



- (51) **International Patent Classification:**  
*H04L 12/24* (2006.01) *H04L 29/02* (2006.01)
- (21) **International Application Number:**  
PCT/US2014/057642
- (22) **International Filing Date:**  
26 September 2014 (26.09.2014)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (71) **Applicant:** HEWLETT PACKARD ENTERPRISE DEVELOPMENT LP [US/US]; 11445 Compaq Center Drive West, Houston, TX 77070 (US).
- (72) **Inventors:** GIBSON, Jonathan; 14231 Tandem Blvd, Austin, Texas 78728 (US). MILLER, Joseph; 13600 EDS Drive, Herndon, Virginia 20171 (US). WILKE, Clifford A.; 13600 EDS Drive, Herndon, Virginia 20171 (US). GAYDOS, Scott A.; 4707 Canterbury Drive, Emmaus, Pennsylvania 18049 (US).
- (74) **Agent:** LIOU, Lisa W.; 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:**

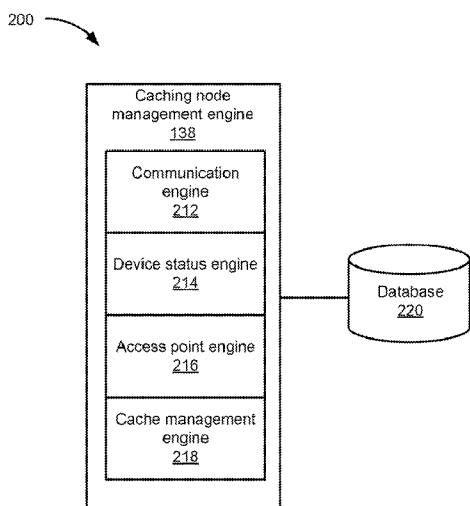
— as to the identity of the inventor (Rule 4.17(i))

**Published:**

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

— with amended claims (Art. 19(1))

(54) **Title:** CACHING NODES



(57) **Abstract:** In examples provided herein, upon receiving notification of data to be cached for a package executed in response to a user-initiated experience, where the user-initiated experience originated from a first networked wearable device (NWD) associated with the user, a caching node management engine identifies one or more caching nodes to cache the data and determines available storage resources for each caching node, where each caching node resides at other NWDs associated with the user. The caching node management engine further transmits different portions of the data to the one or more of the caching nodes for caching based upon available storage resources.

**FIG. 2A**



## CACHING NODES

### BACKGROUND

**[0001]** In many arenas, disparate tools can be used to achieve desired goals. The desired goals may be achieved under changing conditions by the disparate tools.

### BRIEF DESCRIPTION OF THE DRAWINGS

**[0002]** The accompanying drawings illustrate various examples of the principles described below. The examples and drawings are illustrative rather than limiting.

**[0003]** FIG. 1 depicts an example environment in which a context-aware platform that performs remote management of caching nodes may be implemented.

**[0004]** FIG. 2A depicts a block diagram of example components of a caching node management engine.

**[0005]** FIG. 2B depicts a block diagram depicting an example memory resource and an example processing resource for a caching node management engine.

**[0006]** FIG. 3A depicts a block diagram of example components of a caching node, such as at a networked wearable device or access point.

**[0007]** FIG. 3B depicts a block diagram depicting an example memory resource and an example processing resource for a caching node.

**[0008]** FIG. 4 depicts a block diagram of an example context-aware platform.

**[0009]** FIG. 5A depicts a flow diagram illustrating an example process of identifying and selecting one or more caching nodes at a networked wearable device for caching data for use by a package for providing a user experience. FIG. 5B depicts a flow diagram illustrating an example process of registering networked

wearable devices and selecting a caching node at an access point for caching additional data for use by the package.

**[0010]** FIG. 6 depicts a flow diagram illustrating an example process of caching data at a caching node.

**[0011]** FIG. 7 depicts an example system including a processor and non-transitory computer readable medium of a caching node management engine.

**[0012]** FIGS. 8A and 8B depict implementations of example systems including a processor and non-transitory computer readable medium of a caching node.

#### DETAILED DESCRIPTION

**[0013]** As technology becomes increasingly prevalent, it can be helpful to leverage technology to integrate multiple devices, in real-time, in a seamless environment that brings context to information from varied sources without requiring explicit input. Various examples described below provide for a context-aware platform (CAP) that supports remote caching node management of one or more caching nodes, hosted at a networked wearable device (NWD) associated with a user or other device in close proximity to a user's networked devices. The user can be a person, an organization, or a machine, such as a robot. The caching nodes provide storage resources that can allow for faster access to data, for example, as used by computationally intense tasks that provide a seamless experience to the user. As used herein, "CAP experience" and "experience" are used interchangeably and intended to mean the interpretation of multiple elements of context in the right order and in real-time to provide information to a user in a seamless, integrated, and holistic fashion. In some examples, an experience or CAP experience can be provided by executing instructions on a processing resource at a computing node. Further, an "object" can include anything that is visible or tangible, for example, a machine, a device, and/or a substance.

**[0014]** The CAP experience is created through the interpretation of one or more packages. Packages can be atomic components that execute functions related to devices or integrations to other systems. As used herein, "package" is intended to mean components that capture individual elements of context in a given situation. In some examples, the execution of packages provides an experience. For example, a package could provide a schedule or a navigation component, and an experience could be provided by executing a schedule package to determine a user's schedule, and subsequently executing a navigation package to guide a user to the location of an event or task on the user's schedule. As another example, another experience could be provided by executing a facial recognition package to identify a face in an image by comparing selected facial features from the image with data stored in a facial database.

**[0015]** In some examples, the platform includes one or more experiences, each of which correspond to a particular application, such as a user's occupation or a robot's purpose. In addition, the example platform may include a plurality of packages which are accessed by the various experiences. The packages may, in turn, access various information from a user or other resources and may call various services, as described in greater detail below. As a result, the user can be provided with contextual information seamlessly with little or no input from the user. The CAP is an integrated ecosystem that can bring context to information automatically and "in the moment." For example, CAP can sense, retrieve, and provide information from a plurality of disparate sensors, devices, and/or technologies, in context, and without input from a user.

**[0016]** Elements shown in the various figures herein can be added, exchanged, and/or eliminated so as to provide a number of additional examples of the present disclosure. In addition, the proportion and the relative scale of the elements provided in the figures are intended to illustrate the examples of the present disclosure, and should not be taken in a limiting sense.

**[0017]** FIG. 1 depicts an example environment in which a context-aware platform (CAP) 130 may be implemented, where the CAP 130 includes a caching node management engine 138 for managing caching of data at remote caching nodes.

**[0018]** Wearable devices can include any number of portable devices associated with a user of the devices that have a processor and memory and are capable of communicating wirelessly by using a wireless protocol, such as WiFi or Bluetooth. Examples of wearable devices include a smartphone, laptop, smart watch, electronic key fob, smart glass, and any other device or sensor that can be attached to or worn by a user. When a user's wearable devices are configured to communicate with each other, for example, as indicated by wearable device communication network 111 in FIG. 1, the devices are referred to herein as networked wearable devices (NWDs) 110.

**[0019]** Access point 120 can be a standalone access point device; however, examples are not so limited, and access point 120 can be embedded in a stationary device, for example, a printer, a point of sale device, etc. The access point 120 can include a processor and memory configured to communicate with the device in which it is embedded and to communicate with the CAP 130 and/or networked wearable devices 110 within wireless communication range. While only one access point 120 is shown in the example of FIG.1 for clarity, multiple access points can be located within wireless communication range of the one or more NWDs associated with a user.

**[0020]** A caching node used for caching data for a package to provide an experience to a user can reside at a NWD 110 associated with that user or at an access point 120 within wireless communication range of the user's NWDs 110. Each caching node includes components, to be described below, that support caching data for the experience by using the available storage resources of the NWD 110 or access point 120.

**[0021]** In the example of FIG. 1, the CAP 130 can communicate through a network 105 with one or more of the caching nodes at the NWDs 110 and/or a caching node at the access point 120. The network 105 can be any type of network, such as the Internet, or an intranet. The CAP 130 includes a caching node management engine 138, among other components to be described below with reference to FIG. 4. The caching node management engine 138 supports the selection and remote management of caching nodes in close proximity to the user to provide access to data to support providing an experience to the user. The experience can be user-initiated or automatically performed

**[0022]** FIG. 2A depicts a block diagram 200 including example components of a caching node management engine 138. The caching node management engine 138 can include a communication engine 212, a device status engine 214, an access point engine 216, and a cache management engine 218. Each of the engines 212, 214, 216, 218 can access and be in communication with a database 220.

**[0023]** Communication engine 212 may be configured to receive notification of data to be cached as requested by a package for providing an experience to a user.

**[0024]** The device status engine 214 may be configured to register and identify caching nodes at NWDs 110 associated with a user. When data is to be cached to support an experience to be provided to a particular user, the device status engine 214 can determine available storage resources at the caching nodes at each NWD 110 associated with the user and select one or more of the caching nodes to cache portions of the data.

