or communicating with virtualization server 301. Physical memory 316 in the hardware layer 310 may include any type of memory. Physical memory 316 may store data, and in some embodiments may store one or more programs, or set of executable instructions. FIG. 3 illustrates an embodiment where firmware 312 is stored within the physical memory 316 of virtualization server 301. Programs or executable instructions stored in the physical memory 316 can be executed by the one or more processors 308 of virtualization server 301.

[0053] Virtualization server 301 may also include a hypervisor 302. In some embodiments, hypervisor 302 may be a program executed by processors 308 on virtualization server 301 to create and manage any number of virtual machines 332. Hypervisor 302 may be referred to as a virtual machine monitor, or platform virtualization software. In some embodiments, hypervisor 302 can be any combination of executable instructions and hardware that monitors virtual machines executing on a computing machine. Hypervisor 302 may be Type 2 hypervisor, where the hypervisor executes within an operating system 314 executing on the virtualization server 301. Virtual machines may then execute at a level above the hypervisor 302. In some embodiments, the Type 2 hypervisor may execute within the context of a user's operating system such that the Type 2 hypervisor interacts with the user's operating system. In other embodiments, one or more virtualization servers 301 in a virtualization environment may instead include a Type 1 hypervisor (not shown). A Type 1 hypervisor may execute on the virtualization server 301 by directly accessing the hardware and resources within the hardware layer 310. That is, while a Type 2 hypervisor 302 accesses system resources through a host operating system 314, as shown, a Type 1 hypervisor may directly access all system resources without the host operating system 314. A Type 1 hypervisor may execute directly on one or more physical processors 308 of virtualization server 301, and may include program data stored in the physical memory 316.

[0054] Hypervisor 302, in some embodiments, can provide virtual resources to operating systems 330 or control programs 320 executing on virtual machines 332 in any manner that simulates the operating systems 330 or control programs 320 having direct access to system resources. System resources can include, but are not limited to, physical devices 306, physical disks 304, physical processors 308, physical memory 316, and any other component included in hardware layer 310 of the virtualization server 301. Hypervisor 302 may be used to emulate virtual hardware, partition physical hardware, virtualize physical hardware, and/or execute virtual machines that provide access to computing environments. In still other embodiments, hypervisor 302 may control processor scheduling and memory partitioning for a virtual

machine 332 executing on virtualization server 301. Hypervisor 302 may include those manufactured by VMWare, Inc., of Palo Alto, California; HyperV, VirtualServer or virtual PC hypervisors provided by Microsoft, or others. In some embodiments, virtualization server 301 may execute a hypervisor 302 that creates a virtual machine platform on which guest operating systems may execute. In these embodiments, the virtualization server 301 may be referred to as a host server. An example of such a virtualization server is the Citrix Hypervisor provided by Citrix Systems, Inc., of Fort Lauderdale, FL.

[0055] Hypervisor 302 may create one or more virtual machines 332B-C (generally 332) in which guest operating systems 330 execute. In some embodiments, hypervisor 302 may load a virtual machine image to create a virtual machine 332. In other embodiments, the hypervisor 302 may execute a guest operating system 330 within virtual machine 332. In still other embodiments, virtual machine 332 may execute guest operating system 330.

[0056] In addition to creating virtual machines 332, hypervisor 302 may control the execution of at least one virtual machine 332. In other embodiments, hypervisor 302 may present at least one virtual machine 332 with an abstraction of at least one hardware resource provided by the virtualization server 301 (e.g., any hardware resource available within the hardware layer 310). In other embodiments, hypervisor 302 may control the manner in which virtual machines 332 access physical processors 308 available in virtualization server 301. Controlling access to physical processors 308 may include determining whether a virtual machine 332 should have access to a processor 308, and how physical processor capabilities are presented to the virtual machine 332.

[0057] As shown in FIG. 3, virtualization server 301 may host or execute one or more virtual machines 332. A virtual machine 332 is a set of executable instructions that, when executed by a processor 308, may imitate the operation of a physical computer such that the virtual machine 332 can execute programs and processes much like a physical computing device. While FIG. 3 illustrates an embodiment where a virtualization server 301 hosts three virtual machines 332, in other embodiments virtualization server 301 can host any number of virtual machines 332. Hypervisor 302, in some embodiments, may provide each virtual machine 332 with a unique virtual view of the physical hardware, memory, processor, and other system resources available to that virtual machine 332. In some embodiments, the unique virtual view can be based on one or more of virtual machine permissions, application of a policy engine to one or more virtual machine identifiers, a user accessing a virtual machine, the applications executing on a virtual machine, networks accessed by a virtual machine, or any

other desired criteria. For instance, hypervisor 302 may create one or more unsecure virtual machines 332 and one or more secure virtual machines 332. Unsecure virtual machines 332 may be prevented from accessing resources, hardware, memory locations, and programs that secure virtual machines 332 may be permitted to access. In other embodiments, hypervisor 302 may provide each virtual machine 332 with a substantially similar virtual view of the physical hardware, memory, processor, and other system resources available to the virtual machines 332.

[0058] Each virtual machine 332 may include a virtual disk 326A-C (generally 326) and a virtual processor 328A-C (generally 328.) The virtual disk 326, in some embodiments, is a virtualized view of one or more physical disks 304 of the virtualization server 301, or a portion of one or more physical disks 304 of the virtualization server 301. The virtualized view of the physical disks 304 can be generated, provided, and managed by the hypervisor 302. In some embodiments, hypervisor 302 provides each virtual machine 332 with a unique view of the physical disks 304. Thus, in these embodiments, the particular virtual disk 326 included in each virtual machine 332 can be unique when compared with the other virtual disks 326.

[0059] A virtual processor 328 can be a virtualized view of one or more physical processors 308 of the virtualization server 301. In some embodiments, the virtualized view of the physical processors 308 can be generated, provided, and managed by hypervisor 302. In some embodiments, virtual processor 328 has substantially all of the same characteristics of at least one physical processor 308. In other embodiments, virtual processor 308 provides a modified view of physical processors 308 such that at least some of the characteristics of the virtual processor 328 are different than the characteristics of the corresponding physical processor 308.

[0060] With further reference to FIG. 4, some aspects described herein may be implemented in a cloud-based environment. FIG. 4 illustrates an example of a cloud computing environment (or cloud system) 400. As seen in FIG. 4, client computers 411-414 may communicate with a cloud management server 410 to access the computing resources (e.g., host servers 403a-403b (generally referred herein as "host servers 403"), storage resources 404a-404b (generally referred herein as "storage resources 404"), and network elements 405a-405b (generally referred herein as "network resources 405")) of the cloud system.

[0061] Management server 410 may be implemented on one or more physical servers. The management server 410 may run, for example, Citrix Cloud by Citrix Systems, Inc. of Ft.

Lauderdale, FL, or OPENSTACK, among others. Management server 410 may manage various computing resources, including cloud hardware and software resources, for example, host computers 403, data storage devices 404, and networking devices 405. The cloud hardware and software resources may include private and/or public components. For example, a cloud may be configured as a private cloud to be used by one or more particular customers or client computers 411-414 and/or over a private network. In other embodiments, public clouds or hybrid public-private clouds may be used by other customers over an open or hybrid networks.

[0062] Management server 410 may be configured to provide user interfaces through which cloud operators and cloud customers may interact with the cloud system 400. For example, the management server 410 may provide a set of application programming interfaces (APIs) and/or one or more cloud operator console applications (e.g., web-based or standalone applications) with user interfaces to allow cloud operators to manage the cloud resources, configure the virtualization layer, manage customer accounts, and perform other cloud administration tasks. The management server 410 also may include a set of APIs and/or one or more customer console applications with user interfaces configured to receive cloud computing requests from end users via client computers 411-414, for example, requests to create, modify, or destroy virtual machines within the cloud. Client computers 411-414 may connect to management server 410 via the Internet or some other communication network, and may request access to one or more of the computing resources managed by management server 410. In response to client requests, the management server 410 may include a resource manager configured to select and provision physical resources in the hardware layer of the cloud system based on the client requests. For example, the management server 410 and additional components of the cloud system may be configured to provision, create, and manage virtual machines and their operating environments (e.g., hypervisors, storage resources, services offered by the network elements, etc.) for customers at client computers 411-414, over a network (e.g., the Internet), providing customers with computational resources, data storage services, networking capabilities, and computer platform and application support. Cloud systems also may be configured to provide various specific services, including security systems, development environments, user interfaces, and the like.

[0063] Certain clients 411-414 may be related, for example, to different client computers creating virtual machines on behalf of the same end user, or different users affiliated with the same company or organization. In other examples, certain clients 411-414 may be unrelated,

such as users affiliated with different companies or organizations. For unrelated clients, information on the virtual machines or storage of any one user may be hidden from other users.

[0064] Referring now to the physical hardware layer of a cloud computing environment, availability zones 401-402 (or zones) may refer to a collocated set of physical computing resources. Zones may be geographically separated from other zones in the overall cloud of computing resources. For example, zone 401 may be a first cloud datacenter located in California, and zone 402 may be a second cloud datacenter located in Florida. Management server 410 may be located at one of the availability zones, or at a separate location. Each zone may include an internal network that interfaces with devices that are outside of the zone, such as the management server 410, through a gateway. End users of the cloud (e.g., clients 411-414) might or might not be aware of the distinctions between zones. For example, an end user may request the creation of a virtual machine having a specified amount of memory, processing power, and network capabilities. The management server 410 may respond to the user's request and may allocate the resources to create the virtual machine without the user knowing whether the virtual machine was created using resources from zone 401 or zone 402. In other examples, the cloud system may allow end users to request that virtual machines (or other cloud resources) are allocated in a specific zone or on specific resources 403-405 within a zone.

[0065] In this example, each zone 401-402 may include an arrangement of various physical hardware components (or computing resources) 403-405, for example, physical hosting resources (or processing resources), physical network resources, physical storage resources, switches, and additional hardware resources that may be used to provide cloud computing services to customers. The physical hosting resources in a cloud zone 401-402 may include one or more computer servers 403, such as the virtualization servers 301 described above, which may be configured to create and host virtual machine instances. The physical network resources in a cloud zone 401 or 402 may include one or more network elements 405 (e.g., network service providers) comprising hardware and/or software configured to provide a network service to cloud customers, such as firewalls, network address translators, load balancers, virtual private network (VPN) gateways, Dynamic Host Configuration Protocol (DHCP) routers, and the like. The storage resources in the cloud zone 401-402 may include storage disks (e.g., solid state drives (SSDs), magnetic hard disks, etc.) and other storage devices.

[0066] The example cloud computing environment shown in FIG. 4 also may include a virtualization layer (e.g., as shown in FIGS. 1-3) with additional hardware and/or software resources configured to create and manage virtual machines and provide other services to

customers using the physical resources in the cloud. The virtualization layer may include hypervisors, as described above in FIG. 3, along with other components to provide network virtualizations, storage virtualizations, etc. The virtualization layer may be as a separate layer from the physical resource layer, or may share some or all of the same hardware and/or software resources with the physical resource layer. For example, the virtualization layer may include a hypervisor installed in each of the virtualization servers 403 with the physical computing resources. Known cloud systems may alternatively be used, e.g., WINDOWS AZURE (Microsoft Corporation of Redmond Washington), AMAZON EC2 (Amazon.com Inc. of Seattle, Washington), IBM BLUE CLOUD (IBM Corporation of Armonk, New York), or others.

[0067] **ENTERPRISE MOBILITY MANAGEMENT ARCHITECTURE**

[0068] FIG. 5 represents an enterprise mobility technical architecture 500 for use in a "Bring Your Own Device" (BYOD) environment. The architecture enables a user of a mobile device 502 to both access enterprise or personal resources from a mobile device 502 and use the mobile device 502 for personal use. The user may access such enterprise resources 504 or enterprise services 508 using a mobile device 502 that is purchased by the user or a mobile device 502 that is provided by the enterprise to the user. The user may utilize the mobile device 502 for business use only or for business and personal use. The mobile device 502 may run an iOS operating system, an Android operating system, or the like. The enterprise may choose to implement policies to manage the mobile device 502. The policies may be implemented through a firewall or gateway in such a way that the mobile device 502 may be identified, secured or security verified, and provided selective or full access to the enterprise resources (e.g., 504 and 508.) The policies may be mobile device management policies, mobile application management policies, mobile data management policies, or some combination of mobile device, application, and data management policies. A mobile device 502 that is managed through the application of mobile device management policies may be referred to as an enrolled device.

[0069] In some embodiments, the operating system of the mobile device 502 may be separated into a managed partition 510 and an unmanaged partition 512. The managed partition 510 may have policies applied to it to secure the applications running on and data stored in the managed partition 510. The applications running on the managed partition 510 may be secure applications. In other embodiments, all applications may execute in accordance with a set of one or more policy files received separate from the application, and which define one or more

security parameters, features, resource restrictions, and/or other access controls that are enforced by the mobile device management system when that application is executing on the mobile device 502. By operating in accordance with their respective policy file(s), each application may be allowed or restricted from communications with one or more other applications and/or resources, thereby creating a virtual partition. Thus, as used herein, a partition may refer to a physically partitioned portion of memory (physical partition), a logically partitioned portion of memory (logical partition), and/or a virtual partition created as a result of enforcement of one or more policies and/or policy files across multiple applications as described herein (virtual partition). Stated differently, by enforcing policies on managed applications, those applications may be restricted to only be able to communicate with other managed applications and trusted enterprise resources, thereby creating a virtual partition that is not accessible by unmanaged applications and devices.

[0070] The secure applications may be email applications, web browsing applications, software-as-a-service (SaaS) access applications, Windows Application access applications, and the like. The secure applications may be secure native applications 514, secure remote applications 522 executed by a secure application launcher 518, virtualization applications 526 executed by a secure application launcher 518, and the like. The secure native applications 514 may be wrapped by a secure application wrapper 520. The secure application wrapper 520 may include integrated policies that are executed on the mobile device 502 when the secure native application 514 is executed on the mobile device 502. The secure application wrapper 520 may include meta-data that points the secure native application 514 running on the mobile device 502 to the resources hosted at the enterprise (e.g., 504 and 508) that the secure native application 514 may require to complete the task requested upon execution of the secure native application 514. The secure remote applications 522 executed by a secure application launcher 518 may be executed within the secure application launcher 518. The virtualization applications 526 executed by a secure application launcher 518 may utilize resources on the mobile device 502, at the enterprise resources 504, and the like. The resources used on the mobile device 502 by the virtualization applications 526 executed by a secure application launcher 518 may include user interaction resources, processing resources, and the like. The user interaction resources may be used to collect and transmit keyboard input, mouse input, camera input, tactile input, audio input, visual input, gesture input, and the like. The processing resources may be used to present a user interface, process data received from the enterprise resources 504, and the like. The resources used at the enterprise resources 504 by the

virtualization applications 526 executed by a secure application launcher 518 may include user interface generation resources, processing resources, and the like. The user interface generation resources may be used to assemble a user interface, modify a user interface, refresh a user interface, and the like. The processing resources may be used to create information, read information, update information, delete information, and the like. For example, the virtualization application 526 may record user interactions associated with a graphical user interface (GUI) and communicate them to a server application where the server application will use the user interaction data as an input to the application operating on the server. In such an arrangement, an enterprise may elect to maintain the application on the server side as well as data, files, etc. associated with the application. While an enterprise may elect to "mobilize" some applications in accordance with the principles herein by securing them for deployment on the mobile device 502, this arrangement may also be elected for certain applications. For example, while some applications may be secured for use on the mobile device 502, others might not be prepared or appropriate for deployment on the mobile device 502 so the enterprise may elect to provide the mobile user access to the unprepared applications through virtualization techniques. As another example, the enterprise may have large complex applications with large and complex data sets (e.g., material resource planning applications) where it would be very difficult, or otherwise undesirable, to customize the application for the mobile device 502 so the enterprise may elect to provide access to the application through virtualization techniques. As yet another example, the enterprise may have an application that maintains highly secured data (e.g., human resources data, customer data, engineering data) that may be deemed by the enterprise as too sensitive for even the secured mobile environment so the enterprise may elect to use virtualization techniques to permit mobile access to such applications and data. An enterprise may elect to provide both fully secured and fully functional applications on the mobile device 502 as well as a virtualization application 526 to allow access to applications that are deemed more properly operated on the server side. In an embodiment, the virtualization application 526 may store some data, files, etc. on the mobile device 502 in one of the secure storage locations. An enterprise, for example, may elect to allow certain information to be stored on the mobile device 502 while not permitting other information.

