



- (51) **International Patent Classification:**  
*H04L 12/26* (2006.01) *H04L 12/24* (2006.01)
- (21) **International Application Number:**  
PCT/US2015/038258
- (22) **International Filing Date:**  
29 June 2015 (29.06.2015)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**  
2201/CHE/2015 29 April 2015 (29.04.2015) IN
- (71) **Applicant:** HEWLETT PACKARD ENTERPRISE DEVELOPMENT LP [US/US]; 11445 Compaq Center Drive West, Houston, TX 77070 (US).
- (72) **Inventors:** SEBASTIAN, Celestian K.; Sy.No.192, Whitefield Road, Mahadevapura Post, Bangalore 560048 (IN). E P, Vijeesh; Sy.No.192, Whitefield Road, Mahadevapura Post, Bangalore 560048 (IN).
- (74) **Agents:** FERGUSON, Christopher W. et al.; Hewlett Packard Enterprise, 3404 E. Harmony Road, Mail Stop 79, Fort Collins, CO 80528 (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, 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, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

**Declarations under Rule 4.17:**

- as to the identity of the inventor (Rule 4.17(i))
- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))

[Continued on next page]

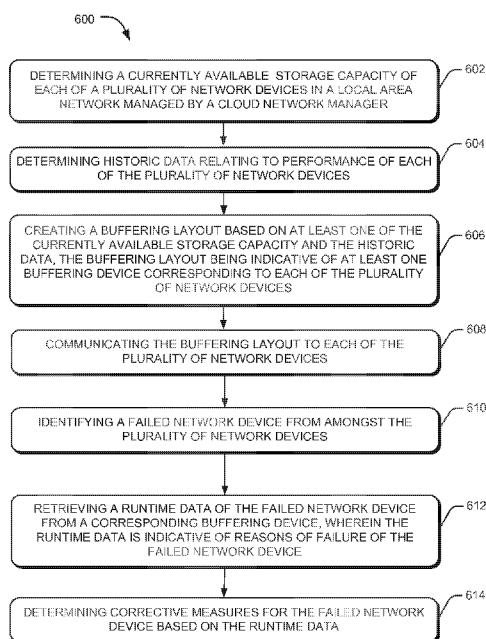
**(54) Title:** FAILURE OF NETWORK DEVICES IN CLOUD MANAGED NETWORKS

Figure 6

(57) **Abstract:** Examples of techniques to manage failure of network devices in cloud managed networks (CMN) are described. A network device of a CMN identifies a first buffering device in the CMN to store runtime data of the network device and provides its runtime data to the first buffering device. The runtime data of the network device is retrievable by a cloud network manager of the CMN to diagnose reasons for failure of the network device in an event of the network device being rendered non-functional.



- 
- *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*
- Published:**
- *with international search report (Art. 21(3))*

## FAILURE OF NETWORK DEVICES IN CLOUD MANAGED NETWORKS

### BACKGROUND

**[0001]** Historically, network administrators have managed a local area network using dedicated systems deployed within the local area network itself. However, with the recent advent of technology, cloud managed networks have emerged. Cloud managed networks allow the network administrators to use a remote server, generally referred to as a cloud network manager, to perform administrative tasks to manage a local area network from a remote location.

**[0002]** Accordingly, in case of an enterprise network having several local area networks distributed in different locations, a cloud network manager eliminates the need for a network administrator to be present at each of the locations. A network administrator, at one location, can manage the several distributed local area networks of the enterprise network using the cloud network manager.

### BRIEF DESCRIPTION OF DRAWINGS

**[0003]** The detailed description is described with reference to the accompanying figures. In the figures, the left-most digit(s) of a reference number identifies the figure in which the reference number first appears. The same numbers are used throughout the figures to reference like features and components:

**[0004]** Figure 1 schematically illustrates a cloud managed network (CMN), according to an example of the present subject matter;

**[0005]** Figures 2A and 2B illustrate a cloud network manager of the CMN, according to an example of the present subject matter;

**[0006]** Figure 3 illustrates a network device of the CMN, according to an example of the present subject matter;

**[0007]** Figure 4, 5, and 6 illustrate methods for managing failure of network devices in the CMN, according to various example implementations of the present subject matter; and

**[0008]** Figure 7 illustrates a network environment for managing failure of the network devices in the CMN, in accordance with an example of the present subject matter.

#### DETAILED DESCRIPTION

**[0009]** In a cloud managed network (CMN), network devices update their latest runtime data repeatedly with a cloud network manager associated with the network devices. Thus, at any instant of time, if a given network device were to fail, the runtime data of the network device is available with the cloud network manager to inspect causes of failure of the network device. Constantly updating the runtime data with the cloud network manager consumes significant Internet bandwidth, and adds to overheads in management plane of the CMN.

**[0010]** Aspects relating to managing failure of network devices in a CMN are described herein. The systems and methods provide for making the runtime data available for a cloud network manager to diagnose failure of a network device without significant Internet bandwidth consumption and without increase in management plane data of the CMN.

**[0011]** In an example, to manage failure of network devices in a CMN, a first network device is nominated from amongst a plurality of network devices in a local area network managed by a cloud network manager. Each of the plurality of network devices stores its runtime data in the first network device. In case of failure of a network device, from amongst a plurality of network devices, the first network device provides the runtime data corresponding to the failed network device to the cloud network manager for diagnosis of the failed network device. Since the plurality of network devices communicates with the first network device over the local area network, the Internet bandwidth is not consumed as is in the case the network devices updates the runtime data to the cloud network manager which is coupled to the network devices over Internet.

**[0012]** In an example implementation, a cloud network manager, comprising a processor and a network monitoring module coupled to the processor, is described. In operation, the network monitoring module may identify a failed network device in a local area network managed by the cloud network manager. A troubleshooting module, coupled to the processor retrieves

the runtime data of the failed network device from another network device, referred to as a first buffering device, coupled to the failed network device, to inspect the cause of the failure.

**[0013]** As would be understood, alike the first network device, the first buffering device is a network device nominated to store runtime data of the failed network device. In a situation where the failed network device is unreachable to the cloud network manager upon its failure, the cloud network manager may obtain the runtime data of the failed network device from the first buffering device. Thus, runtime data of the failed network device is made available to the cloud network manager for diagnostic purposes even in absence of the network device constantly updating its runtime data to the cloud network manager.

**[0014]** The manner in which the systems and methods for managing failure of network devices in a CMN can be implemented shall be explained in details with respect to Figures 1 to 7. While aspects of described systems and methods for managing network device failures in a CMN can be implemented in any number of different computing systems, environments, and/or configurations, the examples are described in the context of the following example system(s).

**[0015]** Figure 1 illustrates a CMN 100, according to an example of the present subject matter. The CMN 100 may be understood as a public or a private network system implementing network devices 102 that enable end user devices to communicate with each other. In an example implementation, the CMN 100 may be an enterprise network that may be distributed in more than one location, such that network devices 102 of the CMN 100 at a given location are coupled to each other in a local area network.

**[0016]** The CMN 100 may be implemented as a wireless network or a wired network, or a combination thereof. The CMN 100 can be an individual network or a collection of many such individual networks, interconnected with each other and functioning as a single large network, e.g., the Internet or an intranet. The CMN 100 may either be a dedicated network or a shared network, which represents an association of the different types of networks that use a

variety of protocols, for example, Hypertext Transfer Protocol (HTTP), Transmission Control Protocol/Internet Protocol (TCP/IP), etc., to communicate with each other. The CMN 100 may also include individual networks, such as, but are not limited to, GSM network, UMTS network, LTE network, PCS network, TDMA network, CDMA network, NGN, PSTN, and ICMN.

**[0017]** In the illustrated example implementation, the CMN 100 comprises a first local area network 104-1 and a second local area network 104-2. Further, the first local area network 104-1 includes network devices N1, N2, N3, and N4 while the second local area network 104-2 includes network devices N5, N6, N7, and N8. The network devices 102 may include, but are not limited to network switches and network routers. Examples of the network devices 102 may also include network hubs, access points, Host Bus Adaptors (HBAs) and other network entities.

**[0018]** Each of the network devices 102 in the CMN 100 are coupled to a cloud network manager 106. The cloud network manager 106 may be a desktop computer, laptop, server, mainframe computer, and the like, belonging to an end user, such as a network administrator, a service provider, an organization, or an enterprise, and implemented to manage the network devices 102. The cloud network manager 106 may communicate with the network devices 102 over a communication network 108, such as a wide area network, for example, the Internet.

**[0019]** The cloud network manager 106 communicates with the network devices 102 for various purposes, such as configuring a firmware into the network devices 102. The cloud network manager 106 also performs maintenance related activities, such as updating the firmware in the network devices 102. The cloud network manager 106 may also perform remote troubleshooting of the network device 102. For example, the cloud network manager 106 may allow a network administration to provide troubleshooting commands to the network devices 102.

**[0020]** In an example implementation, when any of the network devices 102 fail, for example, if any of the network devices are unresponsive or unreachable by the cloud network manager 106, the cloud network manager

106 inspects runtime data of the failed network device to determine the corrective measures to be taken. As explained previously, the runtime data of a network device is indicative of causes of failure of the network device. Examples of runtime data are event details, logs, and statistics relating to performance of the network device.

