



(22) Date de dépôt/Filing Date: 2013/02/15

(41) Mise à la disp. pub./Open to Public Insp.: 2013/08/16

(30) Priorité/Priority: 2012/02/16 (US61/599,769)

(51) Cl.Int./Int.Cl. *G06F 21/30* (2013.01),
G06F 21/62 (2013.01), *H04W 12/08* (2009.01),
G06F 9/445 (2006.01), *H04L 9/00* (2006.01)

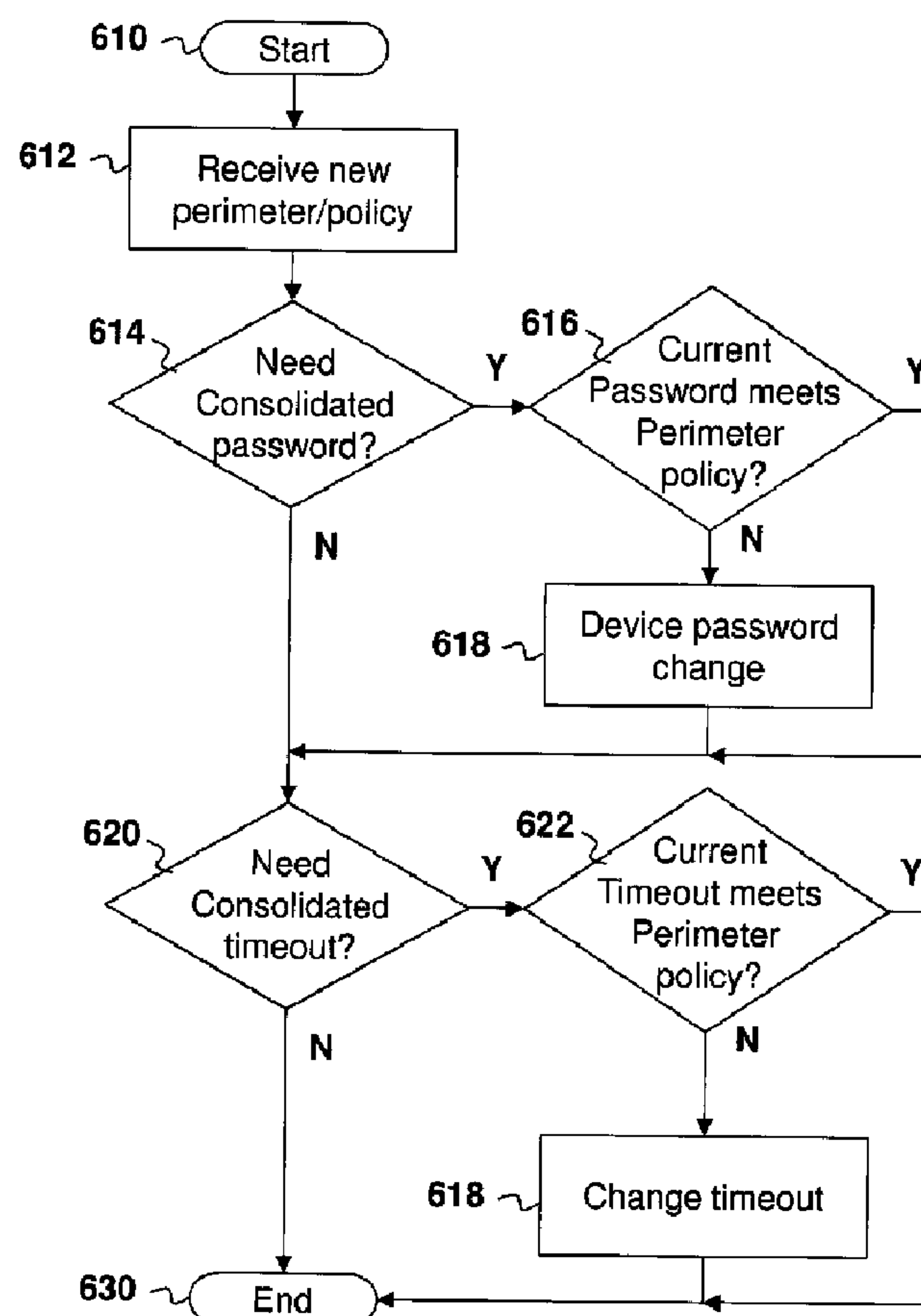
(71) Demandeurs/Applicants:
RESEARCH IN MOTION LIMITED, CA;
QNX SOFTWARE SYSTEMS LIMITED, CA

(72) Inventeurs/Inventors:
NAGARAJAN, SIVAKUMAR, CA;
BENDER, CHRISTOPHER LYLE, CA;
MCCONNAUGHAY, MARK A., CA

(74) Agent: MOFFAT & CO.

(54) Titre : PROCÉDE ET APPAREIL POUR LA GESTION DE RESSOURCES GROUPEES MULTIPLES AU DISPOSITIF

(54) Title: METHOD AND APPARATUS FOR MANAGEMENT OF MULTIPLE GROUPED RESOURCES ON DEVICE



(57) Abrégé/Abstract:

A method and computing device for managing grouped resources comprising receiving, at the computing device, a policy for a set of grouped resources; applying the policy; locking at least one of the computing device or the set of grouped resources associated



(57) **Abrégé(suite)/Abstract(continued):**

with the policy; waiting for receipt of an authentication parameter at the computing device; verifying the authentication parameter; associating the set of grouped resources with the authentication parameter; and unlocking the least one of the computing device or the set of grouped resources.

ABSTRACT

A method and computing device for managing grouped resources comprising receiving, at the computing device, a policy for a set of grouped resources; applying the policy; locking at least one of the computing device or the set of grouped resources associated with the policy; waiting for receipt of an authentication parameter at the computing device; verifying the authentication parameter; associating the set of grouped resources with the authentication parameter; and unlocking the least one of the computing device or the set of grouped resources.

METHOD AND APPARATUS FOR MANAGEMENT OF MULTIPLE GROUPED RESOURCES ON DEVICE

FIELD OF THE DISCLOSURE

[0001] The present disclosure relates to computing devices and in particular relates to application security on the computing devices.

BACKGROUND

[0002] Computing devices, including mobile devices such as personal digital assistants, smart phones, cellular telephones, laptops, tablet computers, among others, as well as desktop or other computing devices, have the capability of running multiple applications at one time. For security purposes, a user or an IT policy for the user may dictate that the computer device lock access to the applications that are running after a period of inactivity. For example, if the user has not touched the computing device for 2 minutes, all of the applications on the device may lock. The password may, for security purposes, have policies associated with it, such as the length of time before timeout, password length, password complexity.

BRIEF DESCRIPTION OF THE DRAWINGS

[0003] The present disclosure will be better understood with reference to the drawings, in which:

Figure 1 is a block diagram showing an example computing device;

Figure 2 is a block diagram showing application and data memory on a mobile device;

Figure 3A is a block diagram showing the connection of a corporate device to a personal device;

Figure 3B is a block diagram showing the connection of a device to a network;

Figure 4 is a flow diagram for soft locking of a mobile device;

Figure 5 is a flow diagram showing the association of a password with a perimeter policy;

Figure 6 is a flow diagram showing consolidation of password and timeout values on a device;

Figure 7 is an example system architecture diagram for a mobile device; and

Figure 8 is a block diagram showing an example mobile device capable of being used with the present disclosure.

DETAILED DESCRIPTION

[0004] The present disclosure provides a method for managing grouped resources on a computing device comprising: receiving, at the computing device, a policy for a set of grouped resources; applying the policy; locking at least one of the computing device or the set of grouped resources associated with the policy; waiting for receipt of an authentication parameter at the computing device; verifying the authentication parameter; associating the set of grouped resources with the authentication parameter; and unlocking the least one of the computing device or the set of grouped resources.

[0005] The present disclosure further provides a computing device capable of managing grouped resources, the computing device comprising: a processor; and a communications subsystem, wherein the computing device is capable of: receiving, at the computing device, a policy for a set of grouped resources; applying the policy; locking at least one of the computing device or the set or grouped resources associated with the policy; waiting for receipt of an authentication parameter at the computing device; verifying the authentication parameter; associating the set of grouped resources with the authentication parameter; and unlocking the least one of the computing device or the set of grouped resources.

[0006] The present disclosure provides for a mobile device, but is not meant to be limited to any particular mobile device. Examples of mobile devices can include smart

phones, personal digital assistants, data enabled cellular telephones, tablet computers, among others.

[0007] Reference is now made to **Figure 1**, which shows a simplified diagram of a computing device **100**. Computing device **100** may comprise a tablet, mobile device, personal computer, laptop computer, among others. The embodiment of **Figure 1** is however not meant to be limiting and other devices could be used.

[0008] Computing device **100** generally includes a processor **138**, which controls the overall operation of the device. Processor **138** interacts with device subsystems such as the display **122**, memory **124**, auxiliary input/output (I/O) subsystems **128**, serial port **130**, one or more keyboards or keypads **132**, where keyboard or keypad **132** may comprise a physical keyboard or a virtual keyboard or both, one or more speakers **134**, microphone **136**, other communication subsystem **140** such as a short-range communications subsystem, including Bluetooth and near field communications, and any other device subsystems generally designated as **142**. Serial port **130** could include a USB port or other port.

[0009] Memory **124** may be segregated into various perimeters, as described below. As used herein, perimeters refer to groups of resources that share a common management scheme governing their use, and perimeters encompass resources and management policies that describe how they may be used. Each perimeter may have policies associated with it, where the policy controls passwords, encryption, inactivity timeouts, network access control, among other items. A perimeter may include file system resources, network (connection profiles and VPN configurations), applications, device and application configuration, certificates and policies, and perimeter policies govern what resources applications can access within and outside of the perimeter. For example, a device may have perimeters for personal applications and data, enterprise or corporate applications or data, applications or data resulting from a connection with a second device, among others. A perimeter is sometimes referred to as a set of grouped resources herein.

[0010] Operating system software used by the processor **138** may be stored in memory **124**. The operating system, specific device applications, or parts thereof, may be temporarily loaded into a volatile memory such as RAM **126**.

[0011] Applications may be loaded onto the device and associated with a perimeter in some cases. Such applications and data for the application may be stored in memory associated with the perimeter.

