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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
20 November 2014 (20.11.2014)

WIPO | PCT

(10) International Publication Number
WO 2014/186331 A1

(51) International Patent Classification:
*G06F 17/30* (2006.01)

(21) International Application Number:
PCT/US2014/037791

(22) International Filing Date:
13 May 2014 (13.05.2014)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
13/893,841 14 May 2013 (14.05.2013) US

(71) Applicant: **GOOGLE INC.** [US/US]; 1600 Amphitheatre Parkway, Mountain View, CA 94043 (US).

(72) Inventors: **BUZBEE, Thomas, P.**; 1600 Amphitheatre Parkway, Mountain View, CA 94043 (US). **PROCOPIO, Michael, J.**; 1600 Amphitheatre Parkway, Mountain View, CA 94043 (US). **SCHNEIDER, Ronald, E.**; 1600 Amphitheatre Parkway, Mountain View, CA 94043 (US).

(74) Agents: **INGERMAN, Jeffrey, H.** et al.; Ropes & Gray LLP, 1211 Avenue of the Americas, New York, NY 10036 (US).

(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, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, 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, SD, SL, 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))*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

(54) Title: SYSTEMS AND METHODS FOR PROVIDING THIRD-PARTY APPLICATION SPECIFIC STORAGE IN A CLOUD-BASED STORAGE SYSTEM

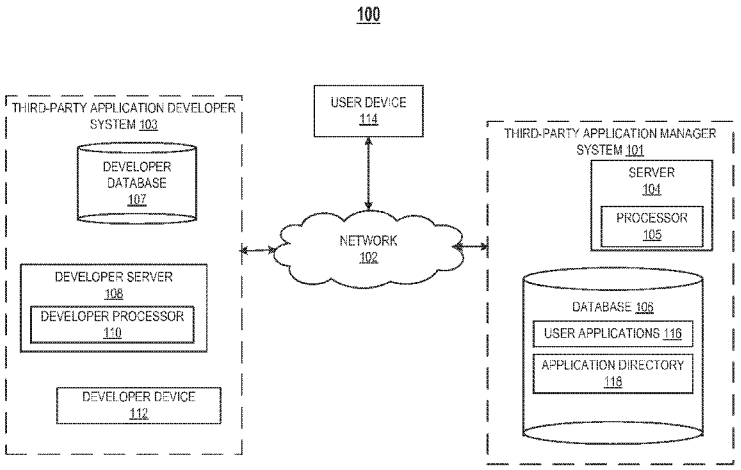


FIG. 1

(57) Abstract: Systems and methods for providing a dedicated storage directory for third-party applications in a cloud storage system may include allocating a user specific and application specific directory for the third-party application and controlling access to the directory. The directory may be used to store application data associated with the third-party application. The directory is made available to the respective third-party application and may be hidden from view from other applications and users.

WO 2014/186331 A1

# SYSTEMS AND METHODS FOR PROVIDING THIRD-PARTY APPLICATION SPECIFIC STORAGE IN A CLOUD-BASED STORAGE SYSTEM

## Technical Field

**[0001]** In general, this disclosure relates to cloud storage systems, in particular, to systems and methods for providing application specific file system directories for third-party applications.

## Background

**[0002]** Cloud storage systems provide users with the ability to store electronic documents and other files on a remote network rather than on the user's local computer or computing device. Such systems allow users to access the remotely stored files from any device that is capable of connecting with the remote network, for example using a web browser over an Internet connection. Cloud storage systems can also provide users with a large amount of memory to store files so that users are not limited by the memory capacity of a particular client or device. To access a cloud storage system, a user typically logs into an account on the cloud storage system using a username and password. The cloud storage system provides a user interface for users to view, edit, and manage files stored on the system.

**[0003]** Third-party applications may be used with cloud storage systems to provide applications and services for user's files on the cloud storage system. Third-party applications may include their own proprietary file formats that may limit access to the specific applications or interfaces by the cloud storage system and other third-party applications. In order for the third-party application to integrate with the cloud storage system, certain application information may be exchanged with the cloud storage system. Handling such application specific information in a streamlined fashion is desirable for efficient integration of third-party applications with cloud storage systems.

## Summary

**[0004]** Thus a need exists in the art to provide a unified system for a third-party application to integrate with a cloud storage system. Systems and methods described herein provide dedicated data storage for a third-party application that is integrated in a cloud storage system to provide access to third-party application supported files.

**[0005]** In one aspect, systems and methods provide dedicated storage for a third-party application in a cloud storage system, by receiving, using a processor, a file for storage in a cloud

storage system, wherein the file is associated with a third-party application and a user of the cloud storage system, identifying the third-party application, automatically allocating a directory for the identified third-party application and the user; and providing access to the directory for the identified third-party application to the third-party application. The directory is configured to store application data for the third-party application. The directory may be associated with a user and also hidden from the user. The directory may be assigned a reference link based on application data for the third-party application and a user identifier. Files for other users and other third-party applications may have respective directories allocated for use by the respective other third-party applications. Access to a file in the directory is authenticated for the requesting third-party application and respective user. In an aspect, if the third-party application is no longer available to the user, the user may be prompted to delete contents of the directory. In another aspect, the user may be presented with information for the directory and be provided an option to modify contents of the directory.

**[0006]** In another aspect, systems and methods provide support for a third-party application in a cloud storage system, by allocating a directory for a third-party application, associating the directory with a user of the cloud storage system, receiving configuration data for the third-party application, storing the configuration data in the directory; and hiding the directory from view by the user.

Brief Description of the Drawings

**[0007]** The methods and systems may be better understood from the following illustrative description with reference to the following drawings in which:

**[0008]** FIG. 1 is a block diagram of a computerized system for providing application-specific directories for third-party applications in accordance with an implementation described herein;

**[0009]** FIGS. 2A and 2B show a storage hierarchy for user files in a cloud storage system in accordance with an implementation as described herein;

**[0010]** FIG. 3 shows a user view of user files in a cloud storage system in accordance with an implementation as described herein;

**[0011]** FIG. 4 shows a method for allocating a third-party application directory in accordance with an implementation as described herein;

[0012] FIG. 5 shows a method for providing access to a third-party application directory in accordance with an implementation as described herein;

[0013] FIG. 6 shows a method for providing a user view of user files in a cloud storage system in accordance with an implementation as described herein; and

[0014] FIG. 7 is a block diagram of a computing device for performing any of the processes described herein, in accordance with an illustrative implementation.

Detailed Description

[0015] To provide an overall understanding of the systems and methods described herein, certain illustrative embodiments will now be described, including systems and methods for a cloud storage system capable of allocating an application specific directory for a third-party application. However, it will be understood by one of ordinary skill in the art that the systems and methods described herein may be adapted and modified as is appropriate for the application being addressed and that the systems and methods described herein may be employed in other suitable applications, and that such other additions and modifications will not depart from the scope thereof. In particular, a server, processor, or system as used herein may be a single computing device or multiple computing devices working collectively and in which the storage of data and the execution of functions are spread out amongst the various computing devices.