**[0021]** Generally, runtime data of a failed network device may be physically retrieved from a local memory component, such as a flash memory of the network device. This involves intervention by a network administrator. In case of cloud managed networks, such as the CMN 100, a network administrator may not be present at the site of each of the local area networks of the CMN 100 to retrieve the runtime data. Rather, in case of the CMN 100, a network administrator may generally manage the network devices 102 in the local area networks through the cloud network manager 106.

**[0022]** Accordingly, the network devices 102 periodically update their runtime data with cloud network manager 106, such that the cloud network manager 106 has the latest runtime data available for inspection in case any of the network devices 102 fail. For example, if an event causes a network device to crash, details of the last event logged by the network device prior to its crashing may be investigated by the cloud network manager 106 to diagnose reasons of failure of the network device. However, frequent updating of the cloud network manager 106 involves transmitting the runtime data of the network devices 102 over the communication network 108 and consumes network bandwidth.

**[0023]** In accordance with one example implementation of the present subject matter, one of the network devices 102 in a local area network is nominated to store the runtime data of the remaining network devices 102. The nominated network device receives and stores the runtime data of the remaining network devices 102. As would be understood, the communication between the nominated network device and the remaining network devices 102 is over the local area network and does not consume bandwidth of the communication network 108. In case of failure of any of the remaining network devices 102, the

nominated network device provides the runtime data of the failed network device to the cloud network manager 106.

**[0024]** To illustrate with an example, network device N4 in the first local area network 104-1 may be nominated to buffer the runtime data of the network devices N1, N2, and N3. The network devices N1, N2, and N3 provide their respective runtime data to network device N4 and in case any of the network devices N1, N2, or N3 crashes, the runtime data corresponding to the failed network device, stored in network device N4, is made available to the cloud network manager 106.

**[0025]** In an implementation, more than one network devices 102 may be nominated to buffer the runtime data of other network devices 102. For example, in the illustrated implementation, the network device N2 is a second network device that is nominated in addition to the first nominated network device N4 in the first local area network 104-1. The second nominated network device may be a back-up to the first nominated network device to store the runtime data of the remaining network devices 102. In one example, the second nominated network device may store the runtime data of the remaining network devices 102 in addition to the first nominated network device to provide redundancy. In another example, the second nominated network device may start storing the runtime data of the remaining network devices 102 in case the first nominated network device fails.

**[0026]** In some example scenarios, the runtime data of a nominated network device may be stored in another network device of the CMN 102. A neighboring network device, i.e., any network device in the CMN 102 which is coupled to the nominated network device is selected to store the runtime data of the nominated network device. The neighboring network device may be in the same local area network as the nominated network device or may be a part of any other local area network managed by the cloud network manager 106.

**[0027]** To explain in reference to the foregoing example illustrated in Figure 1, network device N5 in the second local area network 104-2, which is a neighboring device for the nominated network device N4 of the first local area network 104-1, stores the runtime data of the network device N4. Accordingly, in

case the nominated network device N4 itself fails, the cloud network manager 106 may retrieve the runtime data of the nominated network device N4 from the network device N5. Similarly, in one example network device N7 may store the runtime data of the second nominated network device N2. Further, in some examples, the second nominated network device N2 also being a neighbor of the first nominated network device N2, may store the runtime data of the first nominated network device N2 in addition to being a back-up to the same.

**[0028]** Details relating to implementation and working of the cloud network manager 106 have been elaborated with reference to Figures 2A and 2B that illustrate a cloud network manager, according to an example of the present subject matter.

**[0029]** In accordance with an example implementation illustrated in Figure 2A, the cloud network manager 106 includes a processor 202. The cloud network manager 106 also includes a network monitoring module 204 and a troubleshooting module 206, both coupled to the processor 202.

**[0030]** The processor 202 may be implemented as one or more microprocessors, microcomputers, microcontrollers, digital signal processors, central processing units, state machines, logic circuitries, and/or any devices that manipulate signals based on operational instructions. Among other capabilities, the processor(s) 202 is configured to fetch and execute computer-readable instructions stored in the memory.

**[0031]** The functions of the various elements shown in the figure, including any functional blocks labeled as "processor(s)", may be provided through the use of dedicated hardware as well as hardware capable of executing software in association with appropriate software. When provided by a processor, the functions may be provided by a single dedicated processor, by a single shared processor, or by a plurality of individual processors, some of which may be shared. Moreover, explicit use of the term "processor" should not be construed to refer exclusively to hardware capable of executing software, and may implicitly include, without limitation, digital signal processor (DSP) hardware, network processor, application specific integrated circuit (ASIC), field programmable gate array (FPGA), read only memory (ROM) for storing

software, random access memory (RAM), non-volatile storage. Other hardware, conventional and/or custom, may also be included.

**[0032]** In operation, the network monitoring module 204 communicates with network devices 102 to identify a failed network device in a local area network managed by the cloud network manager 106. In an example, the cloud network manager 106 may identify a network device to have failed if the network device does not respond to a message send by the cloud network manager 106 within a predefined response time. As apparent, once the network device has failed, the network device is unreachable for the cloud network manager 106 to obtain the runtime data of the failed network device from a local memory component of the failed network device.

**[0033]** To make the runtime data of the failed network device available to the cloud network manager 106 for the diagnosis, the troubleshooting module 206 retrieves the runtime data of the failed network device from a first buffering device, the first buffering device being a network device, coupled to the failed network device, that is to store runtime data of the failed network device.

**[0034]** In some example implementations, in case the troubleshooting module 206 determines the first buffering device to be unresponsive, it identifies a second buffering device to retrieve the runtime data of the failed network device. Alike the first buffering device, the second buffering device is another network device that is coupled to the failed network device, and stores runtime data of the failed network device.

**[0035]** Other functionalities of the cloud network manager 106, such as identifying at least one buffering device corresponding to each of the plurality of network devices 102 in the local area is explained in reference to figure 2B that illustrates the cloud network manager 106 in accordance with another example implementation of the present subject matter.

**[0036]** Figure 2B depicts interface(s) 208 coupled to the processor 202. The interfaces 208 may include a variety of software and hardware interfaces that allow the cloud network manager 106 to interact with network devices 102 of the CMN 100. The interfaces 208 may facilitate communications within a wide variety of networks and protocol types, including wire networks, for example,

LAN, cable, etc., and wireless networks, for example WLAN, cellular, satellite-based network, etc. The interface(s) 208 allow the cloud network manager 106 to communicate with the network devices 102 of the CMN 100.

**[0037]** Further, the cloud network manager 106 comprises a memory 210 coupled to the processor 202. The memory 210 may include any computer-readable medium known in the art including, for example, volatile memory (e.g., RAM), and/or non-volatile memory (e.g., EPROM, flash memory, etc.). The cloud network manager 106 also comprises modules 212 and data 214 coupled to the processor 202. In one example, the modules 212 and data 214 may reside in the memory 210.

**[0038]** The modules 212 include routines, programs, objects, components, data structures, and the like, which perform particular tasks or implement particular abstract data types. The modules 212 further include modules that supplement applications on the cloud network manager 106, for example, modules of an operating system. The data 214 serves, amongst other things, as a repository for storing data that may be fetched, processed, received, or generated by one or more of the modules 212. The modules 212 may include a buffering layout creation module 216 and other modules 218 in addition to the aforementioned network monitoring module 204 and troubleshooting module 206. The other module(s) 218 may include programs or coded instructions that supplement applications and functions, for example, programs in the operating system of the cloud network manager 106, and the other data 220 comprise data corresponding to one or more other module(s) 218.

**[0039]** In one example, the troubleshooting module 206 identifies one or more buffering device, such as the first buffering device and the second buffering device based on a buffering layout of the CMN 100. The buffering layout is indicative of at least one buffering device corresponding to each of the plurality of network devices 102 in the local area networks managed by the cloud network manager 106. The buffering layout of a local area network in the CMN 100 may be understood as a mapping of each of the network devices 102 in the local area network, against one or more buffering devices either in the

same local area network, or, in some cases, in another local area network managed by the cloud network manager 106.

**[0040]** The troubleshooting module 206 receives the buffering layout from the buffering layout creation module 216. In some example implementations, the buffering layout creation module 216 performs the task of creating the buffering layout while in some other example implementations that will be explained subsequently, the buffering layout creation module 216 receives the buffering layout from one of the network devices 102 of the CMN 100.

**[0041]** To create the buffering layout, the buffering layout creation module 216 determines a currently available storage capacity of each of the plurality of network devices 102 in the local area network. The currently available storage capacity, as determined, may be stored as storage capacity data 222 within data 214 of the cloud network manager 106. For instance, some of network devices 102 in the CMN 100 may be handling high load since these network devices 102 may be catering to huge number of end user devices. The buffering layout creation module 216 may identify these network devices 102 to have less available storage capacity as compare to other network devices 102 that may be handling less load. Thus, the network devices 102 that handle lesser load and have more currently available storage capacity available may be nominated as buffering devices to store the runtime data of other network devices 102.

**[0042]** In an example, the nomination of a buffering device to buffer runtime data of other network devices 102 may also be based on historic data relating to performance of the buffering device. For the purpose, the buffering layout creation module 216 may use historic data relating to performance of each of the plurality of network devices 102 in the local area network that may be stored as performance data 224 in the cloud network manager 106. In one example, the historic data relating to performance of the network device may reveal the network devices that are more prone to frequent failures. Accordingly, the buffering layout creation module 216 selects a network device that is less prone to failure as a buffering device.