**[0025]** The access point engine 216 may be configured to register and identify access points. Registration information can include a location identifier, such as global positioning (GPS) coordinates. Upon receiving notification of data to be cached for a package for providing an experience to a user, the access point engine 216 may identify one or more suitable access points within wireless communication range of the NWDs 110 associated with the user based on the location of the user. The access point engine 216 can communicate with the appropriately located

access points to determine available storage resources at the respective access points and select one or more access points to cache portions of the data.

**[0026]** The cache management engine 218 may be configured to transmit portions of the data to be cached to one or more caching nodes that reside at a NWD 110. For example, the caching nodes at the NWDs 110 can be used exclusively in a situation when the user is not near any access points, such as when the user is outside.

**[0027]** If the user is near one or more access points 120, for example, inside an office building with caching nodes located at printers or a shopping complex with caching nodes located at point of sale devices, the cache management engine 218 can transmit requests to an access point 120 in close proximity to the user and the user's NWDs 110 to cache portions of the data in addition to, or instead of, at the caching nodes at the NWDs 110. The cache management engine 218 can also transmit to the package requesting caching of the data the location information for the different portions of the data cached at each of the one or more caching nodes at the NWDs 110 and/or the access points 120.

**[0028]** In some instances, the package requesting caching of the data will send an indication that the task using the cached data has been completed, or that the cached data is no longer needed. Then the cache management engine 218 can inform the selected caching nodes that the cached data can be overwritten or deleted from the respective memories.

**[0029]** Database 220 can store data, such as registration information for the NWDs 110 and access points 120.

**[0030]** FIG. 3A depicts a block diagram of example components of a caching node residing at a NWD 110 or access point 120. The caching node can include a node communication engine 302, a caching engine 304, and a security engine 306. Each of engines 302, 304, 306 can interact with a database 310.

**[0031]** Node communication engine 302 may be configured to receive the portion of the data to be cached at the caching node and acknowledge receipt of the data for caching.

**[0032]** The caching engine 304 may be configured to cache the data at a storage resource at the node and respond to requests for cached data.

**[0033]** The security engine 306 may be configured to receive and use authentication information to be used for requests for cached data. Examples of authentication information can include identification information for the package or packages to be allowed to access the data cached at the caching node and a password to be provided with a request for cached data. For example, the security engine 306 can be configured to reject a request for cached data from unauthorized requestors, and allow the caching engine 304 to respond to the request for cached data if the request originates from a previously authorized package. As another example, the security engine 306 can reject a request for cached data if the request includes incorrect passwords, and allow the caching engine 304 to provide the cached data if the request includes a password associated with the cached data, and. The security engine 306 can also be configured to implement different or more stringent security measures for cached data.

**[0034]** Database 310 can store data, such as authentication information for cached data.

**[0035]** The examples of engines shown in FIGS. 2A and 3A are not limiting, as one or more engines described can be combined or be a sub-engine of another engine. Further, the engines shown can be remote from one another in a distributed computing environment, cloud computing environment, etc.

**[0036]** In the above description, various components were described as combinations of hardware and programming. Such components may be implemented in different ways. Referring to FIG. 2B, the programming may be processor executable instructions stored on tangible memory resource 260 and the



hardware may include processing resource 250 for executing those instructions. Thus, memory resource 260 can store program instructions that when executed by processing resource 250, implements caching node management engine 138 of FIG. 2A. Similarly, referring to FIG. 3B, the programming may be processor executable instructions stored on tangible memory resource 360 and the hardware may include processing resource 350 for executing those instructions. So memory resource 360 can store program instructions that when executed by processing resource 350, implements the caching node portion of NWD 110 or access point 120 of FIG. 3A.

**[0037]** Memory resource 260 generally represents any number of memory components capable of storing instructions that can be executed by processing resource 250. Similarly, memory resource 360 generally represents any number of memory components capable of storing instructions that can be executed by processing resource 350. Memory resource 260, 360 is non-transitory in the sense that it does not encompass a transitory signal but instead is made up of one or more memory components configured to store the relevant instructions. Memory resource 260, 360 may be implemented in a single device or distributed across devices. Likewise, processing resource 250 represents any number of processors capable of executing instructions stored by memory resource 260, and similarly for processing resource 350 and memory resource 360. Processing resource 250, 350 may be integrated in a single device or distributed across devices. Further, memory resource 260 may be fully or partially integrated in the same device as processing resource 250, or it may be separate but accessible to that device and processing resource 250, and similarly for memory resource 360 and processing resource 350.

**[0038]** In one example, the program instructions can be part of an installation package that when installed can be executed by processing resource 250 to implement caching node management engine 138 or by processing resource 350 to implement the caching node portion of a NWD 110 or access point 120. In this case, memory resource 260, 360 may be a portable medium such as a compact disc

(CD), digital video disc (DVD), or flash drive or a memory maintained by a server from which the installation package can be downloaded and installed. In another example, the program instructions may be part of an application or applications already installed. Memory resource 260, 360 can include integrated memory, such as a hard drive, solid state drive, or the like.

**[0039]** In the example of FIG. 2B, the executable program instructions stored in memory resource 260 are depicted as communication module 262, device status module 264, access point module 266, and cache management module 268. Communication module 262 represents program instructions that when executed cause processing resource 250 to implement communication engine 212. Device status module 264 represents program instructions that when executed cause processing resource 250 to implement device status engine 214. Access point module 266 represents program instructions that when executed cause processing resource 250 to implement access point engine 216. Cache management module 268 represents program instructions that when executed cause processing resource 250 to implement cache management engine 218.

**[0040]** In the example of FIG. 3B, the executable program instructions stored in memory resource 360 are depicted as node communication module 362, caching module 364, and security module 366. Node communication module 362 represents program instructions that when executed cause processing resource 350 to implement node communication engine 302. Caching module 364 represents program instructions that when executed cause processing resource 350 to implement caching engine 304. Security module 366 represents program instructions that when executed cause processing resource 350 to implement security engine 306.

**[0041]** FIG 4 depicts a block diagram of an example context-aware platform (CAP) 130. The CAP 130 may determine which package among multiple available packages 420 to execute based on information provided by the context engine 456 and the sequence engine 458. In some examples, the context engine 456 can be

provided with information from a device/service rating engine 450, a policy/regulatory engine 452, and/or preferences 454. For example, the context engine 456 can determine what package to execute based on a device/service rating engine 450 (e.g., hardware and/or program instructions that can provide a rating for devices and/or services based on whether or not a device can adequately perform the requested function), a policy/regulatory engine 452 (e.g., hardware and/or program instructions that can provide a rating based on policies and/or regulations), preferences 454 (e.g., preferences created by a user), or any combination thereof. In addition, the sequence engine 458 can communicate with the context engine 456 to identify packages 420 to execute, and to determine an order of execution for the packages 420. In some examples, the context engine 456 can obtain information from the device/service rating engine 450, the policy/regulatory engine 452, and/or preferences 454 automatically (e.g., without any input from a user) and can determine what package 420 to execute automatically (e.g., without any input from a user). In addition, the context engine 456 can determine what package 420 to execute based on the sequence engine 458.

**[0042]** For example, based on information provided to the CAP system 130 from the context engine 456, the sequence engine 458, and the device/service rating engine 450, the experience 410 may call a facial recognition package 422 to perform facial recognition on a digital image of a person's face. In some examples, the experience 410 can be initiated by voice and/or gestures received by a NWD 110 which communicates with the CAP system 130 via network 105 (as shown in FIG. 1) to call the facial recognition package 422, as described above. Alternatively, in some examples, the facial recognition package 422 can be automatically called by the experience 410 at a particular time of day, for example, 10:00 pm, the time scheduled for a meeting with a person whose identity should be confirmed by facial recognition. In addition, the facial recognition package 422 can be called upon determination by the experience 410 that a specific action has been completed, for example, after a digital image has been captured by a digital camera on the NWD 110, such as can be found on a smartphone. Thus, in various examples, the facial

recognition package 422 can be called by the experience 410 without any input from the user. Similarly, other packages 420 that may need the performance of computationally intensive tasks can be called by the experience 410 without any input from the user.

**[0043]** As an example, processing of a task, such as a facial recognition task, can be performed at computing nodes that reside at one or more NWDs 110 associated with the user and/or access points within wireless communication range of the NWDs 110 associated with the user. By selecting computing nodes at the NWDs 110 associated with the user to whom the experience 410 will be provided and access points 120 within close proximity of the NWDs 110, quicker responses to the computationally intense task are obtained in providing the experience 410 to the user because latency in the process is minimized. In contrast, for example, in a centralized computation model in the cloud, the latency in the process can significantly delay the computations.

**[0044]** The storage resources of multiple NWDs and access points near the computing nodes can be used as caching nodes for data needed by a processing task to increase the speed at which the task is performed, such as the facial recognition task. When facial recognition package 422 is executed, it triggers the caching node management engine 138 to call the services 470 to retrieve facial recognition information and/or metadata needed for performing the facial recognition task. The facial recognition information and/or metadata is then transmitted by the caching node management engine 138 via network 105 to the caching nodes selected to cache the information and/or metadata.

