US 20200107164A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2020/0107164 A1**

Lopatin et al. (43) **Pub. Date: Apr. 2, 2020**

(54) **SYSTEM AND METHOD FOR LOCATING WIRELESS ACCESSORIES**

(71) Applicant: **Apple Inc.**, Cupertino, CA (US)

(72) Inventors: **Scott Lopatin**, San Francisco, CA (US); **Emmanuel Lalande**, San Francisco, CA (US); **Lawrence T. Lu**, San Francisco, CA (US); **Robert W. Mayor**, Half Moon Bay, CA (US); **Siva Ganesh Movva**, Cupertino, CA (US); **Raghunandan K. Pai**, Saratoga, CA (US); **Munish K. Poonia**, San Jose, CA (US); **Tommy Rochette**, San Jose, CA (US); **Sam Schaevitz**, Los Gatos, CA (US); **Yannick L. Sierra**, San Francisco, CA (US); **Navin Bindiganavile Suparna**, Santa Clara, CA (US); **Langford Wasada**, San Francisco, CA (US)

(21) Appl. No.: **16/543,227**

(22) Filed: **Aug. 16, 2019**

Related U.S. Application Data

(60) Provisional application No. 62/738,983, filed on Sep. 28, 2018.

Publication Classification

(51) **Int. Cl.**

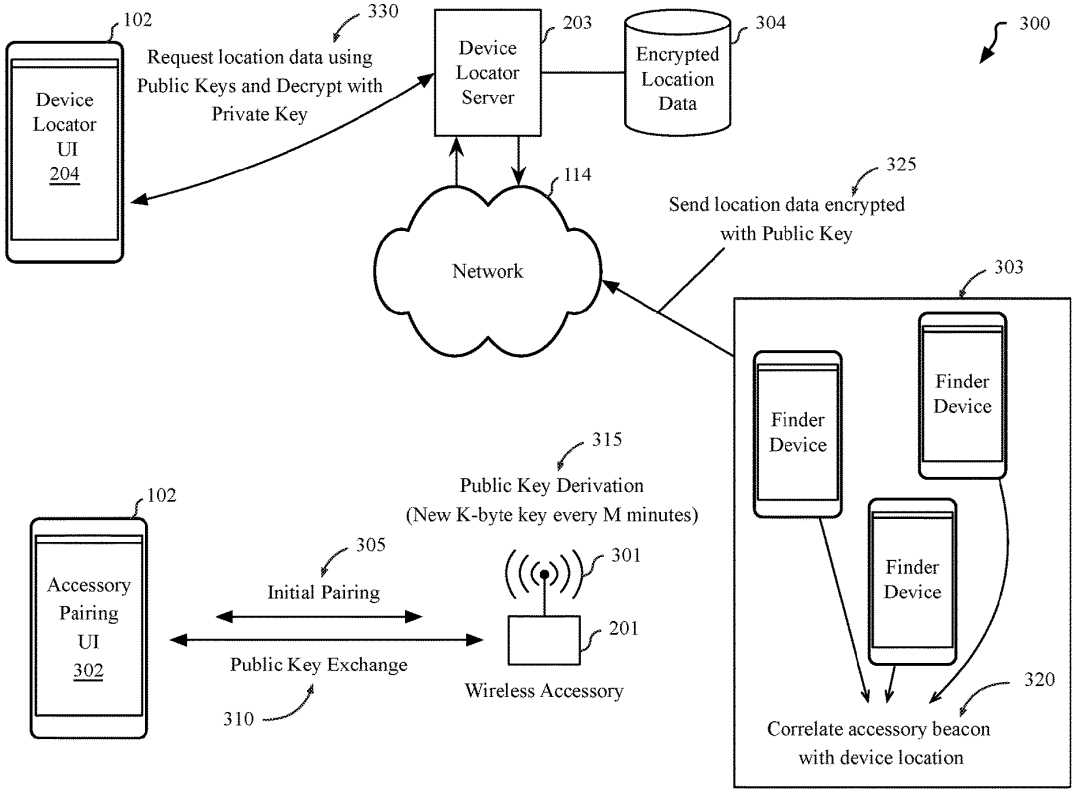
H04W 4/029	(2006.01)
H04W 4/80	(2006.01)
H04W 12/00	(2006.01)
H04W 12/12	(2006.01)

(52) **U.S. Cl.**

CPC ***H04W 4/029*** (2018.02); ***H04W 12/1206*** (2019.01); ***H04W 12/003*** (2019.01); ***H04W 4/80*** (2018.02)

(57) **ABSTRACT**

Embodiments described herein provide for system and methods to crowdsource the location of wireless devices and accessories that lack a connection to a wide area network. One embodiment provides for a data processing system configured to perform operations comprising loading a user interface on an electronic device, the user interface to enable the determination of a location of a wireless accessory that is associated with the electronic device, generating a set of public keys included within a signal broadcast by the wireless accessory, the signal broadcast during a first period, sending the set of public keys to a server with a request to return data that corresponds with a public key in the set of public keys, decrypting the location data using a private key associated with the public key, and processing the location data to determine a probable location for the wireless accessory.