[0016] Aspects of the systems and methods described herein provide an application manager, for example, in a cloud storage system, that is capable of providing and allocating an application specific file system directory for a third-party application. Cloud storage systems may be used to store electronic document or files. Those cloud stored files are accessible from the cloud storage via a network from any client or computing device. As such, a user of a cloud storage system is not limited by storage capacity of a local device, or the availability of the local device. Different kinds of third-party applications may be used for the files that the user wishes to store on the cloud storage system. Since most applications have their own proprietary formats, integration of the third-party applications in another environment can require different application-specific solutions.

[0017] One way to improve efficient handling of third-party application files on a cloud storage system is to provide a third-party application file system directory on the cloud storage system that is accessible and used only by the third-party application. The third-party

application directory may be allocated for each user of a third-party application on the cloud storage system. The third-party application directory may be used to store data relevant to the third-party application or originated by the third-party application, for example, application configuration data, default file types, user preferences, file state information, temporary files, file logs, or other data for the third-party application. In some implementations, such data is preferably protected from access by other applications, and also from the user. For example, if the user sees a data directory for the third-party application and modifies it, the contents of the directory may be unusable by the third-party application. Accordingly, the third-party application directory may be made hidden or un-modifiable to the user.

**[0018]** The third-party application directory may be provided on a cloud storage system that is accessible by any number of client computers and other electronic devices that are capable of connecting with remote networks, for example the Internet. In some implementations, the third-party application may be locally installed on a client computer or other device. In other implementations, the cloud storage system may be connected to the third-party application server or otherwise make the third-party application available to a cloud storage system user. Some examples of third-party applications or services include word processing applications, photo and video storing and sharing services, spreadsheet applications, graphic design services, presentation applications, database applications, gaming and entertainment services, music services, e-mail services, cloud storage services, and a variety of other categories.

**[0019]** FIG. 1 depicts an example of a network and database structure that may be used to implement the systems and methods herein. FIG. 1 is a block diagram of a computerized system 100 for providing application specific directories. The system 100 includes a third-party application manager system 101, a user device 114, and a third-party application developer system 103. The third-party application manager system 101 includes a server 104 including a processor 105 and a database 106 that stores data, including a user's web data, and data related to a list of user applications 116, and application directory 118. As used herein, the term "processor" or "computing device" refers to one or more computers, microprocessors, logic devices, servers, or other devices configured with hardware, firmware, and software to carry out one or more of the computerized techniques described herein. Processors and processing devices may also include one or more memory devices for storing inputs, outputs, and data that are currently being processed. An illustrative computing device 700, which may be used to

implement any of the processors and servers described herein, is described in detail with reference to FIG. 7. As used herein, "developer device" and "user device" include, without limitation, any suitable combination of one or more input devices (e.g., keypads, touch screens, trackballs, voice recognition systems, etc.) and/or one or more output devices (e.g., visual displays, speakers, tactile displays, printing devices, etc.). As used herein, "server" includes, without limitation, any suitable combination of one or more devices configured with hardware, firmware, and software to carry out one or more of the computerized techniques described herein. Only one third-party application manager system 101, one user device 114, and one third-party application developer system 103 are shown in FIG. 1 to avoid complicating the drawing. However, in general, the system 100 can support multiple third-party application manager systems, third-party developer systems, servers, databases, developer devices, and user devices.

**[0020]** The third-party application manager system 101 allows for third-party application developers to associate third-party application data directories 118 with a user. In particular, the third-party application developer may have certain default data to store in a third-party application data directory that may be used to integrate use of a third-party application file in another environment, for example, a cloud storage environment.

**[0021]** In one implementation, the third-party application manager system 101 is a part of a cloud storage system, which is a file hosting system that allows users to store, retrieve, and modify data via one or more user devices such as the user device 114. This data may be referred to as a user's web data. As an example, the cloud storage system 101 may be stored on a single server system or in a distributed system. In particular, the cloud storage system 101 may use cloud storage to store user data.

**[0022]** The database 106 stores several types of data. In particular, the database 106 stores data associated with user applications 116 and application directories 118. The user applications 116 may be stored as a list of applications associated with a user. For example, the set of user applications 116 may correspond to a list of all applications that the user has ever used to interact with a file, or with which the user has selected to be associated. As an example, when the third-party application manager system 101 is a part of a cloud storage system, the user's web data includes a set of files with various application formats. In this case, the list of user applications 116 may correspond to a set of applications that the user has used to access any file in the user's

web data, or a set of applications that are configured to support any file within the user's web data. In general, the list of user applications 116 may be associated with the user, or a set of files associated with the user.

**[0023]** In addition, the database 106 also stores an application directory 118 that is associated with a particular user as well as the user application 116. The application directory 118 may be part of a data object family associated with the respective user application 116 and/or user. In some implementations, any user files related to the third-party application may also be included in the data object family. An alias associated with the third-party application may be assigned to the data object family and used for identification and authentication purposes as well as in a reference link for mapping the file system directory.

**[0024]** The third-party application developer system 103 communicates with the third-party application manager system 101 over the network 102 to initiate use of the third-party application. As depicted in FIG. 1, the developer system 103 includes a developer database 107, a developer server 108, a developer processor 110, and a developer device 112. The developer database 107 may store third-party application files or data. In an example, the developer system 103 is associated with a particular third-party application, and the developer system 103 is configured to transmit a request to the third-party application manager system 101 to allocate an application specific directory. If the third-party application manager system 101 grants the request, the database 106 is updated with the particular third-party application a respective third-party application directory.

**[0025]** The third-party application directory may be allocated per-user and per-application. In some implementations the third-party application directory 118 may not be visible in a user interface for accessing user files, for example, in the cloud storage system. In other implementations, the third-party application directory 118 may be visible to the user, but the contents of the directory may not be modifiable. To the third-party application developer system 103, however, the application directory 118 is visible and available as any other directory. The application directory 118 may be made available using existing API methods using, for example, a standard identifier. The third-party application manager system 101 may receive the identifier from the developer system 103 to encode respective user and application identifiers. The third-party application directory 118 is generally created during the first use of the third-party application by the user. The third-party application directory 118 and its contents may be

assigned certain authorization fields that are used to ensure that any requesting or accessing application is authorized to access the directory. Typically, only one third-party application is able to access its own application specific directory. The authorization fields may also be copied from a parent directory for the application to every directory and file that is created using the third-party application.

**[0026]** One example of a file manager interface is shown in FIG. 2A. As shown, a user may have certain files stored on a cloud storage system, for example my files 202. The user may have one or more third-party applications installed or available to the user, for example a PhotoApp 206 and a WritingApp 208. Each of the applications may have an application directory, for example, PhotoAppDirectory 207. The user's third-party application directories may include any third-party application data, for example temporary files, log data, configuration data, or other application-originated or application-specific data. For example, a writing application directory may include user preferences for document margins, views or fonts. A photography application directory may include user preferences for filter types. In a games application directory, game state information may be stored. The view shown in FIG. 2A may be one that is visible by the third-party application. In the example of FIG. 2A, the view is for the PhotoApp application, which shows the PhotoAppDirectory 207. A corresponding WritingAppDirectory would exist, however, it would not be viewable by another third-party application. User files may also be shown in any hierarchy, minimized or maximized, and include metadata for the file. In general, any file associated with a particular application will have an authorization field that is propagated from the parent or top-level directory.