**[0045]** Caching data for the facial recognition package 422 is one example in which data can be cached at caching nodes near where a processing task is being performed for a particular user. Other packages can also cache data at caching nodes near computing nodes to speed up a processing task. For example, a user can initiate a shopping experience that may provide a listing of current sales in the store in which the user is located. The shopping experience is provided by a

shopping package 424, and the shopping package 424 can trigger the caching node management engine 138 to call services 470 to retrieve shopping and/or marketing information, such as current sales and advertisements related to the sales. In this example, in addition to caching the shopping and/or marketing information at caching nodes of NWDs associated with the user, the caching node management engine 138 can select a caching node residing at a local point of sale device for caching additional information that might not be capable of being stored at the caching nodes of the NWDs due to limitations in storage resources at the NWDs.

**[0046]** As another example, a checklist package 426 can trigger the caching node management engine 138 to call services 470 to retrieve checklist information, such as used in a service technician's call with multiple actions to be performed on different devices. In this example, in addition to caching the checklist information at caching nodes of NWDs associated with the user, the caching node management engine 138 can select a caching node residing at a local printer for caching additional information that might not be capable of being stored at the caching nodes of the NWDs due to limitations in storage resources at the NWDs.

**[0047]** Although caching resources may be limited at a user's NWDs 110, use of other local caching nodes allow the user to operate essentially in a hands-free mode. Thus, while a user may have a laptop device that has sufficient storage resources to cache any information to be used in a processing task for a package, the user may not want to carry the laptop device and merely rely on the memory available in a smart watch attached to the user's wrist and a smartphone carried on the user's belt for storage resources. Data to be cached that does not fit on the memory resources of the smart watch and smart phone may be cached at a local caching node, such as a printer or point of sale device.

**[0048]** FIG. 5A depicts a flow diagram illustrating an example process 500 of identifying and selecting one or more caching nodes at a networked wearable device for caching data for use by a package for providing a user experience.

**[0049]** At block 505, upon receiving notification of data to be cached for a package executed in response to a user-initiated experience, where the user-initiated experience originated from a first networked wearable device (NWD) associated with the user, the caching node management engine identifies one or more caching nodes to cache the data and determines available storage resources at each caching node. The notification for caching of data for the package can include the user who will be provided the experience and a current location of the user. Each caching node resides at other NWDs associated with the user or access points; thus the selected caching nodes are in close proximity to the user.

**[0050]** At block 510, based upon available storage resources, the caching node management engine transmits different portions of the data to one or more of the caching nodes for caching. In some instances, the caching node of the first NWD can also cache a portion of the data in the memory resources of the local NWD.

**[0051]** FIG. 5B depicts a flow diagram illustrating an example process 550 of registering networked wearable devices and selecting an access node for caching additional data for use by the package.

**[0052]** At block 555, the caching node management engine registers each of the NWDs, where registration information obtained during registration includes an identification of a specific associated user.

**[0053]** Then at block 560, the caching node management engine identifies an access point within wireless communication range of the NWDs based on a location of the user. Next, at block 565, the caching node management engine communicates with the access point to determine available storage resources.

**[0054]** At block 570, the caching node management engine, based upon available storage resources at the access point, transmits a separate portion of the data to the access point for caching.

**[0055]** At block 575, the caching node management engine transmits to the package location information for the different portions of the data cached at each of

the one or more caching nodes and for the separate portions of the data cached at the access point.

**[0056]** Then at block 580, the caching node management engine registers each of the access points, where registration information obtained during registration includes a location identifier for the access point.

**[0057]** FIG. 6 depicts a flow diagram illustrating an example process 600 of caching data at a caching node.

**[0058]** At block 605, the caching node receives and caches a first portion of data for use by a package to provide an experience to a user. The caching node resides at a networked wearable device of the user.

**[0059]** Then at block 610, the caching node authenticates a request for at least a part of the first portion of the data prior to responding to the request. At block 615, the caching node responds to the request by the package by retrieving the first portion of the requested data.

**[0060]** Next, at block 620, the caching node permits the first portion of the data to be overwritten after a contextual change occurs, such as a predetermined period of time elapses from the request with no further requests for the first portion of the data. Alternatively, the caching node can store the first portion of the data until directed by the caching node management engine to discard the first portion of the data, for example, upon completion of the task for which the data was cached.

**[0061]** FIG. 7 illustrates an example system 700 including a processor 703 and non-transitory computer readable medium 781 according to the present disclosure. For example, the system 700 can be an implementation of an example system such as caching node management engine 138 of FIG. 2A.

**[0062]** The processor 703 can be configured to execute instructions stored on the non-transitory computer readable medium 781. For example, the non-transitory computer readable medium 781 can be any type of volatile or non-volatile memory or storage, such as random access memory (RAM), flash memory, or a hard disk.

When executed, the instructions can cause the processor 703 to perform a method of selecting one or more caching nodes to cache data for use by a package to provide an experience to a user.

**[0063]** The example medium 781 can store instructions executable by the processor 703 to perform remote management of caching nodes. For example, the processor 703 can execute instructions 782 to register NWDs associated with a user and determine the available storage resources at the NWDs. In addition, the processor 703 can execute instructions 782 to perform blocks 565 and 580 of the method of FIG. 5B.

**[0064]** The example medium 781 can further store instructions 784. The instructions 784 can be executable to register access points capable of caching data requested by a package and determine the available storage resources at the access points. In addition, the processor 703 can execute instructions 784 to perform block 505 of the method of FIG. 5A and block 555 of FIG. 5B.

**[0065]** The example medium 781 can further store instructions 786. The instructions 786 can be executable to select one or more of the caching nodes for caching data for use by a package to provide an experience to a user.

**[0066]** The example medium 781 can further store instructions 788. The instructions 788 can be executable to transmit portions of data to be cached to one or more caching nodes. In addition, the processor 703 can execute instructions 788 to perform block 510 of the method of FIG. 5A.

**[0067]** FIG. 8A illustrates an example system 800A including a processor 803A and non-transitory computer readable medium 881A according to the present disclosure. For example, the system 800A can be an implementation of an example system such as a caching node 320 of FIG. 3A residing at a NWD 110 or access point 120.

**[0068]** The processor 803A can be configured to execute instructions stored on the non-transitory computer readable medium 881A. For example, the non-



transitory computer readable medium 881A can be any type of volatile or non-volatile memory or storage, such as random access memory (RAM), flash memory, or a hard disk. When executed, the instructions can cause the processor 803 to perform a method of caching data for use by a package to provide an experience to a user.

**[0069]** The example medium 881A can store instructions executable by the processor 803A to cache data at a computing node, such as the method described with respect to FIG. 6. For example, the processor 803A can execute instructions 882A to cache data for a package. In addition, the processor 803A can execute instructions 882A to perform block 605 of the method of FIG. 6.

**[0070]** The example medium 881A can further store instructions 884A. The instructions 884A can be executable to respond to a request for cached data. Additionally, the processor 803A can execute instructions 884A to perform block 615, of the method of FIG. 6.

**[0071]** FIG. 8B illustrates an example system 800B including a processor 803B and non-transitory computer readable medium 881B according to the present disclosure. For example, the system 800B can be another implementation of an example system such as a caching node 320 of FIG. 3A residing at a NWD 110 or access point 120.

**[0072]** The processor 803B can be configured to execute instructions stored on the non-transitory computer readable medium 881B. For example, the non-transitory computer readable medium 881B can be any type of volatile or non-volatile memory or storage, such as random access memory (RAM), flash memory, or a hard disk. When executed, the instructions can cause the processor 803B to perform a method of caching data for use by a package to provide an experience to a user.

**[0073]** Similar to the example of FIG. 8A, the example medium 881B can store instructions executable by the processor 803B to cache data at a computing node,

such as the method described with respect to FIG. 6. For example, the processor 803B can execute instructions 882B to cache data for a package, and instructions 884B can be executable to respond to a request for cached data.

**[0074]** The example medium 881B can further store instructions 886B. The instructions 886B can be executable to authenticate a request for cached data. In addition, the processor 803B can execute instructions 886B to perform block 610 of the method of FIG. 6.

**[0075]** The example medium 881B can further store instructions 888B. The instructions 888B can be executable to overwrite cached data. In addition, the processor 803B can execute instructions 888B to perform block 620 of the method of FIG. 6.

**[0076]** Not all of the steps, features, or instructions presented above are used in each implementation of the presented techniques.

## CLAIMS

What is claimed is:

1. A system comprising:
  - a communication engine to receive notification of data to be cached for a package to provide an experience to a user;
  - a device status engine to determine one or more caching nodes for caching the data, wherein each caching node resides at a networked wearable device (NWD) associated with the user;
  - and
  - a cache management engine to transmit different portions of the data to the one or more caching nodes for caching.
  
2. The system of claim 1, further comprising:
  - an access point engine to determine one or more additional caching nodes for caching the data,
  - wherein each additional caching node resides at an access point, and each access point is within wireless communication range of the NWDs, and
  - wherein the cache management engine further transmits additional portions of the data to each of the one or more additional caching nodes for caching.
  
3. The system of claim 2, wherein the access point engine is further to register each of the access points, wherein registration information includes a location identifier for the access point.