[0071] In connection with the virtualization application 526, as described herein, the mobile device 502 may have a virtualization application 526 that is designed to present GUIs and then record user interactions with the GUI. The virtualization application 526 may communicate the user interactions to the server side to be used by the server side application

as user interactions with the application. In response, the application on the server side may transmit back to the mobile device 502 a new GUI. For example, the new GUI may be a static page, a dynamic page, an animation, or the like, thereby providing access to remotely located resources.

[0072] The secure applications 514 may access data stored in a secure data container 528 in the managed partition 510 of the mobile device 502. The data secured in the secure data container may be accessed by the secure native applications 514, secure remote applications 522 executed by a secure application launcher 518, virtualization applications 526 executed by a secure application launcher 518, and the like. The data stored in the secure data container 528 may include files, databases, and the like. The data stored in the secure data container 528 may include data restricted to a specific secure application 530, shared among secure applications 532, and the like. Data restricted to a secure application may include secure general data 534 and highly secure data 538. Secure general data may use a strong form of encryption such as Advanced Encryption Standard (AES) 128-bit encryption or the like, while highly secure data 538 may use a very strong form of encryption such as AES 256-bit encryption. Data stored in the secure data container 528 may be deleted from the mobile device 502 upon receipt of a command from the device manager 524. The secure applications (e.g., 514, 522, and 526) may have a dual-mode option 540. The dual mode option 540 may present the user with an option to operate the secured application in an unsecured or unmanaged mode. In an unsecured or unmanaged mode, the secure applications may access data stored in an unsecured data container 542 on the unmanaged partition 512 of the mobile device 502. The data stored in an unsecured data container may be personal data 544. The data stored in an unsecured data container 542 may also be accessed by unsecured applications 546 that are running on the unmanaged partition 512 of the mobile device 502. The data stored in an unsecured data container 542 may remain on the mobile device 502 when the data stored in the secure data container 528 is deleted from the mobile device 502. An enterprise may want to delete from the mobile device 502 selected or all data, files, and/or applications owned, licensed or controlled by the enterprise (enterprise data) while leaving or otherwise preserving personal data, files, and/or applications owned, licensed or controlled by the user (personal data). This operation may be referred to as a selective wipe. With the enterprise and personal data arranged in accordance to the aspects described herein, an enterprise may perform a selective wipe.

[0073] The mobile device 502 may connect to enterprise resources 504 and enterprise services 508 at an enterprise, to the public Internet 548, and the like. The mobile device 502

may connect to enterprise resources 504 and enterprise services 508 through virtual private network connections. The virtual private network connections, also referred to as microVPN or application-specific VPN, may be specific to particular applications (as illustrated by microVPNs 550, particular devices, particular secured areas on the mobile device (as illustrated by O/S VPN 552), and the like. For example, each of the wrapped applications in the secured area of the mobile device 502 may access enterprise resources through an application specific VPN such that access to the VPN would be granted based on attributes associated with the application, possibly in conjunction with user or device attribute information. The virtual private network connections may carry Microsoft Exchange traffic, Microsoft Active Directory traffic, HyperText Transfer Protocol (HTTP) traffic, HyperText Transfer Protocol Secure (HTTPS) traffic, application management traffic, and the like. The virtual private network connections may support and enable single-sign-on authentication processes 554. The single-sign-on processes may allow a user to provide a single set of authentication credentials, which are then verified by an authentication service 558. The authentication service 558 may then grant to the user access to multiple enterprise resources 504, without requiring the user to provide authentication credentials to each individual enterprise resource 504.

[0074] The virtual private network connections may be established and managed by an access gateway 560. The access gateway 560 may include performance enhancement features that manage, accelerate, and improve the delivery of enterprise resources 504 to the mobile device 502. The access gateway 560 may also re-route traffic from the mobile device 502 to the public Internet 548, enabling the mobile device 502 to access publicly available and unsecured applications that run on the public Internet 548. The mobile device 502 may connect to the access gateway via a transport network 562. The transport network 562 may use one or more transport protocols and may be a wired network, wireless network, cloud network, local area network, metropolitan area network, wide area network, public network, private network, and the like.

[0075] The enterprise resources 504 may include email servers, file sharing servers, SaaS applications, Web application servers, Windows application servers, and the like. Email servers may include Exchange servers, Lotus Notes servers, and the like. File sharing servers may include ShareFile servers, and the like. SaaS applications may include Salesforce, and the like. Windows application servers may include any application server that is built to provide applications that are intended to run on a local Windows operating system, and the like. The enterprise resources 504 may be premise-based resources, cloud-based resources, and the like.

The enterprise resources 504 may be accessed by the mobile device 502 directly or through the access gateway 560. The enterprise resources 504 may be accessed by the mobile device 502 via the transport network 562.

[0076] The enterprise services 508 may include authentication services 558, threat detection services 564, device manager services 524, file sharing services 568, policy manager services 570, social integration services 572, application controller services 574, and the like. Authentication services 558 may include user authentication services, device authentication services, application authentication services, data authentication services, and the like. Authentication services 558 may use certificates. The certificates may be stored on the mobile device 502, by the enterprise resources 504, and the like. The certificates stored on the mobile device 502 may be stored in an encrypted location on the mobile device 502, the certificate may be temporarily stored on the mobile device 502 for use at the time of authentication, and the like. Threat detection services 564 may include intrusion detection services, unauthorized access attempt detection services, and the like. Unauthorized access attempt detection services may include unauthorized attempts to access devices, applications, data, and the like. Device management services 524 may include configuration, provisioning, security, support, monitoring, reporting, and decommissioning services. File sharing services 568 may include file management services, file storage services, file collaboration services, and the like. Policy manager services 570 may include device policy manager services, application policy manager services, data policy manager services, and the like. Social integration services 572 may include contact integration services, collaboration services, integration with social networks such as Facebook, Twitter, and LinkedIn, and the like. Application controller services 574 may include management services, provisioning services, deployment services, assignment services, revocation services, wrapping services, and the like.

[0077] The enterprise mobility technical architecture 500 may include an application store 578. The application store 578 may include unwrapped applications 580, pre-wrapped applications 582, and the like. Applications may be populated in the application store 578 from the application controller 574. The application store 578 may be accessed by the mobile device 502 through the access gateway 560, through the public Internet 548, or the like. The application store 578 may be provided with an intuitive and easy to use user interface.

[0078] A software development kit 584 may provide a user the capability to secure applications selected by the user by wrapping the application as described previously in this description. An application that has been wrapped using the software development kit 584 may

then be made available to the mobile device 502 by populating it in the application store 578 using the application controller 574.

[0079] The enterprise mobility technical architecture 500 may include a management and analytics capability 588. The management and analytics capability 588 may provide information related to how resources are used, how often resources are used, and the like. Resources may include devices, applications, data, and the like. How resources are used may include which devices download which applications, which applications access which data, and the like. How often resources are used may include how often an application has been downloaded, how many times a specific set of data has been accessed by an application, and the like.

[0080] FIG. 6 is another illustrative enterprise mobility management system 600. Some of the components of the mobility management system 500 described above with reference to FIG. 5 have been omitted for the sake of simplicity. The architecture of the system 600 depicted in FIG. 6 is similar in many respects to the architecture of the system 500 described above with reference to FIG. 5 and may include additional features not mentioned above.

[0081] In this case, the left hand side represents an enrolled mobile device 602 with a client agent 604, which interacts with gateway server 606 (which includes Access Gateway and application controller functionality) to access various enterprise resources 608 and services 609 such as Exchange, Sharepoint, public-key infrastructure (PKI) Resources, Kerberos Resources, Certificate Issuance service, as shown on the right hand side above. Although not specifically shown, the mobile device 602 may also interact with an enterprise application store (StoreFront) for the selection and downloading of applications.

[0082] The client agent 604 acts as the UI (user interface) intermediary for Windows apps/desktops hosted in an Enterprise data center, which are accessed using the High-Definition User Experience (HDX)/ICA display remoting protocol. The client agent 604 also supports the installation and management of native applications on the mobile device 602, such as native iOS or Android applications. For example, the managed applications 610 (mail, browser, wrapped application) shown in the figure above are all native applications that execute locally on the mobile device 602. Client agent 604 and application management framework of this architecture act to provide policy driven management capabilities and features such as connectivity and SSO (single sign on) to enterprise resources/services 608. The client agent 604 handles primary user authentication to the enterprise, normally to Access Gateway (AG)

606 with SSO to other gateway server components. The client agent 604 obtains policies from gateway server 606 to control the behavior of the managed applications 610 on the mobile device 602.

[0083] The Secure InterProcess Communication (IPC) links 612 between the native applications 610 and client agent 604 represent a management channel, which may allow a client agent to supply policies to be enforced by the application management framework 614 "wrapping" each application. The IPC channel 612 may also allow client agent 604 to supply credential and authentication information that enables connectivity and SSO to enterprise resources 608. Finally, the IPC channel 612 may allow the application management framework 614 to invoke user interface functions implemented by client agent 604, such as online and offline authentication.

[0084] Communications between the client agent 604 and gateway server 606 are essentially an extension of the management channel from the application management framework 614 wrapping each native managed application 610. The application management framework 614 may request policy information from client agent 604, which in turn may request it from gateway server 606. The application management framework 614 may request authentication, and client agent 604 may log into the gateway services part of gateway server 606 (for example, Citrix Gateway). Client agent 604 may also call supporting services on gateway server 606, which may produce input material to derive encryption keys for the local data vaults 616, or may provide client certificates which may enable direct authentication to PKI protected resources, as more fully explained below.

[0085] In more detail, the application management framework 614 "wraps" each managed application 610. This may be incorporated via an explicit build step, or via a post-build processing step. The application management framework 614 may "pair" with client agent 604 on first launch of an application 610 to initialize the Secure IPC channel 612 and obtain the policy for that application. The application management framework 614 may enforce relevant portions of the policy that apply locally, such as the client agent login dependencies and some of the containment policies that restrict how local OS services may be used, or how they may interact with the managed application 610.

[0086] The application management framework 614 may use services provided by client agent 604 over the Secure IPC channel 612 to facilitate authentication and internal network access. Key management for the private and shared data vaults 616 (containers) may be also

managed by appropriate interactions between the managed applications 610 and client agent 604. Vaults 616 may be available only after online authentication, or may be made available after offline authentication if allowed by policy. First use of vaults 616 may require online authentication, and offline access may be limited to at most the policy refresh period before online authentication is again required.

[0087] Network access to internal resources may occur directly from individual managed applications 610 through Access Gateway 606. The application management framework 614 may be responsible for orchestrating the network access on behalf of each managed application 610. Client agent 604 may facilitate these network connections by providing suitable time limited secondary credentials obtained following online authentication. Multiple modes of network connection may be used, such as reverse web proxy connections and end-to-end VPN-style tunnels 618.

[0088] The Mail and Browser managed applications 610 have special status and may make use of facilities that might not be generally available to arbitrary wrapped applications. For example, the Mail application 610 may use a special background network access mechanism that allows it to access an Exchange server 608 over an extended period of time without requiring a full AG logon. The Browser application 610 may use multiple private data vaults 616 to segregate different kinds of data.

[0089] This architecture may support the incorporation of various other security features. For example, gateway server 606 (including its gateway services) in some cases may not need to validate active directory 622 (AD) passwords. It can be left to the discretion of an enterprise whether an AD password may be used as an authentication factor for some users in some situations. Different authentication methods may be used if a user is online or offline (i.e., connected or not connected to a network). A group management service (GMS) 624 may be deployed along with AD 622 to manage group information associated with a plurality of managed applications 610. For example, GMS 624 may communicate with the plurality of managed applications 610 (e.g., Outlook, Slack, Teams, etc.) to synchronize the applications 610 with updated member information (e.g., removal of a membership, addition a membership, etc.) of the group.

[0090] Step up authentication is a feature wherein gateway server 606 may identify managed native applications 610 that are allowed to have access to highly classified data requiring strong authentication, and ensure that access to these applications is only permitted

after performing appropriate authentication, even if this means a re-authentication is required by the user after a prior weaker level of login.

[0091] Another security feature of this solution is the encryption of the data vaults 616 (containers) on the mobile device 602. The vaults 616 may be encrypted so that all on-device data including files, databases, and configurations are protected. For on-line vaults, the keys may be stored on the server (gateway server 606), and for off-line vaults, a local copy of the keys may be protected by a user password or biometric validation. If or when data is stored locally on the mobile device 602 in the secure container 616, it may be preferred that a minimum of AES 256 encryption algorithm be utilized.

[0092] Other secure container features may also be implemented. For example, a logging feature may be included, wherein security events happening inside a managed application 610 may be logged and reported to the backend. Data wiping may be supported, such as if or when the managed application 610 detects tampering, associated encryption keys may be written over with random data, leaving no hint on the file system that user data was destroyed. Screenshot protection may be another feature, where an application may prevent any data from being stored in screenshots. For example, the key window's hidden property may be set to YES. This may cause whatever content is currently displayed on the screen to be hidden, resulting in a blank screenshot where any content would normally reside.

[0093] Local data transfer may be prevented, such as by preventing any data from being locally transferred outside the application container, e.g., by copying it or sending it to an external application. A keyboard cache feature may operate to disable the autocorrect functionality for sensitive text fields. SSL certificate validation may be operable so the application specifically validates the server SSL certificate instead of it being stored in the keychain. An encryption key generation feature may be used such that the key used to encrypt data on the mobile device 602 is generated using a passphrase or biometric data supplied by the user (if offline access is required). It may be XORed with another key randomly generated and stored on the server side if offline access is not required. Key Derivation functions may operate such that keys generated from the user password use KDFs (key derivation functions, notably Password-Based Key Derivation Function 2 (PBKDF2)) rather than creating a cryptographic hash of it. The latter makes a key susceptible to brute force or dictionary attacks.

[0094] Further, one or more initialization vectors may be used in encryption methods. An initialization vector will cause multiple copies of the same encrypted data to yield different

cipher text output, preventing both replay and cryptanalytic attacks. This will also prevent an attacker from decrypting any data even with a stolen encryption key. Further, authentication then decryption may be used, wherein application data is decrypted only after the user has authenticated within the application. Another feature may relate to sensitive data in memory, which may be kept in memory (and not in disk) only when it's needed. For example, login credentials may be wiped from memory after login, and encryption keys and other data inside objective-C instance variables are not stored, as they may be easily referenced. Instead, memory may be manually allocated for these.

[0095] An inactivity timeout may be implemented, wherein after a policy-defined period of inactivity, a user session is terminated.

[0096] Data leakage from the application management framework 614 may be prevented in other ways. For example, if or when a managed application 610 is put in the background, the memory may be cleared after a predetermined (configurable) time period. When backgrounded, a snapshot may be taken of the last displayed screen of the application to fasten the foregrounding process. The screenshot may contain confidential data and hence should be cleared.

[0097] Another security feature may relate to the use of an OTP (one time password) 620 without the use of an AD (active directory) 622 password for access to one or more applications. In some cases, some users do not know (or are not permitted to know) their AD password, so these users may authenticate using an OTP 620 such as by using a hardware OTP system like SecurID (OTPs may be provided by different vendors also, such as Entrust or Gemalto). In some cases, after a user authenticates with a user ID, a text may be sent to the user with an OTP 620. In some cases, this may be implemented only for online use, with a prompt being a single field.