**[0027]** An example of a user's view of their files is shown in FIG. 2B. As shown, a user may have files, such as my files 703 and include a list of each of their respective files. The view shown in FIG. 2B is simply illustrative of a simple file manager user interface. As shown in FIG. 2B, there is no third-party application directory shown to the user. Instead, only the user files are visible to the user. In some implementations, hiding the data directory is useful to avoid the user being confused by unknown or unrecognizable files, which could lead to the user modifying the data directories rendering the contents unusable by the third-party application. In some implementations, a third-party application data directory is visible to the user (not shown in FIG. 2B), but the directory is not modifiable by the user. In such implementation, the visible third-party-application data directory may include information indicating the storage volume

utilized by the directory. In another implementation, the contents of the directory may be deleted or modified, for example, when the third-party application is no longer available or used by the user. In such implementations, the directory may continue to exist, without any content, for use if the third-party application becomes available to the user again.

**[0028]** Turning to FIG. 3, which depicts a user interface for application management, a user may wish to view a list of applications associated with the user. Such a list of my apps 703 may include any applications available to the user. In addition, storage volumes utilized by each application may also be visible to the user. Total application storage volume information may include all application data associated with the user, including data in certain third-party application directories which may not be included in the application totals. As shown, in some implementations, the user may be able to see total storage utilization, but not specific application directory storage volumes.

**[0029]** FIGS. 4-6 are flowcharts of methods that may be used by a processor such as processor 105 for performing the techniques described herein. In particular, FIG. 4 is a flowchart of a high level method 400 for allocating an application specific directory, in accordance with an illustrative implementation. As shown, FIG. 4 includes the steps of receiving a file on a cloud storage system (step 402), identifying a third-party application for the file (step 404), allocating a third-party application directory (step 406) and authorizing third-party application access to the third-party application directory (step 408).

**[0030]** Method 400 may begin at step 402, when a file is received on a cloud storage system. In some implementations, the file may be received when a user seeks to store the file on a cloud storage system, or otherwise interact with the file using the cloud storage system. A processor for the cloud storage system, or a third-party application manager system may identify a third-party application associated with the file at step 404. The identification process may include consulting a list of third-party applications known to the cloud storage system, determining application information that is associated with the file, or may require communication with a third-party application developer. An allocation of a third-party application directory may be made by the processor for the identified third-party application at step 406. The third-party application directory may be a data directory that is used to store particular data for the third-party application, for example, configuration data, and which is stored in association with the respective user in the cloud storage system. At step 408 the processor authorizes the third-party

application to access the third-party application directory. The authorization process may include communicating authenticating information between the processor and the third-party application. Authentication information may be included in the third-party application directory. The third-party application directory as well as its contents may be assigned an authorization field so that only the specific third-party application may access the directory. The authorization fields are copied from a parent file to each file when they are added.

**[0031]** FIG. 5 depicts a method for authenticating a third-party application requestor, in accordance with an implementation. Method 500 includes receiving a request to access a third-party application specific directory (step 502), determining application data (step 504), determining that the requestor is authorized to access the directory (step 506) and providing access to the authorized third-party application to the third-party application specific directory (step 508).

**[0032]** Authentication of a third-party application, in particular, a requesting third-party application may be performed using a processor in the cloud storage system or the third-party application manager system. A third-party application may request access to its third-party application directory on the cloud storage system by communicating such request to the cloud storage system, generally in connection with a user accessing a file. At step 502, a request is received to access a third-party directory. The processor may obtain certain information from the third-party application as well as the third-party directory to determine application data, at step 504. Typically, the third-party directory will have an authorization field that is associated with the application data. That authorization field is also included in all of the respective application files so that the application is explicitly authorized to access its files on the cloud storage system. The application data of the application may be matched to the third-party application directory at step 506 to determine whether the requestor application is authorized to access the directory. Application data, alias information, application identifiers and other authentication information may be stored in a database accessible to the processor, for example, database 106. If the application data and identifiers match, the processor will provide access to the third-party application specific directory at step 508.

**[0033]** A user is generally freely available to manage his or her files stored on the cloud storage system. However, as mentioned herein, application-specific data directories should not be modified by the user. Although application-specific directories can contain any kind of data

relevant to an application, such data is typically not useful on a standalone basis. As such, user modifications may be made out of context, or without knowledge of the data in the directory. A user may find it useful to know that some data directory exists that has a storage volume that goes towards the user's storage quota, and so an indication of the size of a data file may be made known to the user. In some implementations, the contents of the data directory are not modifiable by the user. A method for limiting access to an application specific directory is shown in FIG. 6.

**[0034]** FIG. 6 depicts a method 600 for providing a file manager based on a requestor, according to an implementation. As shown, method 600 includes receiving a request to view files (step 602), identify requestor associated files (step 604), limit interaction of files and directories based on the requestor type (step 606), and present a list of requestor's associated files and application directories (step 608).

**[0035]** The steps of method 600 may be performed by a processor in the cloud storage system. At step 602, a requestor may select to view the requestor's files by making a selection in a user device, a developer device, or other device to view an interface for a file manager or an application manager. The processor may receive the request to view the user's files and search its database for any available files associated with the requestor. The files may be identified at step 604 by the processor using known techniques for identifying files associated with a particular user. The processor will also obtain information about the user type, for example, whether the user is an individual, or an application developer. Based on the user type, the processor will limit the interaction of the files and the application directories at step 606. For example, an individual user may be limited from viewing the contents of a third-party application directory, or seeing such directory at all. In some implementations, the individual user may see that there is a directory that has a certain stored amount. In such implementation, the user may clear the contents of the directory, or make limited modifications to the directory contents. For an application developer user, however, such limitations would not be applied to its respective third-party application specific directory. Other third-party application directories would not be available for view or interaction by a different third-party application user. At step 608, the processor will present a list of the requestor's file and directories, including the limitations based on the requestor type, to the requestor using for example, a file manager user interface display.

**[0036]** The components of the system 100 of FIG. 1 may be arranged, distributed, and combined in any of a number of ways. FIG. 7 is a block diagram of a computing device, such as any of the components of the system of FIG. 1, for performing any of the processes described herein, according to an illustrative embodiment. Each of the components of these systems may be implemented on one or more computing devices 700. In certain aspects, a plurality of the components of these systems may be included within one computing device 700. In certain implementations, a component and a storage device may be implemented across several computing devices 700.