[0012] In some embodiments, computing device **100** may optionally include a communications subsystem **111** capable of communication with a data access point. Such data access point may include a cellular network or Wi-Fi or WiMAX network, among others. In further embodiments, computing device **100** may be capable of voice communications.

[0013] In certain situations, a dual or plural mode of operation or perimeters may exist for a mobile device, where the mobile device may run certain applications and access certain data in one portion that is not accessible or cannot be run in other portion or portions of the device.

[0014] Reference is now made to **Figure 2**, which shows a block diagram of the memory **210** of a mobile device. The memory is configured to store applications and application data, such combination of stored applications and data being referred to herein as an application space. The memory **210** is divided into a first perimeter **220**, a second perimeter **230**, and a perimeter X **240** in the example of **Figure 9**, where X is the number of perimeters on the device. In one embodiment of the present disclosure, it is possible to have only two perimeters on the device. In other embodiments, there are more than two perimeters on the device.

[0015] Second perimeter **230** may, for example, be associated with a corporate mode, and thus generally comprises a portion of memory on the mobile device segregated for data, applications, or both, which may be considered sensitive to a business,

corporation, enterprise, government, non-profit organization, a user of the device or any other entity setting an information technology policy for the computing device.

[0016] First perimeter **220** may, for example, be associated with a personal mode, and thus generally comprises a portion of memory segregated for personal applications and data, where personal applications or data may be considered outside of, or separate from, an information technology policy.

[0017] Other perimeters may be associated with different levels of corporate data, where certain data is considered more sensitive and thus the perimeter for this data may have a policy for passwords and inactivity timeouts that is more secure than a second corporate perimeter. Other perimeters could be associated with a bridge or connection to a device trusted by a corporate IT policy. Other examples of perimeters are possible.

[0018] Within first perimeter **220**, a plurality of applications **222** can communicate with first data **224**.

[0019] Similarly, in second perimeter **230**, a plurality of applications **232** communicate with second data **234**. In perimeter **240**, a plurality of applications **242** communicate with data **244**.

[0020] By segregating various applications from others and data associated with each, IT policies can be implemented on the device for certain data, thereby protecting the data, while still allowing for flexibility for other applications and data.

[0021] A perimeter manager **260** manages the policies associated with each perimeter as described in more detail below.

[0022] In one embodiment, applications **232** could be provided with additional security over applications **222**. For example, before an application **232** could be launched, the user may need to enter a password. Further, inactivity timers could be implemented to

lock applications **232** after a period of inactivity while leaving personal applications unlocked. A locked application may require a user to initially enter a password to unlock the application and interact with and access data from the application.

[0023] The designation of the application may further limit what data that application has access to. Thus, for example, applications **232** may run in their own mode where any data that they write can never be accessed by applications **222**. The limitation would be that applications **222** are not able to read data **234**, nor is an application **232** capable of writing to personal data **224**. In other embodiments, applications **232** may be able to read data **224**.

[0024] Data **234** may be encrypted for security. Such encryption and the storing of encryption keys may be managed on the device or externally on a trusted device or server.

[0025] Data **234** may also have date of deletion policies in effect on the mobile device. Thus, if data **234** is not accessed within a certain time period, it can be wiped out pursuant to a data reaping timeline. For example, if data is not accessed on the mobile or computing device for seven days, the data may be deleted from the mobile device. The user would then need to download the data again if it was required. This may be implemented through tags or data tables associated with the data.

[0026] An operating system on the mobile device can enforce the above differentiating rules between perimeters. For example, an operating system may implement data access for the various applications **222** and **232**, where each application is given a group permission, similar to UNIX group permissions. In other embodiments, other user permissions or other permission systems may also be used. Data is further designated in files that allow access by certain groups. Thus, the operating system may allow data **234** to be accessed only by applications **232** that have group permissions to access such data. Similarly, data **224** may be written to or read only by applications **222** based on the group permissions of application **222** with regard to data **224**. Applications **232**

however do not have group permissions to write to data **224** in one embodiment, as enforced by the operating system.

[0027] In one embodiment, data **234** may be corporate data and be provided to a device based on a secure connection with the corporate network. For example, this may be done through a virtual private network or other secure connection to an enterprise server. The enterprise server can manage policies for perimeter **230** in the example.

[0028] In some embodiments, a particular device may be connected to a secure (IT Trusted) device. Reference is now made to **Figure 3A**.

[0029] In **Figure 3A**, the IT trusted device is mobile device **305**. However, this is merely an example and other possibilities for IT trusted devices exist.

[0030] In order to run data from mobile device **305** on computing device **300**, a client **310** may be provided on the computing device **300**. Client **310** communicates with a server **320** on mobile device **305** to obtain data from the device.

[0031] Further, computing device **300** may include memory **330**, which has a first perimeter space **334** for storing applications that may be run on computing device **300**. In the example of **Figure 3A**, first perimeter space may be considered a bridge perimeter and have policies set by mobile device **305**.

[0032] Computing device **300** may also have a second perimeter **332** and a third perimeter **335** within memory **330**.

[0033] As seen in the example of **Figure 3A**, the second perimeter space **332** contains applications which may access data **336**. However, in some embodiments no similar data exists for applications in first perimeter space **334**.

[0034] In an alternative embodiment, first perimeter space **334** can have data **338** which can be regulated by the same policies as data **348** on mobile device **305**. Thus, data

338 would be subject to access restrictions to specific applications, garbage collection, restrictions on copying or cutting, among the other restrictions. The client **310** can provide enablement for this functionality.

[0035] On mobile device **305**, the divided modes are similarly provided. In particular, memory **340** contains applications **342**, applications **344** and application **345** in the example of **Figure 3A**. This is similar to the embodiments described above with regard to **Figure 2**.

[0036] In one embodiment, computing device **300** may have a different number of perimeters than mobile device **305**.

[0037] In order to provide security, an application **334** may be started on computing device **300**. As indicated above, a password may be required to start such application, and the password may be verified at mobile device **305**.

[0038] Client **310** recognizes that an application in first perimeter area **334** is running and can communicate with server **320** to indicate that data can be provided. In this way server **320** can access the data that is either in data storage **348** or at a network server (not shown).

[0039] Further, applications in first perimeter area **344** do not necessarily have to be the same as applications in first perimeter area **334**. For example, with a larger display, computing device **300** may be able to run different applications or variations of applications on the mobile device **305**. The data **348** may be the same between the two sets of applications, but can be displayed to the user or used by applications on the computing device differently than the data **348** would be used on mobile device **305**.

[0040] The data **338** may then be provided over a connection **360** between mobile device **305** and computing device **300**. Connection **360** may comprise any short or long range wired or wireless connection, and examples of such connections include

Bluetooth™, USB, Infrared Data Assn (IrDA), Wi-Fi, Radio-frequency identification (RFID), Near Field Communication (NFC) connections, among others.

[0041] Communication over link **360** can be secure. That is, data that is passed to computing device **300** or back to mobile device **305** may be encrypted using a key known to both computing device **300** and mobile device **305**.

[0042] Further, in one embodiment any data that is stored is encrypted. In this case, the encryption key for the stored data in a perimeter may be stored on mobile device **305**, thus necessitating the connection in order to decrypt the data on the computing device **300**.

[0043] Further, it may be a policy that the data is not stored on computing device **300**. Thus, except for some possible caching, corporate data will not be stored on device **300**. Further, client **310** can ensure that the cache is cleared prior to an application shutting down.

[0044] In a further embodiment, client **310** may communicate directly with a network server to access data that may be stored at the network server. In particular, reference is now made to **Figure 3B**.

[0045] A computing device **300** may contain a number of perimeters, as discussed above with regard to **Figures 2** and **3A**. In the example of **Figure 3B**, three perimeters are provided and are each configured with various policies for the perimeter. The perimeters are shown with the same reference numerals as **Figure 3A**, namely a first perimeter having applications **335** and data **339**, a second perimeter having applications **332** and data **336**, and a third perimeter having applications **334** and optionally data **338**.

[0046] In the example of **Figure 3B**, device **300** has a client **310** for managing the perimeters. However, in alternative embodiments, a perimeter manager such as perimeter manager **260** from **Figure 2** may be used at computing device **300**.

[0047] Computing device **300** communicates over a link **365** with a network element **370**. Network element **370** may be any network server or combination of network servers or storage, and may for example be considered a cloud.

[0048] Link **365** may be any short or long range communications connection and may include cellular connections, Bluetooth™, USB, Infrared Data Assn (IrDA), Wi-Fi, Radio-frequency identification (RFID), Near Field Communication (NFC) connections, among others.

[0049] Network element **370** may include a server module **375** for communications with a client **310**. However, such a server module is optional and in other embodiments the network element may manage access to applications and data using other modules or protocols.

[0050] Network element **370** may include applications and/or data associated with the various perimeters on computing device **300** and access to such data or applications may be managed based on perimeter policies for security and access. Thus, in the example of **Figure 3B**, applications **380** and data **382** may be associated with applications **335** and data **339** on device **300**. Similarly, data **384** may be associated with data **336** on device **300**. In this case, applications **332** exist only on the perimeter on device **300** and no similar applications are provided within network element **370**. The association of the data on the device and server could include a subset of data being stored on the device. For example, a complete music collection or picture library, among other example of data, could be stored on the server and only a subset stored on the device. In other examples, the server may contain only a subset of the data on

the device. For example, if data is backed up to the server, only some of the data on the device may be backed up.

[0051] Further, applications **386** and data **388** on network element **370** may be associated with applications **334** and data **338** on computing device **300**.

[0052] In some cases data, such as data **339**, **336** or **338** on computing device **300** may form only a subset of data **382**, **384** and **388** on network element **370**. For example, if data **382** is a music collection on a cloud storage, data **339** on computing device **300** may only contain a subset of the music a user is currently listening to. In other cases, there may be no data **339** stored on computing device **300** and computing device **300** may stream the music directly from network element **370**.

[0053] One example of the embodiment of **Figure 3B** could be cloud storage of data and applications for various perimeters. The perimeters may be broken into personal and corporate perimeters and may also be broken into perimeters based on users of the device. Thus, for example, computing device **300** may be a device used by more than one member of a family, where a first member has one or more personal perimeters and one or more corporate perimeters and a second member has one or more personal perimeters and one or more corporate perimeters. In this case, various data or applications on network element **370** may be shared between users, but other data may be secured and inaccessible by other family members sharing the device. Further, work or corporate perimeters for a particular family member may be managed based on corporate perimeter policies and access to data and applications may be restricted. In this case, a separate password for the perimeter may be established.