[0098] An offline password may be implemented for offline authentication for those managed applications 610 for which offline use is permitted via enterprise policy. For example, an enterprise may want StoreFront to be accessed in this manner. In this case, the client agent 604 may require the user to set a custom offline password and the AD password is not used. Gateway server 606 may provide policies to control and enforce password standards with respect to the minimum length, character class composition, and age of passwords, such as described by the standard Windows Server password complexity requirements, although these requirements may be modified.

[0099] Another feature may relate to the enablement of a client side certificate for certain applications 610 as secondary credentials (for the purpose of accessing PKI protected web resources via the application management framework micro VPN feature). For example, a managed application 610 may utilize such a certificate. In this case, certificate-based authentication using ActiveSync protocol may be supported, wherein a certificate from the client agent 604 may be retrieved by gateway server 606 and used in a keychain. Each managed application 610 may have one associated client certificate, identified by a label that is defined in gateway server 606.

[0100] Gateway server 606 may interact with an enterprise special purpose web service to support the issuance of client certificates to allow relevant managed applications to authenticate to internal PKI protected resources.

[0101] The client agent 604 and the application management framework 614 may be enhanced to support obtaining and using client certificates for authentication to internal PKI protected network resources. More than one certificate may be supported, such as to match various levels of security and/or separation requirements. The certificates may be used by the Mail and Browser managed applications 610, and ultimately by arbitrary wrapped applications 610 (provided those applications use web service style communication patterns where it is reasonable for the application management framework to mediate HTTPS requests).

[0102] Application management client certificate support on iOS may rely on importing a public-key cryptography standards (PKCS) 12 BLOB (Binary Large Object) into the iOS keychain in each managed application 610 for each period of use. Application management framework client certificate support may use a HTTPS implementation with private in-memory key storage. The client certificate may not be present in the iOS keychain and may not be persisted except potentially in "online-only" data value that is strongly protected.

[0103] Mutual SSL or TLS may also be implemented to provide additional security by requiring that a mobile device 602 is authenticated to the enterprise, and vice versa. Virtual smart cards for authentication to gateway server 606 may also be implemented.

[0104] Another feature may relate to application container locking and wiping, which may automatically occur upon jail-break or rooting detections, and occur as a pushed command from administration console, and may include a remote wipe functionality even when a managed application 610 is not running.

[0105] A multi-site architecture or configuration of enterprise application store and an application controller may be supported that allows users to be serviced from one of several different locations in case of failure.

[0106] In some cases, managed applications 610 may be allowed to access a certificate and private key via an API (for example, OpenSSL). Trusted managed applications 610 of an enterprise may be allowed to perform specific Public Key operations with an application's client certificate and private key. Various use cases may be identified and treated accordingly, such as if or when an application behaves like a browser and no certificate access is required, if or when an application reads a certificate for "who am I," if or when an application uses the certificate to build a secure session token, and if or when an application uses private keys for digital signing of important data (e.g. transaction log) or for temporary data encryption.

[0107] **GROUP MANAGEMENT SERVICE**

[0108] Figure 7 depicts an illustrative group management system 700. The group management system may introduce a secondary group management service layer (e.g., a group management service (GMS) 624). For example, GSM 624 may be implemented, by the cloud management server 410, as the secondary group management service layer interfacing with a primary group service layer (e.g., AD 622). The GMS 624 may be deployed along with AD 622 to provide flexible managements on group information associated with a plurality of managed applications 610 (e.g., applications 1-N). AD 622 may, based on a lightweight directory access protocol (LDAP) and via GMS 624, access the plurality of managed applications 610, wherein each of the managed applications 610 may include a distributed data storage that is blocked, by a boundary, from other distributed data storages of other applications of the plurality of managed applications 610. AD 622 may perform, based on user information and a single sign-on (SSO) based authentication process, authentication of a user device 705. GMS 624 may, based on successful completion of the SSO based authentication process, establish a graphical management service user interface (e.g., management board 800). GMS 624 may bypass AD 622 and access, via the management board 800, the distributed data storages without the boundaries for the flexible managements on the group information associated with the plurality of managed applications 610.

[0109] For example, GMS 624 may maintain and store the group information (e.g., cache/store new or modified member information of a group), provide the graphical management service user interface (e.g., management board 800) for performing

administration tasks to manage the member information, or communicate with the plurality of managed applications 610. GMS 624 may receive a user request (e.g., updating group membership information of the user) triggered from one of the managed applications 610 to notify group creator/owner, receive group owner/creator's command (e.g., creating new groups, updating contact group information of existing groups), or synchronize the plurality of managed applications 610 with updated group information. GMS 624 may provide one or more interfaces with the plurality of managed applications 610 to communicate with the plurality of managed applications 610. For example, GMS 624 may send, via the one or more interfaces, send synchronization requests and apply updated/new group information, reschedule any existing events involved with updated group information, notify members affected by the updated group information, or receive a member request to update group information (e.g., removal of a membership, addition a membership, creation of a new group, deletion of an existing group, etc.).

[0110] Figure 8 depicts an illustrative graphical user interface (GUI). The GUI may provide, via the group management service 624, a secure interface with managed applications 610 (e.g., applications 1-N). A user (e.g., system administrator) using the single GUI may centrally monitor or control the plurality of applications 1-N for managing group information associated with the applications 1-N (e.g., Outlook, Slack, Teams, JIRA, Wrike, ShareFile, any SaaS, etc.).

[0111] For example, the GUI may be a management board 800 for monitoring or performing various actions for managing a plurality of groups. The management board 800 may display group information including, for example, a group name (e.g., HDX-Scrum-Masters-NKG), total number of contacts/members of a group (e.g., 15), a group type (e.g., public or private), creator/group owner (e.g., User-A), member information (e.g., group member details such as member names, member addresses/email addresses, tel. numbers, classes, etc.), a creation time (time of a group creation), or a last update time, etc. The group type may be public (e.g., member names are viewable to non-members) or private (e.g., member names are only viewable to group members). The management board 800 may allow a user (e.g., administrator) to search a group by a group name (e.g., entering a group name via input 830), create a new group (e.g., selecting/pressing New Group 810), or modify an existing group (e.g., selecting/pressing Modify Group 820 for adding a new member, deleting an existing member, changing a member information, etc.).

[0112] Figure 9 depicts illustrative actions performed via application(s). The actions may include notifying group members of updated group information via one or more applications used by the group members. The notification may allow group members to become aware of any changes with the group and plan accordingly (e.g., change their meeting schedules). One or more group members may be notified of the updated group information, for example, if the group is one of groups (e.g., marked with CGMS icon 910) controlled by the group management service 624. For example, user A may receive, via Application 1 (e.g., Outlook) a message 920 (e.g., email) indicating that user A is no longer a member of HDX-Scrum-Masters-NKG group and cancelation of a previously scheduled meeting event that required members of HDX-Scrum-Masters-NKG group. For example, user C may see, via Application 2 (e.g., a graphical application user interface 900 of Slack), a notice (e.g., alarm icon 930) adjacent to a group to indicate that there is updated group information on the group. User C may hover mouse over the alarm icon 930 to see details on the updated group information. For example, a phrase 940 "User A is not a member of HDX-Scrum-Masters-NKG anymore" may pop-up as user C hovers mouse over the alarm icon 930. The notice may disappear next time user C using Application 2. For example, the alarm icon 930 may show only once after the updated group information.

[0113] The group information may be updated by an administrator (e.g., a system administrator, a group creator/owner, etc.) or a user (e.g., an existing group member, a former group member, a new/future group member, etc.). For example, updating group information may involve deleting an existing member (e.g., user A) from a group, adding a new member (e.g., user B) to a group, reinstating a former member of a group, changing a group member information (e.g., members' mailing addresses, phone numbers, memberships, locations, etc.), or changing group membership status (e.g., regular class member or gold class member having more privileges than the regular class member , etc.). Accordingly, different information, based on the updated group information, may be notified via one or more applications to an administrator, an existing group member, a former group member, or a new group member.

[0114] Figure 10 depicts illustrative interactions between application(s) and the GUI. The GUI (e.g., management board 800) may centrally monitor/receive messages coming from different applications or send messages to the different applications. The GUI may enable an administrator to use the group management service 624 for automatically synchronizing the group information across the different applications or keep users informed of updated group information. The user may be a group member. For example, different actions requested by a

group member may include updating group member information (e.g., changing contact address, tel. number, etc.) or canceling membership. The actions may be requested by a group member using different applications (e.g., Outlook, Slack, Teams, JIRA, Wrike, ShareFile, any SaaS, etc.). The GUI may centrally indicate, via the group management service 624, information about the actions requested from the different applications.

[0115] For example, a group member (e.g., user A) may send, from Application 2 (e.g., Slack) and via the group management service 624, a message 1010 indicating an action to be performed on a group 1015 (e.g., quit @ HDX-Scrum-Masters-NKG, update user A profile information, etc.). The group management service 624, based on the message 1010, may indicate, via the management board 800, an action requested by the group member. For example, a text message 1020, "user A is asking to quit this group" may appear next to the group 1015 on the management board 800. The group member information may be modified, via the management board 800, according to the requested action. The group management service 624, based on an input received (e.g., clicking on "Modify Group" 820 button) from the management board 800, may automatically identify one or more applications associated with the group 1015 and synchronize the one or more applications with the updated group member information so that the one or more applications may have the same group member information for the group 1015.

[0116] Figure 11 depicts illustrative message sequences for synchronizing applications 610 initiated by an administrator. The management board 800 may receive a message 1110 indicating action requested (e.g., create a new group, update/modify group information of an existing group, etc.) from a computing device (e.g., device A 410 used by an administrator). The management board 800, based on the message 1110, may send a request 1112 (e.g., request to update group information) to the group management service 624. The group management service 624 may update group information and store the updated group information. The group management may search applications 610 impacted by the updated group information. For example, the group management service 624 may search and identify those applications 610 associated with a group from which a group member is canceled according to the updated group information. The group management service 624, based on the search, may send a message 1114 to those applications 610 (e.g., Applications 1-N) for synchronizing group information associated with those applications 610. For example, those applications 610 may update, based on the message 1114, the group information so that the applications 610 have the same updated group information. The applications 610 may send, to the group management service 624, a

response message 1116 (e.g., sync completed) indicating that the group information is updated. The group management service 624 may send a confirmation 1118 to the management board 800. The management board 800 may send, based on the confirmation 1118, a response message 1120 indicating that the requested action is completed to the device A 410 in response to the message 1110.

[0117] The applications 610 may update information associated with any existing event for a group affected by the updated group information (e.g., inviting additional members to the existing event, deleting members previously invited to the existing event, etc.). The applications 610 may optionally perform task 1130 notifying members of the group the updated information. For example, application 1 (e.g., Outlook) may send, to a user, a delete notice of a previously scheduled event indicating that the user is no longer a member of a group invited by the previously scheduled event. For example, application 1 may send, to another user, an invitation to the previously scheduled event. The applications 610 may perform task 1140 of adding the alarm icon 930 and/or updating members of a group affected by the updated group information. For example, application 2 (e.g., Slack) may add the alarm icon 930 adjacent to a group to indicate that the group information has been updated. A user may interact with the alarm icon 930 to obtain detail information on the updated group information, as described in Figure 9.

[0118] Figure 12 depicts illustrative message sequences for synchronizing applications 610 initiated by a group member. For example, the group member may send, from a device B 705 and to application 2 (e.g., Slack), a message 1210 (e.g., ask for removal from a group). Application 2 may send, based on the message 1210, a request 1212 (e.g., trigger removal request) to the group management service 624. The group management service 624 may send an instruction 1214 (e.g., add update icon) to the management board 800. The management board 800 may add update icon to indicate the removal request 1216 to the administrator. Device A 410 (e.g., administrator device) may send a response 1218 (e.g., update group information) to the management board 800. The management board 800 may send, based on the response 1218, an instruction 1220 to the group management service 624 to store new group information (e.g., membership of the group member being canceled). The group management service 624 may search those applications 610 associated with the updated group information and send a message 1222 (e.g., synchronize updated group information) to those applications 610 for synchronizing group information associated with those applications 610. The applications 610 may update information associated with any existing event for a group

affected by the updated group information (e.g., additional members are to be invited to the existing event, deleting members previously invited to the existing event, etc.). The applications 610 may notify members of the group the updated information by performing the tasks 1130 and 1140, as described with Figure 11.

[0119] Figure 13 depicts illustrative alternative message sequences for synchronizing applications 610 initiated by a group member. The alternative message sequences may include extra messages to inform the group member of impacts/consequences of a requested action before committing to the requested action. The group member may be able to make, based on the information, informed decision either to confirm or cancel initial request. The group member may reconsider to withdraw the requested action after being informed. Specifically, the group management service 624, after receiving the trigger removal request 1212 from application 2 (e.g., Slack), may search other applications 610 associated with the removal request. The group management service 624 may send, to device B 705 (e.g., the group member's device) and via application 2, a list 1310 of the other applications 610 (e.g., Outlook, Teams, JIRA, Wrike, ShareFile, etc.) associated with the removal request. The group member may become aware that the removal request would apply not only to application 2, but also the other applications 610 that may give reasons for the group member to keep the membership. The group member using device B 705 may send, to the group management service 624 and via application 2, a response 1320 (e.g., confirm removal request or cancel removal request). For example, if the response 1320 is confirmation of the removal request, application 2 may send confirmation message 1330 to the group management service 624, and the rest of the alternative message sequences may proceed in a similar manner as shown in Figure. 12. If the response 1320 is the cancelation of the removal request, the group management service 624 may not take further action regarding the removal request 1212.

[0120] Figure 14 depicts an illustrative flowchart showing steps of an example method associated a group management service (e.g., GMS 624). The flowchart may be implemented as an algorithm of GMS 624 to be performed by a processor. For convenience, Figure 14 is described by way of an example in which the steps are performed by GMS 624 of the cloud management server 410. One, some, or all steps of the example method of Figure 14, or portions thereof, may be performed by one or more other computing devices. One, some, or all steps of the example method of Figure 14 may be omitted, performed in other orders, and/or otherwise modified, and/or one or more additional steps may be added.

[0121] At step 1410, GMS 624 may receive an input (e.g., indicating an action requested from an administrator or a group member). At step 1420, GMS 624 may determine that the input is from the management board 800 (e.g., administrator) and proceed to step 1430. At step 1430, GMS 624 may, based on the received input, update group information, and store the updated group information. At step 1432, GMS 624 may search applications 610 that are associated with the updated group information. At step 1434, GMS 624 may synchronize the applications 610 with the updated group information. At step 1436, GMS 624 may send confirmation of completed action.

[0122] At step 1420, GMS 624 may determine that the input is not from the management board 800 (e.g., the input is from a group member via an application) and proceed to step 1422. At step 1422, GMS 624 may send, based on the input and to the management board 800, a requested action from a group member. At step 1424, GMS 624 may determine whether the requested action is authorized/approved by the administrator via the management board 800. If not authorized, at step 1426, GMS 624 may send, to the group member and via the application, a rejection response. If authorized, GMS 624 may perform the steps 1432, 1434, and 1436, as previously described.

[0123] The following paragraphs (M1) through (M9) describe examples of methods that may be implemented in accordance with the present disclosure.

[0124] (M1). A method comprising: authenticating, by a primary group management service layer associated with a server and based on user information, a computing device; granting the computing device, based on the authenticating, access to a plurality of applications managed by a secondary group management service layer associated with the server; receiving, from the computing device via a graphical management service user interface associated with the secondary group management service layer, a message indicating to update data associated with a user group; updating, based on the message, data stored in a centralized data storage managed by the secondary group management service layer; identifying, based on the data stored in the centralized data storage, one or more applications, of the plurality of applications, associated with the user group; synchronizing, by the secondary group management service layer associated with the server, data stored in one or more distributed data storages with the data stored in the centralized data storage, each distributed data storage associated with respective one of the identified one or more applications of the plurality of applications; and based on the synchronized data and via one or more graphical application user interfaces, each

associated with respective one of the identified one or more applications, sending, to one or more members of the user group, a notification of updated data associated with the user group.