**[0037]** The computing device 700 comprises at least one communications interface unit, an input/output controller 710, system memory, and one or more data storage devices. The system memory includes at least one random access memory (RAM 702) and at least one read-only memory (ROM 704). All of these elements are in communication with a central processing unit (CPU 706) to facilitate the operation of the computing device 700. The computing device 700 may be configured in many different ways. For example, the computing device 700 may be a conventional standalone computer or alternatively, the functions of computing device 700 may be distributed across multiple computer systems and architectures. In FIG. 7, the computing device 700 is linked, via network or local network, to other servers or systems.

**[0038]** The computing device 700 may be configured in a distributed architecture, wherein databases and processors are housed in separate units or locations. Some units perform primary processing functions and contain at a minimum a general controller or a processor and a system memory. In distributed architecture implementations, each of these units may be attached via the communications interface unit 708 to a communications hub or port (not shown) that serves as a primary communication link with other servers, client or user computers and other related devices. The communications hub or port may have minimal processing capability itself, serving primarily as a communications router. A variety of communications protocols may be part of the system, including, but not limited to: Ethernet, SAP, SAS™, ATP, BLUETOOTH™, GSM and TCP/IP.

**[0039]** The CPU 706 comprises a processor, such as one or more conventional microprocessors and one or more supplementary co-processors such as math co-processors for offloading workload from the CPU 706. The CPU 706 is in communication with the communications interface unit 708 and the input/output controller 710, through which the CPU 706 communicates

with other devices such as other servers, user terminals, or devices. The communications interface unit 708 and the input/output controller 710 may include multiple communication channels for simultaneous communication with, for example, other processors, servers or client terminals in the network 718.

**[0040]** The CPU 706 is also in communication with the data storage device. The data storage device may comprise an appropriate combination of magnetic, optical or semiconductor memory, and may include, for example, RAM 702, ROM 704, flash drive, an optical disc such as a compact disc or a hard disk or drive. The CPU 706 and the data storage device each may be, for example, located entirely within a single computer or other computing device; or connected to each other by a communication medium, such as a USB port, serial port cable, a coaxial cable, an Ethernet cable, a telephone line, a radio frequency transceiver or other similar wireless or wired medium or combination of the foregoing. For example, the CPU 706 may be connected to the data storage device via the communications interface unit 708. The CPU 706 may be configured to perform one or more particular processing functions.

**[0041]** The data storage device may store, for example, (i) an operating system 712 for the computing device 700; (ii) one or more applications 714 (e.g., computer program code or a computer program product) adapted to direct the CPU 706 in accordance with the systems and methods described here, and particularly in accordance with the processes described in detail with regard to the CPU 706; or (iii) database(s) 716 adapted to store information that may be utilized to store information required by the program.

**[0042]** The operating system 712 and applications 714 may be stored, for example, in a compressed, an uncompiled and an encrypted format, and may include computer program code. The instructions of the program may be read into a main memory of the processor from a computer-readable medium other than the data storage device, such as from the ROM 704 or from the RAM 702. While execution of sequences of instructions in the program causes the CPU 706 to perform the process steps described herein, hard-wired circuitry may be used in place of, or in combination with, software instructions for implementation of the processes of the present disclosure. Thus, the systems and methods described are not limited to any specific combination of hardware and software.

**[0043]** Suitable computer program code may be provided for performing one or more functions in relation to associating an application with one or more file formats using a third-party

application manager system as described herein. The program also may include program elements such as an operating system 712, a database management system and "device drivers" that allow the processor to interface with computer peripheral devices (e.g., a video display, a keyboard, a computer mouse, etc.) via the input/output controller 710.

**[0044]** The term "computer-readable medium" as used herein refers to any non-transitory medium that provides or participates in providing instructions to the processor of the computing device 700 (or any other processor of a device described herein) for execution. Such a medium may take many forms, including but not limited to, non-volatile media and volatile media. Non-volatile media include, for example, optical, magnetic, or opto-magnetic disks, or integrated circuit memory, such as flash memory. Volatile media include dynamic random access memory (DRAM), which typically constitutes the main memory. Common forms of computer-readable media include, for example, a floppy disk, a flexible disk, hard disk, magnetic tape, any other magnetic medium, a CD-ROM, DVD, any other optical medium, punch cards, paper tape, any other physical medium with patterns of holes, a RAM, a PROM, an EPROM or EEPROM (electronically erasable programmable read-only memory), a FLASH-EEPROM, any other memory chip or cartridge, or any other non-transitory medium from which a computer can read.

**[0045]** Various forms of computer readable media may be involved in carrying one or more sequences of one or more instructions to the CPU 706 (or any other processor of a device described herein) for execution. For example, the instructions may initially be borne on a magnetic disk of a remote computer (not shown). The remote computer can load the instructions into its dynamic memory and send the instructions over an Ethernet connection, cable line, or even telephone line using a modem. A communications device local to a computing device 700 (e.g., a server) can receive the data on the respective communications line and place the data on a system bus for the processor. The system bus carries the data to main memory, from which the processor retrieves and executes the instructions. The instructions received by main memory may optionally be stored in memory either before or after execution by the processor. In addition, instructions may be received via a communication port as electrical, electromagnetic or optical signals, which are exemplary forms of wireless communications or data streams that carry various types of information.

**[0046]** It will be apparent to one of ordinary skill in the art that aspects of the systems and methods described herein may be implemented in many different forms of software, firmware,

and hardware in the implementations illustrated in the figures. The actual software code or specialized control hardware used to implement aspects consistent with the principles of the systems and method described herein is not limiting. Thus, the operation and behavior of the aspects of the systems and methods were described without reference to the specific software code – it being understood that one of ordinary skill in the art would be able to design software and control hardware to implement the aspects based on the description herein.

**[0047]** Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be advantageous.

What is claimed is:

1. A method for providing dedicated storage for a third-party application in a cloud storage system, the method comprising:

receiving, by a processor, a file for storage in a cloud storage system, wherein the file is associated with a third-party application and a user of the cloud storage system;

identifying by the processor, the third-party application;

automatically allocating, by the processor, a directory for the identified third-party application and the user; and

providing, by the processor, access to the directory for the identified third-party application to the third-party application.

2. The method of claim 1 further comprising hiding the contents of the directory from the user.

3. The method of claim 1 further comprising assigning, by the processor, a reference link to the directory, the reference link based on application data associated with the third-party application and an identifier for the user.

4. The method of claim 1 further comprising receiving at the processor, an access request from a second user to access a second file associated with a second third-party application, and associating a second directory with the second third-party application and the second user.

5. The method of claim 1 further comprising receiving at the processor, a request from a third-party application to access a file in the directory; and determining whether the third-party application is authorized to access the directory.

6. The method of claim 5 wherein determining whether the third-party application is authorized to access the file in the directory comprises determining that the third-party application is associated with the directory and the user.

7. The method of claim 1 wherein the directory is configured to store application data for the third-party application.

8. The method of claim 1 wherein the directory is configured to store user specified data for the third-party application.

9. The method of claim 1 further comprising: detecting by the processor that the third-party application is no longer available to the user, and prompting the user to delete contents of the directory.