[0054] In other cases, perimeters may be established as individual perimeters and other perimeters may be established as shared perimeters. For example, a music application and data may be in a perimeter shared by various family members, but each family member may have a personal perimeter containing an email application and address

book, thereby allowing the various users to keep addresses and emails separate, but to share the music collection.

[0055] Policies for the various perimeters may be set on computing device **300**, network element **370**, or some other network element such as an enterprise server. In the case of an enterprise server or other network element setting the policies for the perimeter, such policies may be sent to one or both of computing device **300** and network element **370**.

[0056] Other examples are possible.

[0057] When enterprise perimeters are created on the device, they are associated with policies that are used to control the access to those perimeters. Additional perimeters can be pushed to the device. In some embodiments, the perimeters are created as encrypted perimeters with access to these perimeters controlled by a password. The encryption may be done, for example, when the perimeters contain data that is enterprise sensitive.

[0058] The encrypted file system perimeters are available only when an appropriate password that can be used to generate a key is entered and has been authenticated to unlock the file system underneath. Each individual perimeter requires separate authentication to be able to generate a key to unlock the file system corresponding to that perimeter. The generation of the key may utilize a password for either the device, the perimeters, or for a group of perimeters.

[0059] When the system comes out of a reboot, the file system is locked, and is considered to be in a state of “hard lock”. At this point, no data can be read from the file system. Once the system has started, and the user has authenticated to the system at least once, the perimeters will move between states of being “unlocked” to being “soft-locked”. When the perimeters are “soft locked”, none of the applications that provide

access to the perimeters can be accessed until the appropriate password is entered for the perimeter. The perimeters can be locked (soft-locked) by either the user, or because of inactivity timeouts. When a device is “soft-locked”, system services that have been previously authenticated by the system, that are already running, may still be able to access the perimeters in a similar manner to when the perimeter is “unlocked”. Such system services can provide aggregation services (like e-mail) or notification services, even when the perimeter is “soft-locked”.

[0060] Reference is now made to **Figure 4**. **Figure 4** shows an example of a soft lock process on the device. In particular, the process of **Figure 4** starts at block **410**, where the device is in a hard lock state, meaning that the data on the device is encrypted and inaccessible.

[0061] The process then proceeds to block **412** in which the device is unlocked. The unlocking of the device is typically performed by receiving an authentication parameter such as a password, which is then typically hashed and compared with a stored hash value on the device. However, other password verification or security verification is possible, including physical verification such as fingerprint, gesture, among others.

[0062] The process then proceeds to block **420** in which a check is made to determine whether a lock indication has been received on the device for either the device or a perimeter on the device. The lock indication can be an inactivity timeout expiring or a manual lock from a user interface, among other examples.

[0063] If a lock indication is not received at block **420**, the process continues to loop until a lock indication is received. Once the lock indication is received the process proceeds to block **430** in which the device, or subset of the device, is soft locked. As indicated above, applications that are running and have specific permissions may still access data and write data while the device is in a soft lock, while other applications are prevented from running or accessing data.

[0064] The process then proceeds to block **440** in which a check is made to determine whether the device or perimeter has been unlocked. The unlocking of the device is again typically based on a password entry on the device.

[0065] From block **440** the process proceeds to block **442** if the device or perimeter is not unlocked. At block **442** a check is made to determine whether any running application with privileges has made a data request. If yes the process proceeds to block **444** in which the data request is applied. The application of the data request at block **444** may include reading or writing to or from data locations on the device.

[0066] From block **444** the process loops back to block **440**. Similarly from block **442**, if no applications are making data requests the process loops to block **440**.

[0067] From block **440** once an unlock is received the process proceeds to block **420** and continues to loop.

[0068] As indicated above with regard to **Figures 2 and 3**, there can be multiple perimeters on the device. These may, for example include bridge or enterprise perimeters. Each of these can have a different policy for control.

[0069] For example, enterprise perimeters may have policies pushed to the device from a remote endpoint. The policies are managed and validated on the device, by the perimeter manager. The remote endpoint can send management commands to the device, updating policies for the password, or locking the device, or even changing/resetting the password, without user intervention. In some cases, policies can be consolidated to have a secure policy for a perimeter to provide overriding access control to data on the computing device. In other embodiments, each perimeter may be managed individually based on a policy for that perimeter. The device may be switched between the security settings dynamically.

[0070] Whenever policy changes are pushed to the device, the perimeter transitions to a “soft-locked” state, requiring the user authentication and validation of passwords against the password policy rules.

[0071] In a further example, a bridge perimeter may have policies managed on a mobile device, such as mobile device **305** from **Figure 3**, with password verification and authentication provided by the mobile device. All of the password policy, validation and authentication is performed on the mobile device, while the perimeter manager on the computing device controls the state of the perimeter (e.g. hard-lock, soft-lock, unlocked).

[0072] Individual perimeters may be controlled and managed individually by the perimeter manager. They can be managed on device or remotely.

[0073] In one embodiment, perimeters can time out and lock or be locked manually independent of each other. Enterprise policy pushed down to a device that is managed locally on the device by a remote endpoint, can also choose to enforce password policies on the device password also. When this is the case, enterprise policies for the password on the device may need to be adhered to. In this case, the password for the perimeter is synchronized with the device password. Unlocking the device also unlocks the perimeter.

[0074] When a change is made to a password on the perimeter on the device, the device password is also automatically changed, keeping the passwords synchronized. The enterprise may choose to change the policy that manages the device, and when these policies are pushed down to the perimeter, the perimeter transitions to a “soft” locked state that requires the user unlock the perimeter to access any perimeter resources.

[0075] When the policies apply to the device equally, the device and perimeter are locked at the same time. Anytime the policy changes, authentication is required based on an existing password with the new policy before access is granted to any perimeter resources. This requirement is enforced even across a reboot, until the password is validated at least once against the new policies.

[0076] When there are multiple perimeters deployed to the device, there can be cases when the managing enterprise for these perimeters is in fact the same. For example, an enterprise pushes the enterprise perimeter to the device, and the enterprise also manages the phone associated with the bridge perimeter. In this scenario, the enterprise and bridge may be “equivalent”. When this is in fact the case, any applications that are launched with access to the either one of these perimeters is provided all the privileges to access the other perimeter also. Since the two perimeters are managed by different passwords, the applications may still require authentication with the appropriate password to access resources in a specific perimeter. This will be required whenever the perimeter is “hard” or “soft” locked

[0077] Reference is now made to **Figure 5**. The process of **Figure 5** starts at block **510** and proceeds to block **512** in which a new perimeter policy is received by the policy manager on the device. The new perimeter policy can relate to an existing perimeter on the device or relate to a new perimeter for the device.

[0078] From block **512** the process proceeds to block **514** in which a soft lock is performed for the perimeter and/or the device. As used herein, a soft lock prevents access to data by applications in general, but permits certain applications which are already running and which have correct permissions specified to continue to read or write data. Examples of the certain applications with correct permissions may include system services or core applications, among others.

[0079] The process at block **514** may simply lock applications or data relating to a perimeter. In this case the device may continue to function for other perimeters. This

may be implemented, for example, when a new policy is provided for an existing perimeter. Conversely, the process at block **514** may lock the entire device requiring a password to be entered to unlock the device.

[0080] From block **514** the process proceeds to block **516** in which a prompt is provided to enter a password. The password may be for the device, perimeter or both.

[0081] Once the password has been successfully entered the process proceeds to block **518** in which the password is associated with the perimeter policy. The password may be used to generate a key to encrypt or decrypt data relating to the perimeter.

[0082] In accordance with the above the association of the password at block **518** may allow the device password to unlock the perimeter.

[0083] The process then proceeds from block **518** to block **520** and ends.

[0084] In certain embodiments of the present disclosure it may be desirable for a password and timeout period to be consolidated among the various perimeters on the device. Such password policies and time out policies for each perimeter are typically found within the perimeter policies.

[0085] The consolidation of the password and timeout period allows one password to be used among the plurality of perimeters while ensuring that the password meets the policy requirements for all of the perimeters. Similarly, one inactivity time out can be used for the device which conforms to the inactivity timeout policies of all of the perimeters. This consolidated inactivity time out would be such that it adheres to the strictest of all the individual inactivity time out policies of all the perimeters.

[0086] Specifically, reference is now made to **Figure 6**. The process of **Figure 6** starts at block **610** and proceeds to block **612** in which a new perimeter policy is received at

the device. The new perimeter policy may be associated with an existing perimeter on the device or may be for a new perimeter for the device.

[0087] From block **612**, the process proceeds to block **614** in which a determination is made regarding the need for a consolidated password on the device. Specifically, in some embodiments different perimeters can have different passwords. However in other embodiments a single password for the device may be used.

[0088] From block **614**, if a consolidated password is needed, the process proceeds to block **616** in which a check is made to determine whether a current device password meets the password policies within the perimeter policy. Specifically, the current device password may need to be a certain length, contain a variety of character types, or other factors may need to be present within the password.

[0089] In particular, various parameters of the policy may include password complexity, inactivity timeouts, policy governing the movement of data between perimeters, etc. This list is not exhaustive, and can include other policy parameters as well.

[0090] From block **616**, if the current device password does not meet the perimeter policy, the process proceeds to block **618** in which a new password for the device is set. The new password would need to meet the requirements under the consolidated perimeter policies.

[0091] From block **614**, if a consolidated password is not required, from block **616** if the current device password meets the perimeter policy, or from block **618**, the process proceeds to block **620** in which a check is made to determine whether a consolidated inactivity timeout is required. For example, a policy may require that the device lock after five minutes of inactivity. Another policy may require that the device lock after fifteen minutes of inactivity. In order to have a single inactivity timeout, a consolidated inactivity time period may be established on the device. Further, a user of the device can set an inactivity period that it is shorter than the consolidated value. In other

embodiments, only a subset of perimeters fall under the consolidated timeout. Thus, for example, all perimeters associated with an enterprise may have a timeout that is consolidated, but a personal perimeter might have a different or no timeout.