[0125] (M2). The method of paragraph (M1), further comprising, based on a lightweight directory access protocol (LDAP) and via the secondary group management service layer, accessing, by the primary group management service layer, the plurality of applications, wherein each of the plurality of applications comprises a distributed data storage that is blocked, by a boundary, from other distributed data storages of other applications of the plurality of applications.

[0126] (M3). The method of paragraph (M2), further comprising: performing, by the primary group management service layer, a single sign-on (SSO) based authentication process with the computing device; based on successful completion of the SSO based authentication process, bypassing boundaries, set by the primary group management service layer, between the distributed data storages associated with the plurality of applications; establishing, by the secondary group management service layer, the graphical management service user interface between the computing device and the plurality of applications; and bypassing the primary group management service layer and managing, via the graphical management service user interface, the distributed data storages without the boundaries.

[0127] (M4). The method of paragraphs (M1) through (M3), wherein the updating data comprises updating member information of the user group stored in the centralized data storage by: adding one or more new members to the user group, deleting one or more members of the user group, changing member contact information of the user group, or changing a class of one or more members of the user group.

[0128] (M5). The method of paragraphs (M1) through (M4), further comprising: determining, based on the synchronized data, that one or more members of the user group are changed; and resending, based on the determining, a notification of a previously scheduled event to the one or more members of the user group.

[0129] (M6). The method of paragraphs (M1) through (M5), further comprising: detecting a member of the user group searching, via one or more of the identified one or more applications, for the user group; and based on the detecting, indicating, to the member, a changed status of one or more members of the user group.

[0130] (M7). The method of paragraphs (M1) through (M6), further comprising: prompting, based on the synchronized data and via one or more of the identified one or more

applications, a text message indicating one or more changed statuses of one or more members of the user group and one or more names of the user group.

[0131] (M8). The method of paragraphs (M1) through (M7), wherein the sending the notification of updated data comprises a message indicating a name of the user group and that the one or more members no longer belongs to the user group.

[0132] (M9). The method of paragraphs (M1) through (M7), wherein the centralized data storage, managed by the secondary group management service layer, comprises a data structure storing a name of the user group, a creator of the user group, a type of the user group, member information of the user group, a creation time of the user group, and a last update time of the user group.

[0133] The following paragraphs (A1) through (A6) describe examples of apparatuses that may be implemented in accordance with the present disclosure.

[0134] (A1). An apparatus comprising: a processor; and a memory storing computer readable instructions that, when executed by the processor, cause the apparatus to: authenticate, by a primary group management service layer associated with the apparatus and based on user information, a computing device; grant the computing device, based on the authentication, access to a plurality of applications managed by a secondary group management service layer associated with the apparatus; receive, from the computing device via a graphical management service user interface associated with the secondary group management service layer, a message indicating to update data associated with a user group; update, based on the message, data stored in a centralized data storage managed by the secondary group management service layer; identify, based on the data stored in the centralized data storage, one or more applications, of the plurality of applications, associated with the user group; synchronize, by the secondary group management service layer associated with the apparatus, data stored in one or more distributed data storages with the data stored in the centralized data storage, each distributed data storage associated with respective one of the identified one or more applications of the plurality of applications; and based on the synchronized data and via one or more graphical application user interfaces, each associated with respective one of the identified one or more applications, send, to one or more members of the user group, a notification of updated data associated with the user group.

[0135] (A2). The apparatus of paragraph (A1), wherein the computer readable instructions, when executed by the processor, cause the apparatus to, based on a lightweight directory access

protocol (LDAP) and via the secondary group management service layer, access, by the primary group management service layer, the plurality of applications, wherein each of the plurality of applications comprises a distributed data storage that is blocked, by a boundary, from other distributed data storages of other applications of the plurality of applications.

[0136] (A3). The apparatus of paragraph (A1), wherein the computer readable instructions, when executed by the processor, cause the apparatus to: perform, by the primary group management service layer, a single sign-on (SSO) based authentication process with the computing device; based on successful completion of the SSO based authentication process, bypass boundaries, set by the primary group management service layer, between the distributed data storages associated with the plurality of applications; establish, by the secondary group management service layer, the graphical management service user interface between the computing device and the plurality of applications; and bypass the primary group management service layer and manage, via the graphical management service user interface, the distributed data storages without the boundaries.

[0137] (A4). The apparatus of paragraphs (A1) through (A3), wherein the computer readable instructions, when executed by the processor, cause the apparatus to update member information of the user group stored in the centralized data storage by: adding one or more new members to the user group, deleting one or more members of the user group, changing member contact information of the user group, or changing a class of one or more members of the user group.

[0138] (A5). The apparatus of paragraphs (A1) through (A4), wherein the computer readable instructions, when executed by the processor, cause the apparatus to: determine, based on the synchronized data, that one or more members of the user group are changed; and resend, based on the determination, a notification of a previously scheduled event to the one or more members of the user group.

[0139] (A6). The apparatus of paragraphs (A1) through (A5), wherein the computer readable instructions, when executed by the processor, cause the apparatus to: detect a member of the user group searching, via one or more of the identified one or more applications, for the user group; and based on the detection, indicate, to the member, a changed status of one or more members of the user group.

[0140] The following paragraphs (CRM1) through (CRM5) describe examples of computer-readable media that may be implemented in accordance with the present disclosure.

[0141] (CRM1). A non-transitory computer readable medium storing computer readable instructions thereon that, when executed by a computer, causes the computer to perform a method comprising: authenticating, by a primary group management service layer associated with the computer and based on user information, a user device; granting the user device, based on the authenticating, access to a plurality of applications managed by a secondary group management service layer associated with the computer; receiving, from the user device via a graphical management service user interface associated with the secondary group management service layer, a message indicating to update data associated with a user group; updating, based on the message, data stored in a centralized data storage managed by the secondary group management service layer; identifying, based on the data stored in the centralized data storage, one or more applications, of the plurality of applications, associated with the user group; synchronizing, by the secondary group management service layer associated with the computer, data stored in one or more distributed data storages with the data stored in the centralized data storage, each distributed data storage associated with respective one of the identified one or more applications of the plurality of applications; and based on the synchronized data and via one or more graphical application user interfaces, each associated with respective one of the identified one or more applications, sending, to one or more members of the user group, a notification of updated data associated with the user group.

[0142] (CRM2). The non-transitory computer readable medium of paragraph (CRM1), when executed by the computer, causes the computer to perform the method further comprising, based on a lightweight directory access protocol (LDAP) and via the secondary group management service layer, accessing, by the primary group management service layer, the plurality of applications, wherein each of the plurality of applications comprises a distributed data storage that is blocked, by a boundary, from other distributed data storages of other applications of the plurality of applications.

[0143] (CRM3). The non-transitory computer readable medium of paragraph (CRM1), when executed by the computer, causes the computer to perform the method further comprising: performing, by the primary group management service layer, a single sign-on (SSO) based authentication process with the user device; based on successful completion of the SSO based authentication process, bypassing boundaries, set by the primary group management service layer, between the distributed data storages associated with the plurality of applications; establishing, by the secondary group management service layer, the graphical management service user interface between the user device and the plurality of applications;

and bypassing the primary group management service layer and managing, via the graphical management service user interface, the distributed data storages without the boundaries.

[0144] (CRM4). The non-transitory computer readable medium of paragraphs (CRM1) through (CRM3), wherein the updating data comprises updating member information of the user group stored in the centralized data storage by: adding one or more new members to the user group, deleting one or more members of the user group, changing member contact information of the user group, or changing a class of one or more members of the user group.

[0145] (CRM5). The non-transitory computer readable medium of paragraphs (CRM1) through (CRM4), when executed by the computer, causes the computer to perform the method further comprising: determining, based on the synchronized data, that one or more members of the user group are changed; and resending, based on the determining, a notification of a previously scheduled event to the one or more members of the user group.

[0146] Although the subject matter has been described in language specific to structural features and/or methodological acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the specific features or acts described above. Rather, the specific features and acts described above are described as example implementations of the following claims.

CLAIMS

What is claimed is:

1. A method comprising:

authenticating, by a primary group management service layer associated with a server and based on user information, a computing device;

granting the computing device, based on the authenticating, access to a plurality of applications managed by a secondary group management service layer associated with the server;

receiving, from the computing device via a graphical management service user interface associated with the secondary group management service layer, a message indicating to update data associated with a user group;

updating, based on the message, data stored in a centralized data storage managed by the secondary group management service layer;

identifying, based on the data stored in the centralized data storage, one or more applications, of the plurality of applications, associated with the user group;

synchronizing, by the secondary group management service layer associated with the server, data stored in one or more distributed data storages with the data stored in the centralized data storage, each distributed data storage associated with respective one of the identified one or more applications of the plurality of applications; and

based on the synchronized data and via one or more graphical application user interfaces, each associated with respective one of the identified one or more applications, sending, to one or more members of the user group, a notification of updated data associated with the user group.

2. The method of claim 1, further comprising, based on a lightweight directory access protocol (LDAP) and via the secondary group management service layer, accessing, by the primary group management service layer, the plurality of applications, wherein each of the plurality of applications comprises a distributed data storage that is blocked, by a boundary, from other distributed data storages of other applications of the plurality of applications.

3. The method of claim 1, further comprising:

performing, by the primary group management service layer, a single sign-on (SSO) based authentication process with the computing device;

based on successful completion of the SSO based authentication process, bypassing boundaries, set by the primary group management service layer, between the distributed data storages associated with the plurality of applications;

establishing, by the secondary group management service layer, the graphical management service user interface between the computing device and the plurality of applications; and

bypassing the primary group management service layer and managing, via the graphical management service user interface, the distributed data storages without the boundaries.

4. The method of claim 1, wherein the updating data comprises updating member information of the user group stored in the centralized data storage by:

adding one or more new members to the user group,

deleting one or more members of the user group,

changing member contact information of the user group, or

changing a class of one or more members of the user group.

5. The method of claim 1, further comprising:

determining, based on the synchronized data, that one or more members of the user group are changed; and

resending, based on the determining, a notification of a previously scheduled event to the one or more members of the user group.

6. The method of claim 1, further comprising:

detecting a member of the user group searching, via one or more of the identified one or more applications, for the user group; and

based on the detecting, indicating, to the member, a changed status of one or more members of the user group.

7. The method of claim 1, further comprising:

prompting, based on the synchronized data and via one or more of the identified one or more applications, a text message indicating one or more changed statuses of one or more members of the user group and one or more names of the user group.

8. The method of claim 1, wherein the sending the notification of updated data comprises a message indicating a name of the user group and that the one or more members no longer belongs to the user group.

9. The method of claim 1, wherein the centralized data storage, managed by the secondary group management service layer, comprises a data structure storing a name of the user group, a creator of the user group, a type of the user group, member information of the user group, a creation time of the user group, and a last update time of the user group.

10. An apparatus comprising:

a processor; and

a memory storing computer readable instructions that, when executed by the processor, cause the apparatus to:

authenticate, by a primary group management service layer associated with the apparatus and based on user information, a computing device;

grant the computing device, based on the authentication, access to a plurality of applications managed by a secondary group management service layer associated with the apparatus;

receive, from the computing device via a graphical management service user interface associated with the secondary group management service layer, a message indicating to update data associated with a user group;

update, based on the message, data stored in a centralized data storage managed by the secondary group management service layer;

identify, based on the data stored in the centralized data storage, one or more applications, of the plurality of applications, associated with the user group;

synchronize, by the secondary group management service layer associated with the apparatus, data stored in one or more distributed data storages with the data stored in the centralized data storage, each distributed data storage associated with respective one of the identified one or more applications of the plurality of applications; and

based on the synchronized data and via one or more graphical application user interfaces, each associated with respective one of the identified

one or more applications, send, to one or more members of the user group, a notification of updated data associated with the user group.

11. The apparatus of claim 10, wherein the computer readable instructions, when executed by the processor, cause the apparatus to, based on a lightweight directory access protocol (LDAP) and via the secondary group management service layer, access, by the primary group management service layer, the plurality of applications, wherein each of the plurality of applications comprises a distributed data storage that is blocked, by a boundary, from other distributed data storages of other applications of the plurality of applications.

12. The apparatus of claim 10, wherein the computer readable instructions, when executed by the processor, cause the apparatus to:

perform, by the primary group management service layer, a single sign-on (SSO) based authentication process with the computing device;

based on successful completion of the SSO based authentication process, bypass boundaries, set by the primary group management service layer, between the distributed data storages associated with the plurality of applications;

establish, by the secondary group management service layer, the graphical management service user interface between the computing device and the plurality of applications; and

bypass the primary group management service layer and manage, via the graphical management service user interface, the distributed data storages without the boundaries.

13. The apparatus of claim 10, wherein the computer readable instructions, when executed by the processor, cause the apparatus to update member information of the user group stored in the centralized data storage by:

adding one or more new members to the user group,

deleting one or more members of the user group,

changing member contact information of the user group, or

changing a class of one or more members of the user group.

14. The apparatus of claim 10, wherein the computer readable instructions, when executed by the processor, cause the apparatus to:

determine, based on the synchronized data, that one or more members of the user group are changed; and

resend, based on the determination, a notification of a previously scheduled event to the one or more members of the user group.

15. The apparatus of claim 10, wherein the computer readable instructions, when executed by the processor, cause the apparatus to:

detect a member of the user group searching, via one or more of the identified one or more applications, for the user group; and

based on the detection, indicate, to the member, a changed status of one or more members of the user group.

16. A non-transitory computer readable medium storing computer readable instructions thereon that, when executed by a computer, causes the computer to perform a method comprising:

authenticating, by a primary group management service layer associated with the computer and based on user information, a user device;

granting the user device, based on the authenticating, access to a plurality of applications managed by a secondary group management service layer associated with the computer;

receiving, from the user device via a graphical management service user interface associated with the secondary group management service layer, a message indicating to update data associated with a user group;

updating, based on the message, data stored in a centralized data storage managed by the secondary group management service layer;

identifying, based on the data stored in the centralized data storage, one or more applications, of the plurality of applications, associated with the user group;

synchronizing, by the secondary group management service layer associated with the computer, data stored in one or more distributed data storages with the data stored in the centralized data storage, each distributed data storage associated with respective one of the identified one or more applications of the plurality of applications; and

based on the synchronized data and via one or more graphical application user interfaces, each associated with respective one of the identified one or more applications, sending, to one or more members of the user group, a notification of updated data associated with the user group.

17. The non-transitory computer readable medium of claim 16, when executed by the computer, causes the computer to perform the method further comprising, based on a lightweight directory access protocol (LDAP) and via the secondary group management service layer, accessing, by the primary group management service layer, the plurality of applications, wherein each of the plurality of applications comprises a distributed data storage that is blocked, by a boundary, from other distributed data storages of other applications of the plurality of applications.

18. The non-transitory computer readable medium of claim 16, when executed by the computer, causes the computer to perform the method further comprising:

performing, by the primary group management service layer, a single sign-on (SSO) based authentication process with the user device;

based on successful completion of the SSO based authentication process, bypassing boundaries, set by the primary group management service layer, between the distributed data storages associated with the plurality of applications;

establishing, by the secondary group management service layer, the graphical management service user interface between the user device and the plurality of applications; and

bypassing the primary group management service layer and managing, via the graphical management service user interface, the distributed data storages without the boundaries.

19. The non-transitory computer readable medium of claim 16, wherein the updating data comprises updating member information of the user group stored in the centralized data storage by:

adding one or more new members to the user group,

deleting one or more members of the user group,

changing member contact information of the user group, or

changing a class of one or more members of the user group.

