

(51) International Patent Classification:
H04L 29/06 (2006.01) *H04L 29/08* (2006.01)(21) International Application Number:
PCT/US2016/015664(22) International Filing Date:
29 January 2016 (29.01.2016)

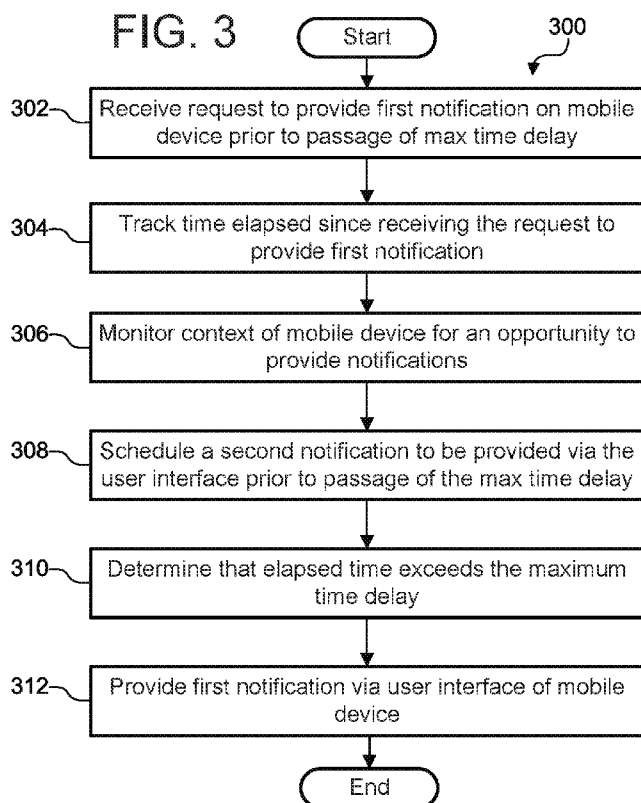
(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **HEWLETT PACKARD ENTERPRISE DEVELOPMENT LP** [US/US]; 11445 Compaq Center Drive W., Houston, Texas 77070 (US).(72) Inventors: **PRADHAN, Swadhin**; 1501 Page Mill Rd., Palo Alto, California 94304-1100 (US). **PARATE, Abhinav**; 1501 Page Mill Rd., Palo Alto, California 94304-1100 (US). **KIM, Kyu-Han**; 1501 Page Mill Rd., Palo Alto, California 94304-1100 (US).(74) Agents: **COOK, Justin M.** et al.; Hewlett Packard Enterprise, 3404 East Harmony Road Mail Stop 79, Fort Collins, Colorado 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,

[Continued on next page]

(54) Title: MOBILE DEVICE NOTIFICATION SCHEDULING



(57) Abstract: In one example, a technique involves receiving a request to provide a first notification via a user interface of a mobile device prior to passage of a maximum time delay. The technique may involve tracking an elapsed time since receiving the request to provide the first notification. The technique may involve monitoring a context of the mobile device for an opportunity to provide notifications via the user interface. The technique may involve scheduling a second notification to be provided via the user interface prior to passage of the maximum time delay based on the monitored context. The technique may involve determining that the elapsed time exceeds the maximum time delay. The technique may involve providing the first notification via the user interface responsive to determining that the elapsed time exceeds the maximum time delay.



SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG). **Published:**

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

Declarations under Rule 4.17:

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

MOBILE DEVICE NOTIFICATION SCHEDULING

BACKGROUND

[0001] Mobile computing devices provide a computing platform for a variety of applications. These installed applications may use notifications to alert a user when some new information is available. A typical mobile device user has many applications installed on their device, and so notifications from these applications can be overwhelming given finite user attention and finite user interface resources (e.g., screen size).

BRIEF DESCRIPTION OF THE DRAWINGS

[0002] The following detailed description references the drawings, wherein:

[0003] FIG. 1 is a block diagram of an example system for managing and scheduling notifications on a mobile device;

[0004] FIG. 2A is a block diagram of an example notification scheduler system illustrating data inputs and outputs during an example notification scheduling process;

[0005] FIG. 2B is a timeline of an example notification scheduling process;

[0006] FIG. 3 is a flowchart of an example notification scheduling process; and

[0007] FIG. 4 is a flowchart of an example notification scheduling process.

DETAILED DESCRIPTION

[0008] The following description makes reference to the accompanying drawings, in which similar symbols identify similar components, unless context dictates otherwise. The descriptions herein, as well as the drawings, present examples of the subject matter of the present disclosure and are in no way limiting in regard to the subject matter disclosed herein. Throughout the description, the singular forms of “a”, “an”, and “the” mean “one or more”. Thus, various examples in which a component is described in singular form also apply to examples having multiple of those components. Moreover, some aspects of the examples presented herein may

be modified, re-arranged, re-ordered substituted, combined, and/or separated in a variety of different configurations without departing from the subject matter of the present disclosure.

[0009] The present disclosure presents a technique for scheduling notifications based in part on the relative importance of each notification to the user. To determine the relative importance, training data is used to identify notifications that are associated with a high degree of user engagement as inferred based on user interaction with a given notification. For instance, a notification that is dismissed immediately may be associated with a low degree of user engagement, whereas a notification that is selected and/or viewed for some duration prior to being dismissed may be associated with a higher degree of user engagement.

[0010] Moreover, notifications may be characterized based on their respective content and/or based on the application that generated them. Such content information for a given notification may then be used to characterize that notification and use that characterization as a basis to predict the degree of user engagement with the given notification. For example, a notification for a text message from a particular sender may be characterized similarly to other text message notifications from the same sender. The new notification can then be assigned a predicted degree of user engagement similar to the degree of user engagement for those other text messages, which may be inferred from training data, machine learning, etc.

[0011] In addition, to assist in scheduling notifications, each new notification may be associated with a respective maximum delay time. For instance, urgent notifications related to operation of the device or a transient, non-repeating event, may be associated with a relatively brief maximum delay time, because delaying deliver of those notifications may limit the usefulness of such notifications. On the other hand, notifications related to a non-critical application update or a non-transient condition may be associated with a relatively long maximum delay time, because relatively little usefulness will be lost by delaying delivery of such notifications. A timer for each new notification can then be started upon generation of the notification. The system may then use the timer for each notification (i.e., time until maximum delay time) when scheduling notifications.

[0012] For example, the system may monitor the device context for an opportunity to interrupt the user with one or more notifications having timers that have not yet reached their respective maximum delay times and, upon identifying an opportunity, provide a set of notifications with the highest predicted degrees of user engagement. However, upon determining that any notifications have reached their maximum delay time, the system may force that notification to be provided, regardless of predicted degree of user engagement and device context. Other examples are also possible. For instance, the remaining time to maximum delay of undelivered notifications may be used as a basis to prioritize which notifications to provide during a given notification opportunity. Thus, notifications approaching maximum delay may be gradually assigned higher priority as they approach their maximum delay time in an attempt to avoid providing such notifications outside of context-based interruption opportunities.

[0013] Finally, notifications may be presented with a user interface modality that is based at least in part on the context of the mobile device. For example, a device context associated with a user being in a meeting may be used as a basis for providing a notification via a silent modality (e.g., by blinking lights on the device). In another example, a device context associated with a user jogging or walking may be used as a basis for providing a notification via a sound-alert modality (e.g., by activating a chime or ring). In another example, a device context associated with a user being engaged with their device may be used as a basis for providing a notification via a display modality (e.g., by displaying a pop-up window and/or a selectable symbol that indicates the notification is ready for viewing). Many other examples are also possible.

[0014] In brief, the present disclosure presents a technique that leverages machine learning techniques to manage and schedule notifications on a mobile device. The technique may involve associating each notification with a predicted degree of user engagement based in part on content of the notification and the originating application. The technique may involve monitoring the device context for notification provision opportunities (e.g., interruption opportunities). Upon identifying an opportunity, the technique may involve selecting one or more notifications to display based on the predicted degree of user engagement of those notifications.

The technique may also involve tracking a respective maximum delay time for each notification. Upon determining that a maximum delay time has been reached, the technique may involve providing the corresponding notification via the user interface. Moreover, the technique may involve selecting a user interface modality for a given notification based in part on the device context. In some cases, the present disclosure presents an approach that specifies: which notifications to provide, when to provide them, and how to do so via the user interface.