4. The system of claim 1, wherein the communication engine further transmits to the package location information for the different portions of the data and the additional portions of the data.
5. The system of claim 1, wherein the device status engine is further to register each of the NWDs, wherein registration information includes identification of a specific associated user.
6. A method comprising:
  - upon receiving notification of data to be cached for a package executed in response to a user-initiated experience, wherein the user-initiated experience originated from a first networked wearable device (NWD) associated with the user, identifying one or more caching nodes to cache the data and determining available storage resources at each caching node, wherein each caching node resides at other NWDs associated with the user; and
  - based upon available storage resources, transmitting different portions of the data to the one or more of the caching nodes for caching.
7. The method of claim 6, further comprising:
  - registering each of the NWDs, wherein registration information includes an identification of a specific associated user.
8. The method of claim 6, further comprising:
  - based on a location of the user, identifying an access point within wireless communication range of the NWDs;
  - communicating with the access point to determine available storage resources; and

based upon available storage resources at the access point, transmitting a separate portion of the data to the access point for caching.

9. The method of claim 8, further comprising transmitting to the package location information for the different portions of the data cached at each of the one or more caching nodes and for the separate portions of the data cached at the access point.

10. The method of claim 9, further comprising registering each of the access points, wherein registration information includes a location identifier for the access point.

11. The method of claim 9, wherein the access point is embedded in at least one of: a printer and a point of sale device.

12. A non-transitory computer readable medium storing instructions executable by a processing resource of a caching node to:

cache a first portion of data for use by a package to provide an experience to a user; and

respond to a request by the package for at least a part of the first portion of the data,

wherein the caching node resides at a networked wearable device of the user.

13. The non-transitory computer readable medium of claim 12, wherein the stored instructions further cause the processing resource to:

authenticate the request for the at least a part of the first portion of the data prior to responding to the request.

14. The non-transitory computer readable medium of claim 12, wherein the stored instructions further cause the processing resource to:

    permit the first portion of the data to be overwritten after a contextual change occurs.

15. The non-transitory computer readable medium of claim 12, wherein a different portion of the data is cached at an access point embedded in at least one of: a printer and a point of sale device.

## AMENDED CLAIMS

received by the International Bureau on 14 July 2015 (14.07.2015)

What is claimed is:

1. A system comprising:
  - a communication engine to receive notification of data to be cached for a package to provide an experience to a user;
  - a device status engine to determine one or more caching nodes for caching the data, wherein each caching node resides at a networked wearable device (NWD) associated with the user; and
  - a cache management engine to transmit different portions of the data to the one or more caching nodes for caching,wherein at least a portion of the data cached at the determined one or more caching nodes is used by a computing node at a NWD associated with the user, and  
further wherein the NWDs associated with the user communicate with each other via a wearable device communication network.
2. The system of claim 1, further comprising:
  - an access point engine to determine one or more additional caching nodes for caching the data,wherein each additional caching node resides at an access point, and each access point is within wireless communication range of the NWDs, and

- wherein the cache management engine further transmits additional portions of the data to each of the one or more additional caching nodes for caching.
3. The system of claim 2, wherein the access point engine is further to register each of the access points, wherein registration information includes a location identifier for the access point.
  4. The system of claim 1, wherein the communication engine further transmits to the package location information for the different portions of the data and the additional portions of the data.
  5. The system of claim 1, wherein the device status engine is further to register each of the NWDs, wherein registration information includes identification of a specific associated user.
  6. A method comprising:
    - upon receiving notification of data to be cached for a package executed in response to a user-initiated experience, wherein the user-initiated experience originated from a first networked wearable device (NWD) associated with the user, identifying one or more caching nodes to cache the data and determining available storage resources at each caching node, wherein each caching node resides at other NWDs associated with the user; and
    - based upon available storage resources, transmitting different portions of the data to the one or more of the caching nodes for caching,



- wherein the NWDs associated with the user communicate with each other via a wearable device communication network.
7. The method of claim 6, further comprising:  
registering each of the NWDs, wherein registration information includes an identification of a specific associated user.
  8. The method of claim 6, further comprising:  
based on a location of the user, identifying an access point within wireless communication range of the NWDs;  
communicating with the access point to determine available storage resources; and  
based upon available storage resources at the access point, transmitting a separate portion of the data to the access point for caching.
  9. The method of claim 8, further comprising transmitting to the package location information for the different portions of the data cached at each of the one or more caching nodes and for the separate portions of the data cached at the access point.
  10. The method of claim 9, further comprising registering each of the access points, wherein registration information includes a location identifier for the access point.
  11. The method of claim 9, wherein the access point is embedded in at least one of: a printer and a point of sale device.

12. A non-transitory computer readable medium storing instructions executable by a processing resource of a caching node to:

cache a first portion of data for use by a package to provide an experience to a user; and  
respond to a request by the package for at least a part of the first portion of the data,  
wherein the caching node resides at one of a plurality of networked wearable device of the user,  
and further wherein the plurality of NWDs communicate with each other via a wearable device communication network.

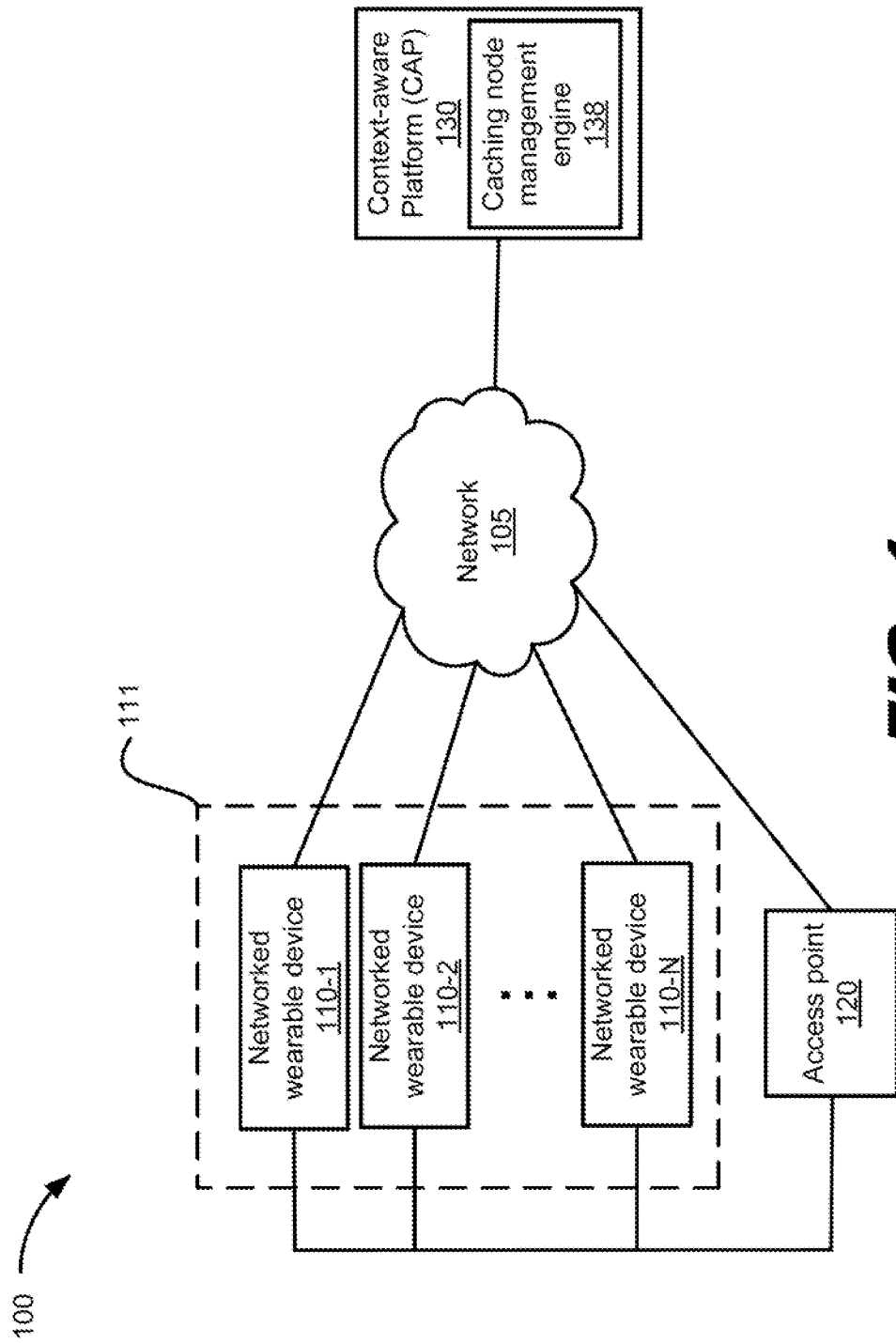
13. The non-transitory computer readable medium of claim 12, wherein the stored instructions further cause the processing resource to:

authenticate the request for the at least a part of the first portion of the data prior to responding to the request.