10. The method of claim 1 further comprising: presenting to the user information for the directory; and providing an option to modify the contents of the directory.

11. A system for providing dedicated storage for a third-party application in a cloud storage system, the system comprising a processor and storage accessible via a network, the processor configured to:

receive a file for storage, wherein the file is associated with a third-party application and a user of the cloud storage system;

identify the third-party application;

automatically allocate a directory for the identified third-party application and the user; and

provide access to the directory for the identified third-party application to the third-party application.

12. The system of claim 11 wherein the processor is further configured to hide the contents of the directory from the user.

13. The system of claim 11 wherein the processor is further configured to assign a reference link to the directory, the reference link based on application data associated with the third-party application and an identifier for the user.

14. The system of claim 11 wherein the processor is further configured to receive an access request from a second user to access a second file associated with a second third-party application, and associating the second directory with the second third-party application and the second user.

15. The system of claim 11 wherein the processor is further configured to receive an access request to a file in the directory by a third-party application; and determine whether the third-party application is authorized to access the directory.

16. The system of claim 15 wherein determining whether the third-party application is authorized to access the file in the directory comprises determining that the third-party application is associated with the directory and the user.

17. The system of claim 11 wherein the directory is configured to store application data for the third-party application.

18. The system of claim 11 wherein the directory is configured to store user specified data for the third-party application.

19. The system of claim 11 wherein the processor is further configured to detect that the third-party application is no longer available to the user, and prompt the user to delete contents of the directory.

20. The system of claim 11 wherein the processor is further configured to present to the user information for the directory and providing an option to modify the contents of the directory.

21. A method for supporting a third-party application in a cloud storage system, the method comprising:

allocating, by a processor, a directory for a third-party application;

associating, by the processor, the directory with a user of the cloud storage system;

receiving, by the processor, application data for the third-party application;

storing the application data in the directory; and

hiding contents of the directory from view by the user.

22. A system for supporting a third-party application in a cloud storage system, the system comprising a processor and storage accessible via a network, the processor configured to:

allocate a directory for a third-party application;

associate the directory with a user of the cloud storage system;

receive application data for the third-party application;

store the application data in the directory; and

hide contents of the directory from view by the user.

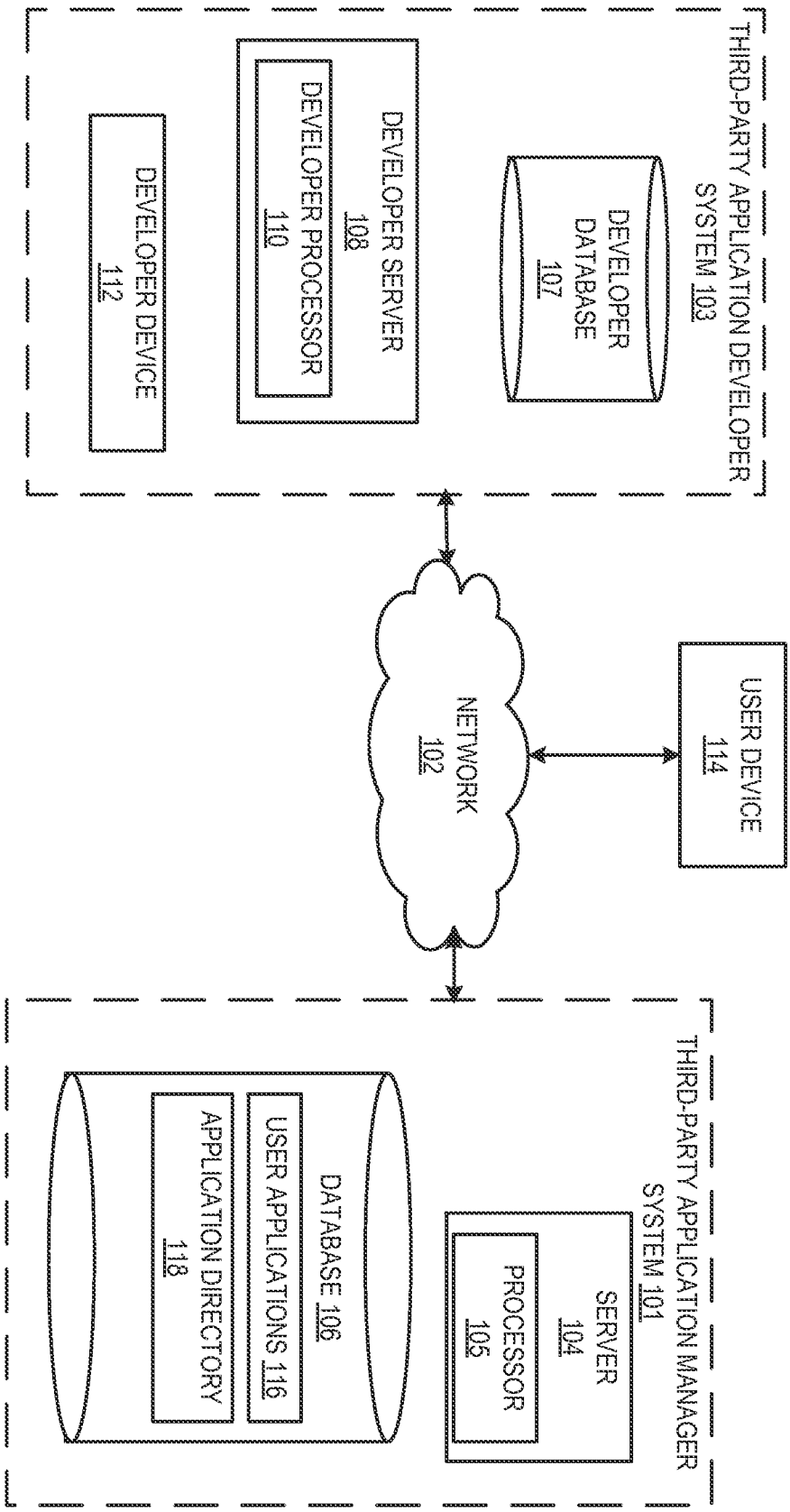
THIRD-PARTY APPLICATION DEVELOPER SYSTEM 103
DEVELOPER DATABASE 107
DEVELOPER SERVER 108
DEVELOPER PROCESSOR 110
DEVELOPER DEVICE 112
NETWORK 102
USER DEVICE 114
THIRD-PARTY APPLICATION MANAGER SYSTEM 101
SERVER 104
PROCESSOR 105
DATABASE 106
USER APPLICATIONS 116
APPLICATION DIRECTORY 118
100