[0015] FIG. 1 is a block diagram of an example system 100 for managing and scheduling notifications on a mobile device 140. System 100 includes a mobile device 140 in communication with a computing system 110. The mobile device 140 and the computing system 110 may be communicatively coupled via network 130. In some cases, the computing system 110 may be a remote server and the mobile device 140 may be a network-connected computing device, such as a mobile computing device (e.g., cellular phone, tablet, etc.) or personal computer (e.g., laptop computer, computer terminal). As described herein, the computing system 110 may send indications of notifications to be displayed on mobile device 140. In some examples, the notifications may be related to particular applications, which each have associated remote computing systems (e.g., the application servers 120, 124) in communication with the computing system 110. For example, the computing system 110 in communication with mobile device 140 may be a remote server associated with an operation system (OS) for the mobile device 140.

[0016] The computing system 110 includes a processing system 112 and data storage 114. The processing system 112 and data storage 114 may be communicatively coupled via a bus and/or network. The processing system 112 may include processor(s) that are used to execute instructions 116 in the data storage 114. In some examples, the processing system 112 may include functional modules that perform predetermined tasks and/or routines (e.g., push notification module 118). For example, functional modules may be defined by processor-executable instructions stored in memory that define processes performed by the computing system 110 upon execution of such instructions by the processor(s). In some examples, computing system 110 may include hardware features to perform

processes described herein, such as logical circuit(s), application specific integrated circuit(s), etc.

[0017] Data storage 114 may be any electronic, magnetic, optical, or other physical storage device that can be non-transiently encoded to store data. Data storage 114 may store data specifying push notification settings. In some examples, settings may include indications of times to deliver push notifications, permissions for particular applications to send push notifications, maximum time delays for particular notifications, user interface delivery modalities, etc. In some examples, the push notification settings stored in data storage 114 may indicate notification settings for different users, such as a first user, a second user, etc. In some cases, the push notification settings may specify user interface delivery modalities, such as notification via display notification on a target device, via haptic feedback, via status indicator lights, etc.

[0018] In addition, the computing system 110 may be in communication with multiple application-associated servers (e.g., servers 120, 124). Such servers may be associated with respective applications that are installed on mobile device 140. For example, some applications that involve sending and receiving communications amongst mobile devices may facilitate such messaging via a server that serves as an intermediary to route communications to and from specific user-associated mobile devices (e.g., device 140). In some cases, such remote servers 120, 124 may communicate directly with mobile device 140 to indicate, for example, that a message is ready for delivery to mobile device 140. In some cases, remote servers 120, 124 may communicate with mobile device 140 by first sending an indication to computing system 110 that a message is ready for delivery to mobile device 140. The computing system 110 may then send push notification(s) to mobile device 140 to pass on the pending notifications and/or redirect the mobile device to server(s) 120, 124 as necessary. Thus, notifications 122, 126 may be indications of various notifications that are generated by application-associated servers 120, 124.

[0019] In some examples, notifications 122, 126 may involve information that an update is available for a particular application. In some examples, notifications 122, 126 may involve information that a pending message is ready for delivery to mobile device 140, such as may occur in communication-related applications in which

another device may originate a message for delivery to mobile device 140. A transmission indicative of such message may then be routed to application server 120, which generates notification 122. Notification 122 may be a transmission indicative of the original message. Notification 122 may be sent to computing system 110, at which point the computing system 110 may send a related push notification to mobile device 140 via network 130. Moreover, OS-associated computing system 110 may send notifications to mobile device 140 that are related to operation of the OS itself. Other examples are also possible in which remote computing systems generate transmissions related to pending notifications for delivery to mobile device 140.

[0020] The mobile device 140 may be a computing device that corresponds to a particular user 10. The user-associated mobile device 140 may be, for example, a cellular phone and/or personal computer that is used by the user 10. In some examples, the description of the mobile device 140 may apply to devices that are used by multiple users that each have a separate login. The separate logins thereby provide separate implementations of a shared device in which personalized features, permissions, access rights, etc., are set according to attributes associated with each user. Thus, as described herein, upon user 10 logging in to such a shared device, that device corresponds to user 10 and may function in a way that is suitable for user 10 (e.g., according to attributes associated with user 10).

[0021] The mobile device 140 includes a communication interface 142, sensor(s) 144, a user interface 146, a processing system 148 and data storage 150. The communication interface 142 may include features that allow the mobile device 140 to send and receive information via network 130 (e.g., for communication with computing system 110). For example, communication interface 142 may include an antenna for sending/receiving wireless signals and/or port(s) for wired network signals as well as related signal processing hardware for generating and interpreting such signals.

[0022] Sensor(s) 144 may include devices that generate data indicative of stimuli and/or status related to the context of the mobile device 140. In some examples, the sensor(s) 144 may include devices that generate data related to position, location, and/or acceleration of the device 140. In some examples, the sensor(s) 144 may

include ambient temperature, incident sound, incident light, applied pressure, and/or touch of the device 140. Moreover, in some examples, the mobile device 140 may not include sensor(s) 144.

[0023] The user interface 146 may include a display device, an audio system, a haptic feedback system, or another system that allows the user 10 to perceive information. In some examples, the user interface 146 may convey information via the user interface 146 (e.g., via a display device and/or audio system). Moreover, user interface 146 may include an input device, such as a touch-sensitive region, a microphone, an image capture device, etc., for receiving inputs from user 10. Thus, the user interface 146 allows the user 10 to operate the device 140 by providing inputs and receiving outputs. Among other features, the user interface 146 may be used to output notification messages for perception by the user 10. Such notification messages may prompt the user 10 to indicate (via the user interface 146) whether to proceed with a given operation (e.g., open an application related to the notification). In some examples, the user interface 146 may include a display device and the confirmation messages may be a pop-up window that is displayed in place of at least a portion of previously-displayed content – the pop-up window may appear to appear “over” the previously-displayed content. Other aspects of the user interface 146 may be used to prompt user 10 to respond to a notification message.

[0024] The processing system 148 may include processor(s) that are used to execute instructions 152 in the data storage 150. In some examples, the processing system 148 may include functional modules that perform predetermined tasks and/or routines. For example, functional modules may be defined by processor-executable instructions stored in memory that define processes performed by the device 140 upon execution of such instructions by the processor(s). For example, instructions 152 may include a notification scheduler module 154.

[0025] Data storage 150 may be any electronic, magnetic, optical, or other physical storage device that can be non-transiently encoded to store data. Data storage 150 may include processor-executable instructions 152 that cause the device 140 to perform operations upon execution of the instructions 152 by the processing system 148.

[0026] Instructions 152 may include an operating system (OS) 156 and applications 158, 160. For example, the OS 156 may be initiated upon initialization of the device 140 and may operate to manage computing resources of the device 140 and regulate operation of applications 158, 160. In some examples, at least some applications installed on the device 140 (e.g., the first application 158, the second application 160, etc.) may occasionally provide notifications for perception by user 10. In some examples, the OS 156 may at least partially regulate how and/or when such notifications are rendered for the user's perception.

[0027] The notification scheduler module 154 may be a functional module that is specified by executable instructions to cause the device 140 to perform the operations described herein. In some cases, the notification scheduler module 154 makes determinations on whether to provide particular pending notifications at particular times and which modality to employ when providing such notifications. In some cases, the notification scheduler may prioritize pending notifications based on their predicted degree of user engagement and also monitor sensor data to identify opportunities for providing notifications via the user interface. In some examples, the notification scheduler 154 may receive incoming notifications, and then determine particular notifications to provide, at particular times, and by particular means.

[0028] In response to a determination made by the module 154, the device 140 may provide a specified notification via user interface 146. The notification scheduler module 154 may be implemented by instructions that form a part of other applications (e.g., applications 158, 160), by instructions that are part of the OS 156 of the device 140, and/or by instructions that define a separate stand-alone application.

[0029] FIG. 2A is a block diagram of an example notification scheduler system 210 illustrating data inputs and outputs during an example process involving scheduling notifications on a mobile device. The notification scheduler 210 may be similar in some respects to the notification scheduler module 154 described in connection with FIG. 1. Thus, the notification scheduler may be a functional module that is resident on a mobile device (e.g., the mobile device 140 of FIG. 1). Further, in some examples, the notification scheduler 140 may be implemented, at least in part, on a remote computing system (e.g., computing system 110 of FIG. 1). The

notification scheduler system 210 of FIG. 2A illustrates incoming and outgoing data and distinct functional modules that are employed to identify select amongst multiple notifications in an example.