[0092] From block **620**, if a consolidated time out is required the process proceeds to block **622** and checks whether the current consolidated time out meets the perimeter policy time out. If no, the process proceeds to block **624** in which the consolidated time out value is changed to the value found in the current perimeter policy.

[0093] From block **620**, if no consolidated time out is required, or from block **622** if the current consolidated time out has a value that is lower than the perimeter policy timeout value, or from block **624**, the process proceeds to block **630** and ends.

[0094] After the password policies have been set and/or a consolidated timeout value established, if a user tries to change the password or inactivity time out period for the device a check is made to determine whether the new password or inactivity time out value falls within the consolidated policies.

[0095] One system architecture capable of being used with the above embodiments is shown with regard to **Figure 7**. The architecture of **Figure 7** is however not meant to be limiting and other system architectures are possible.

[0096] Reference is now made to **Figure 7**, which shows a block diagram of an example wireless data network in accordance with the present disclosure and with which the various embodiments of the methods of the instant disclosure may cooperate. **Figure 7** shows a block diagram of a mobile device **710** and example Code Division Multiple Access (CDMA) 1x network **720**, an example Evolution Data Only (EVDO) network **730**, a public switched telephone network (PSTN) **735**, a data network **740**, wireless gateway **742** and enterprise server **744**. This is shown merely as an example, and other network architectures, such as Global System for Mobile (GSM), GSM Packet Radio Service (GPRS), Universal Mobile Telecommunications Service (UMTS), Long Term Evolution

(LTE), LTE Advanced (LTE-A), High Speed Downlink Packet Access (HSDPA), Wi-Fi, WiMAX, among others, are possible.

[0097] The mobile device **710** may comprise a two-way communication device having data and voice communication capabilities. **Figure 7** further shows an access point **770** for use with an alternative data connection such as a Wi-Fi or WiMAX connection.

[0098] CDMA network **720** is comprised of a base transceiver station (BTS) **722** and a base station controller (BSC) **724**. Base station controller **724** communicates with a mobile switching center **726** which, as will be appreciated, is a circuit switched only component communicating with PSTN **735**. Base station controller **724** further communicates with a packet data serving node (PDSN) **728** which is a packet switched only component. PDSN **728** further communicates with IP network **740**.

[0099] EVDO network **730** contains an EVDO sector **732** which communicates with access node (AN) **734**. Since the EVDO network **730** is a data only network, access node **734** communicates only with PDSN **728** and not with any circuit switch components.

[00100] An authentication, authorization and accounting node **736** is associated with AN **734**, and a similar node **729** is associated with PDSN **728**.

[00101] Operationally, mobile device **710** communicates wirelessly with CDMA network **720** using BTS **722** and BSC **724** to gain access to the CDMA 1x network.

[00102] Mobile device **710** sends and receives both data and voice services through CDMA network **720** until an EVDO network connection with established, at which point data can be transmitted over the EVDO network connection.

[00103] Further, mobile device **710** can be connected to a computing device **754** such as a tablet for a variety of reasons, some of which are provided above. The connection may be through various means such as a Universal Serial Bus (USB) or other serial port, or by short range wireless communications with a computing device

754. Computing device **754** can then gain access to data network **740** and to enterprise server **744** through EVDO network **730** or CDMA network **720** using mobile device **710**. In other embodiments, computing device **754** may also be capable of accessing networks **720**, **730** or **770** directly.

[00104] Mobile device **710** may further have capabilities to communicate through access point **770** using, for example, Wi-Fi. Access point **770** connects to a data network **740** and thus access to wireless gateway **742** and enterprise server **744** are possible through access point **770**

[00105] In one embodiment, enterprise server **744** can provide both the IT policies for the mobile device **710** and also provide access to a permanent store of the corporate data which can be accessed by mobile device **710**.

[00106] The embodiment of **Figure 7** is merely an example and other network architectures are possible for mobile device **710** to connect to enterprise server **744**. The embodiment of **Figure 7** is not meant to be limiting to any particular network architecture.

[00107] Further, mobile device **710** may not be a dual mode or multi-mode device that allows connection to Wi-Fi. In this case, the Wi-Fi connection to access point **770** would be removed from the embodiment of **Figure 7** and all communication may proceed over the cellular network through the base station **722** or **732**. In other embodiments, mobile device **710** may only have access through an access point **770** and thus the cellular network would be removed from **Figure 7**. Other possibilities would be apparent to those skilled in the art having the benefit of the present disclosure.

[00108] Computing device **754**, may, in some embodiments, comprise a personal computing device. For example, computing device **754** may comprise a tablet computer. The user may further wish to use computing device **754** for corporate functions. However, for security reasons, the corporate IT department may not consider

the computing device **754** to be a secure destination for data, since it is a personal device.

[00109] The device of **Figure 1** can be a mobile device. One such example mobile device is illustrated below with reference to **Figure 8**. The mobile device of **Figure 8** is however not meant to be limiting and other mobile devices can also be used.

[00110] Mobile device **800** may comprise a two-way wireless communication device having any of voice communication capabilities, data communication capabilities, or both. Mobile device **800** generally has the capability to communicate with other devices or computer systems. Depending on the exact functionality provided, the mobile device may be referred to as a data messaging device, a two-way pager, a wireless e-mail device, a cellular telephone with data messaging capabilities, a wireless Internet appliance, a wireless device, a user equipment, a tablet, or a data communication device, as examples.

[00111] Where mobile device **800** is enabled for two-way communication, it may incorporate a communication subsystem **811**, including both a receiver **812** and a transmitter **814**, as well as associated components such as one or more antenna elements **816** and **818**, local oscillators (LOs) **813**, and a processing module such as a digital signal processor (DSP) **820**. As will be apparent to those skilled in the field of communications, the particular design of the communication subsystem **811** will be dependent upon the communication network in which the device is intended to operate.

[00112] Network access requirements will also vary depending upon the type of network **819**. In some networks, network access is associated with a subscriber or user of mobile device **800**. A mobile device may require a removable user identity module (RUIM) or a subscriber identity module (SIM) card in order to operate on the network. The SIM/RUIM interface **844** may be similar to a card-slot into which a SIM/RUIM card can be inserted and ejected like a diskette or PCMCIA card. The SIM/RUIM card can

have memory and hold many key configuration **851**, and other information **853** such as identification, and subscriber related information.

[00113] When required network registration or activation procedures have been completed, mobile device **800** may send and receive communication signals over the network **819**. As illustrated in **Figure 8**, network **819** can consist of multiple base stations communicating with the mobile device. For example, in a hybrid CDMA 1x EVDO system, a CDMA base station and an EVDO base station communicate with the mobile station and the mobile device is connected to both simultaneously. In other systems such as Long Term Evolution (LTE) or Long Term Evolution Advanced (LTE-A), multiple base stations may be connected to for increased data throughput. Other systems such as GSM, GPRS, UMTS, HSDPA, among others are possible and the present disclosure is not limited to any particular cellular technology.

[00114] Signals received by antenna **816** through communication network **819** are input to receiver **812**, which may perform such common receiver functions as signal amplification, frequency down conversion, filtering, channel selection and the like, and in the example system shown in **Figure 8**, analog to digital (A/D) conversion. A/D conversion of a received signal allows more complex communication functions such as demodulation and decoding to be performed in the DSP **820**. In a similar manner, signals to be transmitted are processed, including modulation and encoding for example, by DSP **820** and input to transmitter **814** for digital to analog conversion, frequency up conversion, filtering, amplification and transmission over the communication network **819** via antenna **818**. DSP **820** not only processes communication signals, but also provides for receiver and transmitter control. For example, the gains applied to communication signals in receiver **812** and transmitter **814** may be adaptively controlled through automatic gain control algorithms implemented in DSP **820**.

[00115] Mobile device **800** generally includes a processor **838** which controls the overall operation of the device. Communication functions, including data and voice communications, are performed through communication subsystem **811**. Processor

838 also interacts with further device subsystems such as the display **822**, flash memory **824**, random access memory (RAM) **826**, auxiliary input/output (I/O) subsystems **828**, serial port **830**, one or more keyboards or keypads **832**, speaker **834**, microphone **836**, other communication subsystem **840** such as a short-range communications subsystem and any other device subsystems generally designated as **842**. Serial port **830** can include a USB port or other port known to those in the art having the benefit of the present disclosure.

[00116] Some of the subsystems shown in **Figure 8** perform communication-related functions, whereas other subsystems may provide “resident” or on-device functions. Notably, some subsystems, such as keyboard **832** and display **822**, for example, may be used for both communication-related functions, such as entering a text message for transmission over a communication network, and device-resident functions such as a calculator or task list, among other applications.

[00117] Operating system software used by the processor **838** may be stored in a persistent store such as flash memory **824**, which may instead be a read-only memory (ROM) or similar storage element (not shown). Those skilled in the art will appreciate that the operating system, specific device applications, or parts thereof, may be temporarily loaded into a volatile memory such as RAM **826**. Received communication signals may also be stored in RAM **826**.

[00118] As shown, flash memory **824** can be segregated into different areas for both computer programs **858** and program data storage **850**, **852**, **854** and **856**. These different storage types indicate that each program can allocate a portion of flash memory **824** for their own data storage requirements. The applications may be segregated based on the mode or category they fall into. Memory **824** may further provide security for corporate data and if some applications are locked while others are not.

[00119] Processor **838**, in addition to its operating system functions, may enable execution of software applications on the mobile device. A predetermined set of

applications that control basic operations, including data or voice communication applications for example, as well as a predetermined set of certificates, will normally be installed on mobile device **800** during manufacturing. Other applications can be installed subsequently or dynamically.

[00120] Applications and software, such as those described above may be stored on any computer readable storage medium. The computer readable storage medium may be a tangible or intransitory/non-transitory medium such as optical (e.g., CD, DVD, etc.), magnetic (e.g., tape) or other memory known in the art.

[00121] One example software application may be a personal information manager (PIM) application having the ability to organize and manage data items relating to the user of the mobile device such as, but not limited to, e-mail, calendar events, voice mails, appointments, and task items. Further applications, including, but not limited to, a media player, camera, messenger, mail, calendar, address book, web browser, social networking, game, electronic book reader, map, or other application may also be loaded onto the mobile device **800** through the network **819**, an auxiliary I/O subsystem **828**, serial port **830**, short-range communications subsystem **840** or any other suitable subsystem **842**, and installed by a user in the RAM **826** or a non-volatile store (not shown) for execution by the processor **838**. Such flexibility in application installation increases the functionality of the device and may provide enhanced on-device functions, communication-related functions, or both. For example, secure communication applications may enable electronic commerce functions and other such financial transactions to be performed using the mobile device **800**.