FIG. 1

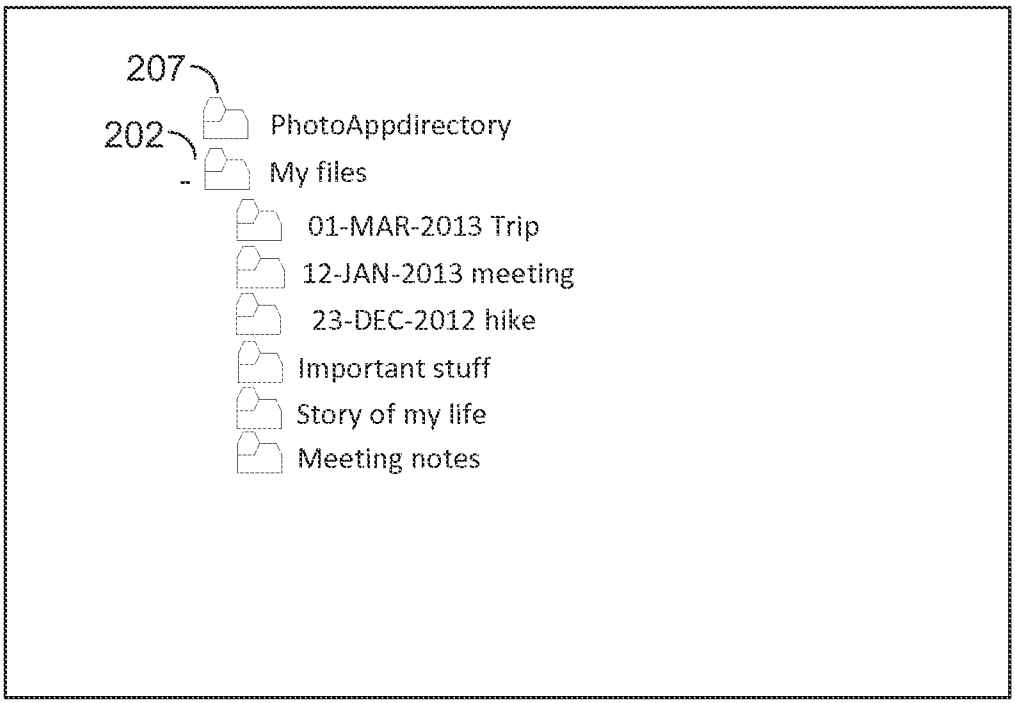
207
202
PhotoAppdirectory
My files
01-MAR-2013 Trip
12-JAN-2013 meeting
23-DEC-2012 hike
Important stuff
Story of my life
Meeting notes

FIG. 2A

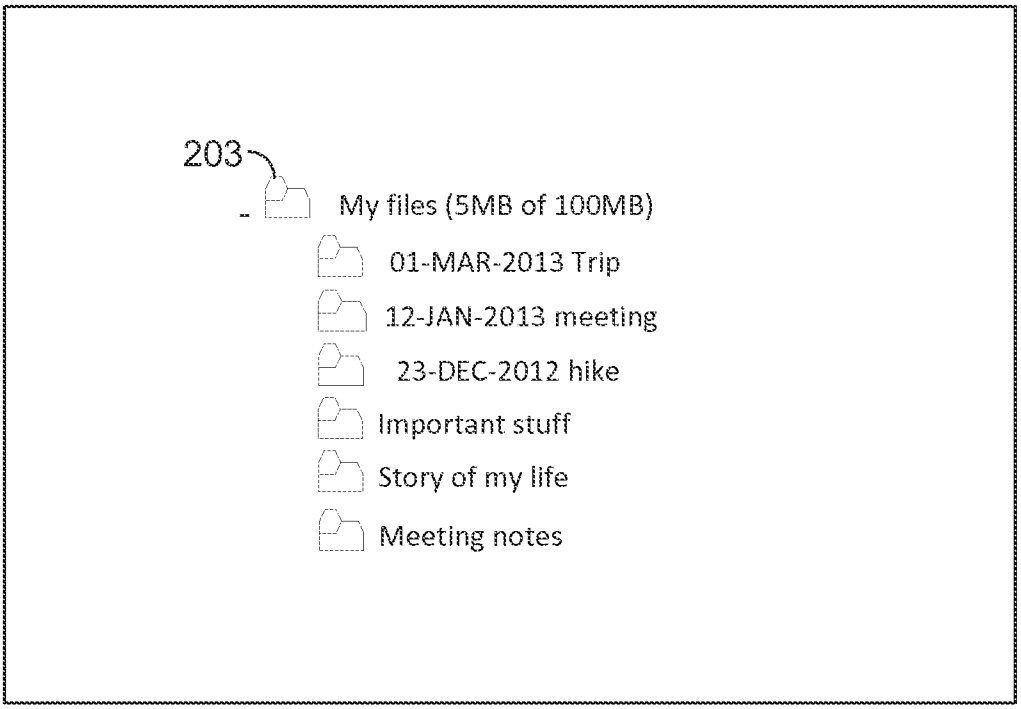
203
My files (5MB of 100MB)
01-MAR-2013 Trip
12-JAN-2013 meeting
23-DEC-2012 hike
Important stuff
Story of my life
Meeting notes

FIG. 2B

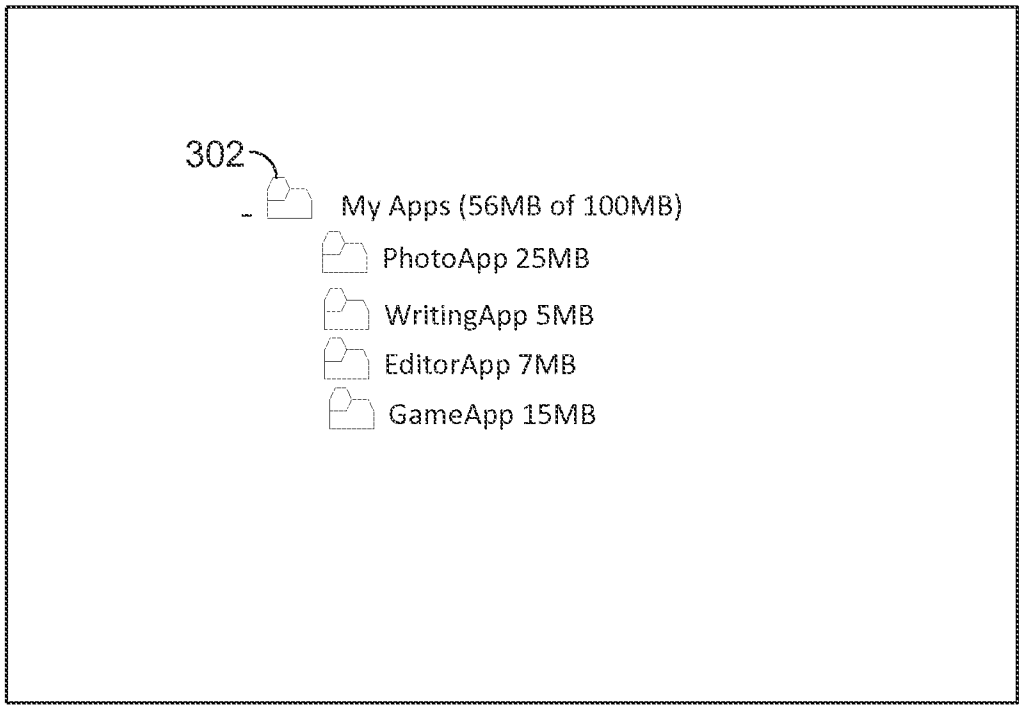
302
My Apps (56MB of 100MB)
PhotoApp 25MB
WritingApp 5MB
EditorApp 7MB
GameApp 15MB