[0030] As shown in FIG. 2A, the notification scheduler 210 includes a sensor analyzer 212, a timer 214, an engagement predictor 215, a notification prioritizer 216, a notification selector 218, an opportunity identifier 220, and a modality selector 222. The notification scheduler 210 receives prior user interaction data 201, sensor data 202, and data indicative of notifications (e.g., notification A 204, notification B 206, and notification C 208). The data indicative of each notification may specify: the associated application, a maximum time delay, and content of the notification. Thus, notification A 204 is associated with application "app A", a maximum time delay "Tmax A", and content "content A". Similarly, notification B 206 is associated with app B, Tmax B, and content B; and notification C 208 is associated with app C, Tmax C, and content C.

[0031] Sensor analyzer 212 analyzes received sensor data 202. For example, sensor analyzer 212 may monitor and analyze data indicative of temperature, ambient noise, acceleration, location, and so forth. The analyzed sensor data may then be combined with the data 201 indicative of prior user interactions with the device to determine the device context and/or predict the degree of user engagement with the device. In short, the data 201 provides information of past user interactions that contextualize the sensor data 202 to allow the system to determine the device context and/or degree of user engagement. For example, when current sensor data 202 corresponds to a particular pattern that is logged in the past user interaction data 201, the current sensor data 202 may be determined to indicate a similar device context and/or degree of user engagement as was found in the previously logged information (i.e., data 201).

[0032] Timer 214 may include a counter or similar device that monitors and tracks elapsed time since an initiating event. Among other aspects, timer 214 may be used to time elapsed time since a given notification is received relative to its specified maximum allowable time delay.

[0033] The engagement predictor 215 may make real time determinations regarding the degree of user engagement with the device. These determinations

may be made based on sensor data 202 contextualized in view of the prior user interaction data 201. In some examples, the engagement predictor 215 may determine the extent to which the user is presently engaged with their device. This information is useful, because during periods of relatively high user engagement (e.g., while the user is actively interacting with the device via the user interface), the user may be more receptive to receiving notifications. On the other hand, during periods of relatively low user engagement (e.g., while the user is engaged in activities that do not involve the device), the user may be less receptive to receiving notifications. The degree of user engagement with the device, as determined the engagement predictor 215, may be based on a model in which the prior user interaction data 201 is used as training data and the analyzed sensor data is used as an input to the model to determine the degree of user engagement with the device in real time. In some examples, the engagement predictor 215 may assign a predicted degree of user engagement with the device having multiple gradations. That is, rather than simply assigning a binary value indicating whether the user is either a) engaged or b) not engaged, the engagement predictor 215 may assign one of multiple values that each indicate a respective predicted degree of user engagement with the device.

[0034] The notification prioritizer 216 may prioritize multiple received notifications based on their respective predicted degree of user engagements. The predicted degree of user engagement for each notification may be determined based on content of each received notification, which is correlated with training data and/or analyzed according to machine learning techniques. Such techniques may involve, for example, analyzing content and/or associated application of a given notification and then comparing such data with predicted degree of user engagement for particular notifications. The predicted degree of user engagement may be based on a model derived from the user's interaction with actual notification, for example. Thus, the model may evolve in time according to machine learning techniques. In some examples, a user's selection and/or dismissal of a given notification may be used as a basis for assigning and/or adjusting a predicted degree of user engagement with a given notification and/or class of notifications. In some examples, a user's viewing of a notification (e.g., as derived via eye tracking and/or accelerometer data) may be used as a basis for assigning and/or adjusting a

predicted degree of user engagement with a given notification and/or class of notifications. Moreover, other user interface elements may be used as a basis for determining that a user is viewing or otherwise interacting with a notification. For instance, if a notification is being displayed, and the user is preventing the display from being dimmed and/or rendered blank (e.g., by intermittently touching the display and/or based on proximity sensors, etc.), it may be inferred that the user is viewing the notification. Such information may be used to adjust a model of predicted degree of user engagement. In some cases, the notification prioritizer 216 may analyze the content of each notification, and use its model of predicted degree of user engagement to generate a list of the pending notifications ordered according to their relative predicted degree of user engagement.

[0035] In addition, the notification prioritizer 216 may receive an indication of the real time degree of user engagement from the engagement predictor 215, and use that information as a basis for assigning priorities to pending notifications. For example, during periods of relatively high user engagement, as indicated by the engagement predictor 215, the notification prioritizer 216 may prioritize a particular subset of pending notifications having content related to, for example, incoming communications from particular users (or another class of pending notifications). On the other hand, during periods of relatively low user engagement, as indicated by the engagement predictor 215, the notification prioritizer 216 may prioritize another subset of pending notifications having content related to another type of notifications. Thus, the output of the notification prioritizer 216 may depend on both the predicted degree of user engagement with the device generally, as indicated by the engagement predictor 215, as well as the predicted degree of user engagement with the particular pending notifications (e.g., notifications A, B, and C), which may be based at least in part on the content of such notifications.

[0036] The opportunity identifier 220 and modality selector 222 each receive information from the sensor analyzer 212 as well as prior user interaction data 201 to perform their respective functions. The opportunity identifier 220 monitors information from the sensor analyzer 212 to identify opportunities for providing notifications (e.g., interruption opportunities). For example, the opportunity identifier 220 may identify times during which sensor information indicates that the user is

transitioning between different activities (e.g., walking to standing, or standing to sitting). Such transition times have been previously associated with times of relatively high user engagement with notifications. In some examples, the opportunity identifier 220 may identify times during which sensor information indicates that the user is at rest, as opposed to in the middle of walking or running. In some examples, the opportunity identifier 220 may also analyze information other than sensor-derived context information, such as information from a calendar application, phone application, etc., to characterize the context of the mobile device and its user and thereby identify opportunities for providing notifications.

[0037] The modality selector 222 may use sensor information to select a modality for providing a notification based on the current context of the device. For example, if sensor information indicates the user is in motion, the modality selector 222 may select a modality in which the user interface of the mobile device uses an audible signal (e.g., a chime), as opposed to a purely visual cue. In addition, if sensor information indicates the user is at rest, the modality selector 222 may select a purely visual cue (e.g., a blinking status light); if sensor information indicates the user is driving, the modality selector may select a modality in which the notification is announced by a text-to-speech application. In some examples, the modality selector may select a modality in which the notification is indicated by vibration (e.g., via a haptic feedback system). Other notification modalities using various aspects of the user interface system and/or combinations thereof may also be selected. In some cases, the modality selector 222 selects from amongst a set of possible notification provision modalities based on sensor information. Many other examples are also possible.

[0038] The notification selector 218 receives information from the timer 214, notification prioritizer 216, and opportunity identifier 220 to schedule pending notifications to be provided at particular times. In addition, the modality selector 222 selects the appropriate modality for that time, and the information is combined to output data 230, which indicates a subset of pending notifications and the modality by which the subset should be provided via the user interface. The notification selector 218 determines whether any pending notifications have been pending for their maximum time delay (as indicated by timer 214), and should therefore be

scheduled to be provided immediately. The notification selector 218 also determines whether an opportunity exists for providing notifications (as indicated by opportunity identifier 220), and then schedules one or more of the highest priority pending notifications (as indicated by notification prioritizer 216). In some cases, moreover, the notification selector 218 may use the remaining time until the maximum time delay is elapsed for a given notification as a factor to be weighed along with its relative priority when scheduling notifications. As such, the notification scheduler 218 may help avoid scheduling some notifications only upon passage of their maximum time delay. This may be desirable, for example, because notifications scheduled upon passage of their maximum time delay may be relatively less likely to be scheduled at opportune moments for interruption (e.g., as identified by opportunity identifier 220). Example operations of the notification selector 210 are described below in connection with the flowcharts of FIGS. 3 and 4.

[0039] FIG. 2B is a timeline of an example process involving scheduling notifications on a mobile device. The timeline shows that notifications A, B, and C 204, 206, 208 are received at times T_A , T_B , and T_C , respectively. As noted above, the received notifications also indicate that they are to be provided prior to passage of respective maximum time delays: $T_{max A}$, $T_{max B}$, and $T_{max C}$, respectively. The notifications are ultimately scheduled to be provided at schedule times: $T_{sch A}$, $T_{sch B}$, and $T_{sch C}$, respectively. In operation, the order of events is as follows: (1) Notification A is received at T_A ; (2) Notification B is received at T_B ; (3) Notification C is received at T_C ; (4) Notification B is provided at $T_{sch B}$, due to passage of $T_{max B}$ from T_B ; (5) Notification C is provided at $T_{sch C}$, due to Notification C being the highest priority pending notification at the identified notification time; (6) Notification A is provided at $T_{sch A}$, due to passage of $T_{max A}$ from T_A .