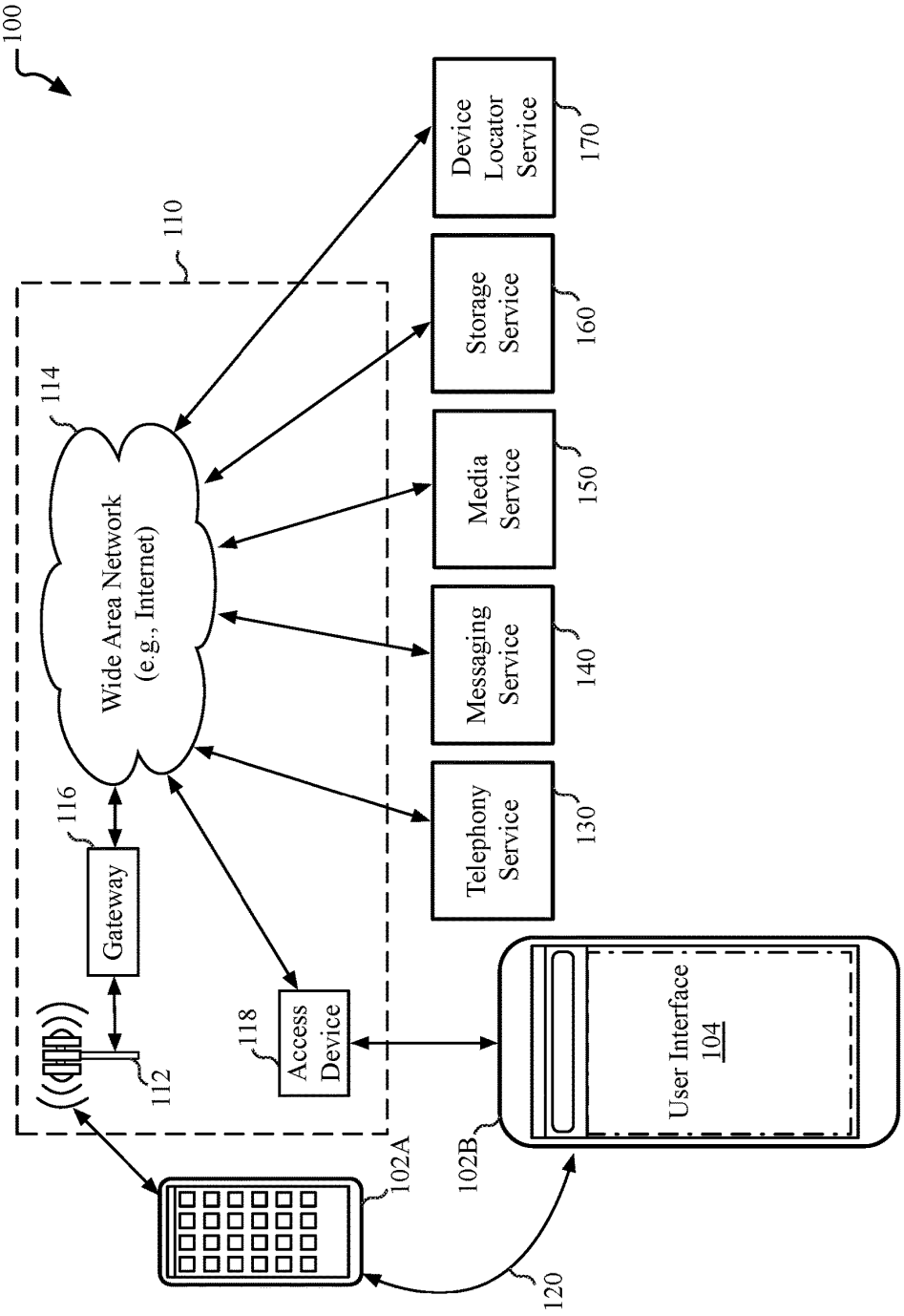

100
110
114
Wide Area Network (e.g., Internet)
116
Gateway
112
118
Access Device
Device Locator Service
170
Storage Service
160
Media Service
150
Messaging Service
140
Telephony Service
130
User Interface 104
102A
102B
120

FIG. 1

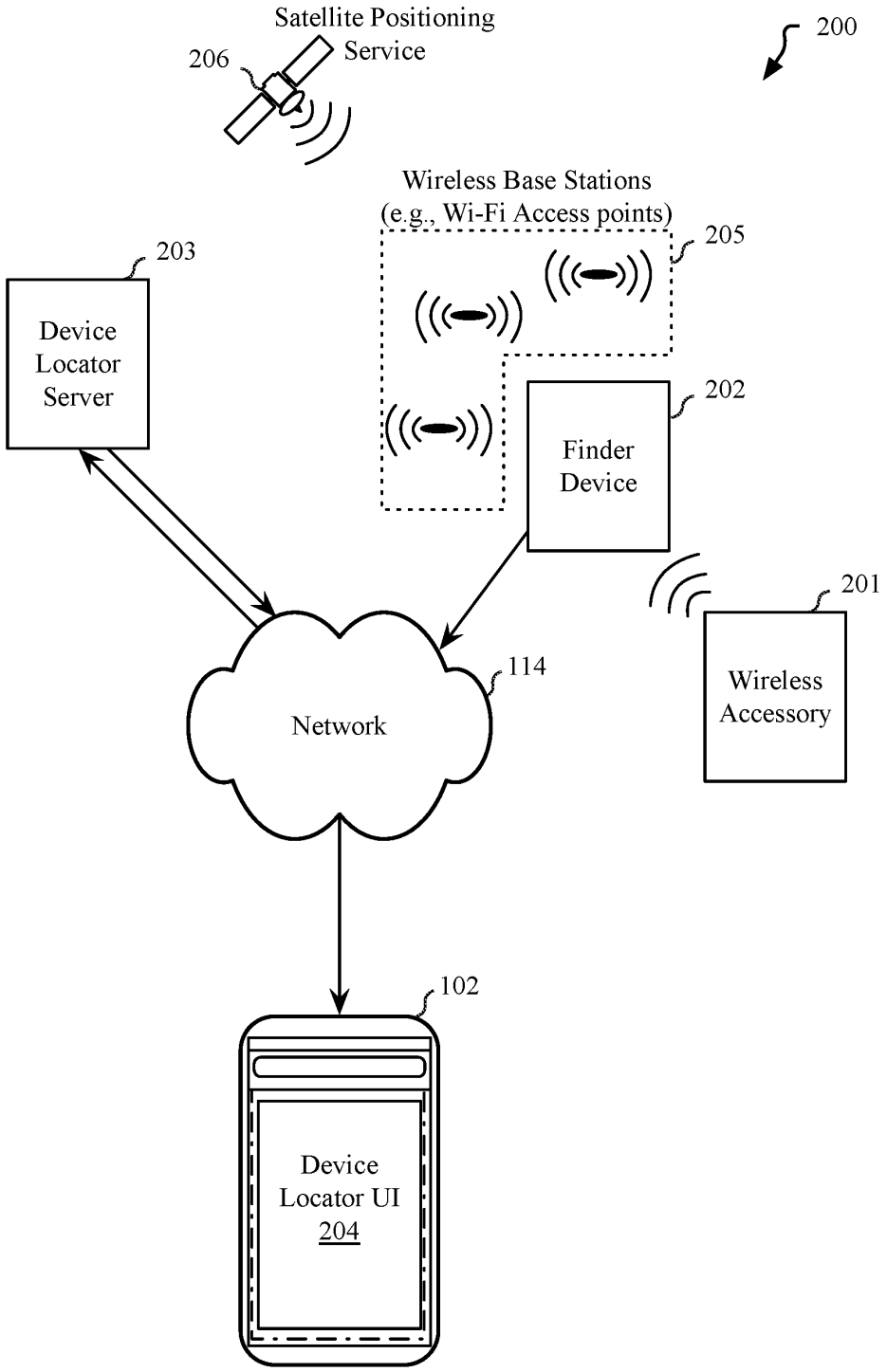

Satellite Positioning Service
206
200
Wireless Base Stations
(e.g., Wi-Fi Access points)
205
203
Device Locator Server
202
Finder Device
201
Wireless Accessory
114
Network
102
Device Locator UI
204