FIG. 3

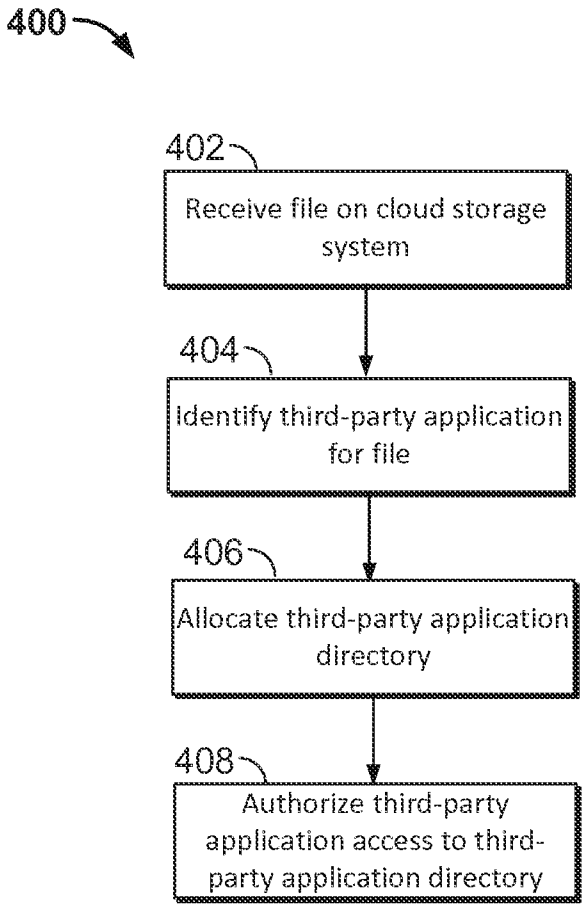
400
402
Receive file on cloud storage system
404
Identify third-party application for file
406
Allocate third-party application directory
408
Authorize third-party application access to third-party application directory

FIG. 4

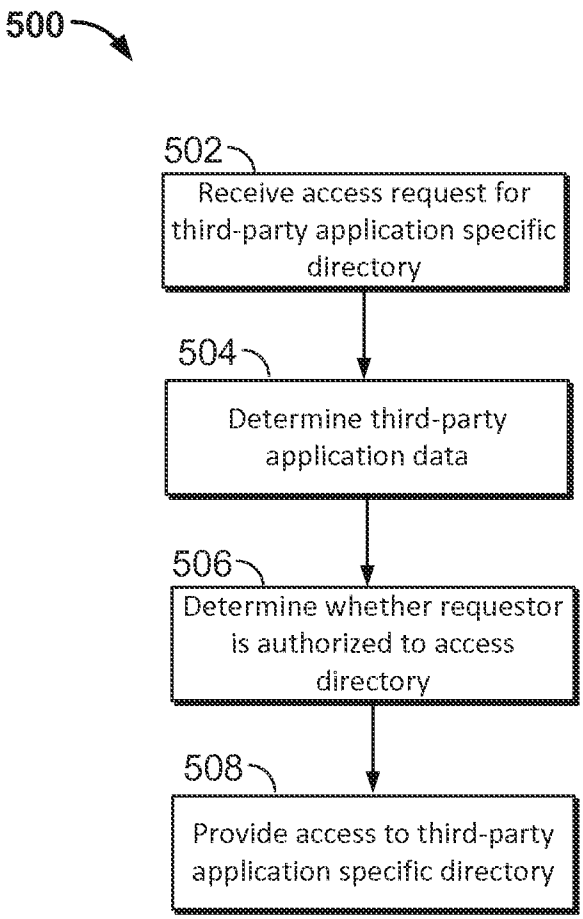
500
502
Receive access request for third-party application specific directory
504
Determine third-party application data
506
Determine whether requestor is authorized to access directory
508
Provide access to third-party application specific directory

FIG. 5

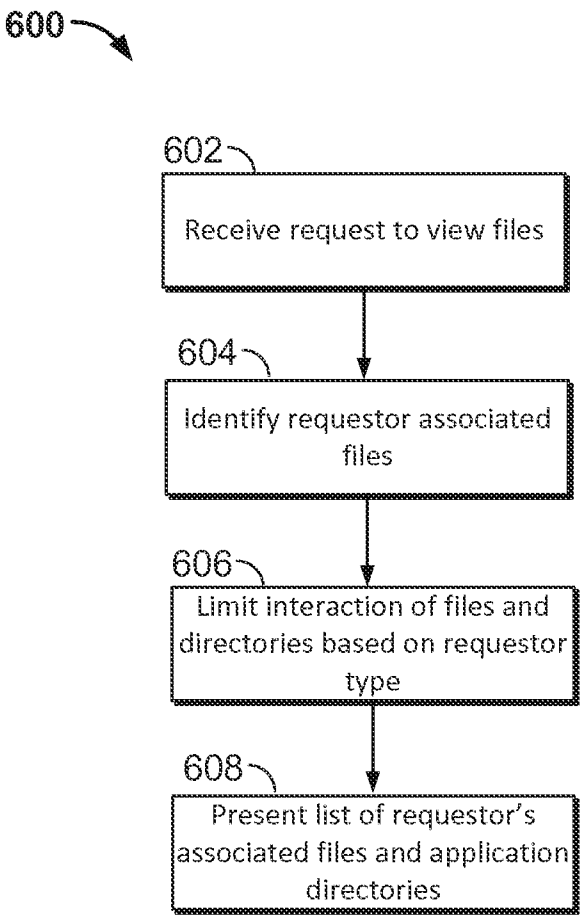
600
602
Receive request to view files
604
Identify requestor associated files
606
Limit interaction of files and directories based on requestor type
608
Present list of requestor's associated files and application directories