[0040] In the example timeline of FIG. 2B, the $T_{max B}$ is relatively short compared to $T_{max A}$ and $T_{max C}$. This may be due to Notification B having a relatively higher degree of urgency than Notifications A and C. In the example timeline of FIG. 2B, the only identified opportunity for providing a notification (e.g., an interruption opportunity) is at $T_{sch C}$. At $T_{sch C}$, the highest priority pending notification is Notification C, and so Notification C is scheduled to be provided while Notification A remains pending. The priority may be derived by notification prioritizer

216 mapping the content of both Notifications A and C to its model of predicted degree of user engagement. In the example timeline, Notifications A, B, and C are not provided in the order in which they were received, nor are they simply provided in accordance with their respective maximum time delays. Rather, a combination of identifying notification provision opportunities, prioritization pending notifications, and monitoring maximum time delay are used to schedule all three.

[0041] Moreover, in some examples, the maximum time delay values may be modified dynamically based on device context and/or user interactions with the device. For instance, if the device context and/or user interactions indicate a period of relatively high user engagement, then the maximum time delays for pending notifications may be reduced to thereby encourage those notifications to be provided within the period of high user engagement. On the other hand, if the device context and/or user interactions indicate a period of relatively low user engagement, then the maximum time delays for pending notifications may be increased to thereby allow those notifications to be delayed beyond the period of low user engagement. In some cases, such dynamic adjustments may involve scaling the maximum time delays of pending notifications and/or adjusting the maximum time delays of pending notifications according to a formula that is based on the real time predicted degree of user engagement.

[0042] FIG. 3 is a flowchart of an example process 300 involving scheduling notifications on a mobile device. Process 300 may be described below as being executed or performed by a system, such as a user computing device (e.g., device 140 described in connection with FIG. 1). Process 300 may be implemented in the form of executable instructions stored on at least one machine-readable storage medium of the system and executed by at least one processor of the system. In some examples, process 300 may be implemented in the form of electronic circuitry (e.g., hardware). The process 300 is depicted with a series of blocks in the flowchart of FIG. 3. In some cases, one or more blocks of process 300 may be executed substantially concurrently or in a different order than shown in FIG. 3. In some cases, process 300 may include more or less blocks than are shown in FIG. 3. In some examples, one or more of the blocks of process 300 may, at certain times, be ongoing and/or may repeat.

[0043] At block 302, a request is received to provide a first notification prior to passage of a maximum time delay. The request may be received from a remote server, for example, and may be related to a notification for a particular application installed on the mobile device. For instance, the notification may indicate an available update, a communication from another user, etc. In some examples, the maximum time delay for the first notification may be indicated in the received request. For instance, the notification may be associated with a maximum time delay by its originating application (e.g., based on the urgency or time-sensitivity of the notification). In some examples, the maximum time delay may be one of a set of possible time delays that is indicated by some aspect of the received request. For instance, each application may be associated with a maximum time delay, and each notification may be associated with one of those applications such that the association between maximum time delays and applications can be used to associate each notification with a maximum time delay. In some examples, moreover, the maximum time delay for at least some received notifications may be set by the mobile device, such as to a default value.

[0044] At block 304, the time elapsed since receiving the request to provide the first notification is tracked. For example, a timer may be initiated upon receiving the first notification, and the elapsed time may be tracked relative to the indicated maximum time delay.

[0045] At block 306, the context of the mobile device may be monitored for an opportunity to provide notifications. For example, sensor data may be analyzed for indications of a time to provide notifications. In some examples, this may involve identifying times in which the context changes between two activities (e.g., standing to sitting, walking to standing, etc.).

[0046] At block 308, a second notification may be scheduled to be provided via the user interface of the device prior to passage of the maximum time delay for the first notification. The second notification may be a notification that has reached its own maximum time delay since being received at the mobile device. In some cases, the second notification may be a notification with a higher predicted degree of user engagement than the first notification (e.g., a higher priority as determined by the notification prioritizer).

[0047] At block 310, the elapsed time since receiving the first notification (at block 302) is determined to exceed the maximum time delay. For example, a timer that tracks the elapsed time since receiving the first notification (e.g., at block 304) may exceed the maximum time delay.

[0048] At block 312, the first notification may be provided via the user interface of the mobile device. The provision of the first notification may be performed in response to the determination of block 310. The provision of the first notification may be carried out in accordance with a notification modality that is selected at least partially based on a context of the device.

[0049] FIG. 4 is a flowchart of an example process 400 involving scheduling notifications on a mobile device. Process 400 may be described below as being executed or performed by a system, such as a user computing device (e.g., device 140 described in connection with FIG. 1). Process 400 may be implemented in the form of executable instructions stored on at least one machine-readable storage medium of the system and executed by at least one processor of the system. In some examples, process 400 may be implemented in the form of electronic circuitry (e.g., hardware). The process 400 is depicted with a series of blocks in the flowchart of FIG. 4. In some cases, one or more blocks of process 400 may be executed substantially concurrently or in a different order than shown in FIG. 4. In some cases, process 400 may include more or less blocks than are shown in FIG. 4. In some examples, one or more of the blocks of process 400 may, at certain times, be ongoing and/or may repeat.

[0050] At block 402, a request is received to provide notifications on a mobile device prior to passage of a maximum time delay. For example, the request may be received from a remote server responsible for providing push notifications. In some examples, the notification may even be generated locally by an application installed on a device. The received request may then be passed to a utility used to schedule and manage provision of notifications on the mobile device.

[0051] At block 404, the time since receiving each request is tracked. For example, a timer may be started upon reception of each request, and the elapsed time can be tracked.

[0052] At block 406, the pending notifications may be prioritized. For example, a notification prioritizer may evaluate each pending notification using its content, associated application, and/or remaining time until passage of maximum time delay using a model of predicted degree of user engagement. The model may specify, for example, predicted degrees of user engagement for notifications having content that corresponds to a particular pattern (e.g., notifications of messages from a particular sender). In some cases, the model may evolve over time via machine learning techniques based on the user's interaction with the mobile device upon provision of particular notifications. For instance, a user selecting a notification may be regarded as a relatively high degree of user engagement whereas a user dismissing a notification may be regarded as a relatively low degree of user engagement. All pending (i.e., not yet provided) notifications may be prioritized according to the model of predicted degree of user engagement with the notification.

[0053] At block 408, the mobile device may determine whether the context of the device indicates an opportunity to provide notifications. As noted above, this determination may involve periodically analyzing sensor data to determine a context of the mobile device. Some contexts may be preferred for provision of notifications. In some cases, transitions between contexts may be preferred for provision of notifications. The opportunity identifier may use the sensor data to analyze opportunities for providing notifications. If, at block 408, it is determined that the context indicates an opportunity to provide notifications, process 400 proceeds to block 412. On the other hand, if it is determined that context does not indicate an opportunity to provide notifications, process 400 proceeds to block 410.

[0054] At block 410, the mobile device may determine whether the elapsed time since receiving any of the pending notifications exceeds its respective maximum time delay. This determination may involve comparing each of several ongoing timers with a respective maximum time delay to determine whether any of the elapsed times has exceeded its maximum time delay. If, at block 410, it is determined that a notification's elapsed time has exceeded its maximum time delay, process 400 proceeds to block 414. On the other hand, if it is determined that none of the notifications have an elapsed time that exceeds its maximum time delay, process 400 returns to block 402 to receive additional notification(s). Thus, if context

does not indicate an opportunity to notify (in block 408) and none of the elapsed times have exceeded their respective time delays (in block 410), the process 400 does not provide any notifications and checks for any newly received notifications to be provided. If no new notifications have been received upon returning to block 402, process 400 may periodically repeat blocks 408 and 410 to determine anew whether any pending notifications should be provided.