FIG. 2

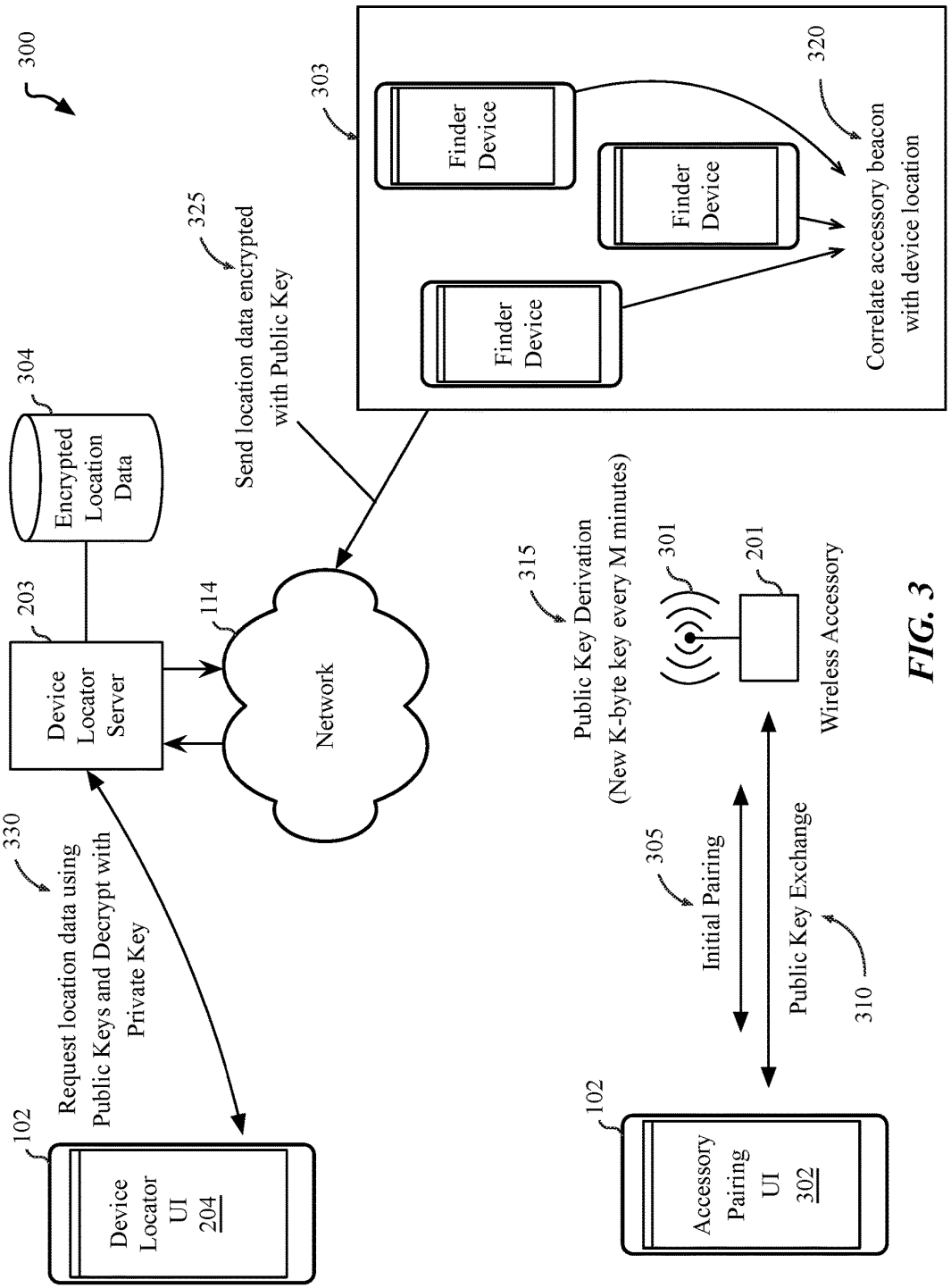

300
303
Finder Device
Finder Device
Finder Device
320
Correlate accessory beacon with device location
325
Send location data encrypted with Public Key
304
Encrypted Location Data
203
Device Locator Server
114
Network
330
Request location data using Public Keys and Decrypt with Private Key
102
Device Locator UI 204
315
Public Key Derivation (New K-byte key every M minutes)
301
201
Wireless Accessory
305
Initial Pairing
Public Key Exchange
310
102
Accessory Pairing UI 302