FIG. 6

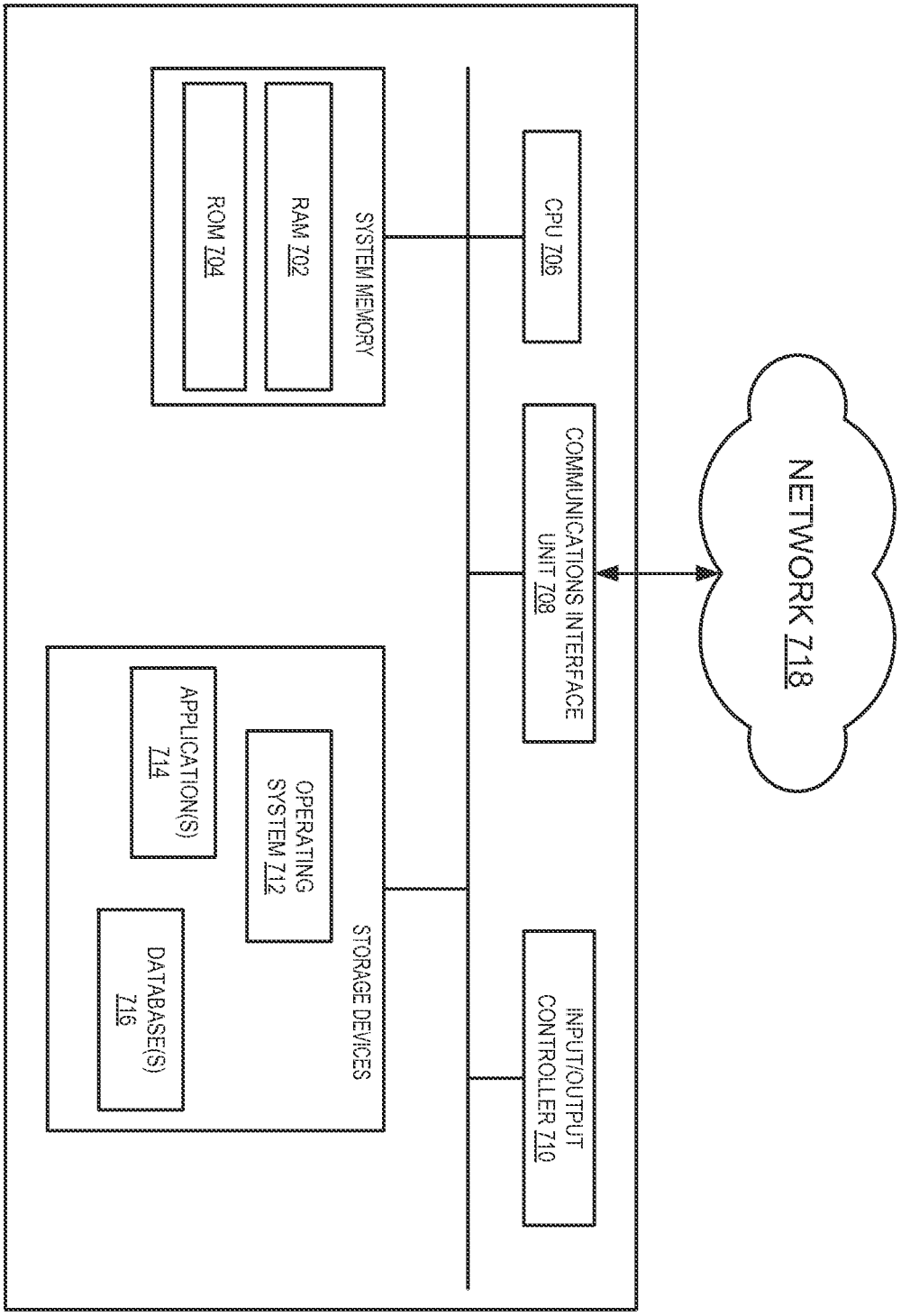
700
NETWORK 718
CPU 706
COMMUNICATIONS INTERFACE UNIT 708
INPUT/OUTPUT CONTROLLER 710
SYSTEM MEMORY
RAM 702
ROM 704
STORAGE DEVICES
OPERATING SYSTEM 712
APPLICATION(S) 714
DATABASE(S) 716

FIG. 7

# INTERNATIONAL SEARCH REPORT

International application No
PCT/US2014/037791

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F17/30
ADD.

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)
G06F

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)
EPO-Internal, WPI Data, INSPEC

C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| X | US 2013/086585 A1 (HUANG YUN SONG [CN] ET AL) 4 April 2013 (2013-04-04) | 1-8, 10-18, 20-22 |
| Y | paragraphs [0039] - [0043], [0063], [0076], [0078], [0096], [0136], [0176] | 9,19 |
| Y | US 2007/226535 A1 (GOKHALE PARAG [US]) 27 September 2007 (2007-09-27) paragraph [0086] | 9,19 |
| A | WO 2010/054374 A1 (GOOGLE INC [US]; UHRHANE ERIC J [US]; PAPAKIPOS MATTHEW [US]) 14 May 2010 (2010-05-14) paragraphs [0005], [0006], [0022] - [0025], [0027] - [0030], [0036], [0037], [0040], [0041], [0045] - [0047], [0053] | 1-3, 7-13, 17-22 |

-/--

[X] Further documents are listed in the continuation of Box C.

[X] 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

"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 |
|---|---|
| 22 September 2014 | 08/10/2014 |

| Name and mailing address of the ISA/ | Authorized officer |
|---|---|
| European Patent Office, P.B. 5818 Patentlaan 2<br>NL - 2280 HV Rijswijk<br>Tel. (+31-70) 340-2040,<br>Fax: (+31-70) 340-3016 | Moran, Matthew |

# INTERNATIONAL SEARCH REPORT

International application No
PCT/US2014/037791

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| A | SYMANTEC CORPORATION: "E-security begins with sound security policies", ANNOUNCEMENT SYMANTEC, XX, XX, 14 June 2001 (2001-06-14), XP002265695, page 10-11, section "Who Writes the Policy?" | 4-6, 14-16 |
| A | LAFERRIERE C ET AL: "Authentication and authorization techniques in distributed systems", SECURITY TECHNOLOGY, 1993 SECURITY TECHNOLOGY, PROCEEDINGS, INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS 1993 INTERNATIONAL CARNAHAN CO NFERENCE ON OTTAWA, ONT., CANADA 13-15 OCT. 1993, NEW YORK, NY, USA,IEEE, 13 October 1993 (1993-10-13), pages 164-170, XP010124730, DOI: 10.1109/CCST.1993.386805 ISBN: 978-0-7803-1479-5 the whole document | 4-6, 14-16 |
| A | LAMPSON B ET AL: "AUTHENTICATION IN DISTRIBUTED SYSTEMS: THEORY AND PRACTICE", ACM TRANSACTIONS ON COMPUTER SYSTEMS (TOCS), ASSOCIATION FOR COMPUTING MACHINERY, INC, US, vol. 10, no. 4, 1 November 1992 (1992-11-01), pages 265-310, XP000336440, ISSN: 0734-2071, DOI: 10.1145/138873.138874 the whole document | 4-6, 14-16 |

# INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2014/037791

| Patent document cited in search report | Publication date | Patent family member(s) | Publication date |
|---|---|---|---|
| US 2013086585 A1 | 04-04-2013 | CN 103034453 A | 10-04-2013 |
| | | US 2013086585 A1 | 04-04-2013 |
| US 2007226535 A1 | 27-09-2007 | NONE | |
| WO 2010054374 A1 | 14-05-2010 | CN 102246168 A | 16-11-2011 |
| | | CN 103605674 A | 26-02-2014 |
| | | EP 2356590 A1 | 17-08-2011 |
| | | US 2010121893 A1 | 13-05-2010 |
| | | US 2013275471 A1 | 17-10-2013 |
| | | WO 2010054374 A1 | 14-05-2010 |