[0055] At block 412, which is undertaken upon determining that context indicates an opportunity to notify, a subset of notifications with greatest priority may be provided. In some examples, the number of notifications provided at any given time may be limited to a maximum number N, which may be 2, 3, 4, 5, etc. The number N may be set based on the maximum number of notifications that a user can sensibly amalgamate and process simultaneously without becoming overwhelmed. The pending notification(s) with the greatest priority may be determined based on the most recent ranking of all pending notifications (at block 406). In some cases, the notifications may be provided via the user interface of the mobile device using a notification modality that is selected based in part on sensor data indicative of the context of the device.

[0056] In some examples, moreover, the system may also limit the number of opportunities for providing notifications that may be identified within a given time interval. That is, the system may limit the rate at which opportunities for providing notifications may be identified. As such, the system may limit both the number of notifications that are presented at any given time (e.g., the number "N") and also the frequency with which sets of notifications are presented. Together, these features may help prevent the user from being overwhelmed with notifications being presented too frequently and/or presented too many at once.

[0057] At block 414, which is undertaken upon determining that the elapsed time since receiving a notification exceeds that notification's maximum time delay, the notification having the elapsed time greater than its maximum time delay may be provided. In some cases, the notifications may be provided via the user interface of the mobile device using a notification modality that is selected based in part on sensor data indicative of the context of the device.

[0058] At block 416, the time since receiving the provided notifications (of blocks 412 and/or 414) may stop being tracked. Once a notification is provided, it is no longer necessary to continue tracking the elapsed time since receiving the notification because that notification will no longer fail to be provided within its specified maximum time delay. Following block 416, process 400 may return again to block 402 to receive a notification once again. If no new notifications have been received upon returning to block 402, process 400 may periodically repeat blocks 408 and 410 to determine anew whether any pending notifications should be provided.

[0059] FIG. 5 is a block diagram of an example OS fault state determination system 500. System 500 may be similar to systems 100 and 210 described in connection with FIGS. 1-4, for example. In FIG. 5, system 500 includes a processor 510 and a non-transitory machine-readable storage medium 520. Although the following descriptions refer to a single processor and a single machine-readable storage medium, the descriptions may also apply to a system with multiple processors and/or multiple machine-readable storage mediums. In such examples, the instructions may be distributed (e.g., stored) across multiple machine-readable storage mediums and the instructions may be distributed (e.g., executed) across multiple processors.

[0060] Processor 510 may incorporate central processing units (CPUs), microprocessors, and/or other hardware devices suitable for retrieval and execution of instructions stored in non-transitory machine-readable storage medium 520. In the example shown in FIG. 5, processor 510 may fetch, decode, and execute instructions 522, 524, 526, 528. In some examples, processor 510 may include electronic circuits having electronic components for performing the processes specified by the instructions in machine-readable storage medium 520. With respect to the executable instruction representations (e.g., boxes) described and shown herein, it should be understood that part or all of the executable instructions and/or electronic circuits included within one box may, in some examples, be included in a different box shown in the figures or in a different box not shown.

[0061] Machine-readable storage medium 520 may be any electronic, magnetic, optical, or other physical storage device that stores executable instructions. Thus,

machine-readable storage medium 520 may be, for example, Random Access Memory (RAM), an Electrically-Erasable Programmable Read-Only Memory (EEPROM), a NAND or NOR flash memory device, a storage drive, an optical disc, and the like. Machine-readable storage medium 520 may be disposed within system 500, as shown in FIG. 5. In this situation, the executable instructions may be “installed” on the system 500. In some examples, machine-readable storage medium 520 may be a portable, external or remote storage medium, for example, that allows system 500 to download the instructions from the portable/external/remote storage medium. In this situation, the executable instructions may be part of an “installation package”. As described herein, machine-readable storage medium 520 may be encoded with executable instructions for comparing image data to a reference image and, based on the comparison, determining whether an OS is in a state associated with the reference image.

[0062] Referring to system 500, notification reception instructions 522, when executed by a processor (e.g., 510), may cause system 500 to receive indications of notifications to be provided to a user via a user interface of a mobile device. Time tracking instructions 524, when executed by a processor (e.g., 510), may cause system 500 to track elapsed time since a given notification is received. In some examples, instructions 524 may cause the system 500 to track elapsed time until a given notification exceeds its maximum time delay. Context monitoring instructions 526, when executed by a processor (e.g., 510), may cause system 500 to analyze sensor data to identify contexts suitable for providing notifications. For example, the instructions 526 may cause the system 500 to identify intervals in which the system 500 changes from one context to another, which may be a desirable interval in which to present notifications. Notification prioritization and scheduling instructions 528, when executed by a process (e.g., 510), may cause system 500 to prioritize pending notifications based on their predicted degree of user engagement and select at least some of those notifications for scheduling. In some examples, the instructions 528 may cause system 500 to select a set of the most highly ranked notifications to be scheduled during each identified opportunity to provide notifications. In some examples, the instructions 528 may cause system 500 to immediately schedule any notifications for which the elapsed time since receiving them exceeds their maximum time delay.

CLAIMS

1. A method comprising:
 - receiving a request to provide a first notification via a user interface of a mobile device prior to passage of a maximum time delay;
 - tracking an elapsed time since receiving the request to provide the first notification;
 - prior to passage of the maximum time delay:
 - monitoring a context of the mobile device for an opportunity to provide notifications via the user interface, and
 - scheduling, based on the monitored context, a second notification to be provided via the user interface prior to passage of the maximum time delay;
 - determining that the elapsed time exceeds the maximum time delay; and
 - responsive to determining that the elapsed time exceeds the maximum time delay, providing the first notification via the user interface.
2. The method of claim 1, further comprising:
 - prior to passage of the maximum time delay, making a determination, based at least in part on content of the first and second notifications, that the second notification has a higher predicted degree of user engagement than the first notification.
3. The method of claim 1, wherein the second notification is one of multiple additional notifications, the method further comprising:
 - prior to passage of the maximum time delay, making a determination that the multiple additional notifications have a higher predicted degree of user engagement than the first notification, wherein the determination is made based on content of the multiple additional notifications and content of the first notification; and

scheduling, based on the monitored context, the multiple additional notifications to be provided via the user interface prior to passage of the maximum time delay.

4. The method of claim 1, further comprising:

after receiving the request to provide the first notification, receiving a request to provide the second notification prior to passage of a second maximum time delay.

5. The method of claim 1, wherein monitoring the context comprises:

receiving sensor data indicative of motion of the mobile device; and
determining that the indicated motion is associated with a period of high user engagement, and

wherein scheduling the second notification comprises:

selecting a subset of notifications from amongst a plurality of notifications that includes the first notification and the second notification, wherein the subset is selected from amongst the plurality based on content of the plurality of notifications, wherein the selected subset includes the second notification and does not include the first notification; and

providing the selected subset of notifications via the user interface during the period of high user engagement.

6. The method of claim 1, further comprising:

selecting one from amongst multiple notification provision modalities based on the monitored context of the mobile device, wherein each of the multiple notification provision modalities specifies a respective operation of the user interface, and

wherein the provision of the first notification via the user interface is performed in accordance with the selected one of the notification provision modalities.

7. The method of claim 1, further comprising:
- receiving an indication of a predicted degree of user engagement with a group of notifications, wherein the group is defined at least in part based on a similarity of content of the notifications in the group;
 - determining that the first notification is included in the group based at least in part on content of the first notification;
 - making a comparison between a predicted degree of user engagement with the second notification and the indicated predicted degree of user engagement with the group of notifications; and
 - determining, based on the comparison, that the second notification has a greater predicted degree of user engagement than the first notification, and
 - wherein scheduling the second notification, rather than the first notification, is based in part on determining that the second notification has a greater predicted degree of user engagement than the first notification.
8. A mobile device comprising:
- a user interface;
 - a processing system coupled to the user interface, the processing system to:
- receive a request to provide a first notification via the user interface prior to passage of a maximum time delay;
 - track an elapsed time since receiving the request to provide the first notification;
 - monitor a context of the mobile device for an opportunity to provide notifications via the user interface, and
 - schedule a second notification, rather than the first notification, to be provided via the user interface prior to passage of the maximum time delay;
 - determine that the elapsed time exceeds the maximum time delay;
 - and
 - responsive to determining that the elapsed time exceeds the maximum time delay, provide the first notification via the user interface.