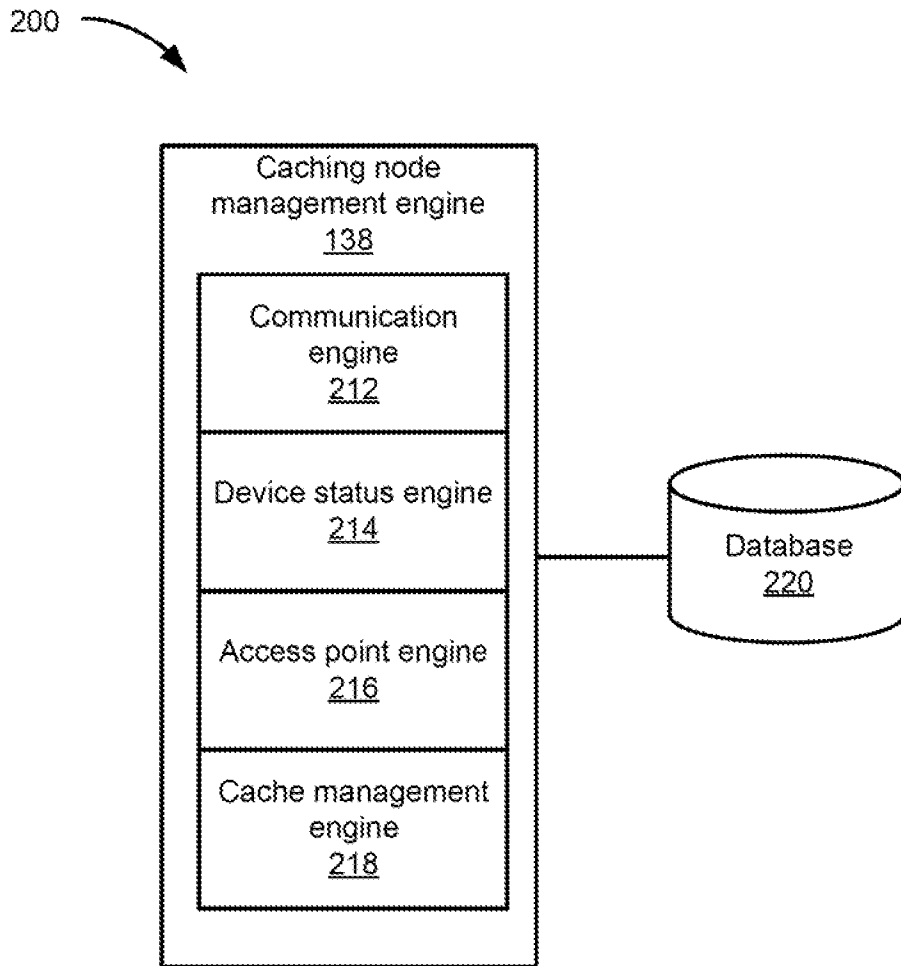
14. The non-transitory computer readable medium of claim 12, wherein the stored instructions further cause the processing resource to:

permit the first portion of the data to be overwritten after a contextual change occurs.

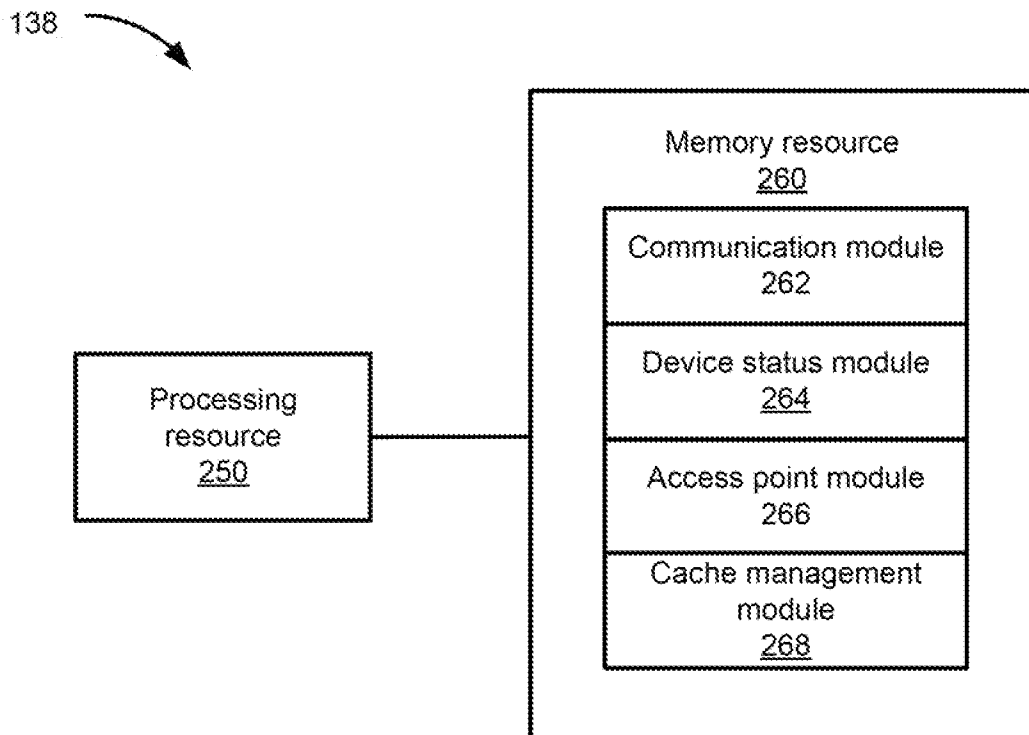
15. The non-transitory computer readable medium of claim 12, wherein a different portion of the data is cached at an access point embedded in at least one of: a printer and a point of sale device.