FIG. 3

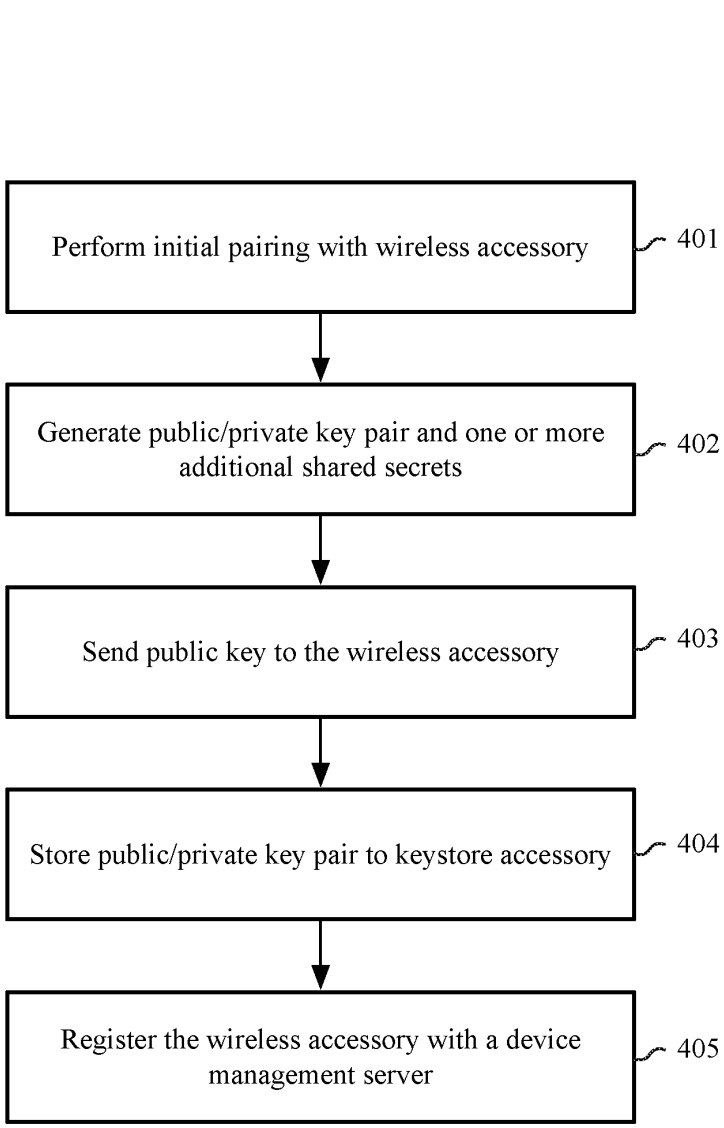

400
Perform initial pairing with wireless accessory
401
Generate public/private key pair and one or more additional shared secrets
402
Send public key to the wireless accessory
403
Store public/private key pair to keystore accessory
404
Register the wireless accessory with a device management server
405

FIG. 4A

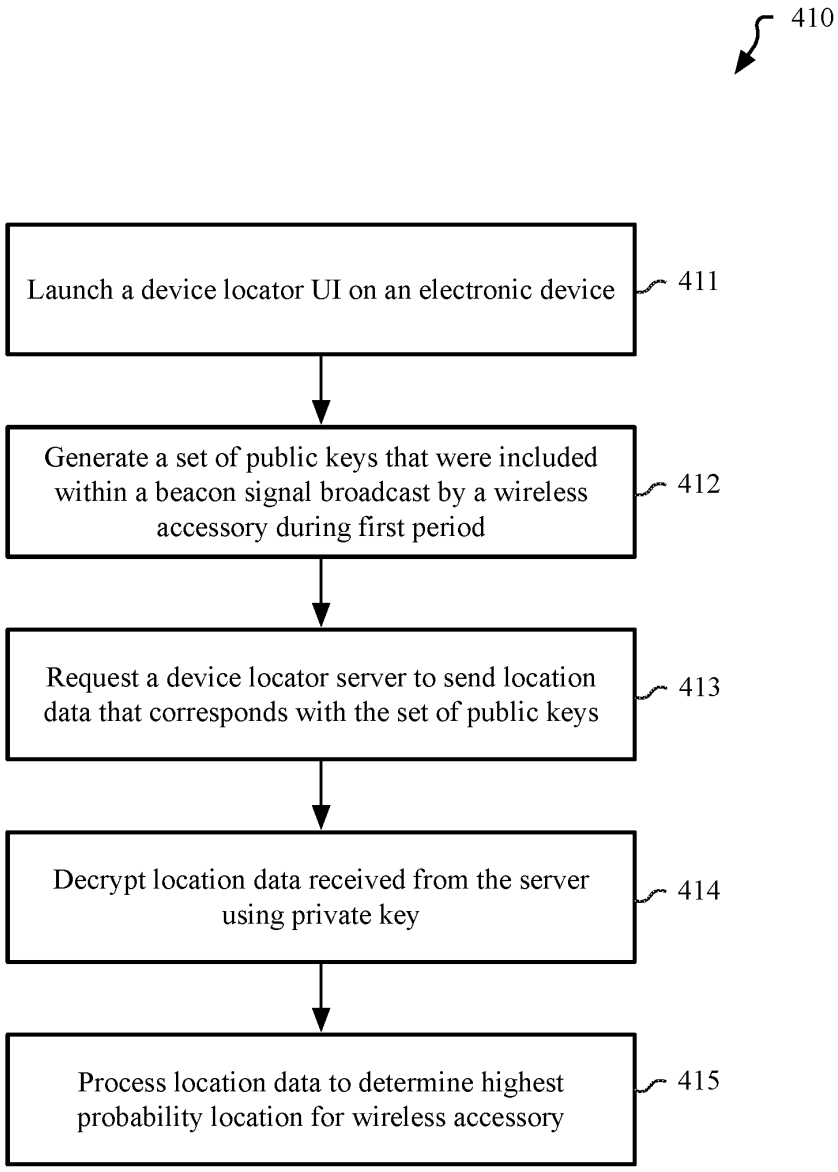

410
Launch a device locator UI on an electronic device
411
Generate a set of public keys that were included within a beacon signal broadcast by a wireless accessory during first period
412
Request a device locator server to send location data that corresponds with the set of public keys
413
Decrypt location data received from the server using private key
414
Process location data to determine highest probability location for wireless accessory
415