9. The mobile device of claim 8, wherein the processing system is to:
- prior to passage of the maximum time delay, make a determination, based at least in part on content of the first and second notifications, that the second notification has a higher predicted degree of user engagement than the first notification.
10. The mobile device of claim 8, wherein the second notification is one of multiple additional notifications, and wherein the processing system is to:
- prior to passage of the maximum time delay, make a determination that the multiple additional notifications have a higher predicted degree of user engagement than the first notification, wherein the determination is made based on content of the multiple additional notifications and content of the first notification; and
 - schedule, based on the monitored context, the multiple additional notifications to be provided via the user interface prior to passage of the maximum time delay.
11. The mobile device of claim 8, wherein the processing system is to:
- after receiving the request to provide the first notification, receive a request to provide the second notification prior to passage of a second maximum time delay.
12. A non-transitory machine-readable storage medium encoded with instructions executable by a processor of a system to cause the system to:
- receive a request to provide a first notification via the user interface prior to passage of a maximum time delay;
 - track an elapsed time since receiving the request to provide the first notification;
 - receive sensor data indicative of motion of the mobile device;
 - determine that the indicated motion is associated with a period of high user engagement;
 - select a subset of notifications from amongst a plurality of notifications that includes the first notification and the second notification, wherein the subset is

selected from amongst the plurality based on content of the plurality of notifications, wherein the selected subset includes the second notification and does not include the first notification;

provide the selected subset of notifications via the user interface during the period of high user engagement;

determine that the elapsed time exceeds the maximum time delay; and

responsive to determining that the elapsed time exceeds the maximum time delay, provide the first notification via the user interface.

13. The non-transitory machine-readable storage medium of claim 12, wherein the instructions further cause the system to:

prior to passage of the maximum time delay, make a determination, based at least in part on content of the first and second notifications, that the second notification has a higher predicted degree of user engagement than the first notification.

14. The non-transitory machine-readable storage medium of claim 12, wherein the instructions further cause the system to:

prior to passage of the maximum time delay, make a determination that the multiple additional notifications have a higher predicted degree of user engagement than the first notification, wherein the determination is made based on content of the multiple additional notifications and content of the first notification; and

schedule, based on the monitored context, the multiple additional notifications to be provided via the user interface prior to passage of the maximum time delay.

15. The non-transitory machine-readable storage medium of claim 12, wherein the instructions further cause the system to:

after receiving the request to provide the first notification, receive a request to provide the second notification prior to passage of a second maximum time delay.