[00122] In a data communication mode, a received signal such as a text message or web page download will be processed by the communication subsystem **811** and input to the processor **838**, which may further process the received signal for output to the display **822**, or alternatively to an auxiliary I/O device **828**.

[00123] A user of mobile device **800** may also compose data items such as email messages for example, using a keyboard **832**, which may comprise a virtual or physical

keyboard or both, and may include a complete alphanumeric keyboard or telephone-type keypad, among others, in conjunction with the display **822** and possibly an auxiliary I/O device **828**. Such composed items may then be transmitted over a communication network through the communication subsystem **811**.

[00124] For voice communications, overall operation of mobile device **800** is similar, except that received signals would typically be output to one or more speakers **834** and signals for transmission would be generated by a microphone **836**. Alternative voice or audio I/O subsystems, such as a voice message recording subsystem, may also be implemented on mobile device **800**. Although voice or audio signal output may be accomplished primarily through the one or more speakers **834**, display **822** may also be used to provide an indication of the identity of a calling party, the duration of a voice call, or other voice call related information for example.

[00125] Serial port **830** in **Figure 8** would normally be implemented in a personal digital assistant (PDA)-type mobile device for which synchronization with a user's desktop computer (not shown) may be desirable, but is an optional device component. Such a port **830** would enable a user to set preferences through an external device or software application and would extend the capabilities of mobile device **800** by providing for information or software downloads to mobile device **800** other than through a wireless communication network. The alternate download path may for example be used to load an encryption key onto the device through a direct and thus reliable and trusted connection to thereby enable secure device communication. As will be appreciated by those skilled in the art, serial port **830** can further be used to connect the mobile device to a computer to act as a modem.

[00126] Other communications subsystems **840**, such as a short-range communications subsystem, are further optional components which may provide for communication between mobile device **800** and different systems or devices, which need not necessarily be similar devices. For example, the subsystem **840** may include an infrared device and associated circuits and components, near field communications

(NFC) or a Bluetooth™ communication module to provide for communication with similarly enabled systems and devices.

[00127] The embodiments described herein are examples of structures, systems or methods having elements corresponding to elements of the techniques of this application. This written description may enable those skilled in the art to make and use embodiments having alternative elements that likewise correspond to the elements of the techniques of this application. The intended scope of the techniques of this application thus includes other structures, systems or methods that do not differ from the techniques of this application as described herein, and further includes other structures, systems or methods with insubstantial differences from the techniques of this application as described herein.

CLAIMS

1. A method for managing grouped resources on a computing device comprising:

receiving, at the computing device, a policy for a set of grouped resources;

applying the policy;

locking at least one of the computing device or the set or grouped resources associated with the policy;

waiting for receipt of an authentication parameter at the computing device;

verifying the authentication parameter;

associating the set of grouped resources with the authentication parameter; and

unlocking the least one of the computing device or the set of grouped resources.
2. The method of claim 1, wherein each set of grouped resources on the device has a policy associated therewith.
3. The method of claim 2, wherein a plurality of policies is managed at a manager on the computing device.
4. The method of claim 1, wherein the authentication parameter is a password.
5. The method of claim 1, wherein the locking prevents new applications from being launched on the computing device.

6. The method of claim 5, wherein the locking allows applications already running and having access privileges to access data on the device.
7. The method of claim 1, wherein the associating generates a key based on authentication parameter.
8. The method of claim 7, wherein the key is different for different sets of grouped resources.
9. The method of claim 1, wherein the associating further consolidates policies for a plurality of sets of grouped resources.
10. The method of claim 9, further comprising checking whether a current password on the computing device complies with the consolidated policies.
11. The method of claim 10, further comprising initiating a password change if the current password on the computing device does not comply with the consolidated policies.
12. The method of claim 9, further comprising checking whether an inactivity timer value on the computing device complies with the consolidated policies.

13. The method of claim 12, further comprising initiating an inactivity timer value change if the inactivity timer value on the computing device does not comply with the consolidated policies.
14. The method of claim 1, wherein the verifying is done on the computing device.
15. The method of claim 1, wherein the verifying is done on a remote device or server.
16. A computing device capable of managing grouped resources, the computing device comprising:
- a processor; and
 - a communications subsystem,
- wherein the computing device is capable of:
- receiving, at the computing device, a policy for a set of grouped resources;
 - applying the policy;
 - locking at least one of the computing device or the set or grouped resources associated with the policy;
 - waiting for receipt of an authentication parameter at the computing device;
 - verifying the authentication parameter;
 - associating the set of grouped resources with the authentication parameter; and
 - unlocking the least one of the computing device or the set of grouped resources.

17. The computing device of claim 16, wherein each set of grouped resources on the device has a policy associated therewith.
18. The computing device of claim 17, wherein a plurality of policies is managed at a manager on the computing device.
19. The computing device of claim 16, wherein the authentication parameter is a password.
20. The computing device of claim 16, wherein the locking prevents new applications from being launched on the computing device.
21. The computing device of claim 20, wherein the locking allows applications already running and having access privileges to access data on the device.
22. The computing device of claim 16, wherein the associating generates a key based on authentication parameter.
23. The computing device of claim 22, wherein the key is different for different sets of grouped resources.
24. The computing device of claim 16, wherein the associating further consolidates policies for a plurality of sets of grouped resources.

25. The computing device of claim 24, further configured to check whether a current password on the computing device complies with the consolidated policies.

26. The computing device of claim 25, further configured to initiate a password change if the current password on the computing device does not comply with the consolidated policies.

27. The computing device of claim 24, further configured to check whether an inactivity timer value on the computing device complies with the consolidated policies.

28. The computing device of claim 27, further configured to initiate an inactivity timer value change if the inactivity timer value on the computing device does not comply with the consolidated policies.

29. The computing device of claim 16, wherein the computing device is one of a mobile device, a tablet computer, a laptop computer or a desktop computer.