**[0043]** Based on the currently available storage capacity and historic data relating to performance of each of the plurality of network devices 102, the

buffering layout creation module 216 identifies at least one buffering device corresponding to each of the network devices 102 in the local area network to create the buffering layout. Buffering layout may be stored as buffering layout data 226 within data 214 of the cloud network manager 106. Based on the buffering layout, there may exist one buffering device for storing the runtime data of all the remaining network devices 102 in the local area network or different buffering devices may exist corresponding to each of the network devices 102 as the case may be.

**[0044]** The buffering layout may also be dynamically updated. For example, as load handled by the network devices varies at different instants of time, buffering layout creation module 216 may accordingly appoint different network devices 102 to acts at the buffering device for the other network devices 102 such that runtime data of network devices controlled by the cloud network manager 106 is efficiently buffered at all time instants. The cloud network manager 106 may accesses the runtime data of a network device for diagnosis purposes when required. The runtime data of a network device when retrieved by the cloud network manager 106 may be stored in as network device runtime data 228 in the data 214.

**[0045]** Figure 3 illustrates a network device 300 according to an example of the present subject matter. In one example, the network device 300 comprises a processor 302, interface(s) 304 coupled to the processor 302, and memory 306 coupled to the processor 302. The processor 302, interface(s) 304, and memory 306 of the network device 300 may be implemented in a manner similar to the processor 202, interface(s) 208, and memory 210, respectively, of the cloud network manager 106, and may operate likewise. Similar to the cloud network manager 106, the network device 300 also comprises modules 308 and data 310. The modules 308 include a network manager communication module 312, a peer device communication module 314, a nomination module 316 and other module(s) 318 while the data 310 may include the storage capacity data 222, the buffering layout data 226, other device runtime data 320, and other data 322.

**[0046]** In one example, the network manager communication module 312 of the network device 300 may interact with the cloud network manager 106 through the interface(s) 304 to receive the buffering layout created by the cloud network manager 106. The buffering layout may be stored in the buffering layout data 226 of the network device 300 and be periodically updated. Based on the buffering layout, the network device 300 determines whether it has been nominated as a buffering device to store the runtime data of other the network devices 102 or not.

**[0047]** When the network device 300 is nominated as the buffering device for other network devices 102, the peer device communication module 314 interacts with other network devices 102 through the interface(s) 304 to receive the runtime data of such other network devices 102. The runtime data of the other network devices 102 is stored as other device runtime data 320 in data 310. In case any of the other network devices 102 fails, the runtime data corresponding to the failed network device is provided to the cloud network manager 106. In an example, while the network device 300 receives the runtime data of other network devices 102 on an on-going basis and stores all the runtime data received, the network device 300 may communicate the latest runtime data to the cloud network manager 106 in case any of the network device 106 is rendered non-functional. This minimizes communication network bandwidth consumption.

**[0048]** In an example implementation, the network device 300 may receive the runtime data from each of the other network devices 102 at a predefined frequency. The predefined frequency of receiving a runtime data from a network device depends on how relevant of the runtime data is for diagnosis of the network device. For instance, runtime data, such as information relating to remaining buffer space, details of packets sent and received and information relating to counters of a network device, that are more relevant for diagnosing reasons of failure of the network device, are sent to the network device 300 more frequently than runtime data that are less relevant, for example, details of an Open Shorted Path First (OSPF) event that may have occurred at the network device. Accordingly, each of the other network devices

102 may provide the runtime data having high relevance at a higher frequency to the network device 300 than the runtime data having lower relevance.

**[0049]** Further, based on the buffering layout received from the cloud network manager 106, the nomination module 316 determines a neighboring network device in which the network device 300 is to store its runtime data. Accordingly, the peer device communication module 314 may provide the runtime data of the network device 300 to the identified neighboring network device, i.e., a buffering device corresponding to the network device 300.

**[0050]** The network device 300 too sends its runtime data to its buffering device based on the predefined frequency defined for the runtime data. In an example, the predefined frequency may be a default value that may be preconfigured for the network device 300 for a given runtime data. In some cases, the cloud network manager 106 may set a predefined frequency for each runtime data. Thus, based on the predefined frequency, the network device 300 may send information relating to its remaining buffer space to its corresponding buffering device every second while information relating to OSPF event may be shared every couple of minutes.

**[0051]** As mentioned earlier, in to some example implementations of the present subject matter, the network device 300 may create the buffering layout in consultation with the other network devices 102. For the purpose, the peer device communication module 314 shares information regarding its currently available storage capacity with the other network devices 102. The peer device communication module 314 also communicates with the other network devices 102 to learn about the currently available storage capacity of each of the other network devices 102. The information relating to currently available storage capacity of the other network devices 102 is stored as the storage capacity data 222. As evident, the storage capacity data 222 is refreshed frequently.

**[0052]** Based on the currently available storage capacity of the other network devices 102, the nomination module 316 may identify one or more network devices as its buffering devices. In a similar manner, the network device 300 may also identify buffering devices corresponding to each of the other network devices 102. Alternatively, each of the other network devices 102

may interact amongst themselves to identify their respective buffering devices. Thus, a mapping of at least one buffering devices corresponding to each of the network device 300 and the other network devices 102, i.e., the buffering layout, is created.

**[0053]** In example implementations where the network device 300 does not receive the buffering layout by the cloud network manager 106, rather creates the same using the nomination module 316, the network manager communication module 312 shares the buffering layout with the cloud network manager 106. This enables the cloud network manager 106 to identify a buffering device corresponding to a given network device for retrieving the runtime data of the failed network device when the network device fails.

**[0054]** Figure 4, 5, and 6 illustrate methods 400, 500, and 600, respectively, for managing failures of network devices in a CMN, in accordance with different examples of the present subject matter. The order in which the methods are described is not intended to be construed as a limitation, and any number of the described method blocks can be combined in any order to implement the methods, or alternative methods. Additionally, individual blocks may be deleted from the methods without departing from the spirit and scope of the subject matter described herein. Furthermore, the methods can be implemented in any suitable hardware, software, firmware, or combination thereof.

**[0055]** A person skilled in the art will readily recognize that steps of the methods can be performed by programmed computing devices. Herein, some examples are also intended to cover program storage devices, for example, digital data storage media, which are machine or computer readable and encode machine-executable or computer-executable programs of instructions, wherein said instructions perform some or all of the steps of the described method. The program storage devices may be, for example, digital memories, magnetic storage media, such as a magnetic disks and magnetic tapes, hard drives, or optically readable digital data storage media. The examples are also intended to cover both communication network and communication system configured to perform said steps of the example method.

**[0056]** Further, although the methods 400, 500, and 600 for managing failure of network devices in a CMN may be implemented in a variety of communication network environments, the examples described in Figure 4, 5, and 6 are explained in context of the aforementioned CMN 100 implementing network devices N1 through N8 for the ease of understanding.

**[0057]** Figure 4 describes the method 400 for managing failure of network devices in a local area network managed by a cloud network manager, from the perspective of a network device that has been nominated to store the runtime data of other network devices in the CMN. Referring to the example illustrated in Figure 1, the method 400 may be understood to be implemented in the network device N4 of the CMN 100.

**[0058]** At block 402, a first network device, from amongst a plurality of network devices in the local area network managed by the cloud network manager is nominated. As explained previously, the nomination may be performed by the cloud network manager or by one or more of the plurality of network devices.

**[0059]** At block 404, the first network device stores the runtime data of each of the plurality of network devices and at block 406, in case any one of the plurality of network devices fails, the first network device provides the stored runtime data corresponding to the failed network device, to the cloud network manager. The runtime data of a network device provides for diagnosis of the network device upon failure of the network device. Accordingly, the cloud network manager has access to the runtime data of the failed network device to investigate events that may have occurred in the network device resulting in the failure of the network device.

**[0060]** Referring to the foregoing example of the network device N4 for explanation, when the network device N4 is nominated, the network device N4 stores the runtime data of the other network device N1, N2, and N3 of the first local area network 104-1 while simultaneously storing its own runtime data in a neighboring network device, such network device N1, N2, and N3 in the first local area network 104-1 or network device N5 in the second local area network 104-2. Thus, the runtime data of each of the network device N1, N2, N3, and N4

is retrievable by the cloud network manager from another network device in the CMN in case of a network device failure.

**[0061]** Figure 5 describes method 500 for managing failure of network devices in the local area network managed by the CMN, from the perspective of a network device that stores its runtime data in another network device in the CMN, herein referred to a buffering device. In an example, the network device may identify the buffering device based on a buffering layout of the CMN as explained below.

**[0062]** At block 502, the network device determines the currently available storage capacity of each of the plurality of network devices in the local area network. Based on the determined currently available storage capacity, at block 504, the network device creates the buffering layout. The buffering layout is indicative of at least one buffering device corresponding to each of the plurality of network devices in the local area network. The process of creation of the buffering layout by a network device in consultation of other network devices is explained earlier and is not repeated here to avoid redundancy. The buffering layout so created is communicated to the cloud network manager at block 506 to make the cloud network manager aware of a buffering device corresponding to each of the network devices.

**[0063]** At block 508, the network device identifies a first buffering device and a second buffering device based on the buffering layout. The first and the second buffering device are to store the runtime data of the network device such that the runtime data is available for the cloud network manager for diagnosis of the network device in an event that the network device is rendered non-function.