20. The non-transitory computer readable medium of claim 16, when executed by the computer, causes the computer to perform the method further comprising:

determining, based on the synchronized data, that one or more members of the user group are changed; and

resending, based on the determining, a notification of a previously scheduled event to the one or more members of the user group.

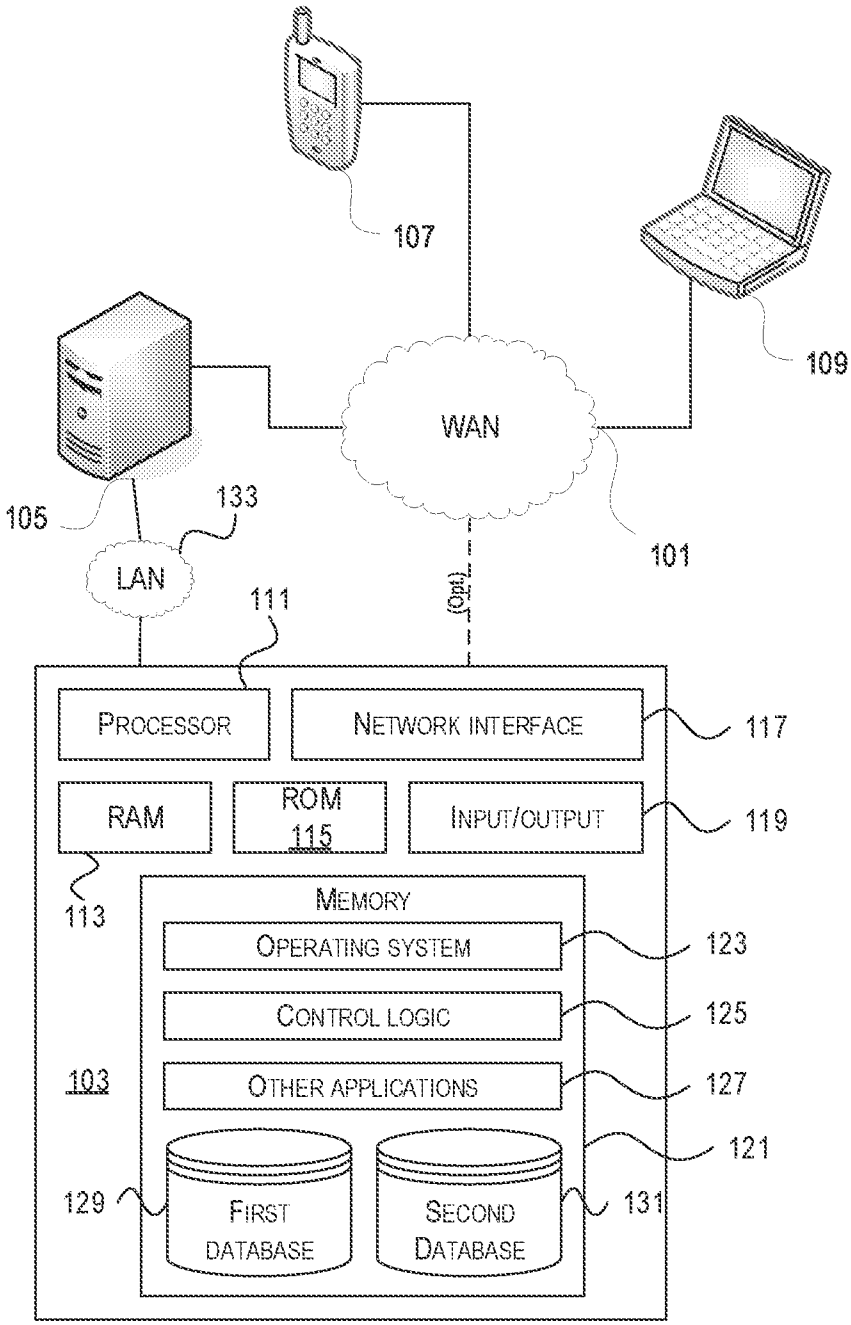

107
109
WAN
105
133
LAN
101
(Opt.)
111
PROCESSOR
NETWORK INTERFACE
117
RAM
ROM
115
INPUT/OUTPUT
119
113
MEMORY
OPERATING SYSTEM
123
CONTROL LOGIC
125
103
OTHER APPLICATIONS
127
121
129
FIRST
DATABASE
SECOND
DATABASE
131

FIG. 1

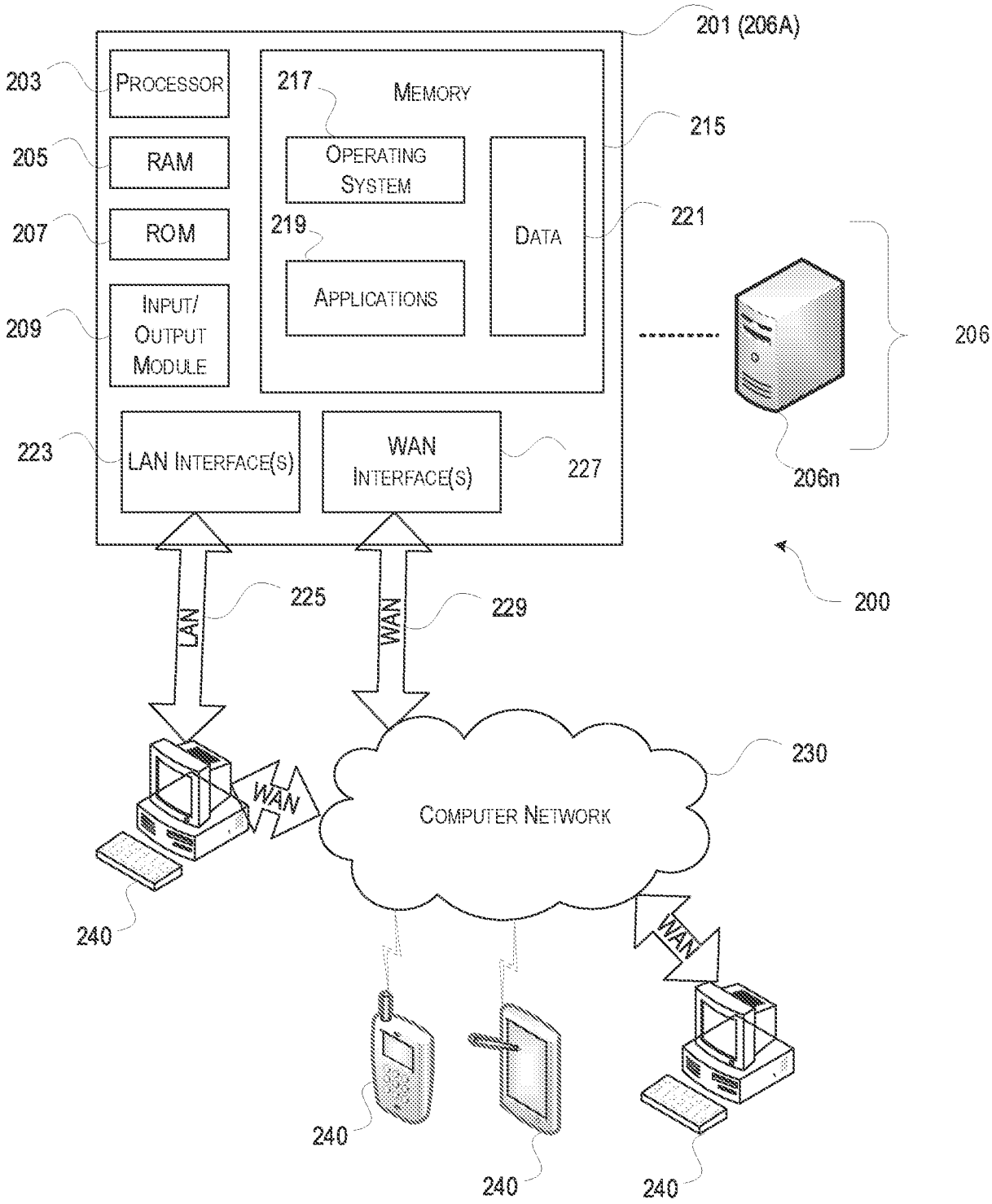

201 (206A)
203
PROCESSOR
217
MEMORY
215
205
RAM
OPERATING SYSTEM
207
ROM
219
221
DATA
APPLICATIONS
209
INPUT/ OUTPUT MODULE
206
206n
223
LAN INTERFACE(S)
WAN INTERFACE(S)
227
200
225
LAN
229
WAN
230
COMPUTER NETWORK
WAN
WAN
240
240
240
240

FIG. 2

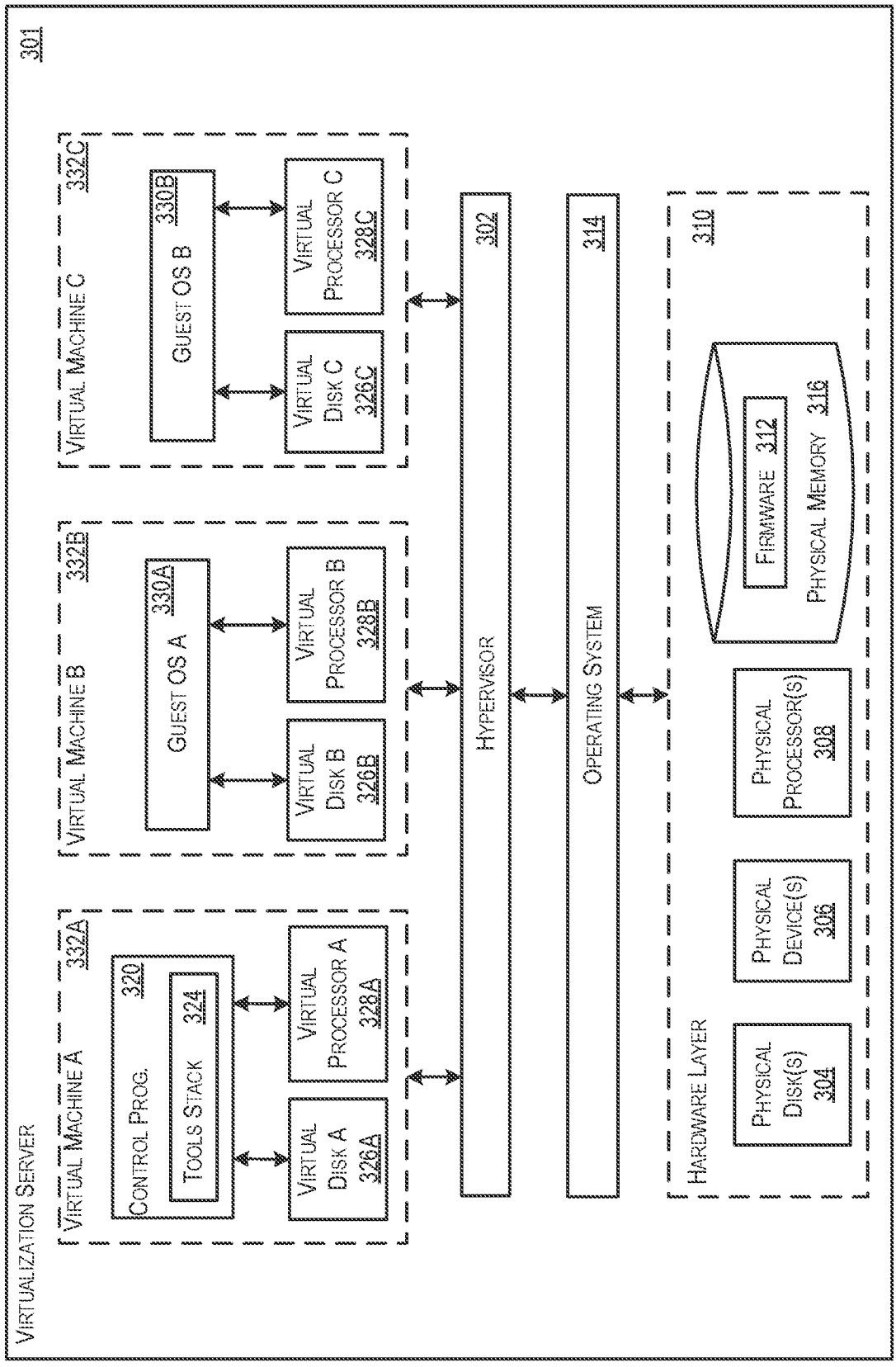

VIRTUALIZATION SERVER
301
VIRTUAL MACHINE A
332A
CONTROL PROG.
320
TOOLS STACK
324
VIRTUAL DISK A
326A
VIRTUAL PROCESSOR A
328A
VIRTUAL MACHINE B
332B
GUEST OS A
330A
VIRTUAL DISK B
326B
VIRTUAL PROCESSOR B
328B
VIRTUAL MACHINE C
332C
GUEST OS B
330B
VIRTUAL DISK C
326C
VIRTUAL PROCESSOR C
328C
HYPERVISOR
302
OPERATING SYSTEM
314
HARDWARE LAYER
310
PHYSICAL DISK(S)
304
PHYSICAL DEVICE(S)
306
PHYSICAL PROCESSOR(S)
308
FIRMWARE
312
PHYSICAL MEMORY
316

FIG. 3

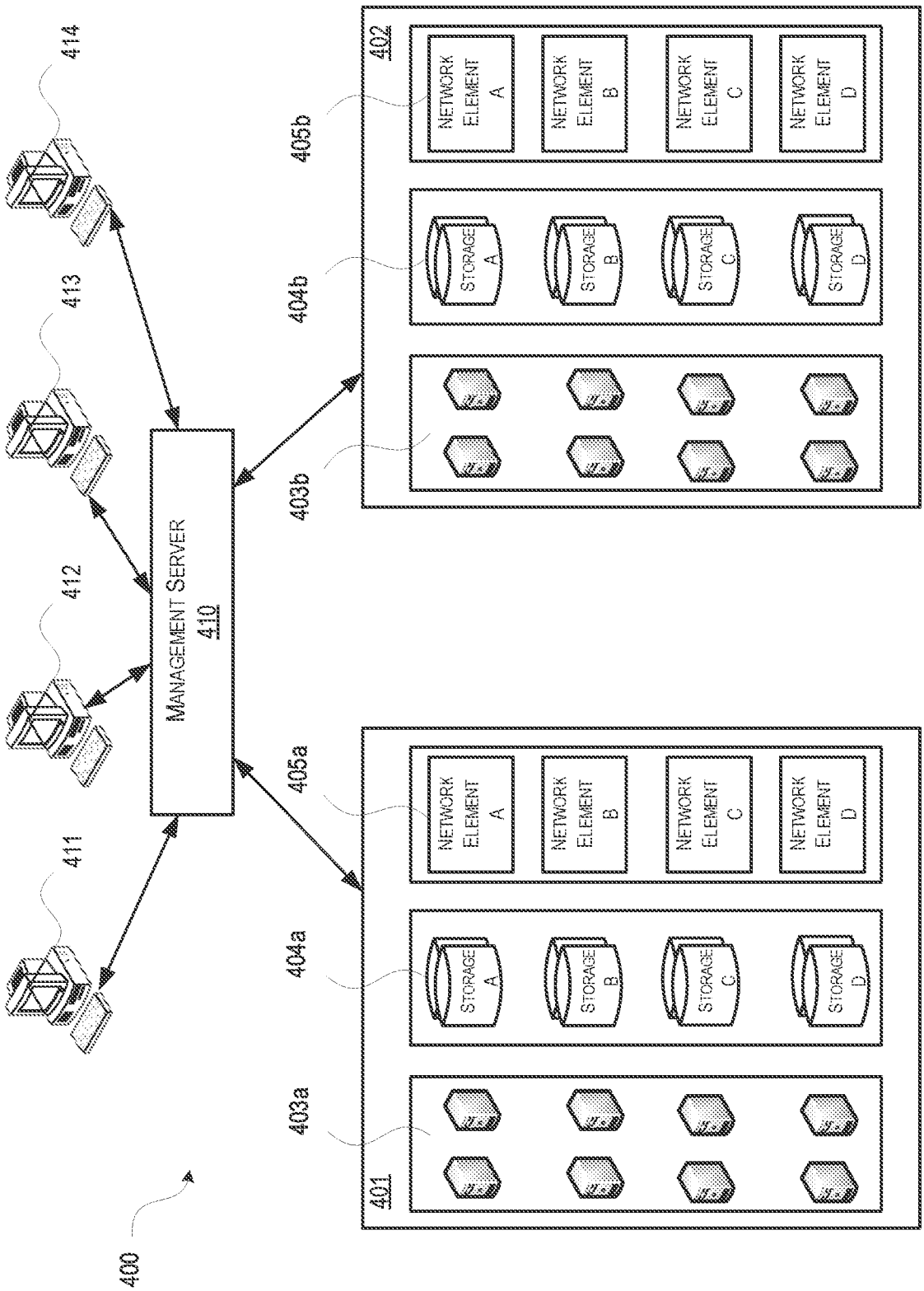

400
401
402
403a
403b
404a
404b
405a
405b
411
412
413
414
MANAGEMENT SERVER
410
NETWORK ELEMENT A
NETWORK ELEMENT B
NETWORK ELEMENT C
NETWORK ELEMENT D
STORAGE A
STORAGE B
STORAGE C
STORAGE D