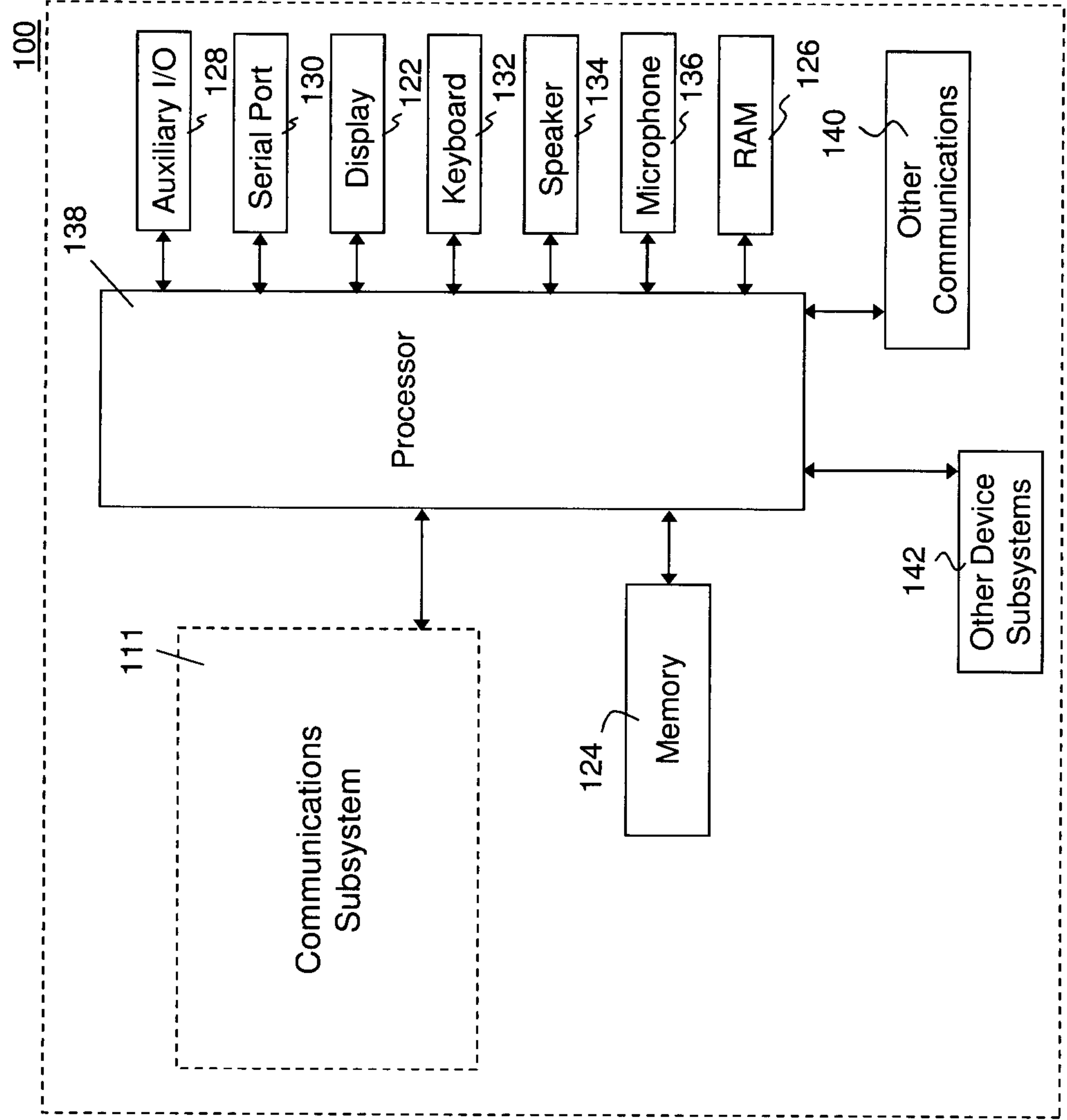


FIG. 1

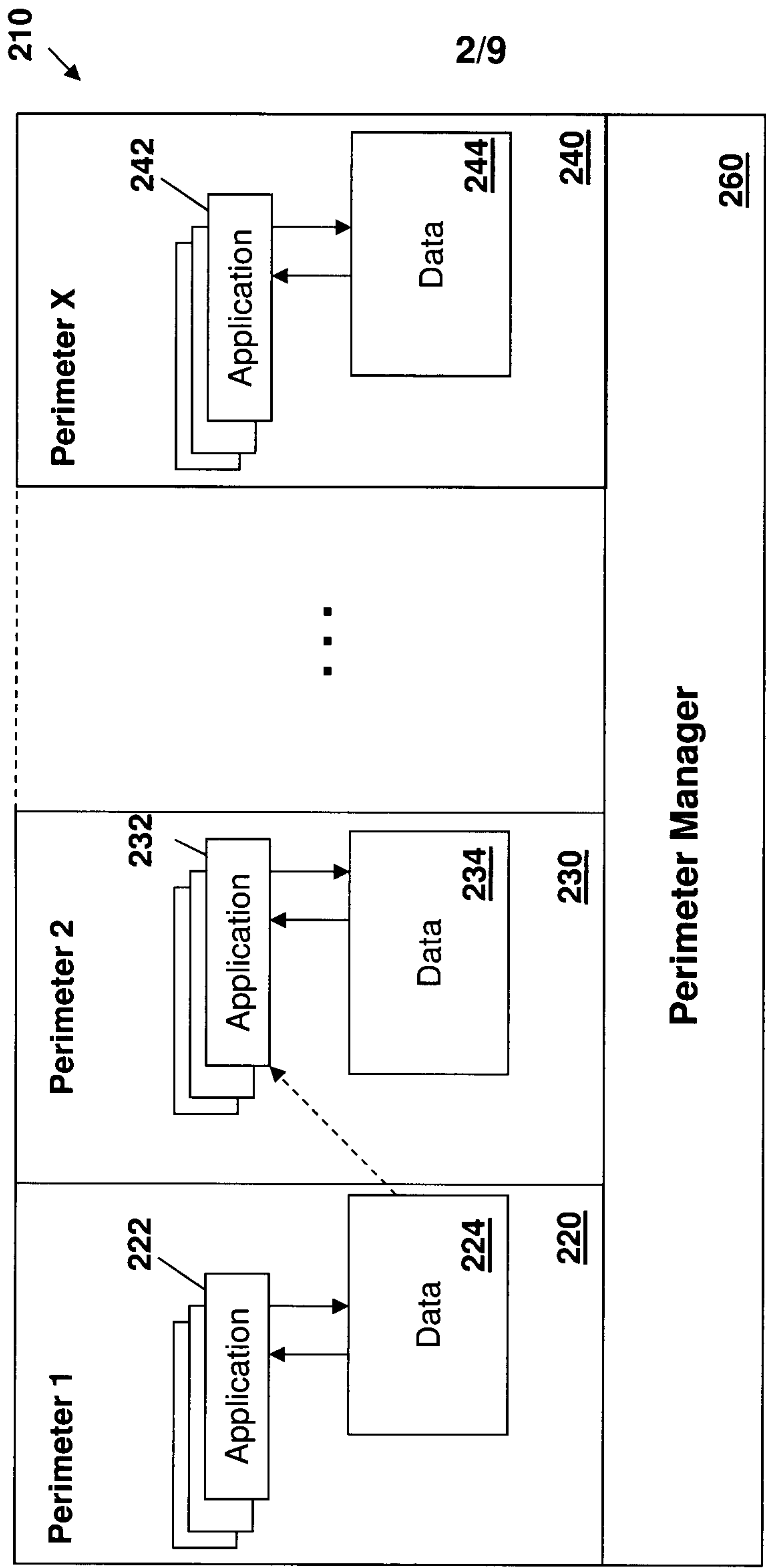


FIG. 2

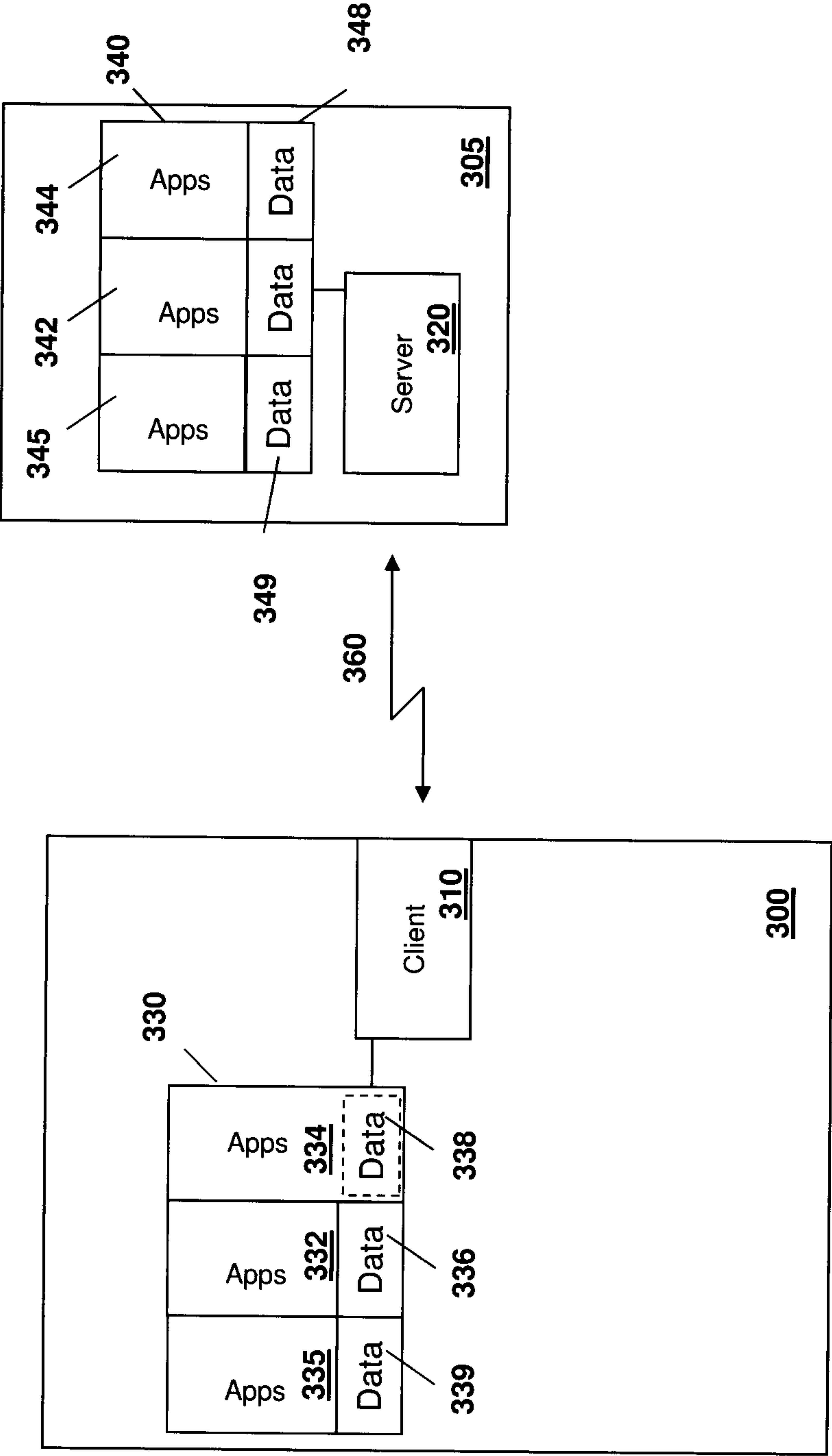