FIG. 4B

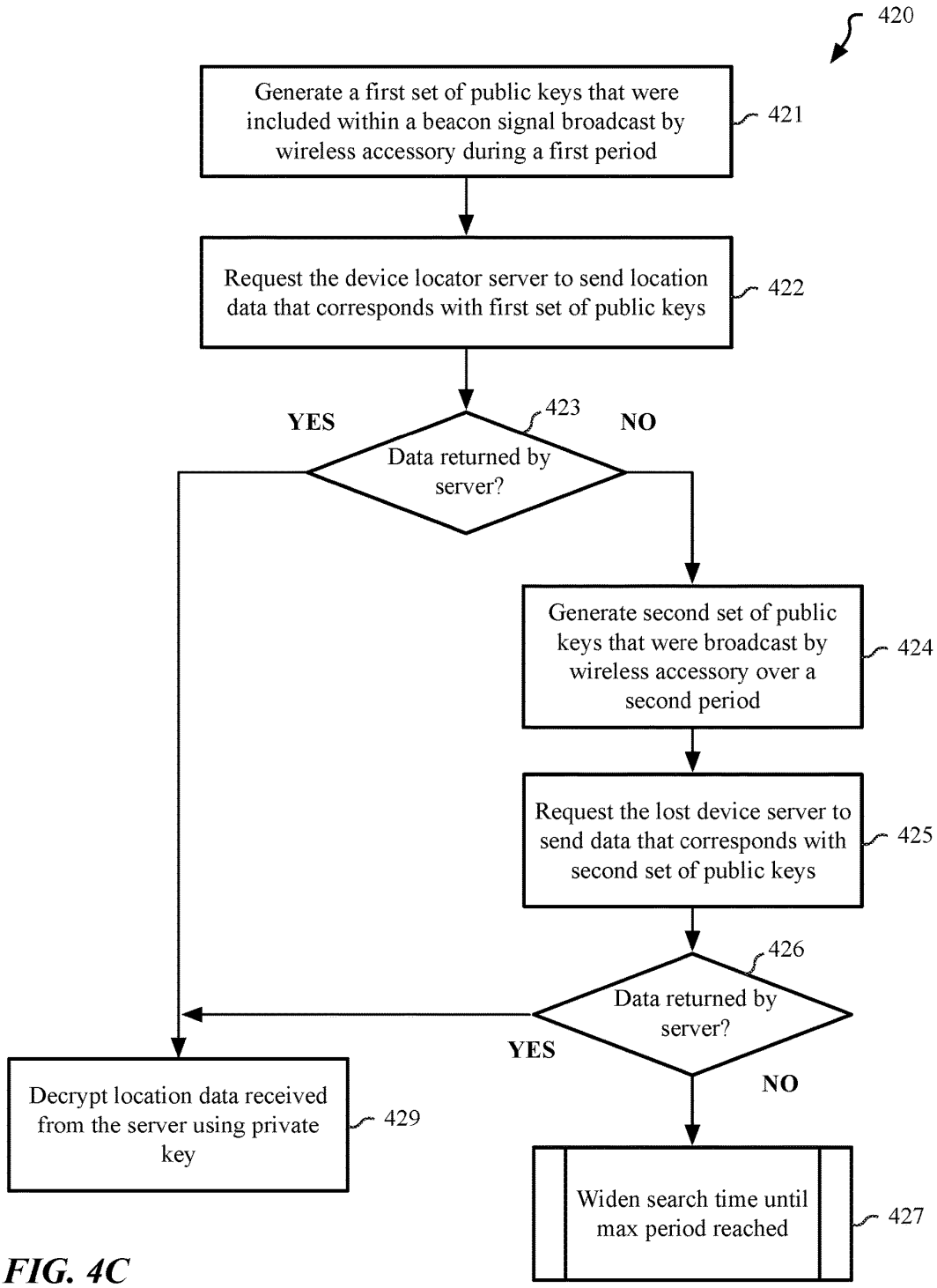

420
Generate a first set of public keys that were included within a beacon signal broadcast by wireless accessory during a first period
421
Request the device locator server to send location data that corresponds with first set of public keys
422
YES
423
NO
Data returned by server?
Generate second set of public keys that were broadcast by wireless accessory over a second period
424
Request the lost device server to send data that corresponds with second set of public keys
425
426
Data returned by server?
YES
NO
Decrypt location data received from the server using private key
429
Widen search time until max period reached
427

FIG. 4C

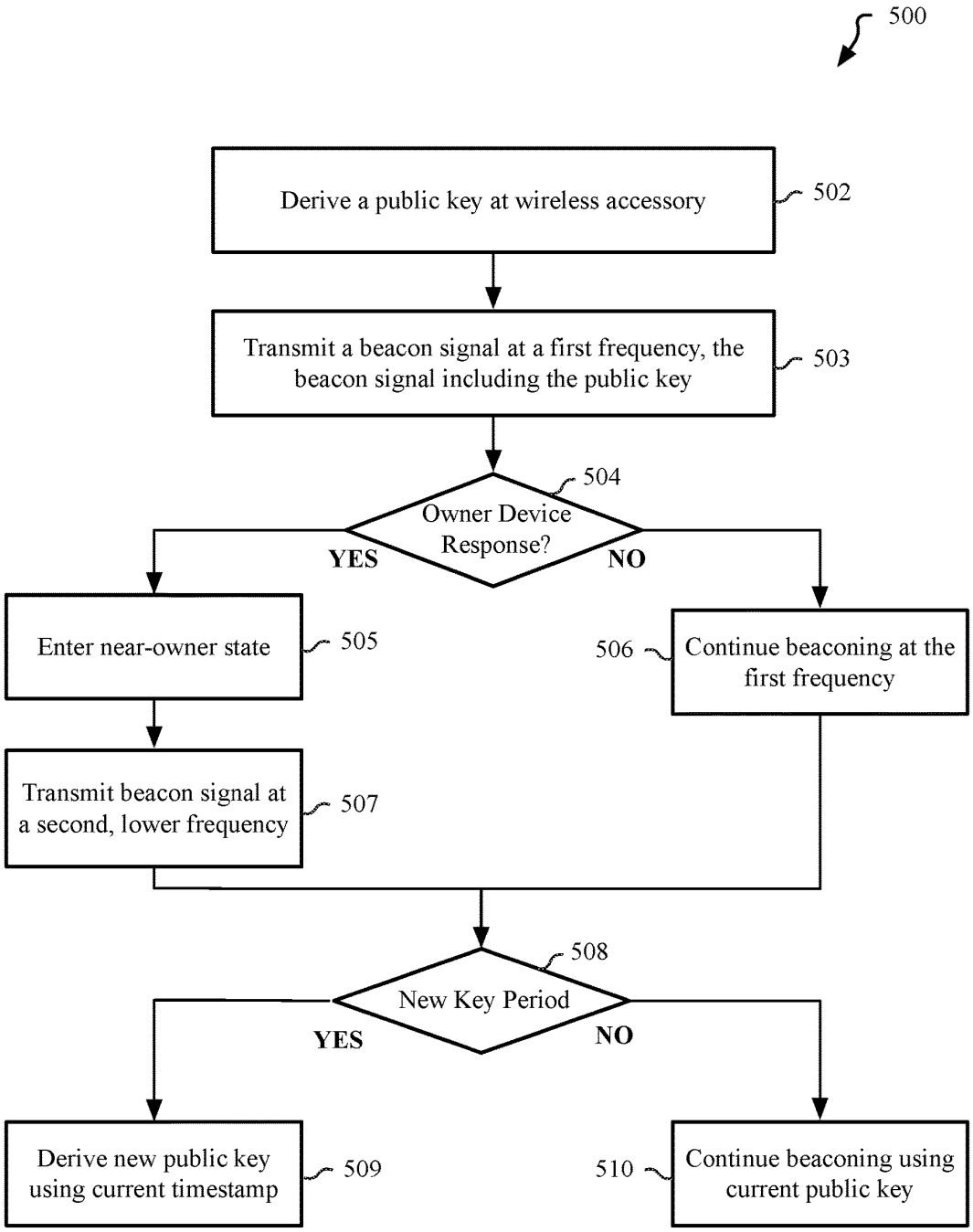

500
Derive a public key at wireless accessory
502
Transmit a beacon signal at a first frequency, the beacon signal including the public key
503
504
Owner Device Response?
YES
NO
Enter near-owner state
505
506
Continue beaconing at the first frequency
Transmit beacon signal at a second, lower frequency
507
508
New Key Period
YES
NO
Derive new public key using current timestamp
509
510
Continue beaconing using current public key