FIG. 4

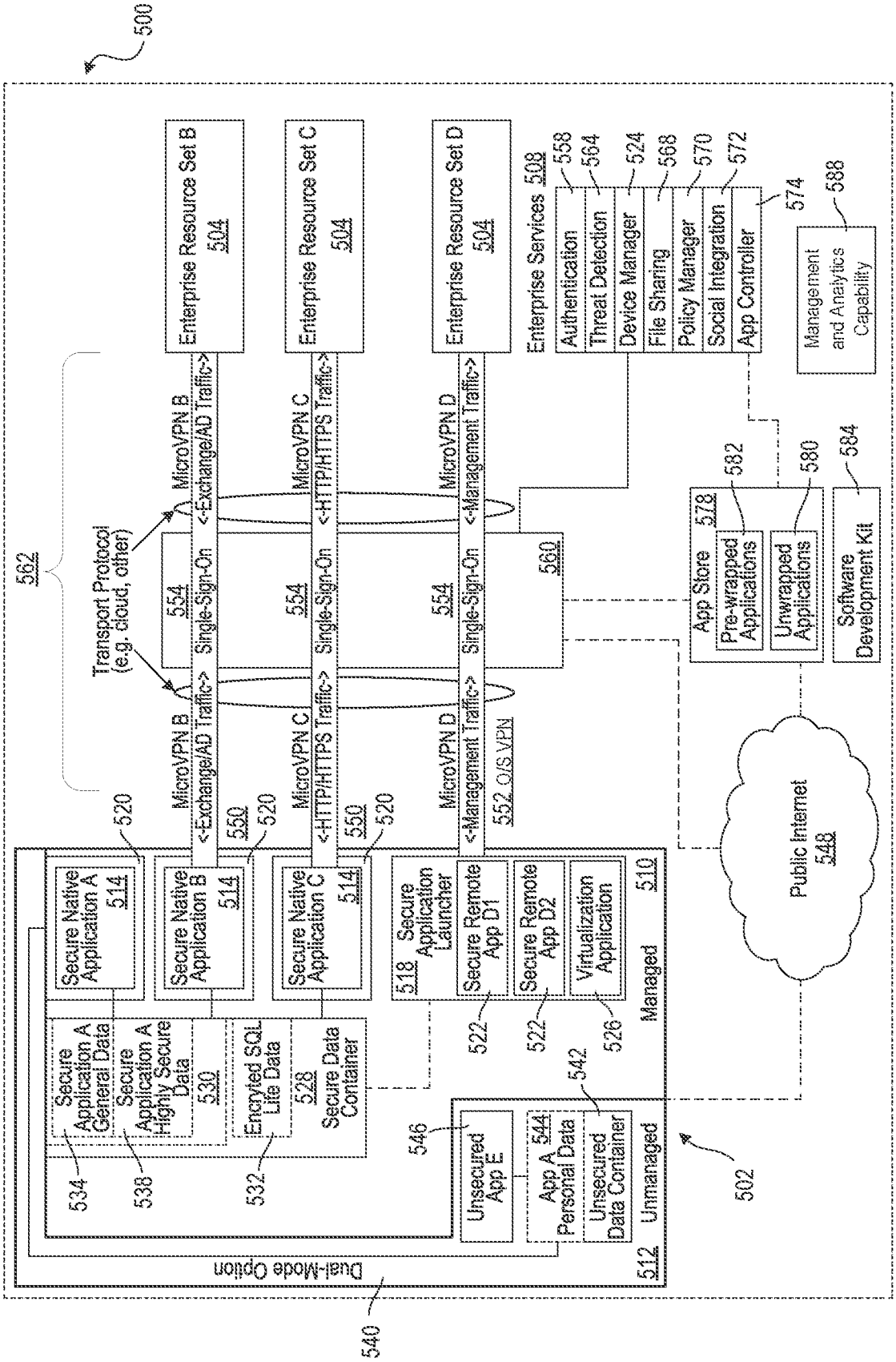

500
562
Transport Protocol (e.g. cloud, other)
MicroVPN B
<-Exchange/AD Traffic->
554
Single-Sign-On
MicroVPN C
<-HTTP/HTTPS Traffic->
MicroVPN D
<-Management Traffic->
552 O/S VPN
560
Enterprise Resource Set B 504
Enterprise Resource Set C 504
Enterprise Resource Set D 504
Enterprise Services 508
Authentication 558
Threat Detection 564
Device Manager 524
File Sharing 568
Policy Manager 570
Social Integration 572
App Controller 574
Management and Analytics Capability 588
App Store 578
Pre-wrapped Applications 582
Unwrapped Applications 580
Software Development Kit 584
Public Internet 548
Secure Native Application A 514
Secure Native Application B 514
Secure Native Application C 514
520
550
518 Secure Application Launcher
Secure Remote App D1
Secure Remote App D2
Virtualization Application
522
526
542
Secure Application A General Data 534
Secure Application A Highly Secure Data 530
538
Encrypted SQL Lite Data 532
Secure Data Container 528
Unsecured App E 546
App A Personal Data 544
Unsecured Data Container
Managed 510
Unmanaged 512
Dual-Mode Option
502
540

FIG. 5

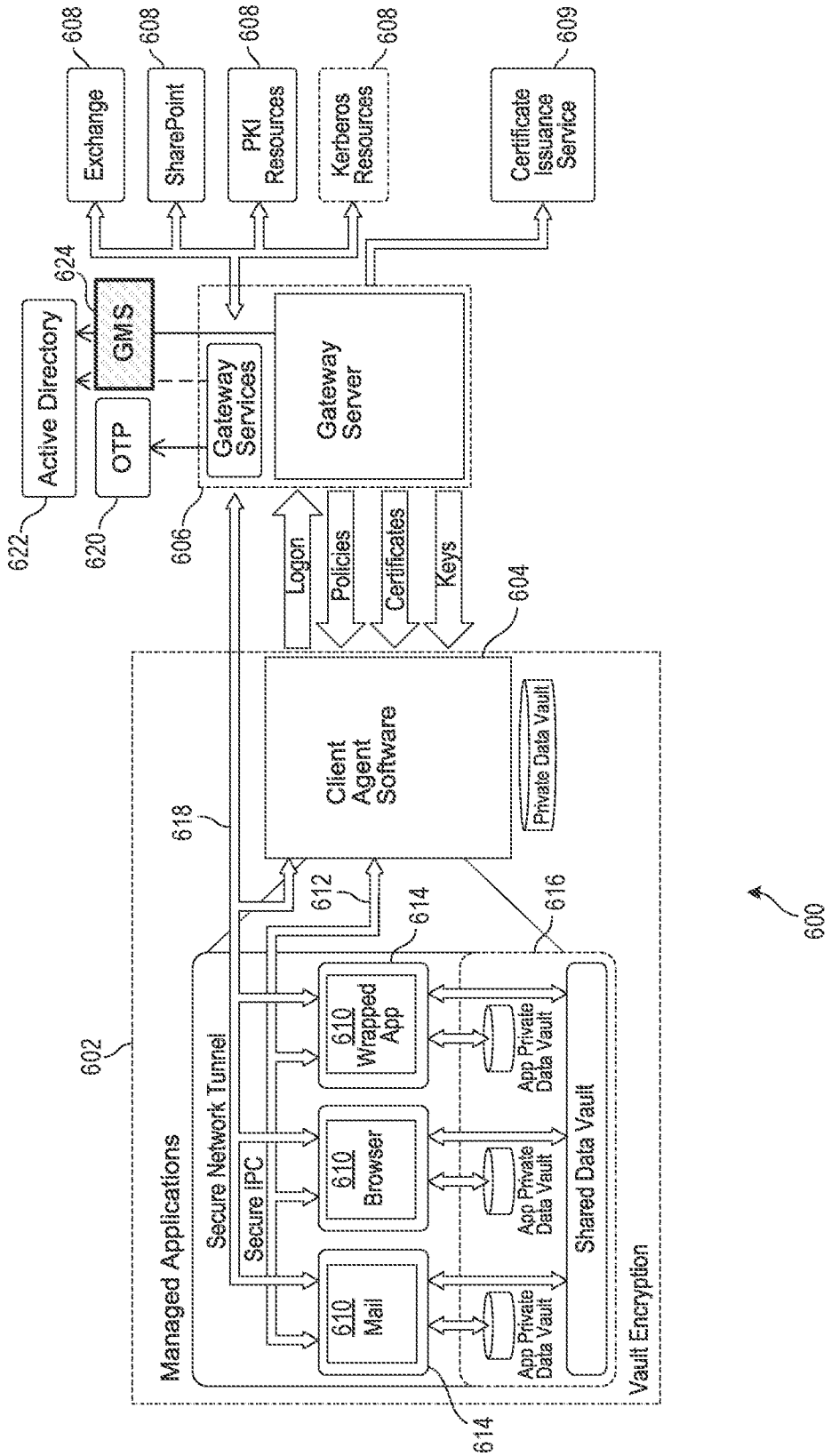

608
Exchange
608
SharePoint
608
PKI Resources
608
Kerberos Resources
609
Certificate Issuance Service
624
GMS
Active Directory
OTP
Gateway Services
Gateway Server
622
620
606
Logon
Policies
Certificates
Keys
604
Client Agent Software
Private Data Vault
618
612
614
616
602
Managed Applications
Secure Network Tunnel
Secure IPC
610 Wrapped App
610 Browser
610 Mail
App Private Data Vault
App Private Data Vault
App Private Data Vault
Shared Data Vault
Vault Encryption
614
600

FIG. 6

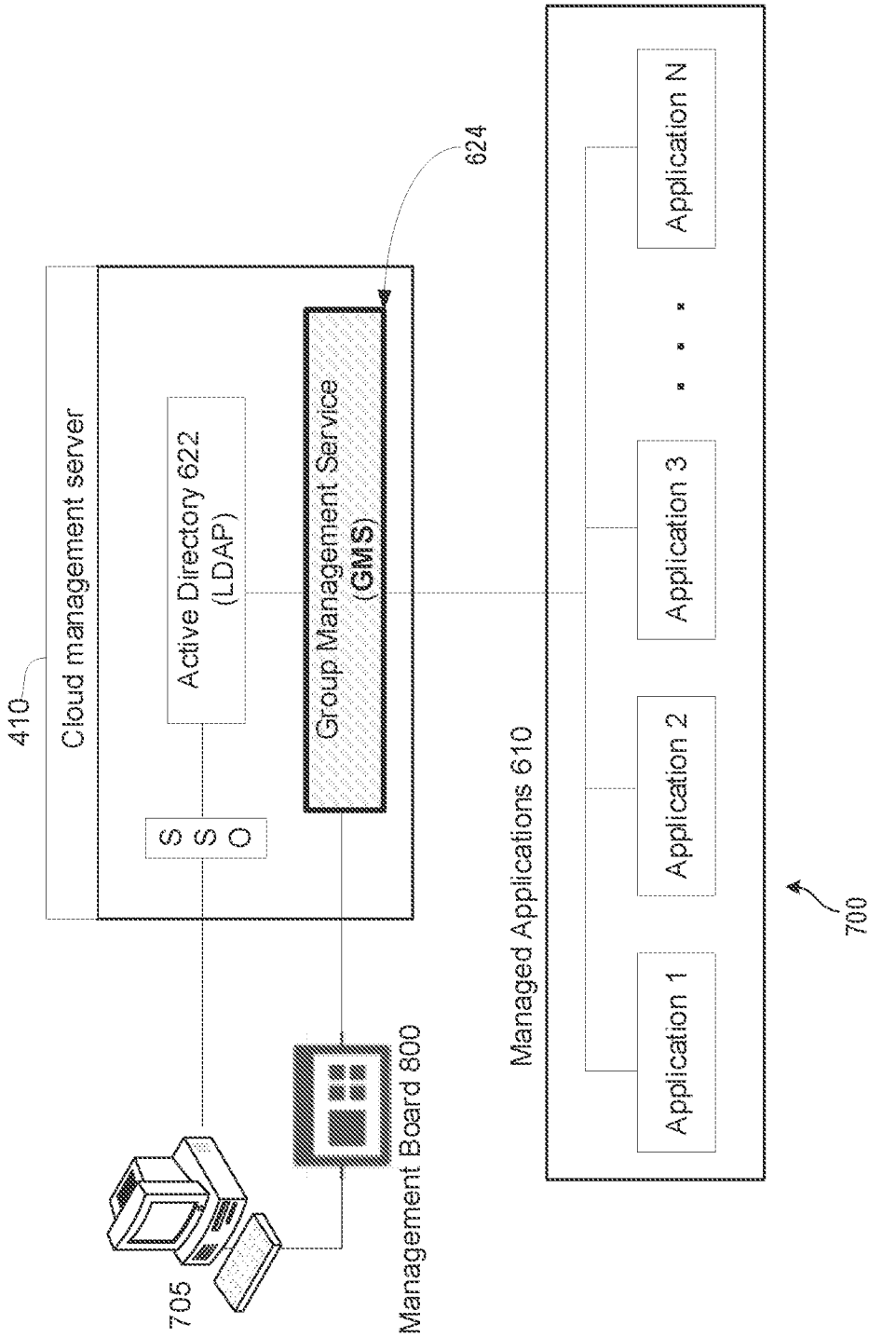

410
Cloud management server
Active Directory 622
(LDAP)
S S O
Group Management Service
(GMS)
624
Management Board 800
705
Managed Applications 610
Application 1
Application 2
Application 3
Application N
700