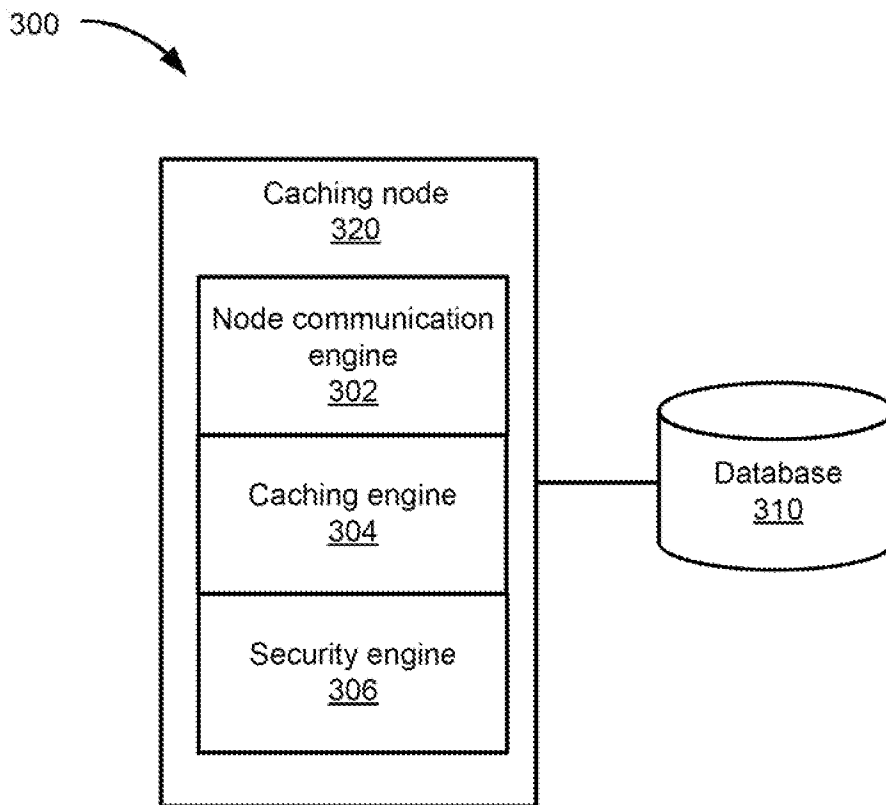
**FIG. 1**



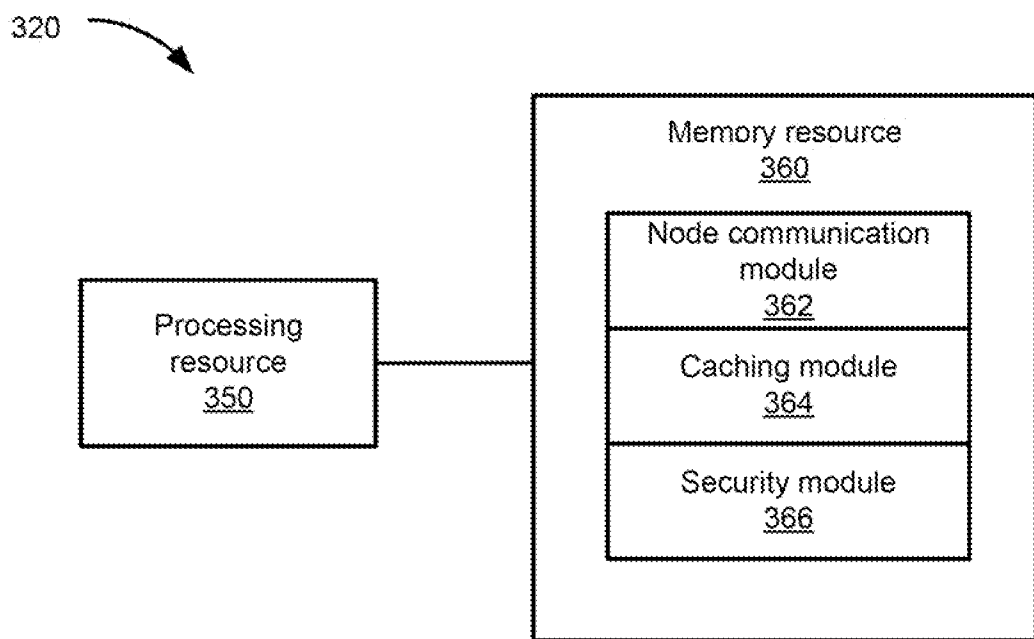
**FIG. 2A**



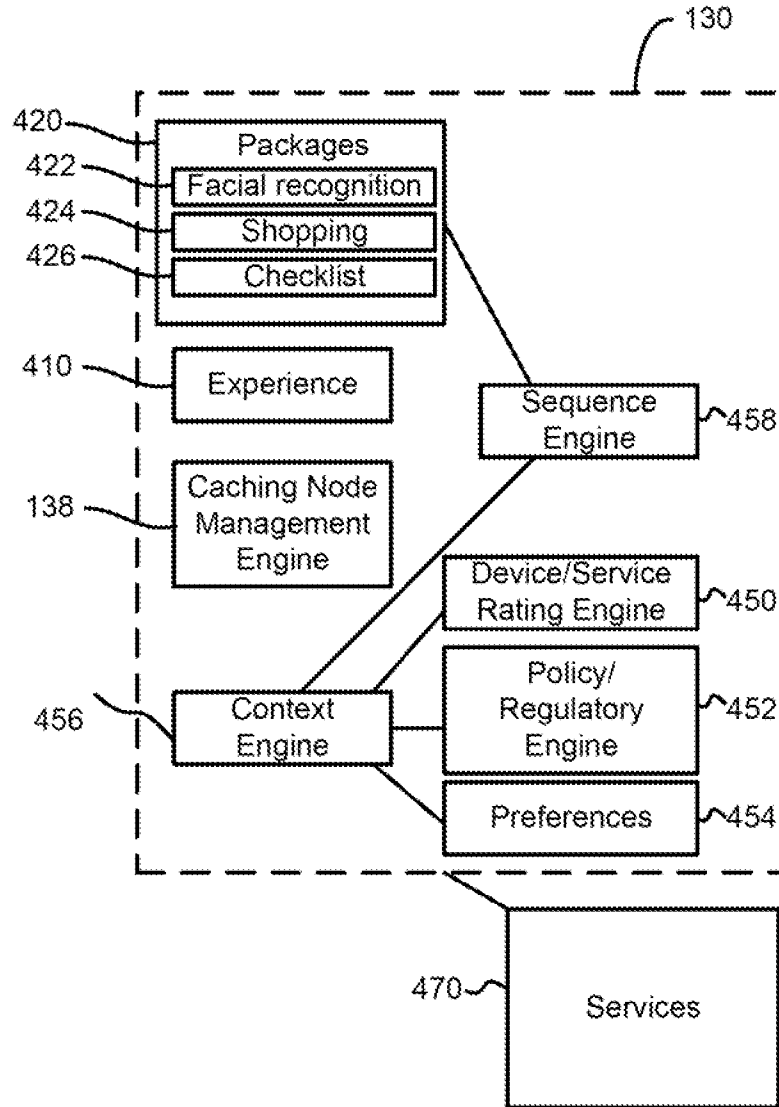
**FIG. 2B**



**FIG. 3A**



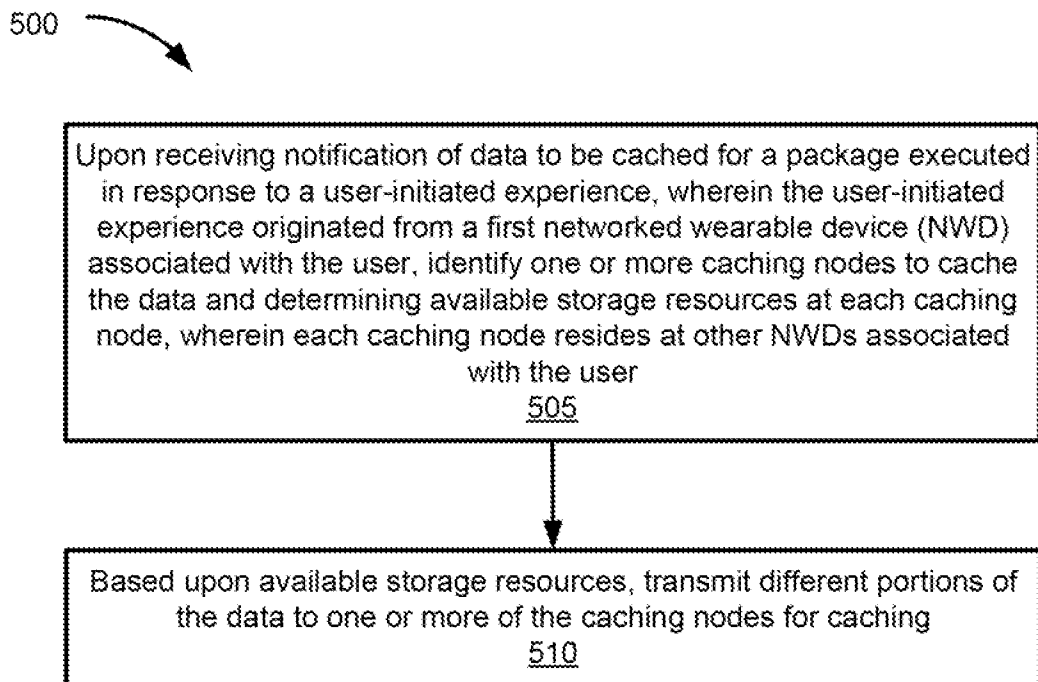
**FIG. 3B**



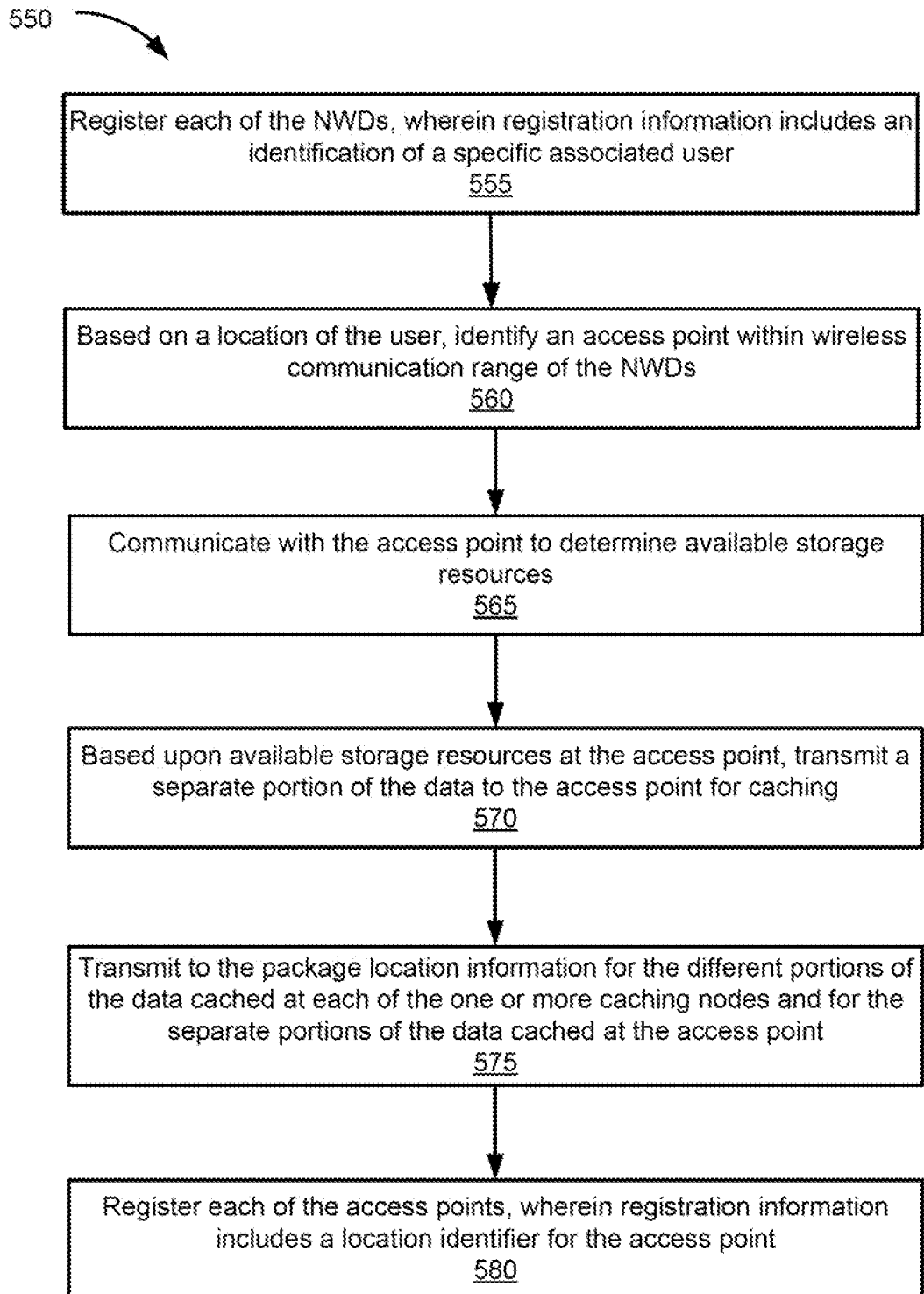
**FIG. 4**



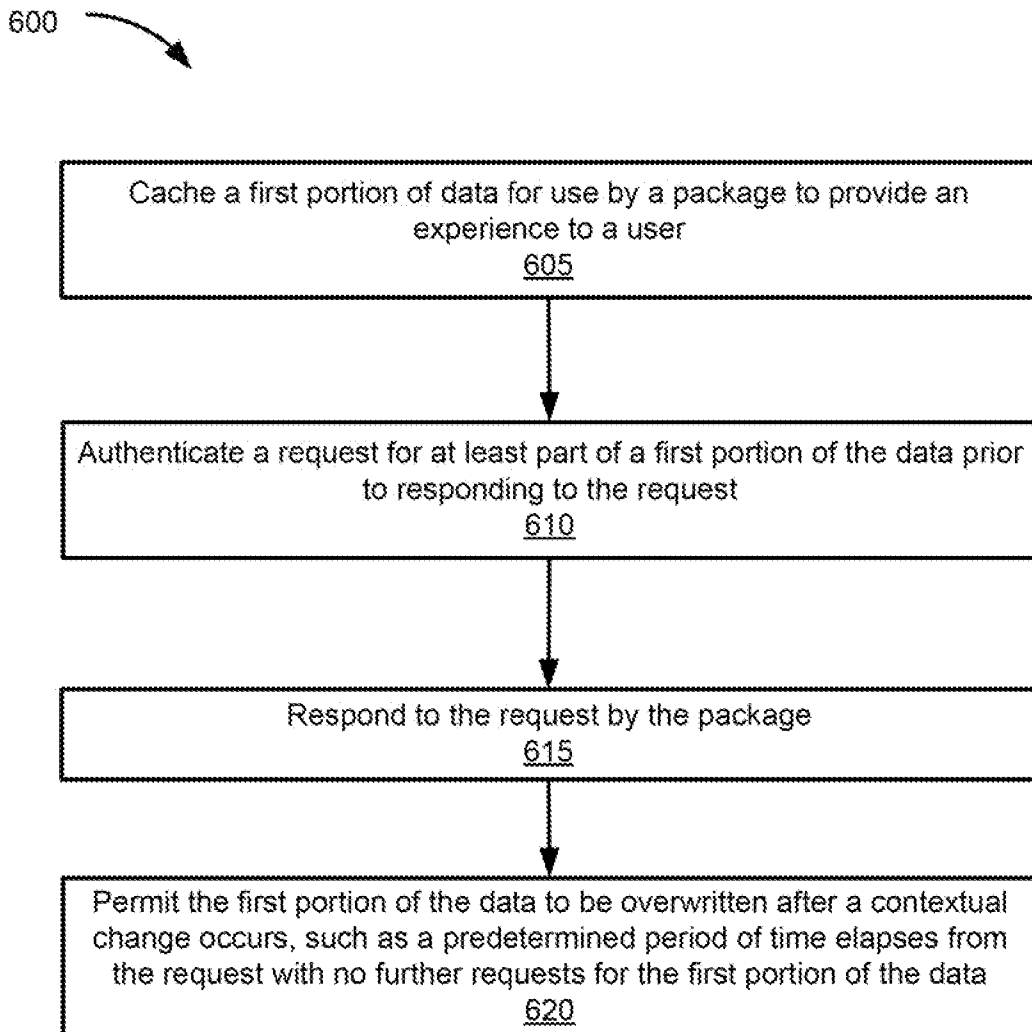
7/12

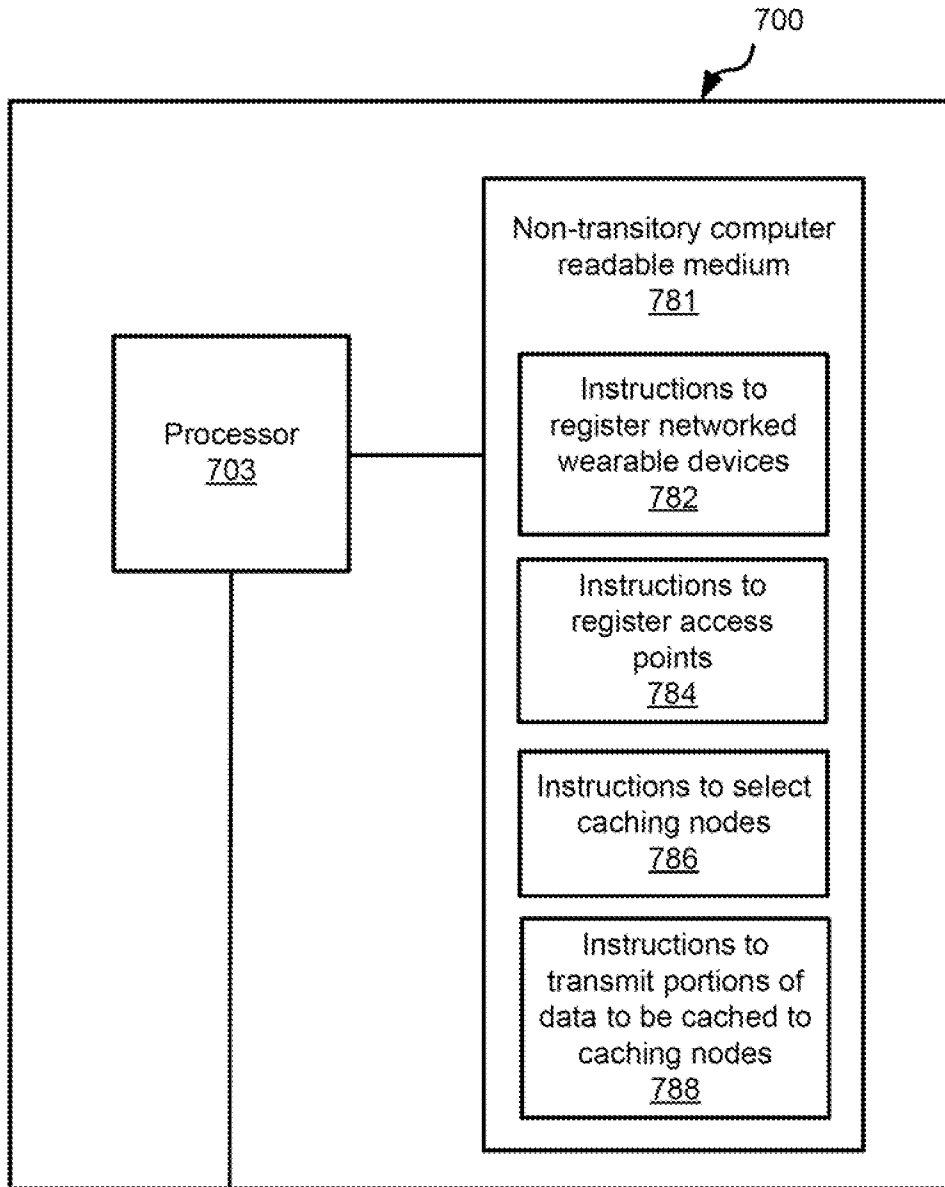
**FIG. 5A**

8/12

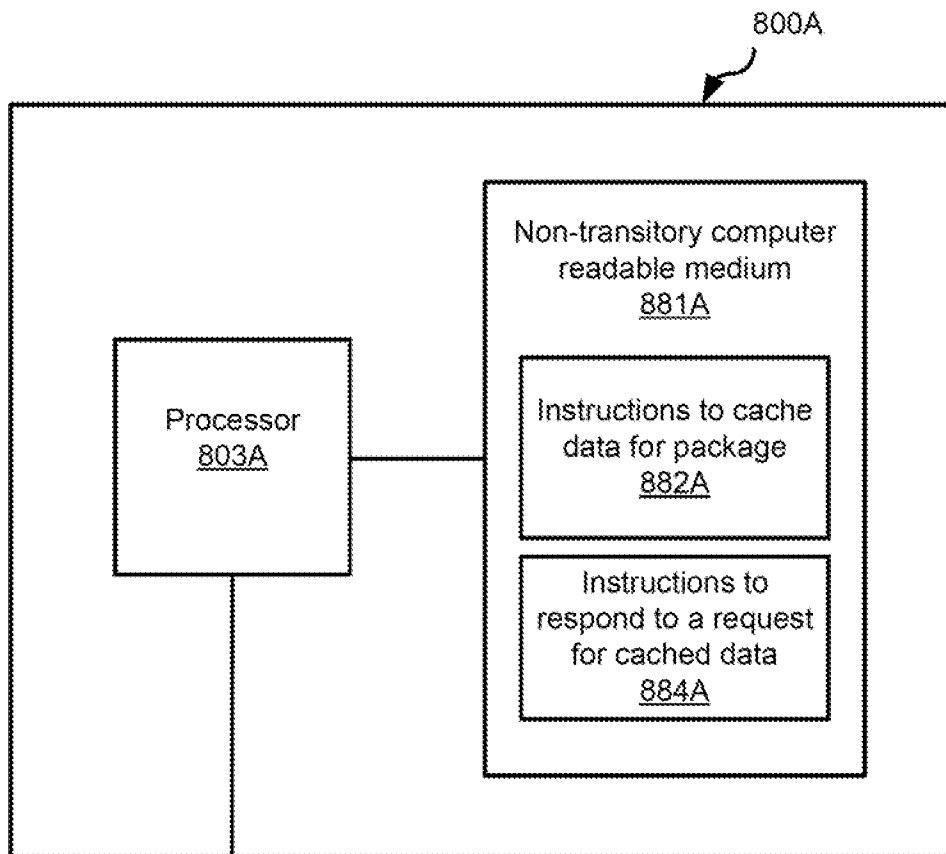
**FIG. 5B**

9/12

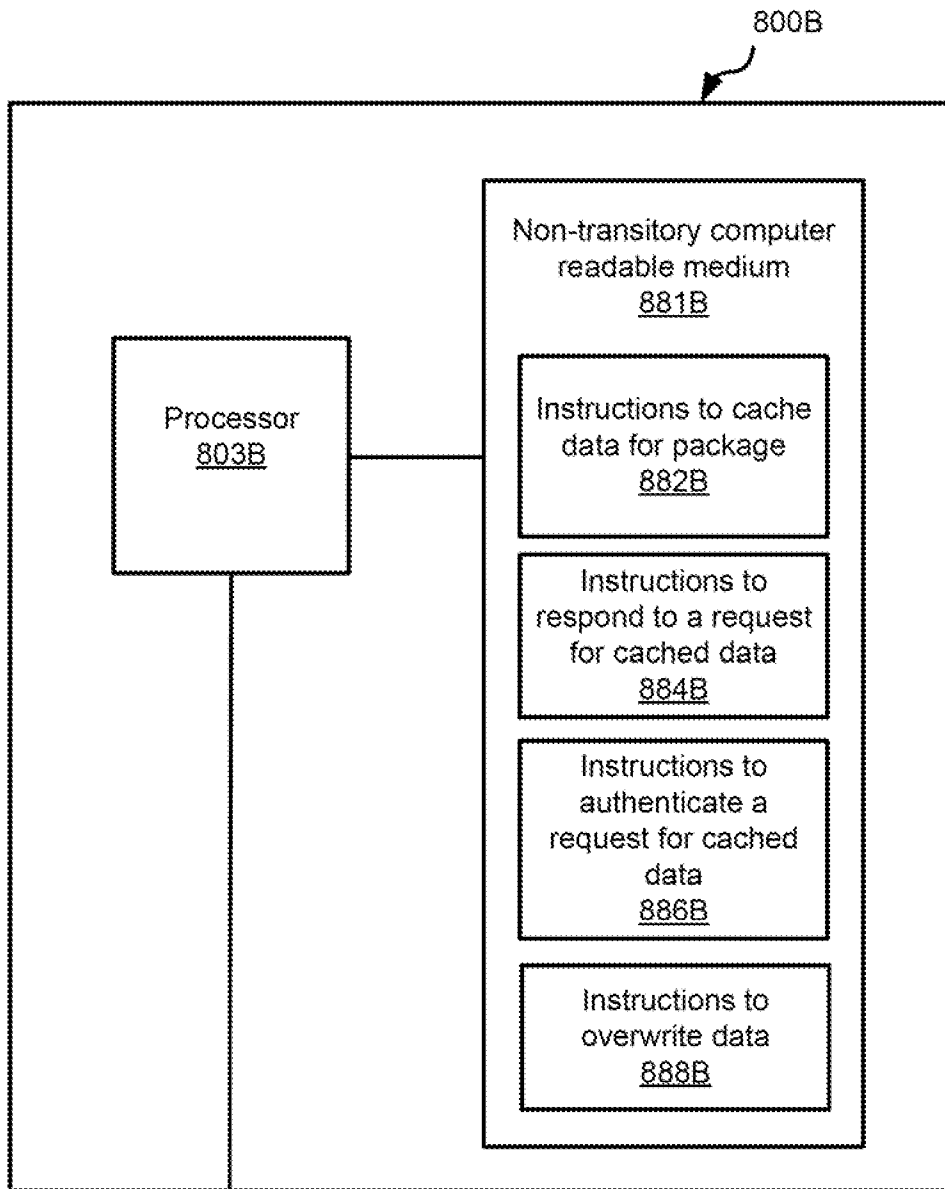
**FIG. 6**



**FIG. 7**



**FIG. 8A**



**FIG. 8B**

## INTERNATIONAL SEARCH REPORT

International application No.  
**PCT/US2014/057642****A. CLASSIFICATION OF SUBJECT MATTER****H04L 12/24(2006.01)i, H04L 29/02(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 12/24; H04L 29/08; A61B 5/02; G08B 23/00; G06F 17/21; G06F 15/173; H04L 29/08; H04L 29/02

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: cache, node, communication, wearable, device

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2014-0189060 A1 (CEDRIC WESTPHAL) 03 July 2014 See paragraphs [0007], [0020], [0032], [0040]-[0047], [0079]-[0082]; claim 5; and figures 2, 8.	1-4, 6, 8-9, 12
Y		5, 7, 10-11, 13-15