FIG. 5

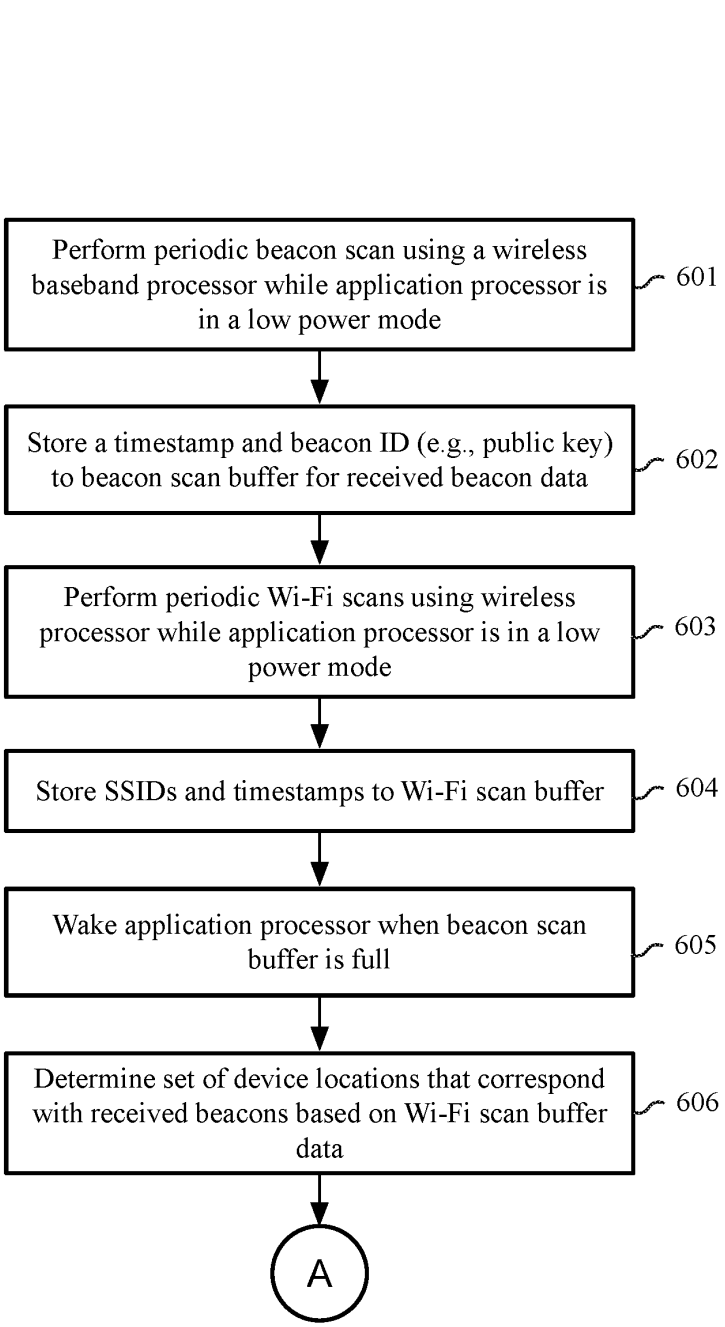

600
Perform periodic beacon scan using a wireless baseband processor while application processor is in a low power mode
601
Store a timestamp and beacon ID (e.g., public key) to beacon scan buffer for received beacon data
602
Perform periodic Wi-Fi scans using wireless processor while application processor is in a low power mode
603
Store SSIDs and timestamps to Wi-Fi scan buffer
604
Wake application processor when beacon scan buffer is full
605
Determine set of device locations that correspond with received beacons based on Wi-Fi scan buffer data
606
A

FIG. 6A

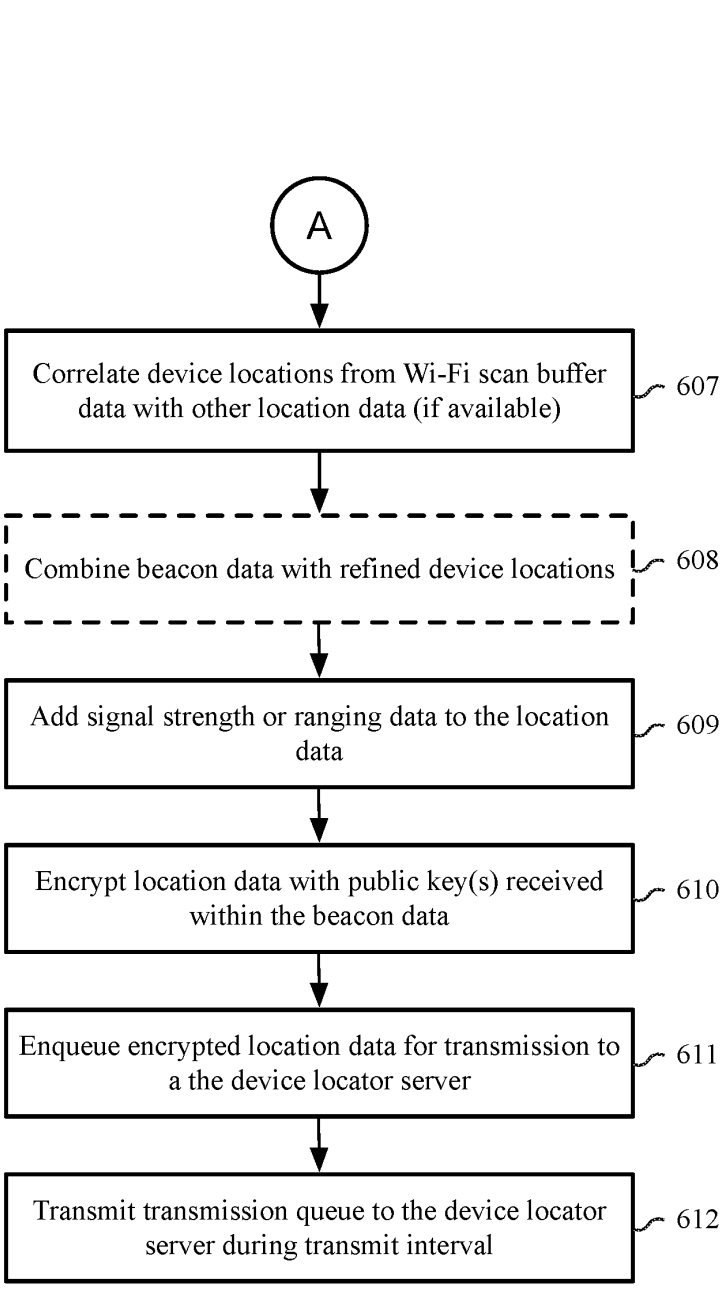

600
A
Correlate device locations from Wi-Fi scan buffer data with other location data (if available)
607
Combine beacon data with refined device locations
608
Add signal strength or ranging data to the location data
609
Encrypt location data with public key(s) received within the beacon data
610
Enqueue encrypted location data for transmission to a the device locator server
611
Transmit transmission queue to the device locator server during transmit interval
612