**[0064]** At block 510, the network device stores its runtime data in the first buffering device identified based on the buffering layout at block 508. At block 512, the network device may determine the first buffering device to have failed. In an example, if the network device sends runtime data to the first buffering and does not receive any acknowledgement from the first buffering device within a predefined response time; the network device may determine the first buffering device to have failed. Accordingly, at block 514, the network device may start

storing its runtime data in the second buffering device identified, at block 508, based on the buffering layout.

**[0065]** Although the above description has been explained with reference to the first buffering device and the second buffering device, it will be appreciated that the concepts explained in context of the first and the second first buffering device may be extended to any number of buffering devices in the CMN in a similar manner. For example, there may be more than two buffering devices corresponding to each network device. Further, in an example implementation, each of these two or more buffering devices may also concurrently store the runtime data of the network device.

**[0066]** Reference is now made to Figure 6 that illustrates the method 600 that may be implemented in a cloud network manager for managing failure of network devices managed by the cloud network manager. In an example, the method 600 may be understood to be implemented in the previously described cloud network manager 106.

**[0067]** At block 602 the cloud network manager determines the currently available storage capacity of each of the plurality of network devices in the local area network. The currently available storage capacity of a network device is based on the amount of traffic being handled by the network device. In an example, the cloud network manager may estimate the amount of traffic each of the plurality of network devices may be handling to determine their respective currently available storage capacity.

**[0068]** The cloud network manager also determines historic data relating to performance of each of the plurality of network devices at block 604. Based on the information collected at block 602 and 604, at block 606, the cloud network manager creates a buffering layout which is indicative of at least one buffering device corresponding to each of the plurality of network devices in the local area network. Multiple buffering devices may be nominated for each of the network devices as a fail-safe option. In some cases, higher number of buffering devices may be nominated for network devices that are found to be more prone to failure based on their historic data compared to other network devices that are relatively robust.

**[0069]** At block 608, the cloud network manager communicates the buffering layout to each of the network devices to enable the network devices to identify their corresponding buffering devices. In operation, based on the buffering layout provided by the cloud network manager, the network devices start buffering their runtime data in their respective buffering devices. In the course of operation of the network devices, at block 610, the cloud network manager may identify a failed network device in the local area network. Accordingly, at block 612, the cloud network manager retrieves the runtime data of the failed network device from a buffering device corresponding to the failed network device. The runtime data is indicative of reasons of failure of the failed network device. At block 614, cloud network manager uses the runtime data of the failed network device, retrieved from the corresponding buffering device, to determine corrective measure that may be taken for the failed network device.

**[0070]** Thus, the cloud network manager is enabled to take measures for managing failed network devices based on the runtime data of the failed network devices retrieved from their respective buffering devices. This approach involves no human intervention to retrieve the runtime data of the failed network devices and consumes significantly low Internet bandwidth.

**[0071]** The words 'during', 'while', 'when', and 'upon' as used herein are not exact terms that mean as action takes place instantly upon an initiating action but that there may be some small but reasonable delay, such as propagation delay, between the initial action and the reaction that is initiated by the initial action. Additionally, the word 'coupled' is used throughout for clarity of the description and can include either a direct coupling or an indirect coupling.

**[0072]** Figure 7 illustrates an example network environment 700 implementing a non-transitory computer readable medium for managing failure of network devices in a CMN, in accordance with an example of the present subject matter. The network environment 700 may be a public networking environment or a private networking environment. In one implementation, the network environment 700 includes a processing resource 702 communicatively coupled to a non-transitory computer readable medium 704 through a communication link 706.

**[0073]** For example, the processing resource 702 can be a processor of a network device, such as the network device 300. The non-transitory computer readable medium 704 can be, for example, an internal memory device or an external memory device. In one implementation, the communication link 706 may be a direct communication link, such as one formed through a memory read/write interface. In another implementation, the communication link 706 may be an indirect communication link, such as one formed through a network interface. In such a case, the processing resource 702 can access the non-transitory computer readable medium 704 through a network 708. The network 708 may be a single network or a combination of multiple networks and may use a variety of different communication protocols.

**[0074]** The processing resource 702 and the non-transitory computer readable medium 704 may also be communicatively coupled to data sources 710 over the network 708. The data sources 710 can include, for example, databases and computing devices. The data sources 710 may be used by an administrator of the CMN and other users to communicate with the processing resource 702.

**[0075]** In one implementation, the non-transitory computer readable medium 704 includes a set of computer readable instructions, such as instructions for implementing the nomination module 316 and the peer device communication module 314. The set of computer readable instructions, referred to as instructions hereinafter, can be accessed by the processing resource 702 through the communication link 706 and subsequently executed to perform acts for managing failure of network devices in the CMN.

**[0076]** For discussion purposes, the execution of the instructions by the processing resource 702 has been described with reference to various components introduced earlier with reference to description of the foregoing Figures.

**[0077]** In an example, the instructions can cause the processing resource 702 to identify a first buffering device in the CMN to store runtime data of a network device and to provide the runtime data to the first buffering device. The runtime data is retrievable by a cloud network manager of the CMN to diagnose

reasons for failure of the network device in case the network device is rendered non-functional. To elaborate with an example, the processing resource 702 may be considered to be the processor 302 of the network device 300. The instructions may cause the processor 302 to implement the nomination module 316 to identify at least one buffering devices and to implement the peer device communication module 314 to store runtime data of the network device 300 in the first buffering device.

**[0078]** In one example implementation, the instructions may cause the processing resource 702 to receive an indication, from the cloud network manager of the CMN, to identify the first buffering device and store the runtime data in the first buffering device. In a similar manner, the instructions may also cause the processing resource 702 to receive an indication for identification of additional buffering devices, such as a second buffering device that may be a back-up to the first buffering device. Referring again to the example of the network device 300, the first and the second buffering device may be identified by the nomination module 316 based on the buffering layout provided to the network device 300 by the cloud network manager 106 of the network device 300. Thus, in one example, the buffering layout may be considered as the indication, from the cloud network manager, for identification of the buffering devices.

**[0079]** In some cases, the instruction may cause the processing resource 702 to store the runtime data in the second buffering device in addition to storing the same in the first buffering device. However, in some cases the instructions may cause the processing resource 702 save the runtime data in the second buffering device if the first buffering device fails. For the purpose, the instructions may cause the processing resource 702 to identify failure of the first buffering device, for example, based on non-receipt of acknowledge from the first buffering device, for previously sent runtime data.

**[0080]** In an approach that may be implemented as an alternate to the example implementations where the indication to identify the buffering devices is received from the cloud network controller, the instructions may cause the processing resource 702 to determine free memory space in neighbouring

network devices that are coupled to the network device and to identify the buffering devices based on the free memory space. Thus, in one example, a neighbouring network device that has the maximum free memory space available is selected as the first buffering device. Likewise, a neighbouring network device, besides the first buffering device, that has the maximum free memory space is chosen as the second buffering device.

**[0081]** Once the buffering devices are identified, the the runtime data of the network device is shared with the buffering device. In an example implementation, runtime data of the network device is shared with a buffering device at a predefined time period which is based on a relevance level of the runtime data. As explained previously, the relevance level may be defined based on a relevance of the runtime data for diagnosis of the network device. For example, if a runtime data, such as an error message is generated in a network device, it may be shared with the buffering device instantaneously or before lapse of a relatively small predefined time period from the instant of generation of the error message. However, in case the runtime data is not an error message but rather an information message, for example, a state change information message, it may be communicated to the buffering device within a relatively longer predefined time period.

**[0082]** In some examples, the relevance level of a runtime data is dependent on performance parameters of the network device wherein the performance parameter is indicative of a performance of the network device. Examples of performance parameters may include system memory usage, central processing (CPU) usage and utilization of packet buffer of the network device. The predefined time period for sharing the runtime data may be in inverse proportion to the performance of the network device as indicated by the performance parameters. As will be understood, when the performance parameters indicate that the network device is performing poorly, the chances of failure of the network device is higher thereby making the runtime data more relevant for the diagnosis.

**[0083]** For instance, in case the system memory or the CPU usage of the network device is higher than a predefined threshold, the likelihood that the

network device may crash due to overload is high. Accordingly, the relevance level of runtime data that is generated when the system memory or the CPU usage is above threshold is more than that generated when the system memory or the CPU usage is normal. Thus, the predefined time period for sharing the runtime data is less in former case. In a similar example, the predefined time period for sharing the runtime data at instants when the utilization of packet buffer of the network device is 95% is significantly low compared to when the utilization is 15%.

**[0084]** The instructions thus cause the processing resource 702 to determine a relevance level associated with the runtime data being shared with the buffering devices, for example, based on the performance parameters of the network device. Based on the relevance level, the instructions cause the processing resource 702 to ascertain the predefined frequency of sharing the runtime data and provide the runtime data to the buffering device accordingly.

**[0085]** Defining different predefined time periods for the runtime data based on their relevance in diagnosis of a network device allows management plane data to be utilized efficiently. Thus, the methods and systems of the present subject matter provide for minimizing the communication network bandwidth consumption involved in managing network device failures in a CMN without adding to any overheads in the management plane of the CMN.