Y	US 2012-0245464 A1 (BAO TRAN) 27 September 2012 See paragraphs [0005], [0038], [0160], [0185]-[0186]; and figure 1.	5, 7, 10-11, 13, 15
Y	US 2004-0123242 A1 (MICHAEL T. MCKIBBEN et al.) 24 June 2004 See paragraphs [0068], [0075]; and figure 1.	14
A	US 2004-0255016 A1 (FRANK HUNDSCHIEDT et al.) 16 December 2004 See paragraphs [0015]-[0017], [0023]; and figure 1.	1-15
A	EP 2779539 A1 (SAMSUNG ELECTRONICS CO., LTD.) 17 September 2014 See paragraphs [0004], [0141]; and figure 1.	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

14 May 2015 (14.05.2015)

Date of mailing of the international search report

**15 May 2015 (15.05.2015)**

Name and mailing address of the ISA/KR

International Application Division  
Korean Intellectual Property Office  
189 Cheongsa-ro, Seo-gu, Daejeon Metropolitan City, 302-701,  
Republic of Korea

Facsimile No. ++82 42 472 7140

Authorized officer

AHN, Jeong Hwan

Telephone No. +82-42-481-8440



## INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

**PCT/US2014/057642**

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2014-0189060 A1	03/07/2014	WO 2014-107602 A1	10/07/2014
US 2012-0245464 A1	27/09/2012	US 07539532 B2	26/05/2009
		US 08121673 B2	21/02/2012
		US 08328718 B2	11/12/2012
		US 08425415 B2	23/04/2013
		US 08708903 B2	29/04/2014