FIG. 7

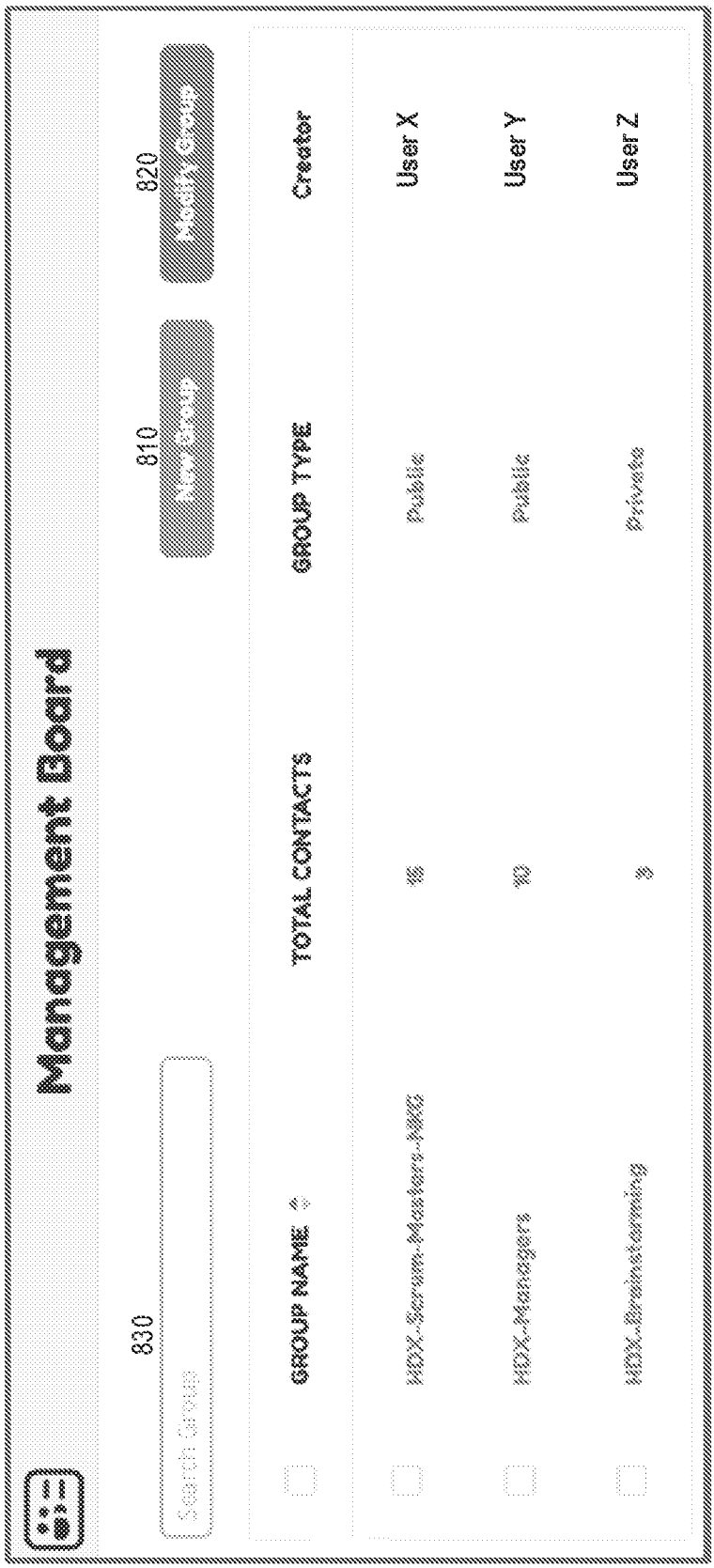

Management Board
830
Search Group
810
New Group
820
Modify Group
GROUP NAME
TOTAL CONTACTS
GROUP TYPE
Creator
HDX-Scrum-Masters-XXX
16
Public
User X
HDX-Managers
10
Public
User Y
HDX-Brainstorming
3
Private
User Z
800

FIG. 8

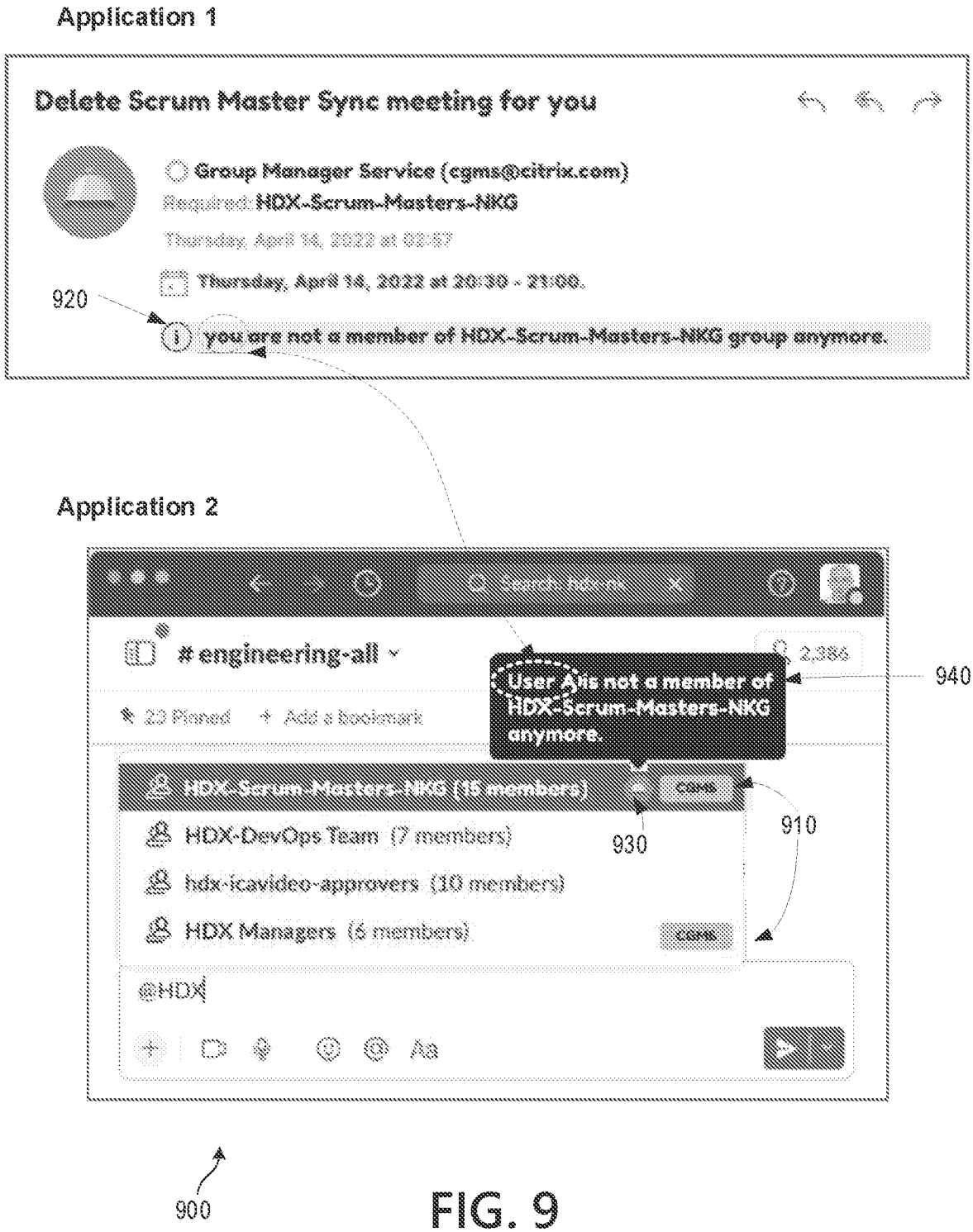

Application 1
Delete Scrum Master Sync meeting for you
Group Manager Service (cgms@citrix.com)
Required: HDX-Scrum-Masters-NKG
Thursday, April 14, 2022 at 20:30 - 21:00.
920
you are not a member of HDX-Scrum-Masters-NKG group anymore.
Application 2
engineering-all
2,386
940
23 Pinned
Add a bookmark
User A is not a member of HDX-Scrum-Masters-NKG anymore.
HDX-Scrum-Masters-NKG (15 members)
Copy
910
930
HDX-DevOps Team (7 members)
hdx-icavideo-approvers (10 members)
HDX Managers (6 members)
Copy
@HDX
Aa
900

FIG. 9

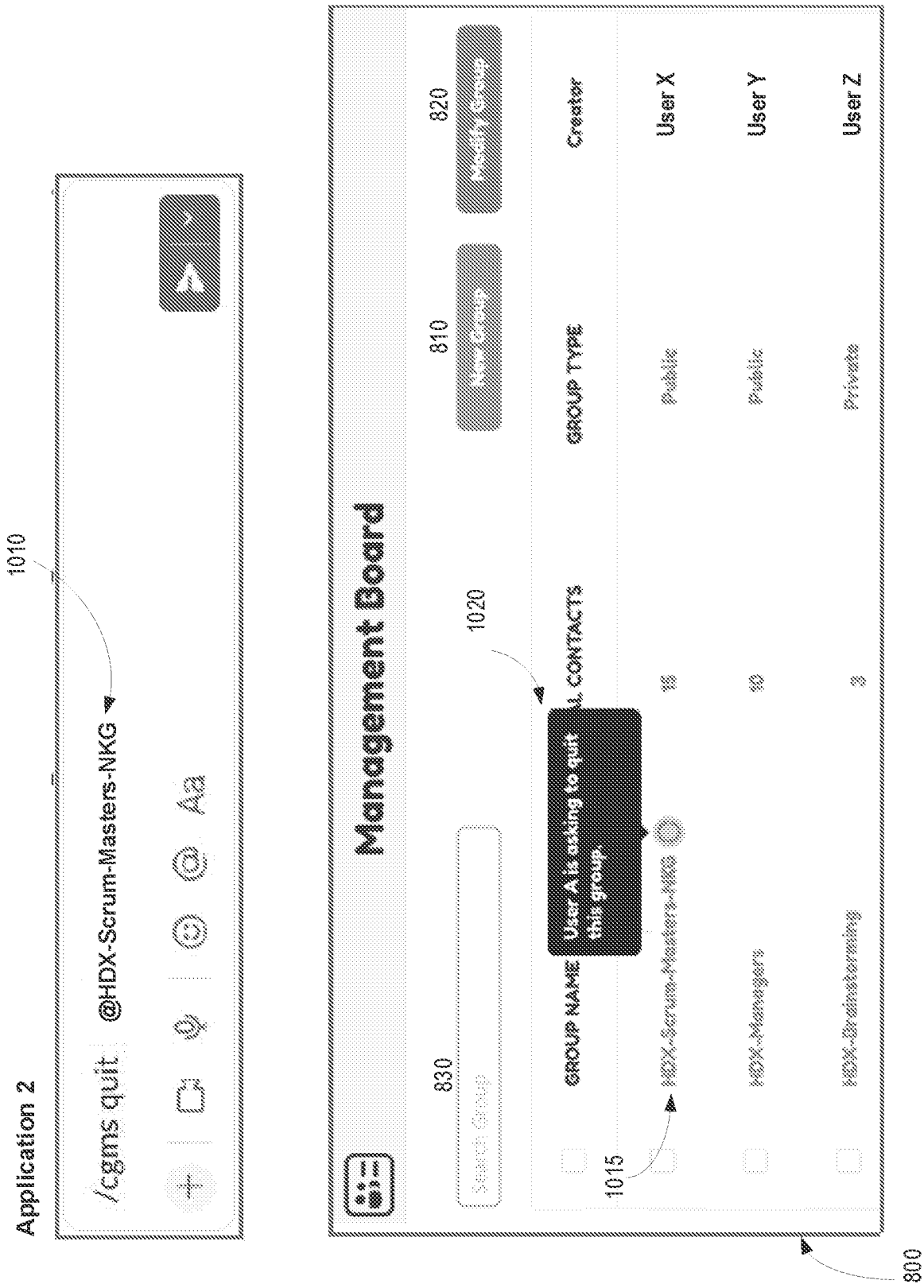

Application 2
/cgms quit @HDX-Scrum-Masters-NKG
1010
Management Board
830
Search Group
1020
810
820
GROUP NAME
GROUP TYPE
Creator
User A is going to quit this group
HDX-Scrum-Masters-NKG
HDX-Managers
HDX-Brainstorming
Public
Public
Private
User X
User Y
User Z
1015
800

FIG. 10

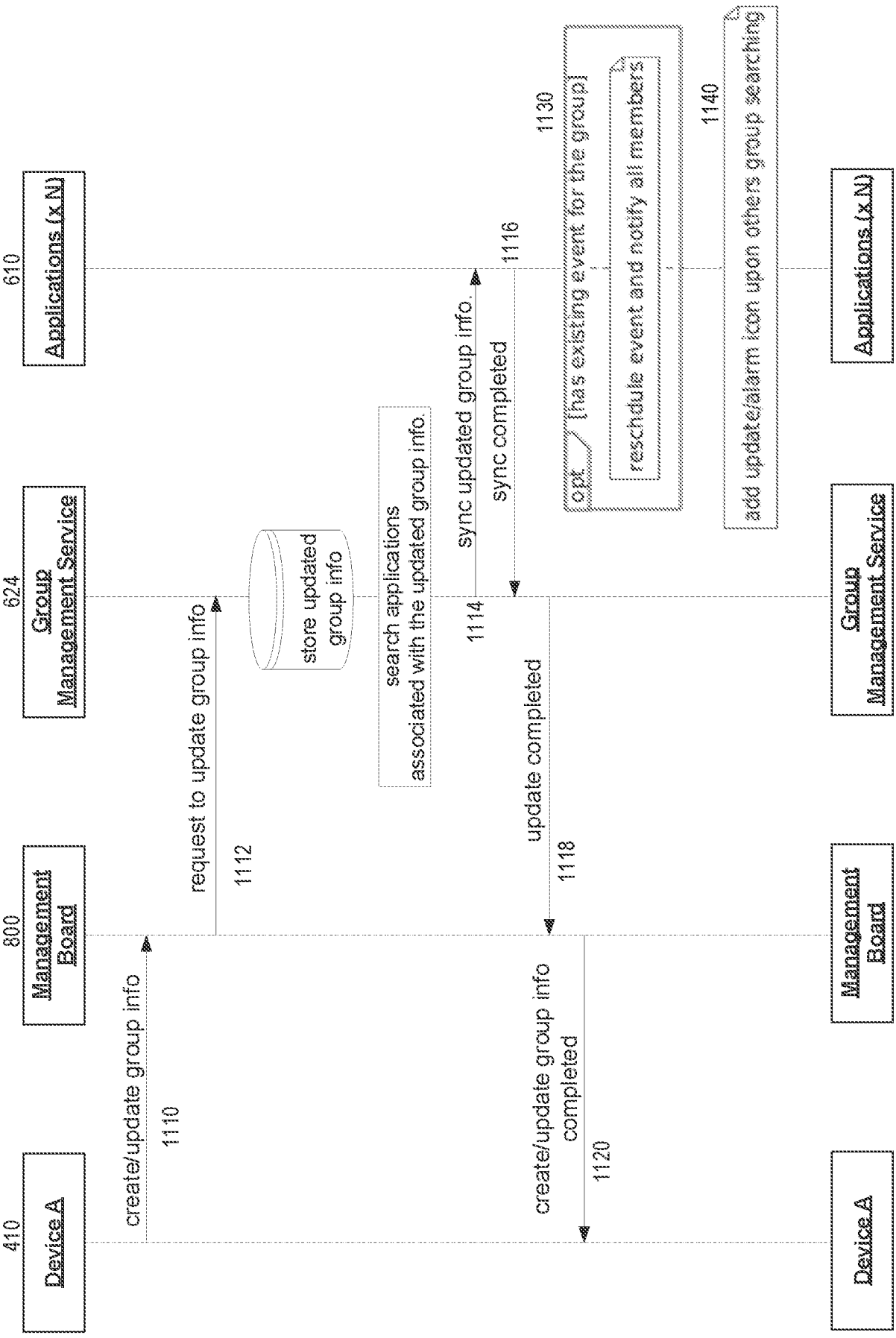

410
Device A
800
Management Board
624
Group Management Service
610
Applications (x,N)
create/update group info
1110
request to update group info
1112
store updated group info
search applications associated with the updated group info
1114
sync updated group info
1116
sync completed
1130
opt [has existing event for the group]
reschdule event and notify all members
update completed
1118
1140
add update/alarm icon upon others group searching
create/update group info completed
1120
Applications (x,N)
Group Management Service
Management Board
Device A