**[0086]** Although implementations for a CMN have been described in language specific to structural features and/or methods, it is to be understood that the appended claims are not necessarily limited to the specific features or methods described. Rather, the specific features and methods are disclosed as example implementations for managing network device failures in a CMN.

I/We claim:

1. A method comprising:

nominating a first network device, from amongst a plurality of network devices in a local area network managed by a cloud network manager;

storing runtime data of each of the plurality of network devices in the first network device, wherein the runtime data of a network device provides for diagnosis of the network device upon failure of the network device; and

providing, the stored runtime data corresponding to at least one of the plurality of network devices, to the cloud network manager upon failure of the at least one of the plurality of network devices.

2. The method as claimed in claim 1 further comprising:

identifying a second network device, from amongst the plurality of network devices, the second network device being a back-up to the first network device to store the runtime data of each of the plurality of network devices.

3. The method as claimed in claim 1, wherein the nominating comprises ascertaining currently available storage capacity of each of the plurality of network devices.

4. The method as claimed in claim 1 further comprising:

storing the runtime data of the first network device in a neighboring network device, wherein the neighboring network device is from amongst the plurality of network devices in the local area network and a network device in another local area network managed by the cloud network manager,

wherein the runtime data of the first network device is retrievable from the neighboring network device on failure of the first network device.

5. The method as claimed in claim 1 wherein the storing comprises receiving the runtime data from each of the plurality of network devices at a predefined frequency, wherein the predefined frequency is based on a

relevance of the runtime data for diagnosis of each of the plurality of network devices.

6. A cloud network manager comprising:
  - a processor;
  - a network monitoring module, coupled to the processor, to:
    - identify a failed network device in a local area network managed by the cloud network manager; and
  - a troubleshooting module, coupled to the processor, to:
    - retrieve runtime data of the failed network device from a first buffering device, wherein the runtime data is indicative of reasons of failure of the failed network device, and wherein the first buffering device is a network device coupled to the failed network device.
7. The cloud network manager as claimed in claim 6 further comprising a buffering layout creation module, coupled to the processor, to:
  - determine at least one of a currently available storage capacity of each of a plurality of network devices in the local area network and historic data relating to performance of each of the plurality of network devices in the local area network;
  - create a buffering layout based on at least one of the currently available storage capacity and the historic data, the buffering layout being indicative of at least one buffering device corresponding to each of the plurality of network devices in the local area network; and
  - communicate the buffering layout to each of the plurality of network devices in the local area network, wherein
    - the troubleshooting module is to identify the first buffering device based on the buffering layout.
8. The cloud network manager as claimed in claim 6 further comprising a buffering layout creation module, coupled to the processor, to receive a buffering layout from a network device, from amongst the plurality of network

devices in the local area network, the buffering layout being indicative of at least one buffering device corresponding to each of the plurality of network devices in the local area network, wherein the troubleshooting module is to identify the first buffering device based on the buffering layout.

9. The cloud network manager as claimed in claim 6, wherein the troubleshooting module is to:

determine the first buffering device to be unresponsive;

identify a second buffering device that is to store runtime data of the failed network device, wherein the second buffering device is another network device coupled to the failed network device; and

retrieve the runtime data of the failed network device from the second buffering device.

10. A non-transitory computer-readable medium comprising instructions for managing failure of a network device in a Cloud Managed Network (CMN), executable by a processing resource to:

identify a first buffering device in the CMN to store runtime data relating to operation of the network device, wherein the runtime data is retrievable by a cloud network manager of the CMN to diagnose reasons for failure of the network device in an event of the network device being rendered non-functional; and

provide the runtime data to the first buffering device.

11. The non-transitory computer-readable medium as claimed in claim 10 further comprising instructions executable to:

determine a relevance level associated with the runtime data, the relevance level being based on a relevance of the runtime data for diagnosis of the network device; and

ascertain a predefined time period of providing the runtime data to the first buffering device based on the relevance level.

12. The non-transitory computer-readable medium as claimed in claim 11 further comprising instructions executable to determine the relevance level based on at least one performance parameter, the performance parameter being indicative of a performance of the network device.

13. The non-transitory computer-readable medium as claimed in claim 10 further comprising instructions executable to:

- identify a second buffering device in the CMN to store the runtime data;
- and
- provide the runtime data to the second buffering device.

14. The non-transitory computer-readable medium as claimed in claim 12 further comprising instructions executable to:

- determine free memory space in each of the neighboring network devices in the CMN coupled to the network device; and
- identify the first buffering device and the second buffering device based on the free memory space.

15. The non-transitory computer-readable medium as claimed in claim 12 further comprising instructions executable to:

- receive an indication of the first buffering device and the second buffering device, from the cloud network manager of the CMN, to identify the first buffering device and the second buffering device.