FIG. 6B

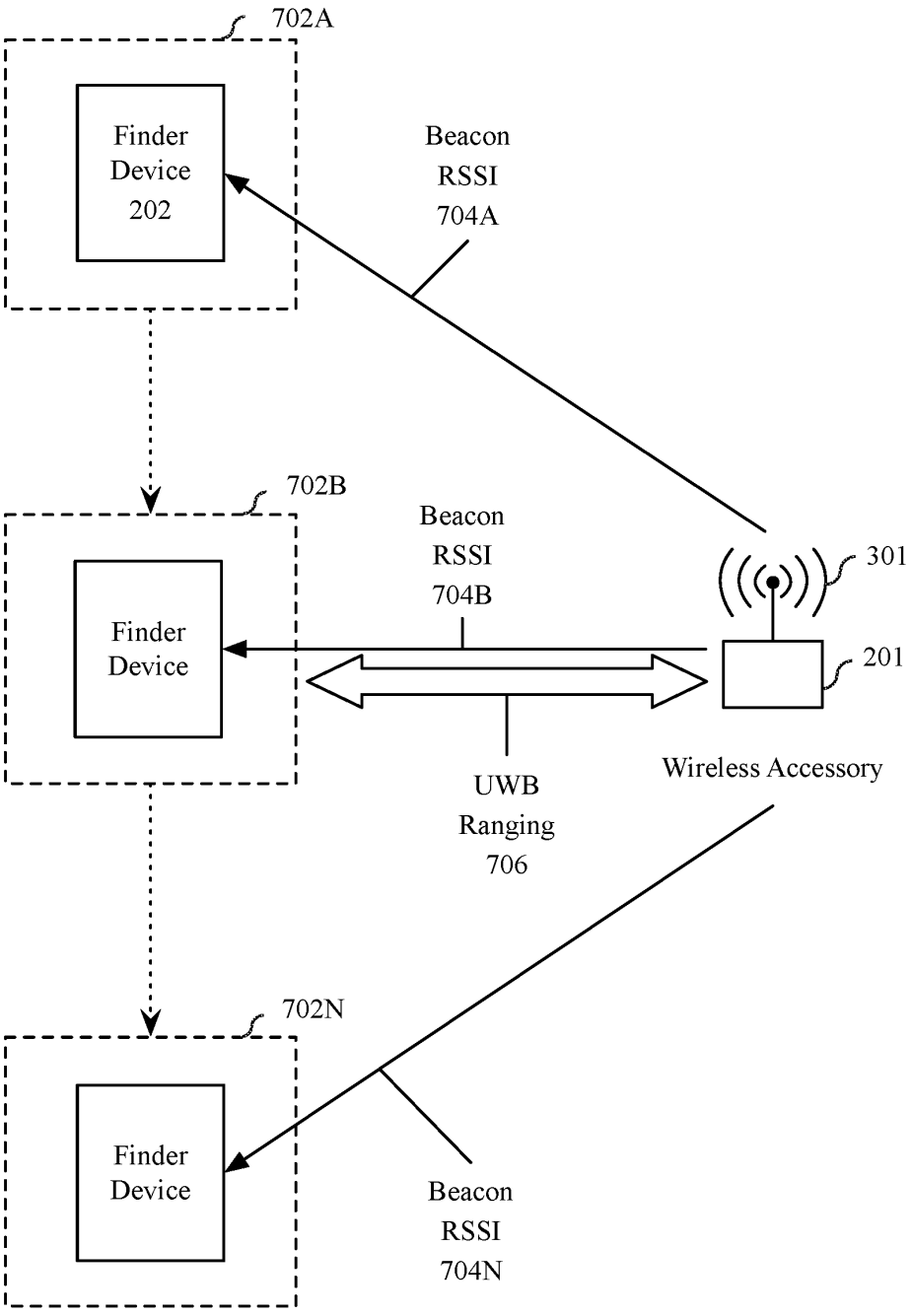

702A
Finder Device 202
Beacon RSSI 704A
702B
Finder Device
Beacon RSSI 704B
301
201
Wireless Accessory
UWB Ranging 706
702N
Finder Device
Beacon RSSI 704N