FIG. 3

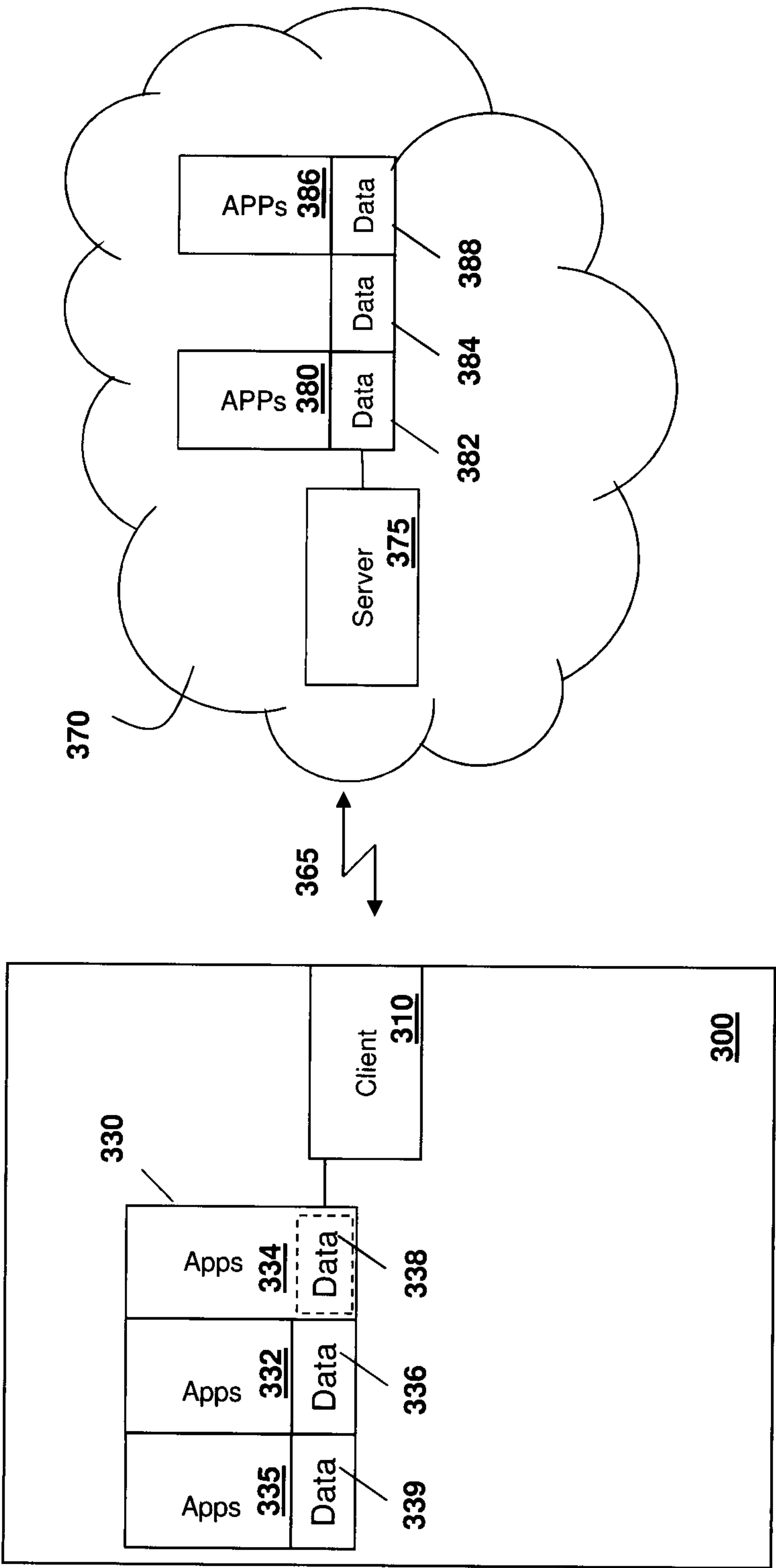


FIG. 3B

5/9

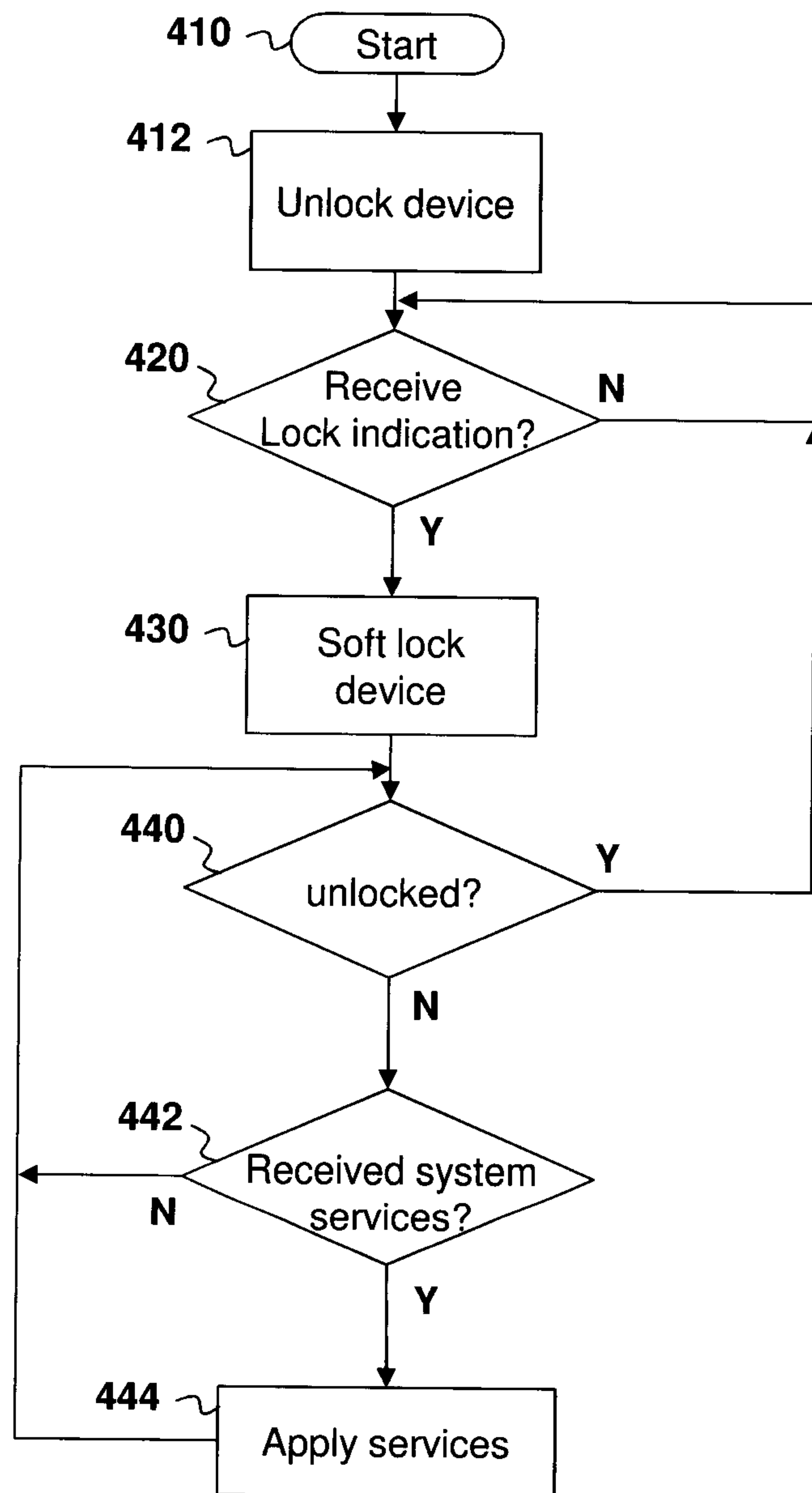
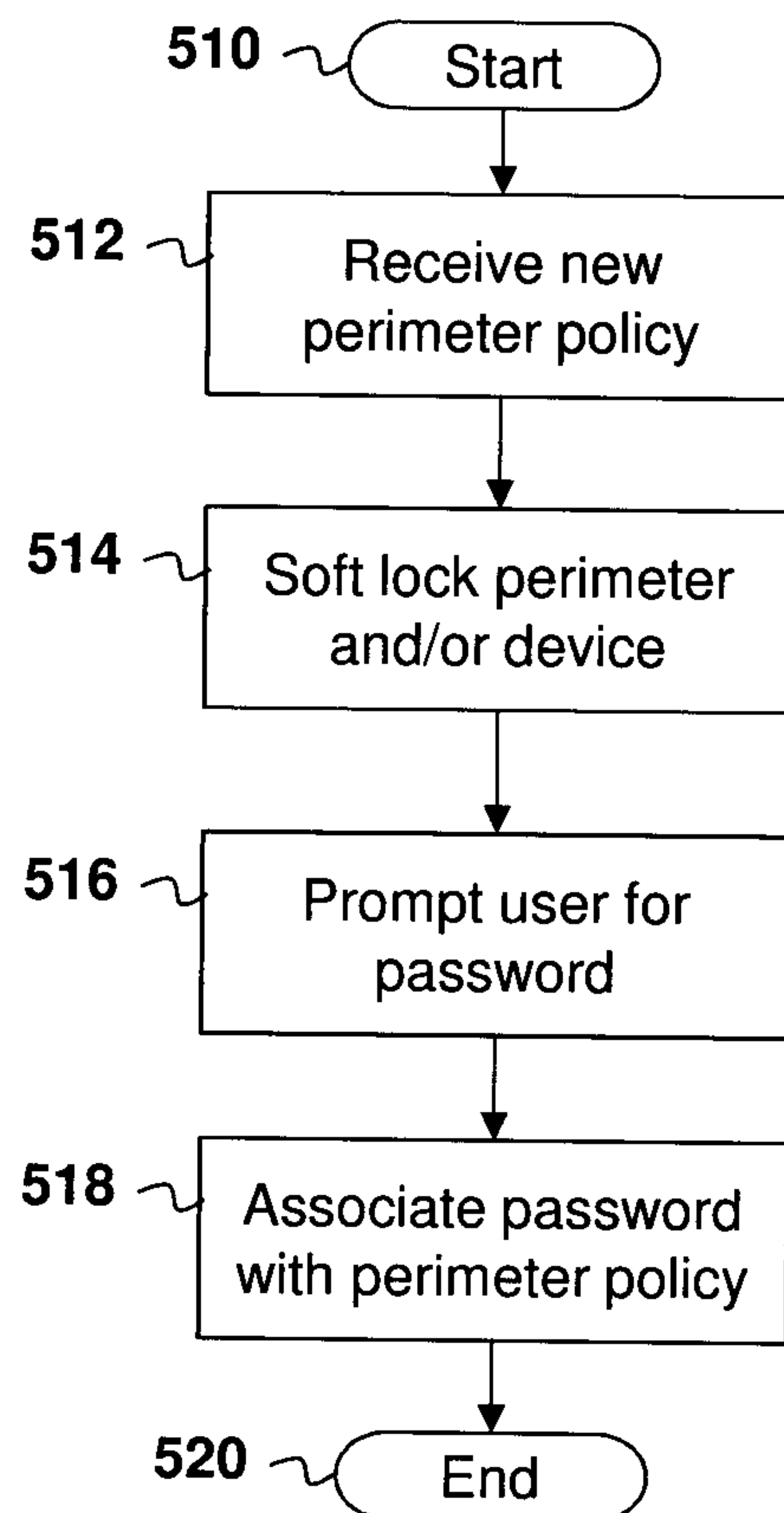