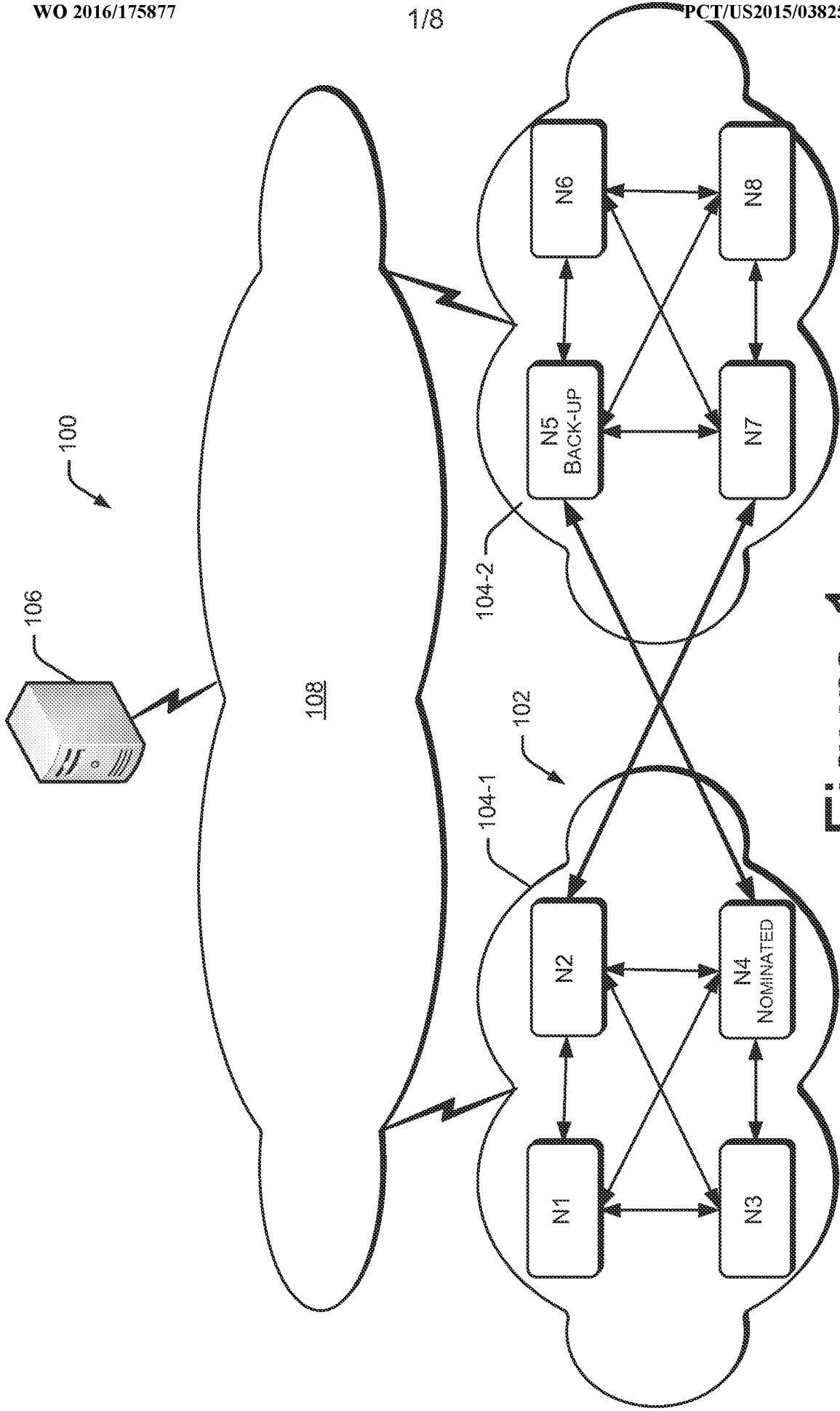


Figure 1

2/8

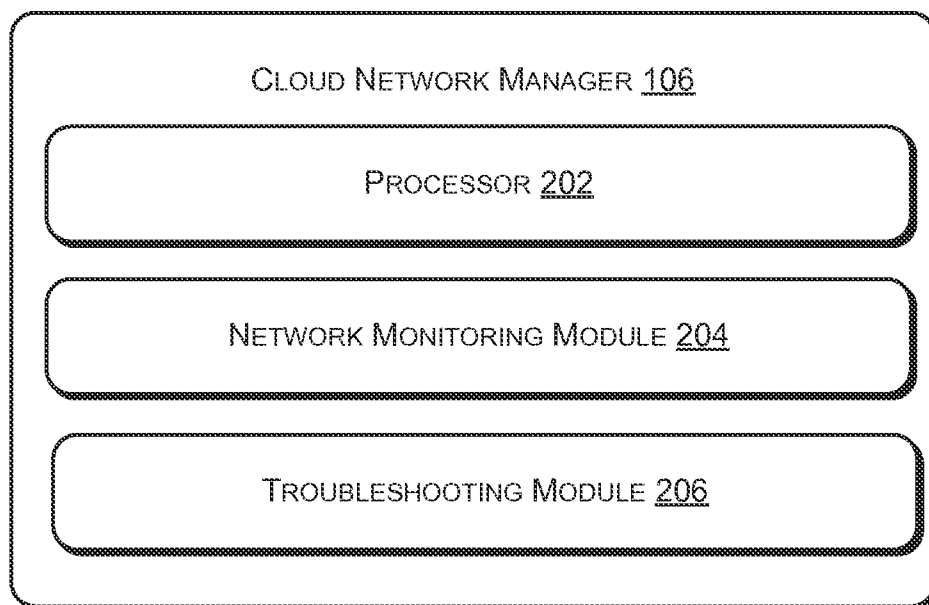
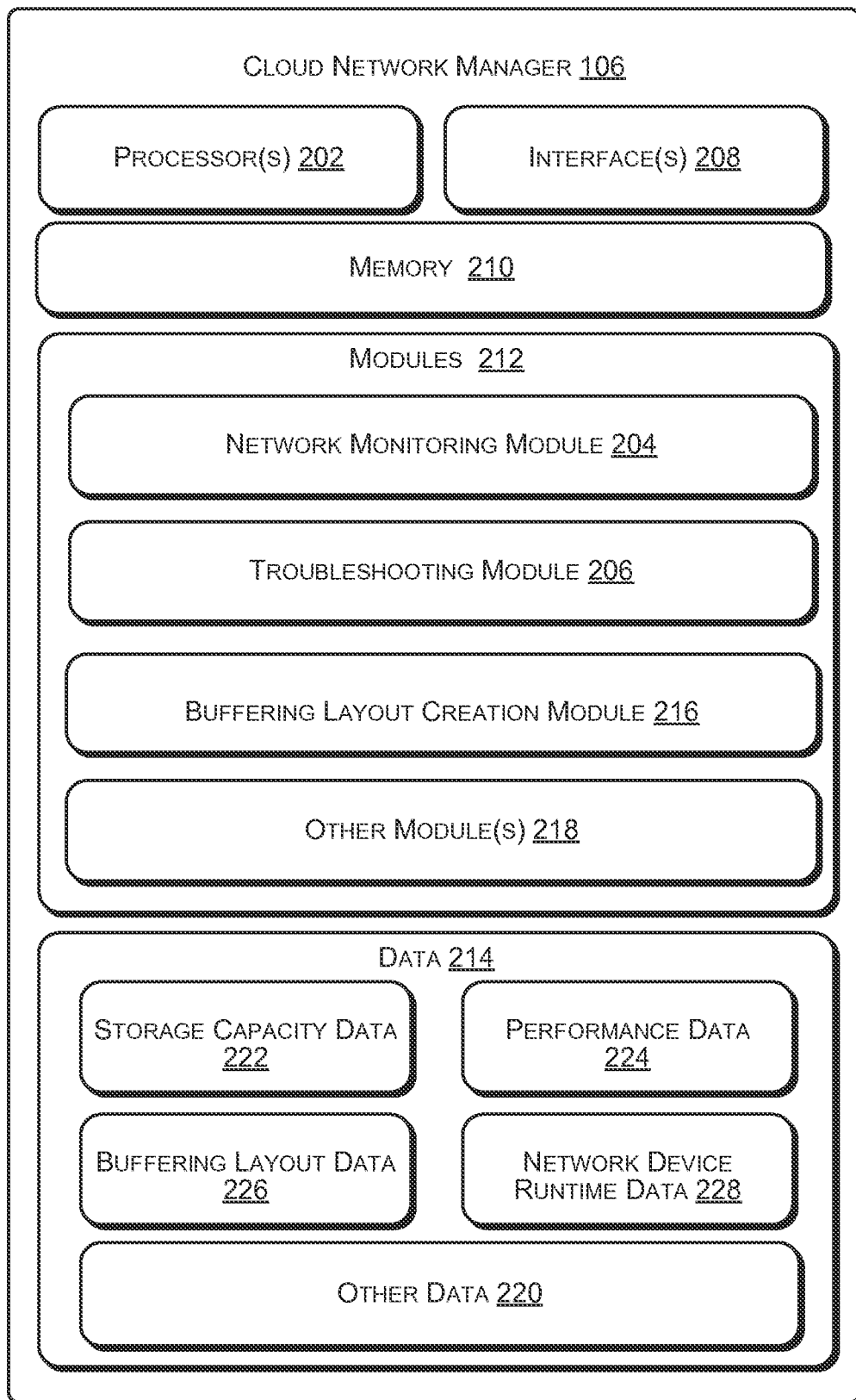
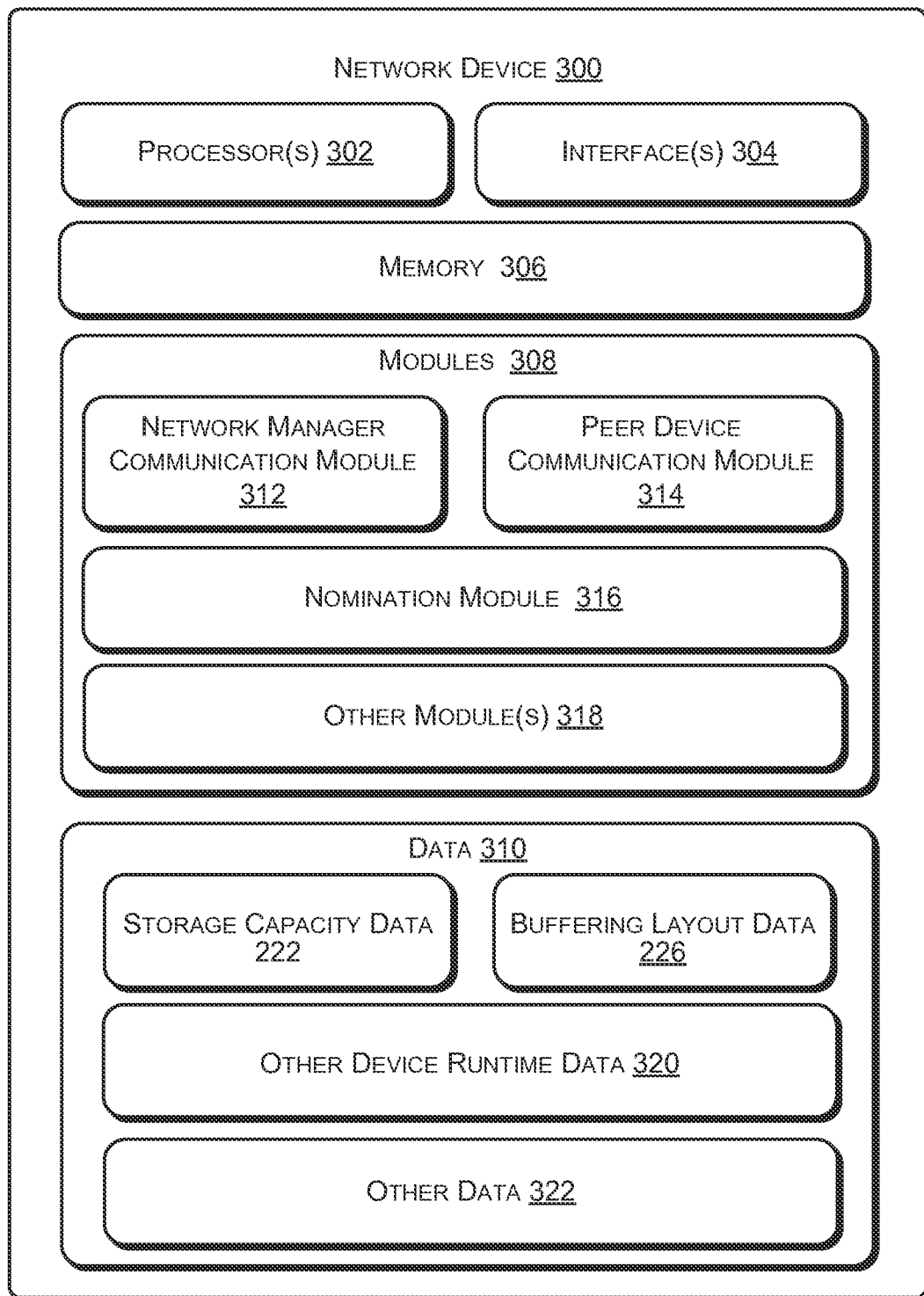