		US 2007-265533 A1	15/11/2007
		US 2009-227877 A1	10/09/2009
		US 2012-095357 A1	19/04/2012
		US 2013-197322 A1	01/08/2013
US 2004-0123242 A1	24/06/2004	AU 2003-297888 A1	30/06/2004
		AU 2003-300866 A1	30/06/2004
		US 07139761 B2	21/11/2006
		US 07925246 B2	12/04/2011
		US 08195714 B2	05/06/2012
		US 2004-0122835 A1	24/06/2004
		US 2005-0063524 A1	24/03/2005
		US 2006-0069726 A1	30/03/2006
		US 2006-0080344 A1	13/04/2006
		US 2007-0127400 A1	07/06/2007
		WO 2004-053658 A2	24/06/2004
		WO 2004-053658 A3	29/07/2004
		WO 2004-053847 A2	24/06/2004
		WO 2004-053847 A3	23/09/2004
		WO 2005-043864 A2	12/05/2005
		WO 2005-043864 A3	30/03/2006
		WO 2006-047561 A2	04/05/2006
		WO 2006-047561 A3	01/03/2007
		WO 2006-047597 A2	04/05/2006
		WO 2006-047597 A3	07/06/2007
		WO 2008-097580 A2	14/08/2008
		WO 2008-097580 A3	16/10/2008
US 2004-0255016 A1	16/12/2004	AT 295043 T	15/05/2005
		AU 2002-340853 A1	14/04/2003
		CN 100539578 C	09/09/2009
		CN 1575575 A	02/02/2005
		DE 60204031 T2	09/03/2006
		EP 1298878 A1	02/04/2003
		EP 1430691 A2	23/06/2004
		EP 1430691 B1	04/05/2005
		US 07860948 B2	28/12/2010
		WO 03-030486 A2	10/04/2003
		WO 03-030486 A3	09/10/2003
EP 2779539 A1	17/09/2014	KR 10-2014-0111485 A	19/09/2014
		US 2014-0258490 A1	11/09/2014