FIG. 4

6/9

**FIG. 5**

7/9

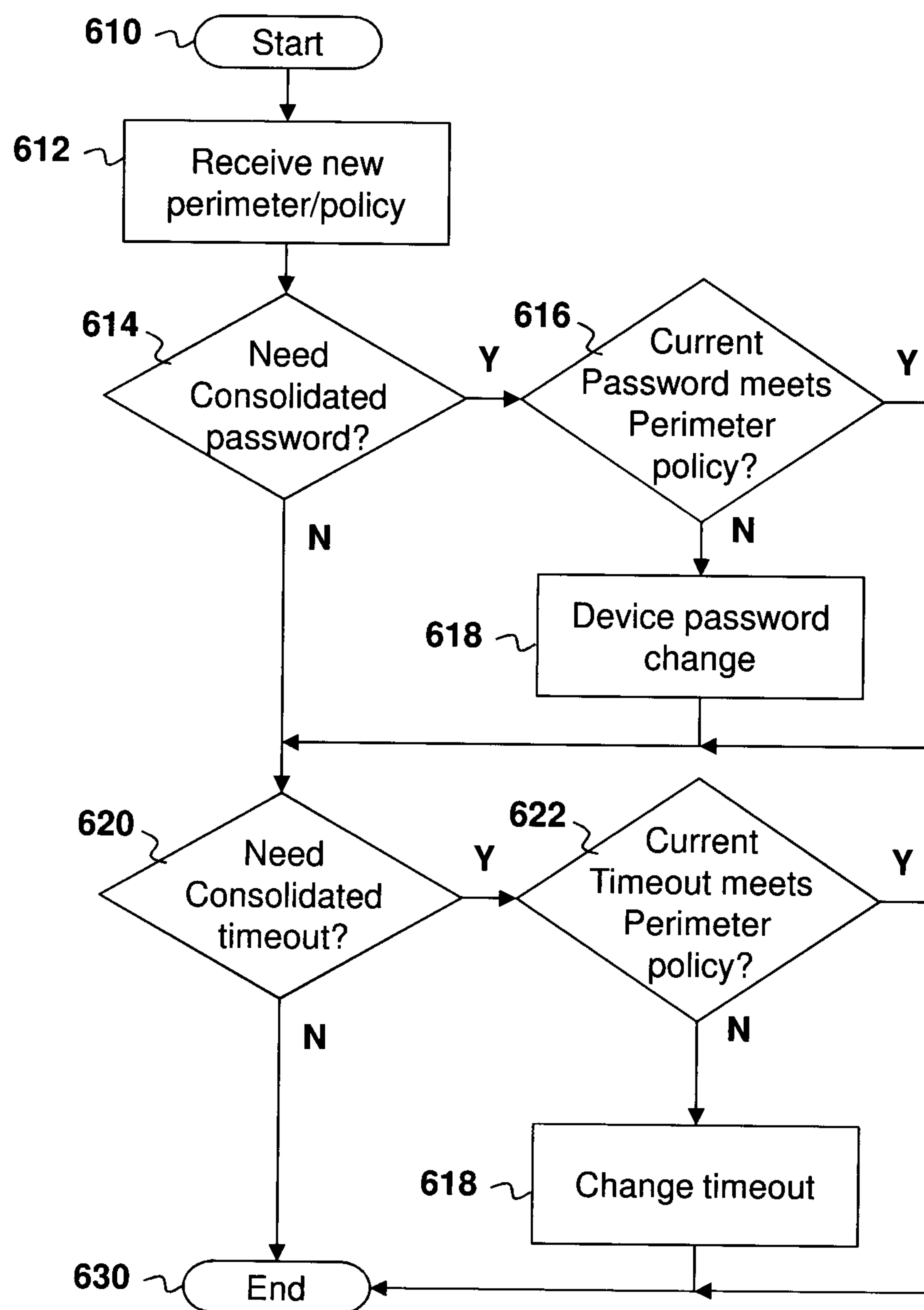


FIG. 6

8/9

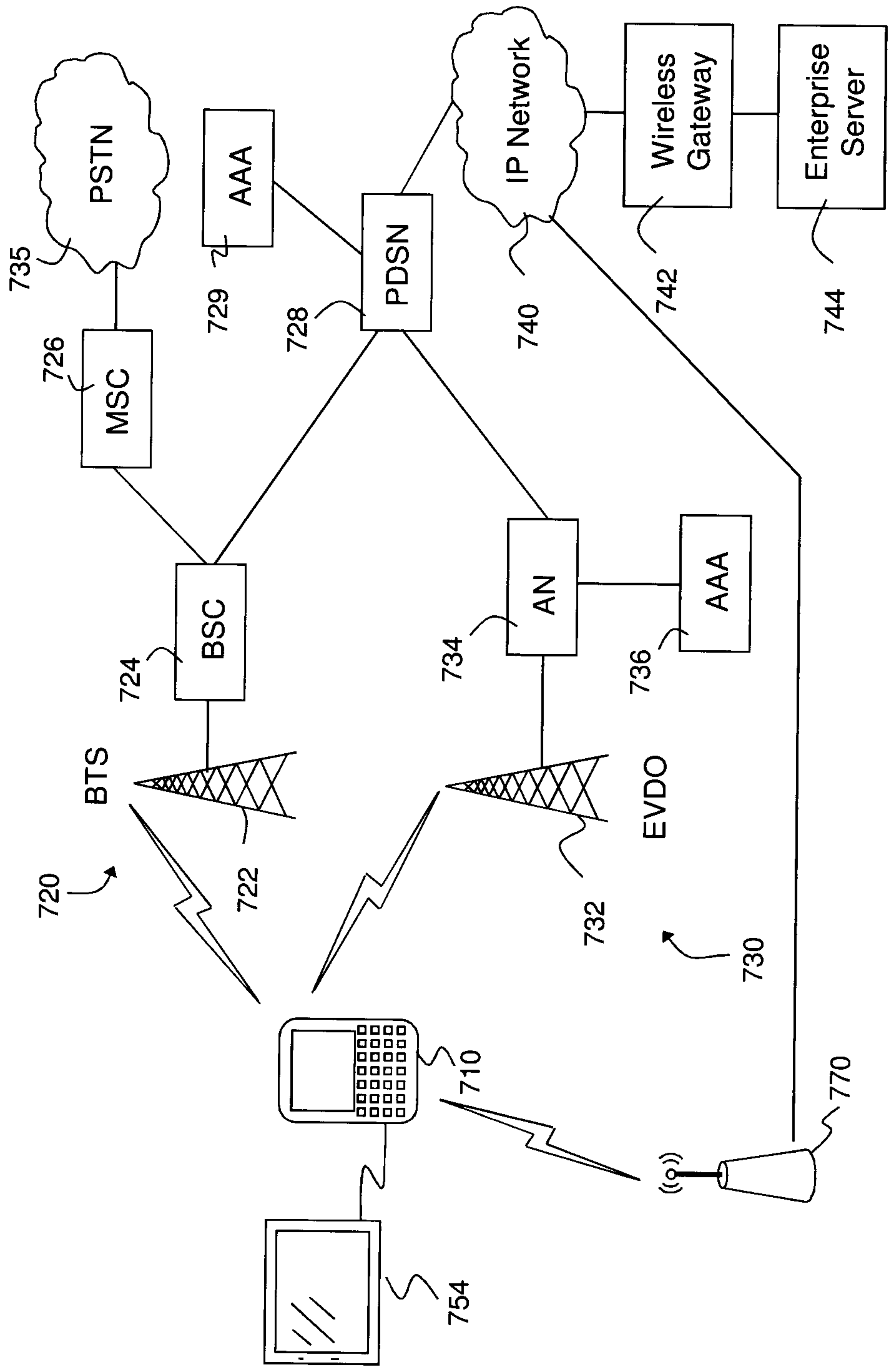


FIG. 7

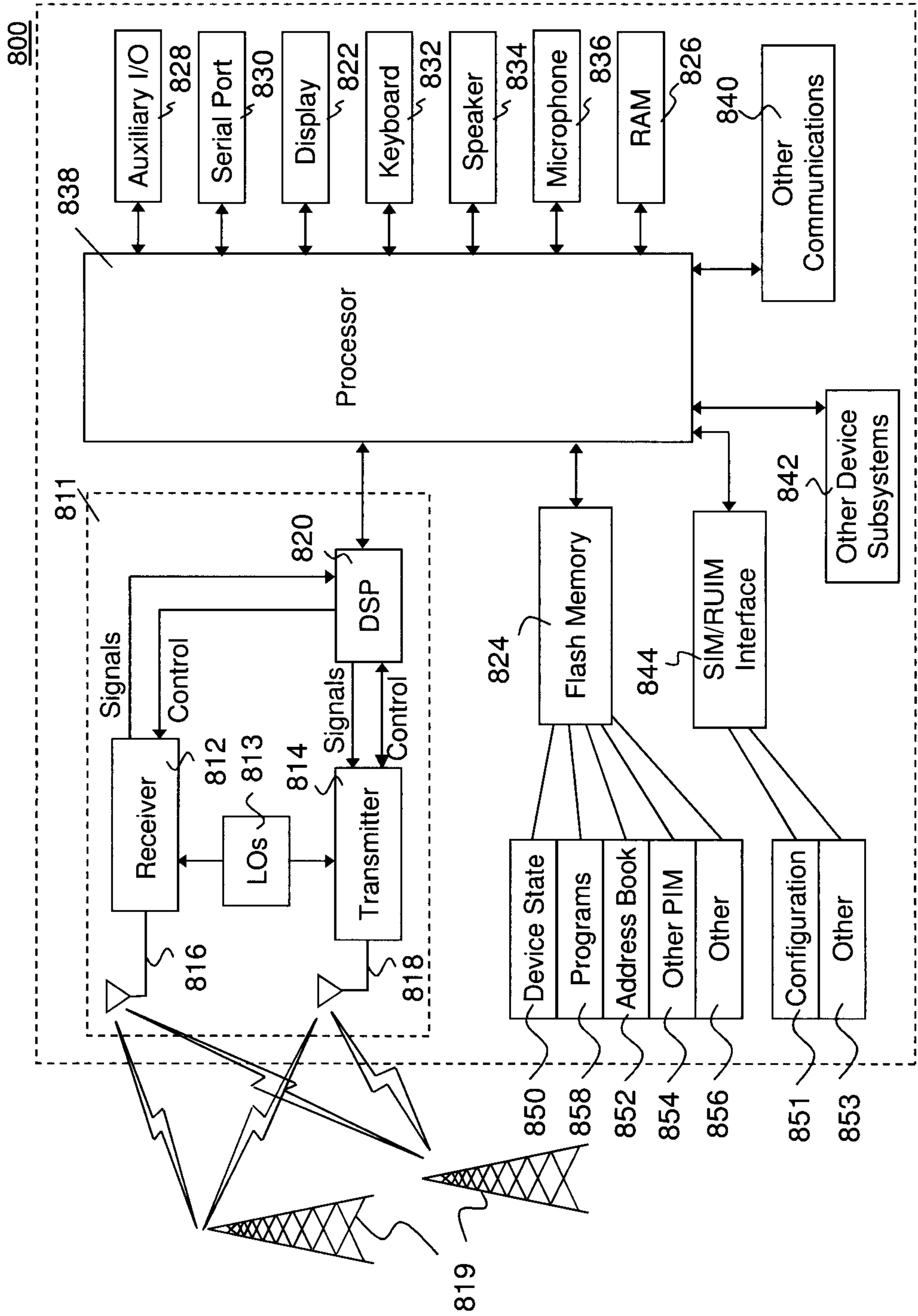


FIG. 8