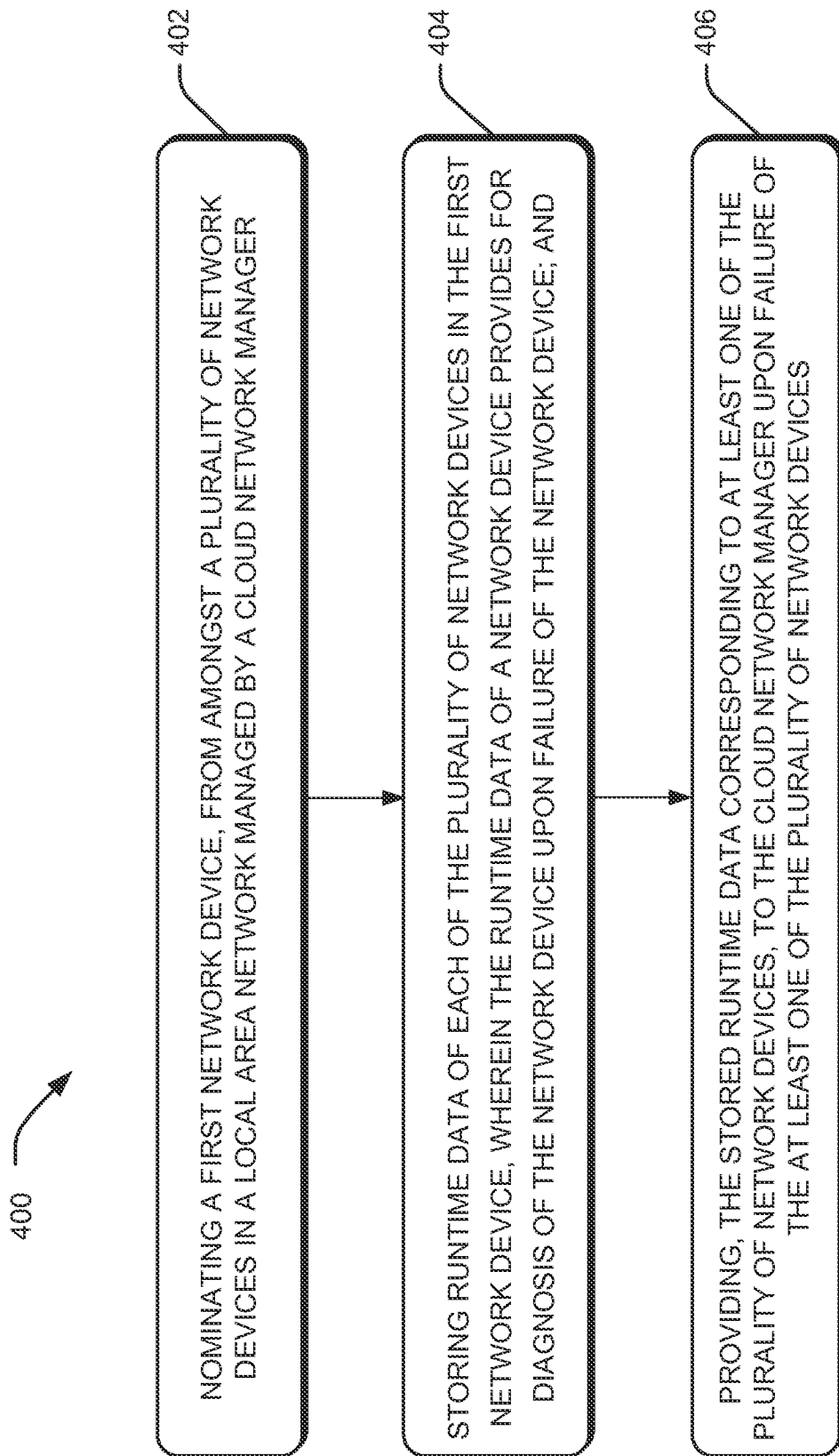
Figure 2A

3/8

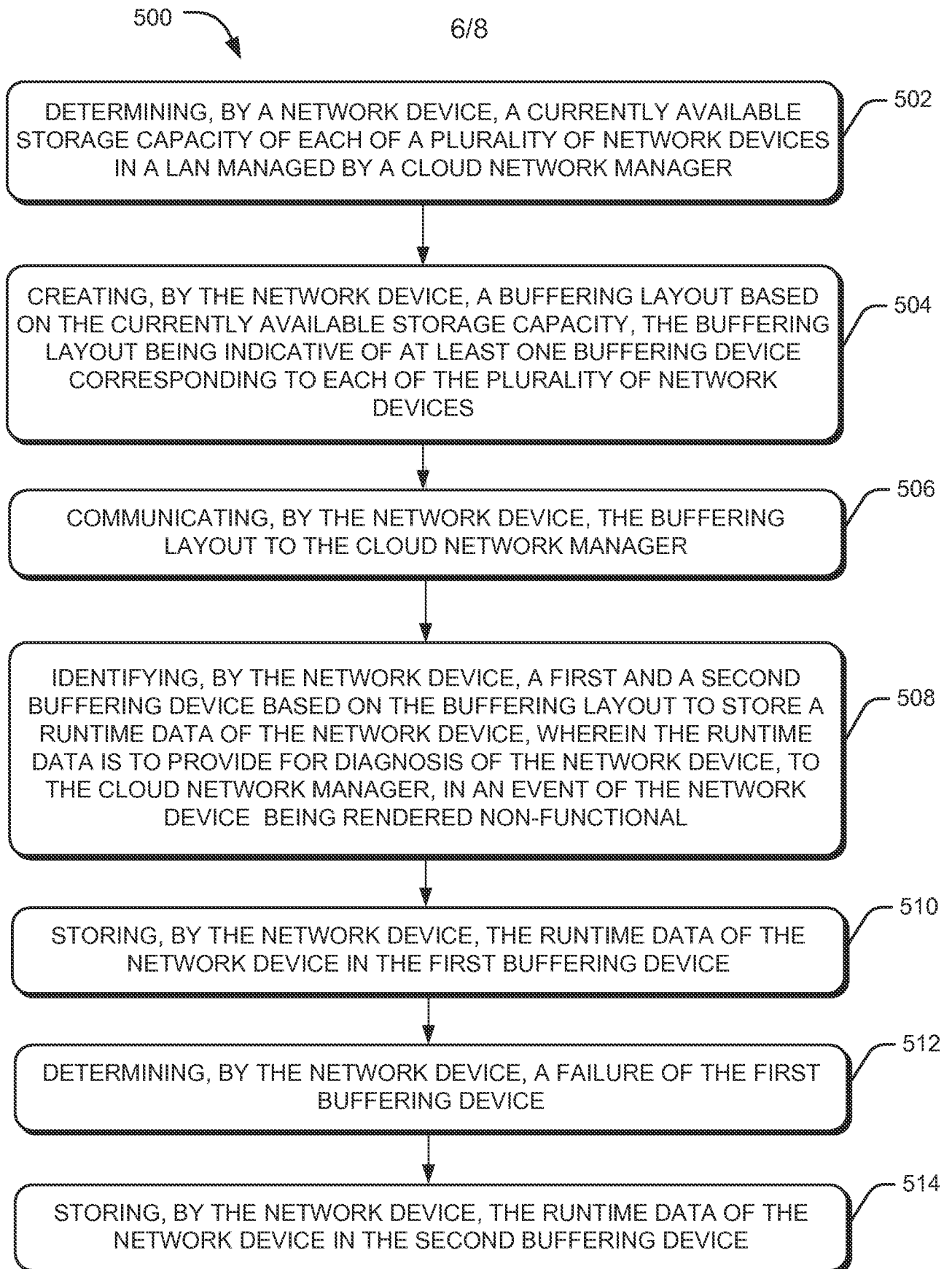
**Figure 2B**

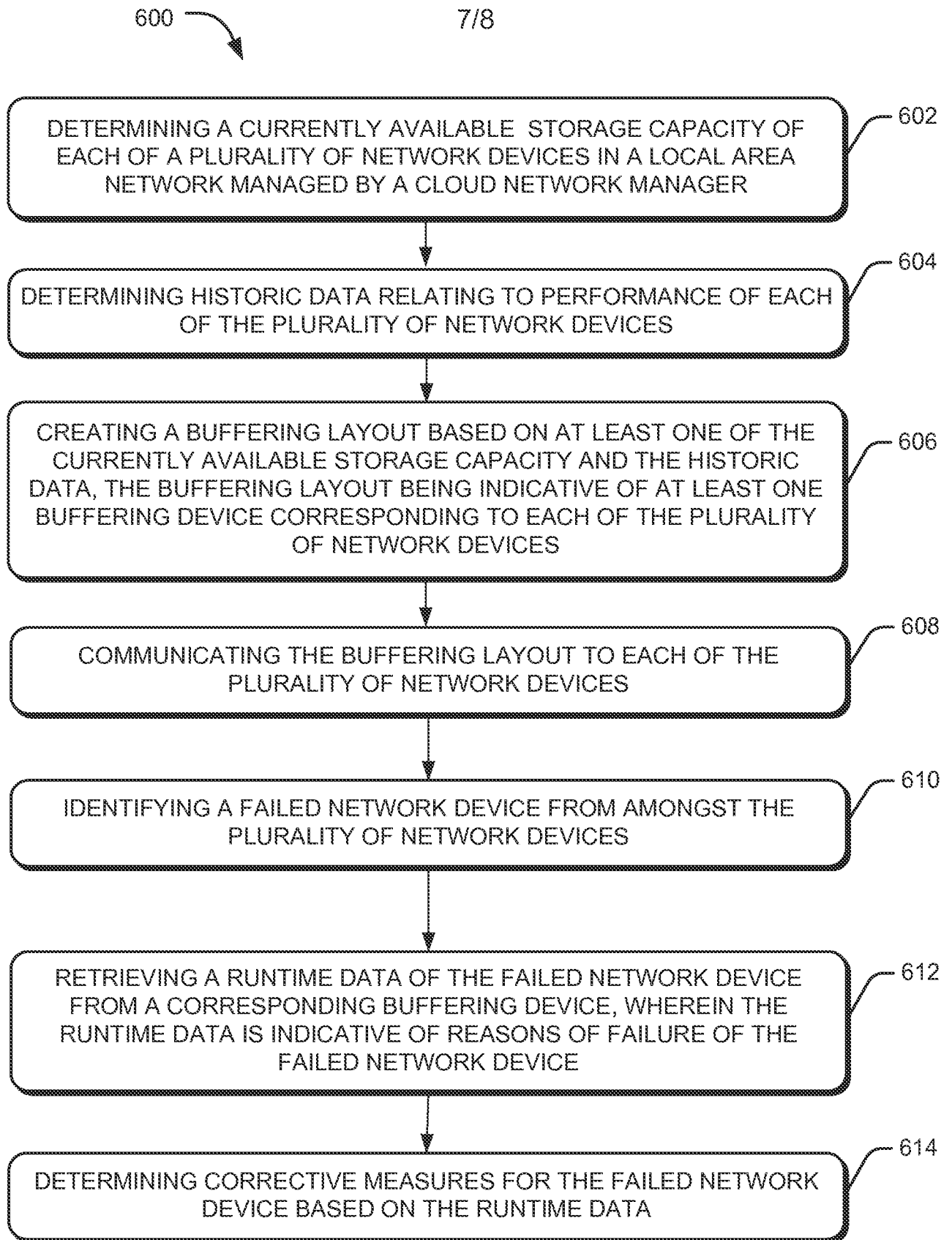
4/8

**Figure 3**



# Figure 4

**Figure 5**



# Figure 6

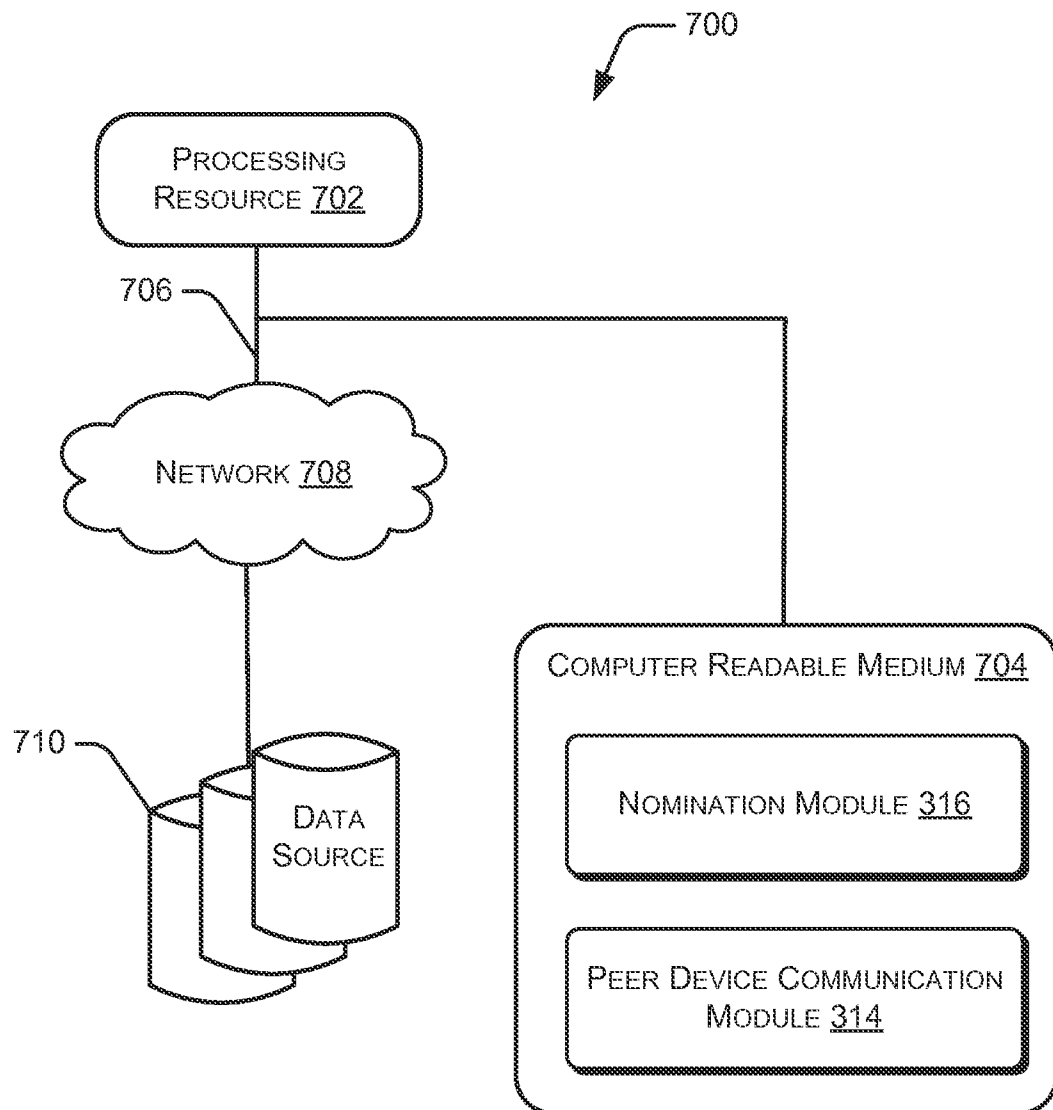


Figure 7

**A. CLASSIFICATION OF SUBJECT MATTER****H04L 12/26(2006.01)i, H04L 12/24(2006.01)i**

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

**B. FIELDS SEARCHED**

Minimum documentation searched (classification system followed by classification symbols)

H04L 12/26; G06F 11/32; H04M 7/00; H04M 3/42; H03K 19/003; G06F 11/36; H02H 3/05; H04L 12/24

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

Korean utility models and applications for utility models

Japanese utility models and applications for utility models

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

eKOMPASS(KIPO internal) &amp; Keywords: runtime, failure, diagnosis, cloud, nominate.

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

| Category* | Citation of document, with indication, where appropriate, of the relevant passages                                                  | Relevant to claim No. |
|-----------|-------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| A         | US 2014-0181592 A1 (MICROSOFT CORPORATION) 26 June 2014<br>See paragraphs [0004], [0010]-[0017]; claim 1; and figures 1-2.          | 1-15                  |
| A         | US 2004-0165710 A1 (SERGIO JASON DELHOYO et al.) 26 August 2004<br>See paragraphs [0006], [0019]; and figure 2.                     | 1-15                  |
| A         | WO 01-022550 A1 (CONCORD COMMUNICATIONS, INC et al.) 29 March 2001<br>See page, 5 line 5 - page 6, line 21; claim 1; and figure 1.  | 1-15                  |
| A         | US 2014-0207942 A1 (INTERNATIONAL BUSINESS MACHINES CO.) 24 July 2014<br>See paragraphs [0041]-[0044]; claims 1-3; and figures 2-4. | 1-15                  |
| A         | US 2002-0162056 A1 (GEORGE H. FORMAN et al.) 31 October 2002<br>See paragraphs [0013]-[0014]; claims 1-2; and figures 1-3.          | 1-15                  |



Further documents are listed in the continuation of Box C.



See patent family annex.

\* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"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

"&amp;" document member of the same patent family

Date of the actual completion of the international search

23 January 2016 (23.01.2016)

Date of mailing of the international search report

**02 March 2016 (02.03.2016)**

Name and mailing address of the ISA/KR

International Application Division

Korean Intellectual Property Office

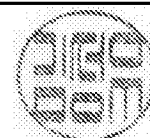
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-472-7140