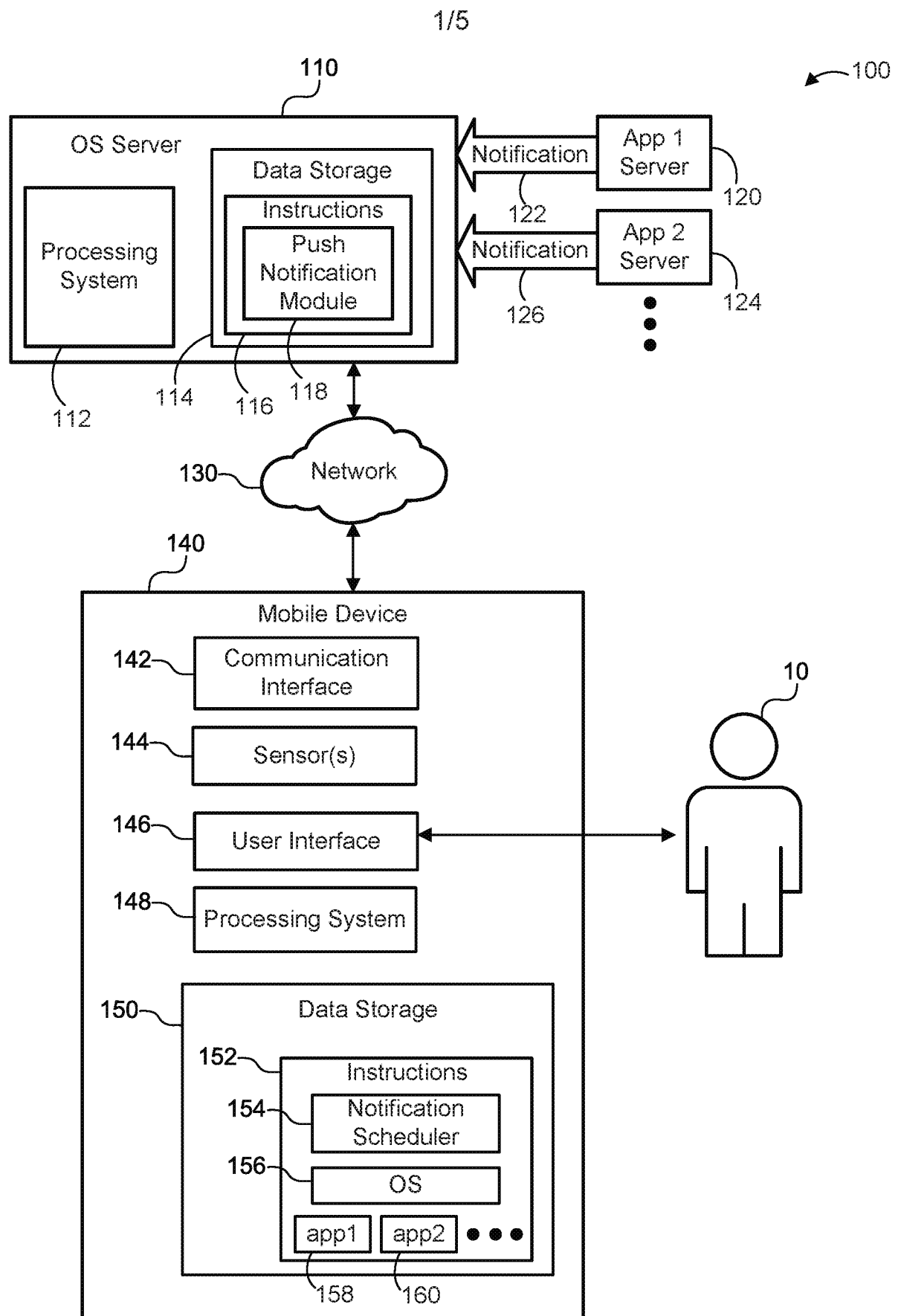


FIG. 1

2/5

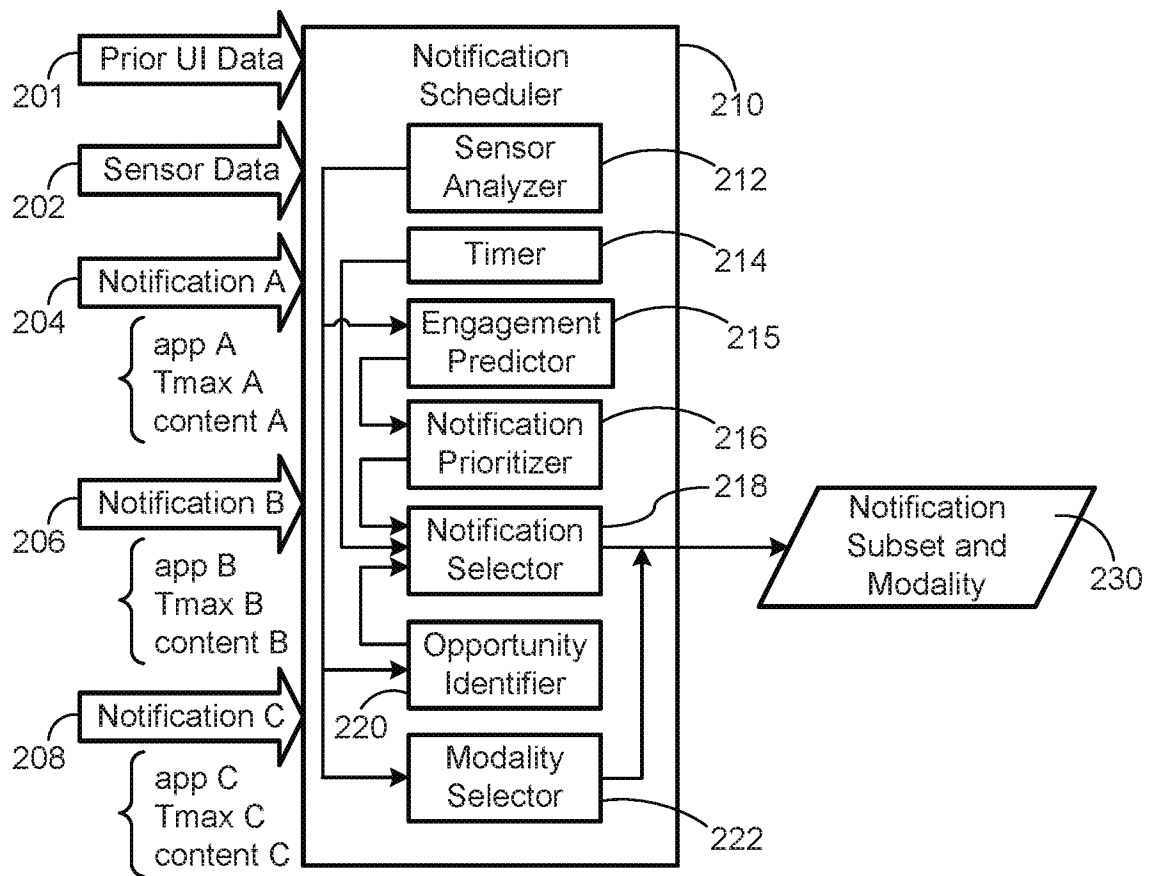


FIG. 2A

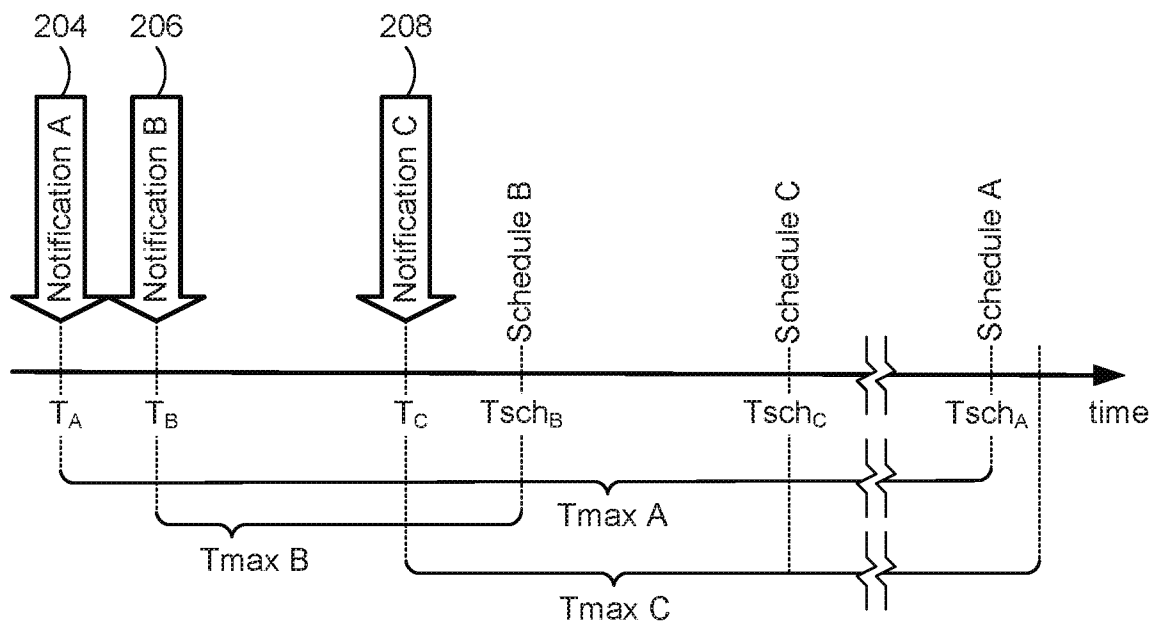


FIG. 2B

3/5

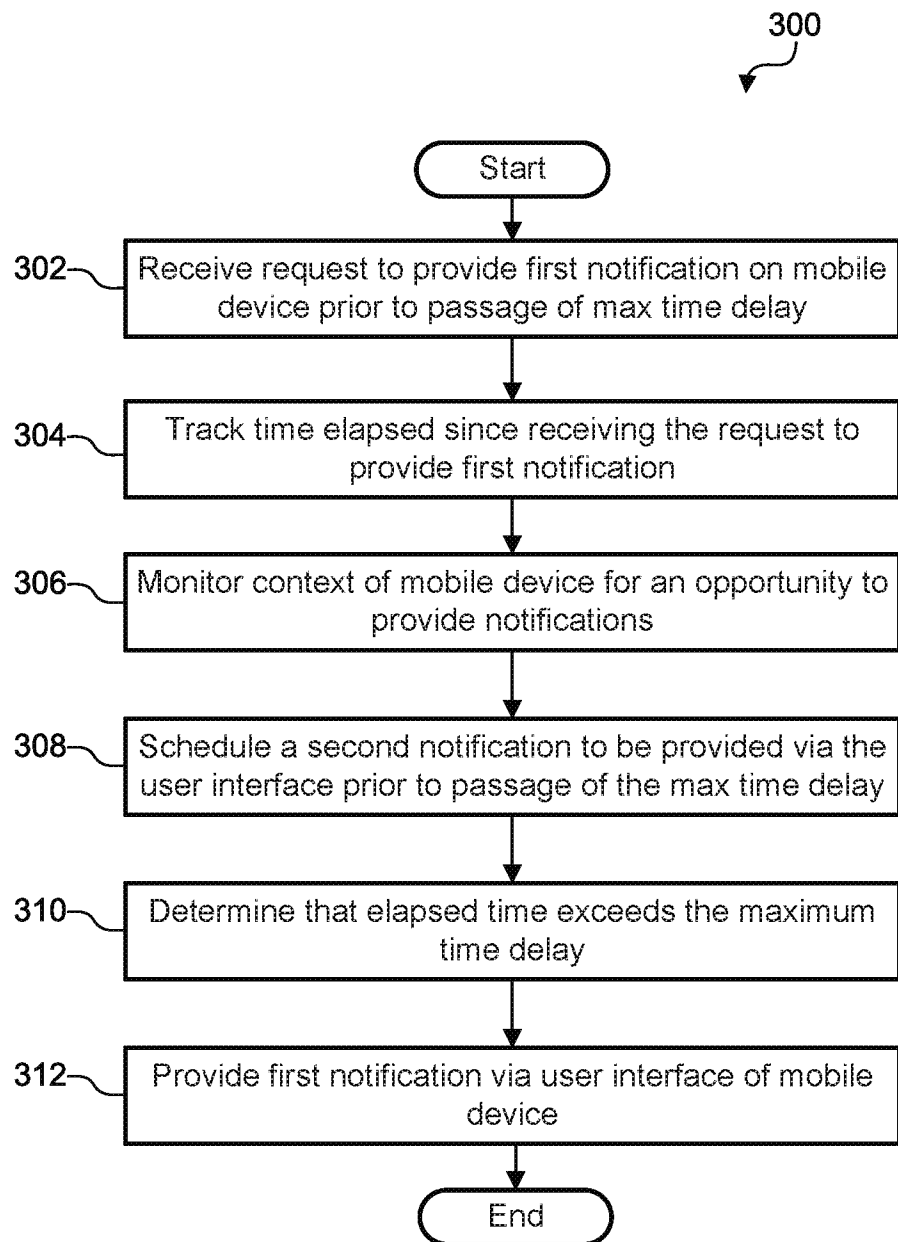


FIG. 3

4/5

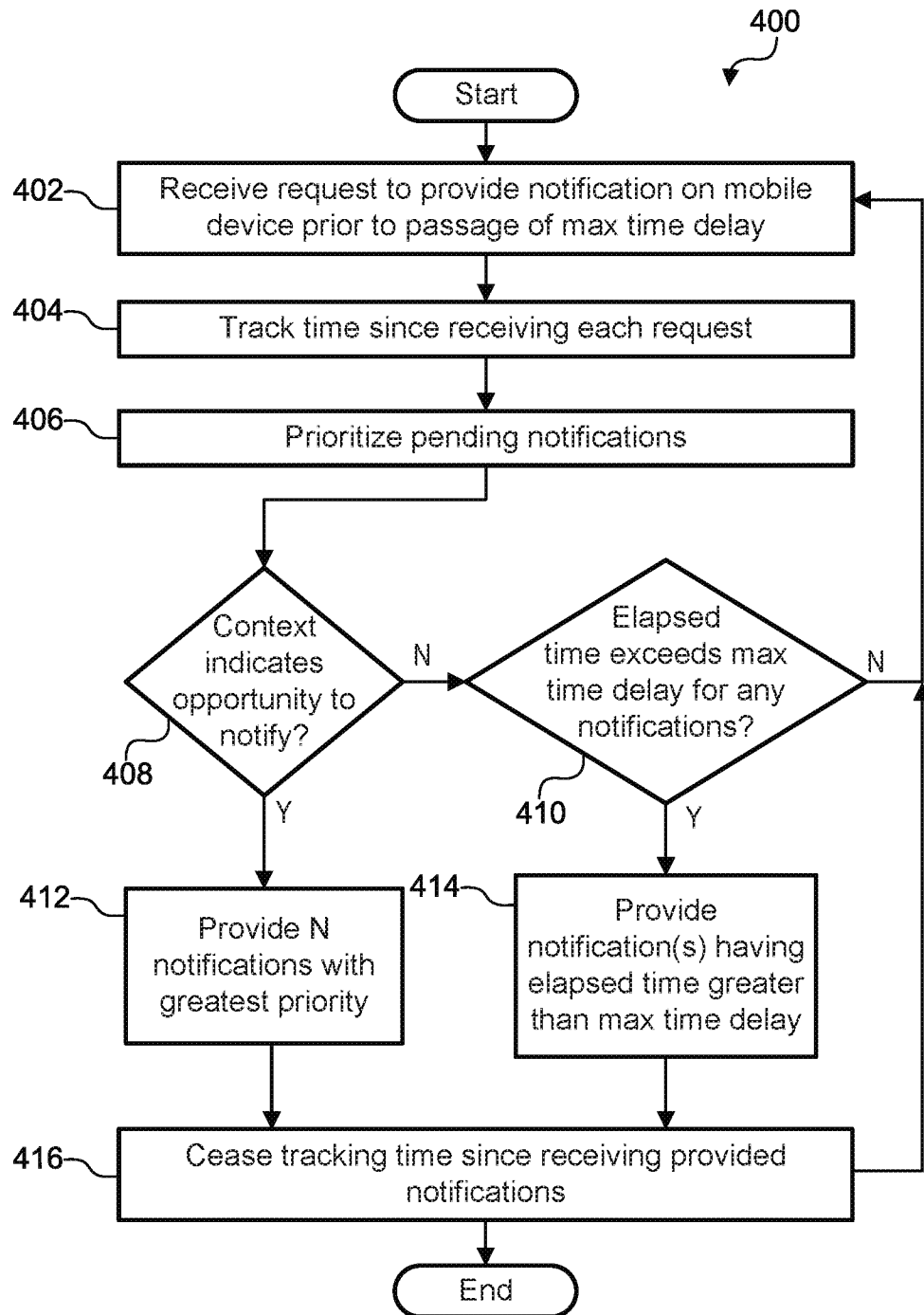


FIG. 4

5/5

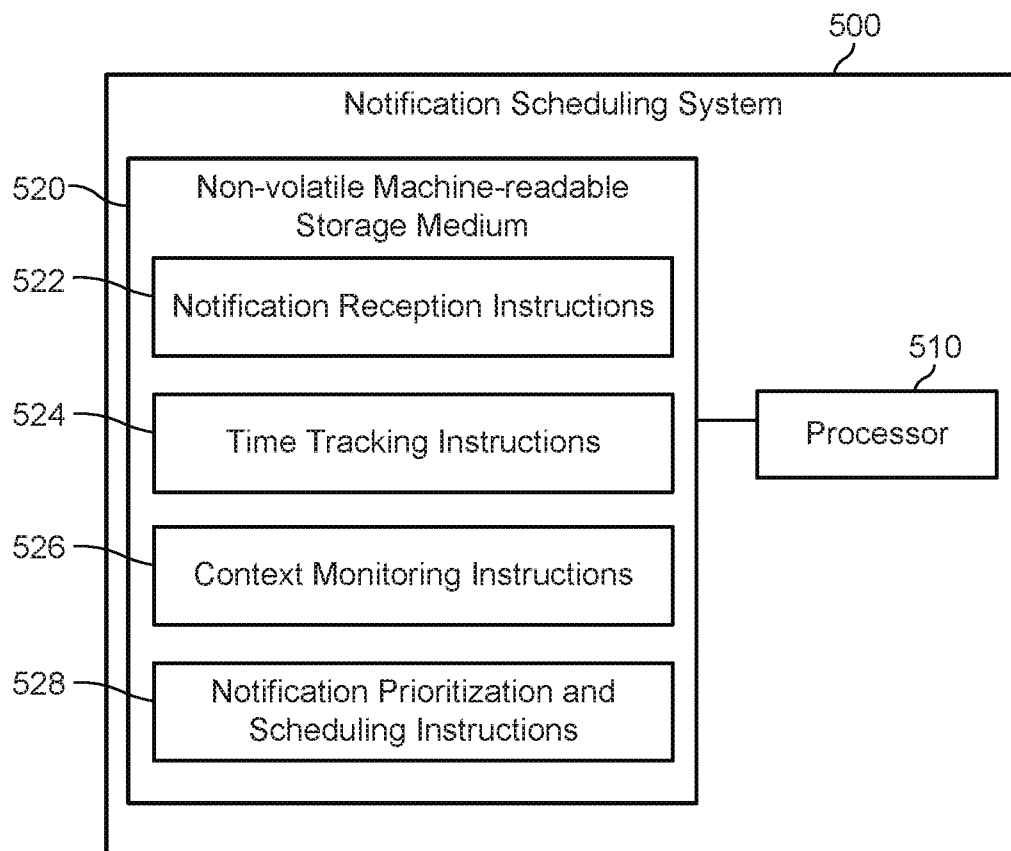


FIG. 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2016/015664**A. CLASSIFICATION OF SUBJECT MATTER****H04L 29/06(2006.01)i, H04L 29/08(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 29/06; H04L 12/58; H04W 4/12; G06F 17/00; G06F 17/30; H04M 19/04; G06F 15/18; H04M 1/663; H04L 29/08

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) & Keywords: notification, time delay, elapsed time, context, user engagement, priority, machine learning

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2015-0195240 A1 (GOOGLE INC.) 09 July 2015 See paragraphs [0057]-[0058], [0082]-[0088]; and figure 5.	1-15
A	US 2014-0229880 A1 (GOOGLE INC.) 14 August 2014 See paragraphs [0025]-[0032]; and figure 2.	1-15
A	US 2012-0149342 A1 (GABRIEL COHEN et al.) 14 June 2012 See paragraph [0121]; and figure 6A.	1-15
A	US 8886576 B1 (PANNAG RAGHUNATH SANKETI et al.) 11 November 2014 See columns 8-9; and figure 1A.	1-15
A	WO 2013-180892 A1 (QUALCOMM INCORPORATED) 05 December 2013 See paragraph [0070]; and figure 9.	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