FIG. 11

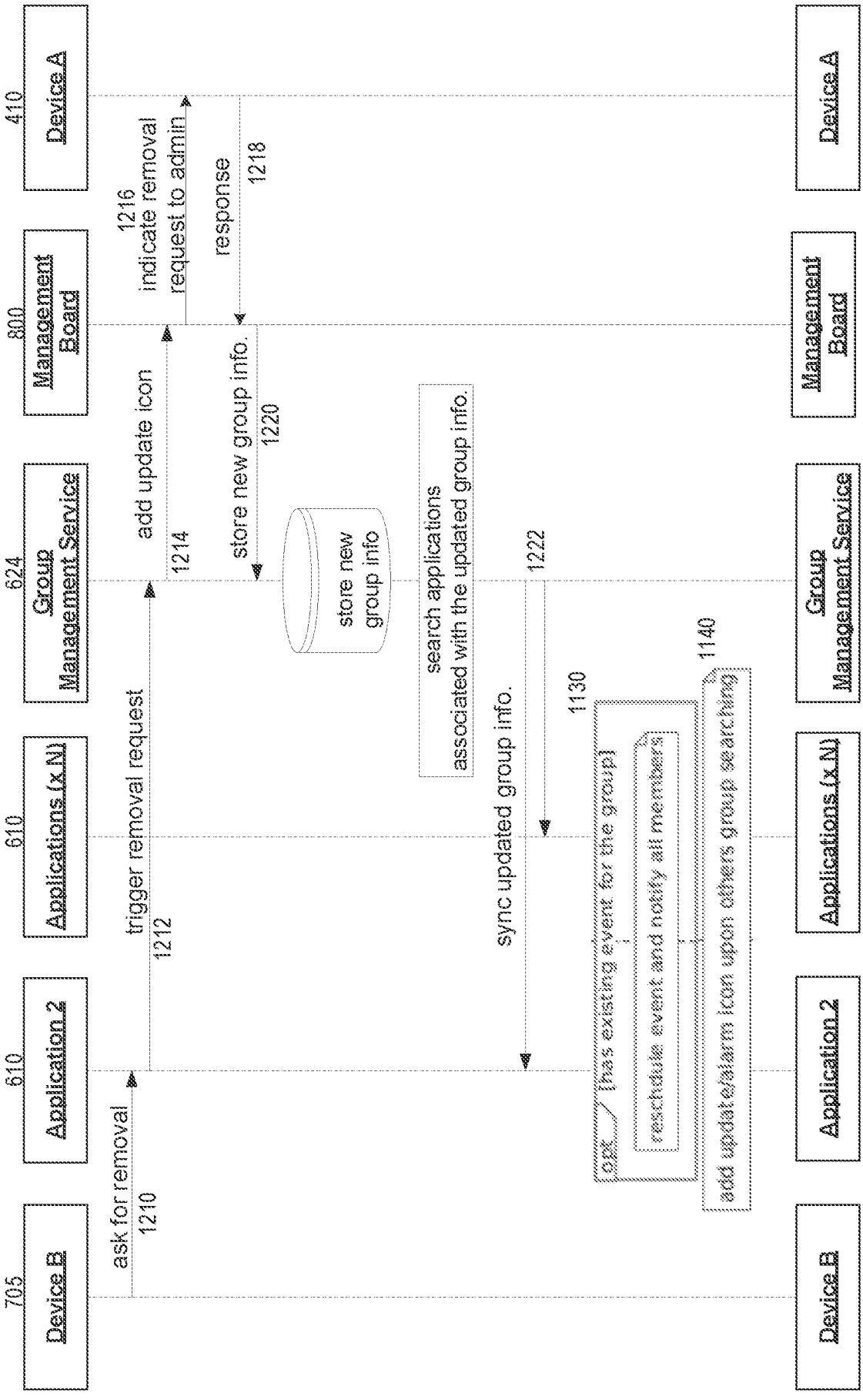

705
Device B
610
Application 2
610
Applications (x N)
624
Group Management Service
800
Management Board
410
Device A
ask for removal
1210
trigger removal request
1212
add update icon
1214
store new group info.
1220
store new group info
search applications associated with the updated group info.
1222
sync updated group info.
1130
opt [has existing event for the group]
reschdule event and notify all members
1140
add update/alarm icon upon others group searching
1216
indicate removal request to admin
response
1218
Device A
Management Board
Group Management Service
Applications (x N)
Application 2
Device B

FIG. 12

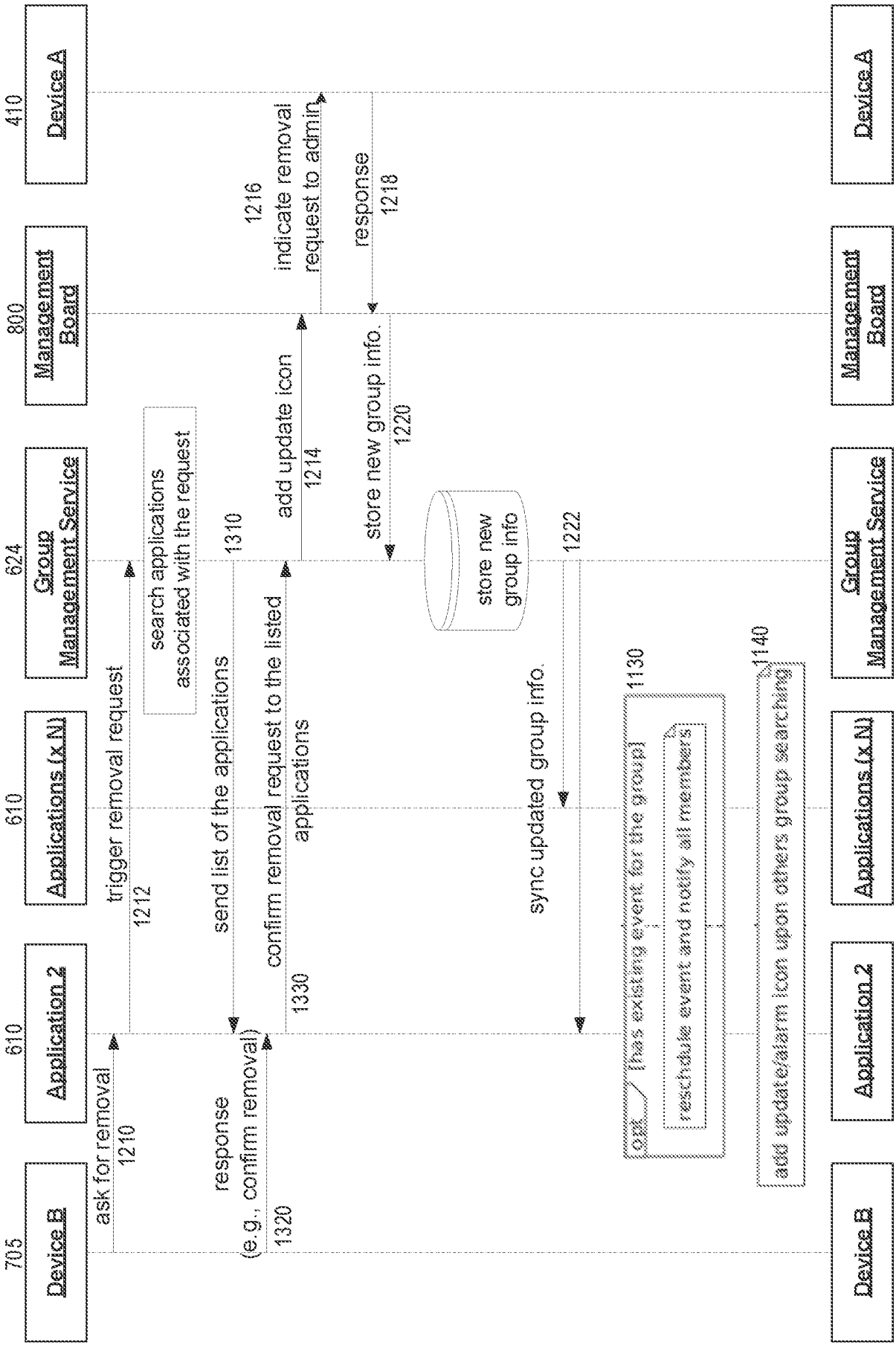

705
Device B
610
Application 2
610
Applications (x N)
624
Group Management Service
800
Management Board
410
Device A
ask for removal
1210
trigger removal request
1212
search applications associated with the request
send list of the applications
1310
confirm removal request to the listed applications
1330
add update icon
1214
1216
indicate removal request to admin
response
1218
store new group info.
1220
store new group info
sync updated group info.
1222
response (e.g., confirm removal)
1320
opt [has existing event for the group]
1130
reschduke event and notify all members
add update/alarm icon upon others group searching
1140
Device B
Application 2
Applications (x N)
Group Management Service
Management Board
Device A

FIG. 13

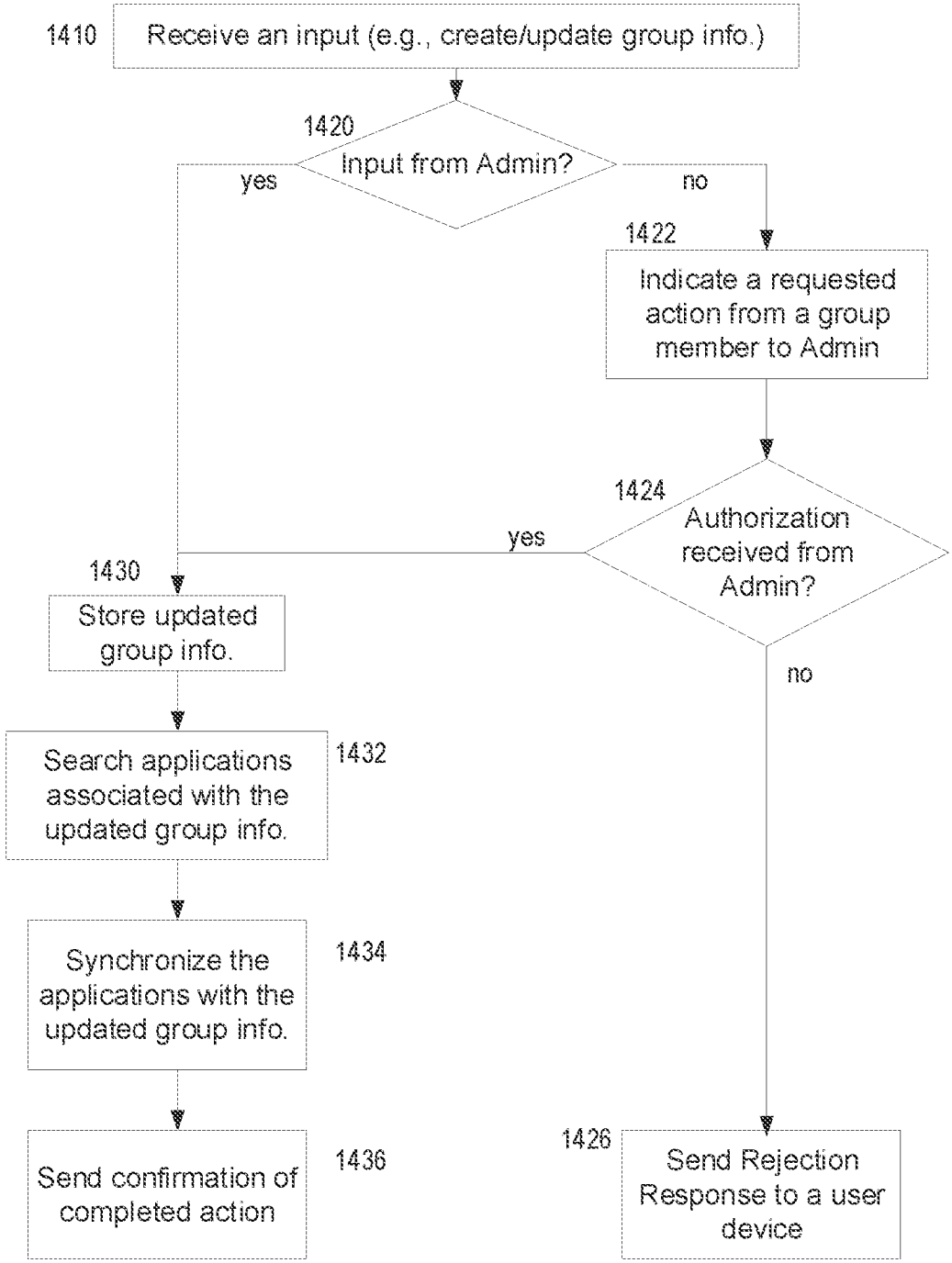

1410
Receive an input (e.g., create/update group info.)
1420
Input from Admin?
yes
no
1422
Indicate a requested action from a group member to Admin
1424
Authorization received from Admin?
yes
no
1430
Store updated group info.
Search applications associated with the updated group info.
1432
Synchronize the applications with the updated group info.
1434
Send confirmation of completed action
1436
1426
Send Rejection Response to a user device

FIG. 14

INTERNATIONAL SEARCH REPORT

International application No.
PCT/CN2022/121549

A. CLASSIFICATION OF SUBJECT MATTER

G06F15/16(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC: G06F, G06Q, H04L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

CNABS, CNKI, CNTXT, EPTXT, WOTXT, VCN, USTXT, IEEE: group, manage+, member, primary, secondary, authenticat+, identif+, grant, creat+, delete, updat+, synchroniz+

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	WO 2018132304 A1 (MICROSOFT TECHNOLOGY LICENSING, LLC) 19 July 2018 (2018-07-19) description, paragraphs 0038-0072, figures 1-2	1-20
A	CN 105830388 A (AMAZON TECHNOLOGIES INC.) 03 August 2016 (2016-08-03) the whole document	1-20
A	CN 111052706 A (CITRIX SYSTEMS, INC.) 21 April 2020 (2020-04-21) the whole document	1-20
A	US 2013297700 A1 (CITRIX SYSTEMS, INC.) 07 November 2013 (2013-11-07) the whole document	1-20
A	US 2022043698 A1 (CITRIX SYSTEMS, INC.) 10 February 2022 (2022-02-10) the whole document	1-20
A	WO 2007100942 A2 (CITRIX SYSTEMS, INC. et al.) 07 September 2007 (2007-09-07) the whole document	1-20
A	US 2013036459 A1 (SAFEFACES LLC) 07 February 2013 (2013-02-07) the whole document	1-20

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"D" document cited by the applicant in the international application
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
18 May 2023	**24 May 2023**

Name and mailing address of the ISA/CN	Authorized officer
CHINA NATIONAL INTELLECTUAL PROPERTY ADMINISTRATION **6, Xitucheng Rd., Jimen Bridge, Haidian District, Beijing 100088, China**	**FAN,YuXia** Telephone No. **(+86) 010-53961331**

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CN2022/121549

Patent document cited in search report			Publication date (day/month/year)	Patent family member(s)			Publication date (day/month/year)
WO	2018132304	A1	19 July 2018	US	2018205676	A1	19 July 2018
				IN	201741001493	A	20 July 2018
CN	105830388	A	03 August 2016	CA	2930255	A1	14 May 2015
				EP	3069464	A2	21 September 2016
				US	2015135272	A1	14 May 2015
				JP	2017503231	A	26 January 2017
				WO	2015070246	A2	14 May 2015
CN	111052706	A	21 April 2020	US	2020314088	A1	01 October 2020
				AU	2018318922	A1	12 March 2020
				US	2019058706	A1	21 February 2019
				EP	3669512	A1	24 June 2020
				WO	2019036337	A1	21 February 2019
				CA	3073086	A1	21 February 2019
				JP	2020531973	A	05 November 2020
US	2013297700	A1	07 November 2013		None		
US	2022043698	A1	10 February 2022	WO	2022027289	A1	10 February 2022
WO	2007100942	A2	07 September 2007		None		
US	2013036459	A1	07 February 2013	WO	2013022780	A1	14 February 2013
				EP	2740067	A1	11 June 2014