Authorized officer

KIM, Seong Woo

Telephone No. +82-42-481-3348



**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No.

**PCT/US2015/038258**

| Patent document<br>cited in search report | Publication<br>date | Patent family<br>member(s)                                                                                                                                                                     | Publication<br>date                                                                                                                                    |
|-------------------------------------------|---------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------|
| US 2014-0181592 A1                        | 26/06/2014          | None                                                                                                                                                                                           |                                                                                                                                                        |
| US 2004-0165710 A1                        | 26/08/2004          | US 7010108 B2<br>WO 2004-077237 A2<br>WO 2004-077237 A3                                                                                                                                        | 07/03/2006<br>10/09/2004<br>26/05/2005                                                                                                                 |
| WO 01-22550 A1                            | 29/03/2001          | AU 2001-40220 A1<br>AU 2001-40220 B2<br>AU 4022001 A<br>CA 2385682 A1<br>DE 60032921 T2<br>EP 1222724 A1<br>EP 1222724 A4<br>EP 1222724 B1<br>HK 1048566 A1<br>US 7287192 B1<br>WO 01-22550 A9 | 24/04/2001<br>12/05/2005<br>24/04/2001<br>29/03/2001<br>25/10/2007<br>17/07/2002<br>26/03/2003<br>10/01/2007<br>18/05/2007<br>23/10/2007<br>03/10/2002 |
| US 2014-0207942 A1                        | 24/07/2014          | None                                                                                                                                                                                           |                                                                                                                                                        |
| US 2002-0162056 A1                        | 31/10/2002          | US 06772374 B2                                                                                                                                                                                 | 03/08/2004                                                                                                                                             |