"&" document member of the same patent family

Date of the actual completion of the international search

07 October 2016 (07.10.2016)

Date of mailing of the international search report

17 October 2016 (17.10.2016)

Name and mailing address of the ISA/KR

International Application Division

Korean Intellectual Property Office

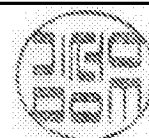
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-481-8578

Authorized officer

KIM, Seong Woo

Telephone No. +82-42-481-3348



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2016/015664

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2015-0195240 A1	09/07/2015	US 2015-0145669 A1 US 9024747 B1 WO 2015-080792 A2 WO 2015-080792 A3	28/05/2015 05/05/2015 04/06/2015 01/10/2015
US 2014-0229880 A1	14/08/2014	US 8707201 B1	22/04/2014
US 2012-0149342 A1	14/06/2012	AU 2011-338267 A1 AU 2011-338267 B2 CN 103493438 A EP 2649762 A1 JP 2014-501979 A JP 2015-015038 A KR 10-1563248 B1 KR 10-2013-0135283 A US 2013-0041968 A1 US 8312096 B2 US 8935347 B2 WO 2012-078920 A1	11/07/2013 14/08/2014 01/01/2014 16/10/2013 23/01/2014 22/01/2015 26/10/2015 10/12/2013 14/02/2013 13/11/2012 13/01/2015 14/06/2012
US 8886576 B1	11/11/2014	None	
WO 2013-180892 A1	05/12/2013	US 2013-0316744 A1 US 8948821 B2	28/11/2013 03/02/2015