FIG. 7

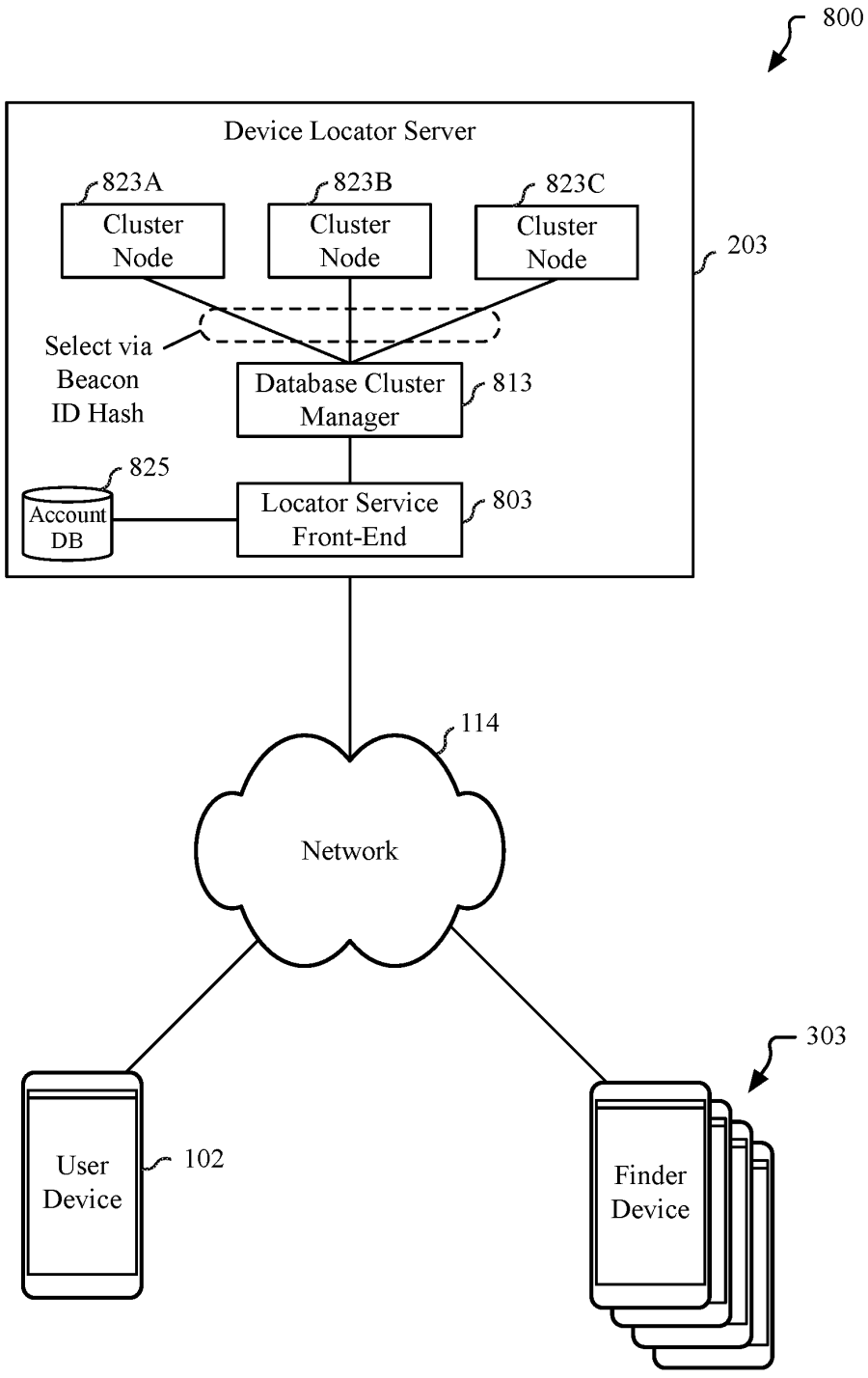

800
Device Locator Server
823A
Cluster Node
823B
Cluster Node
823C
Cluster Node
203
Select via Beacon ID Hash
Database Cluster Manager
813
825
Account DB
Locator Service Front-End
803
114
Network
303
User Device
102
Finder Device

FIG. 8

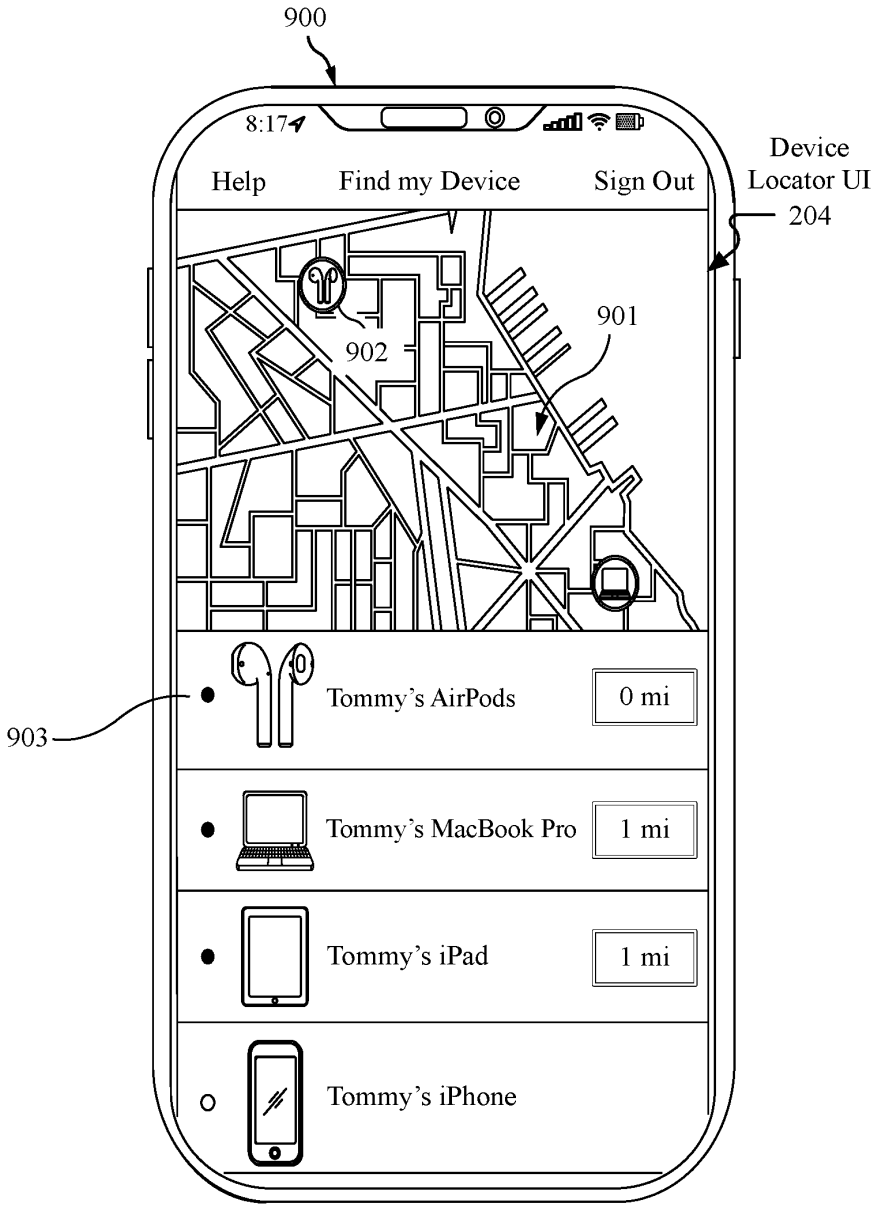

900
8:17
Help
Find my Device
Sign Out
Device Locator UI
204
901
902
903
Tommy's AirPods
0 mi
Tommy's MacBook Pro
1 mi
Tommy's iPad
1 mi
Tommy's iPhone

FIG. 9A

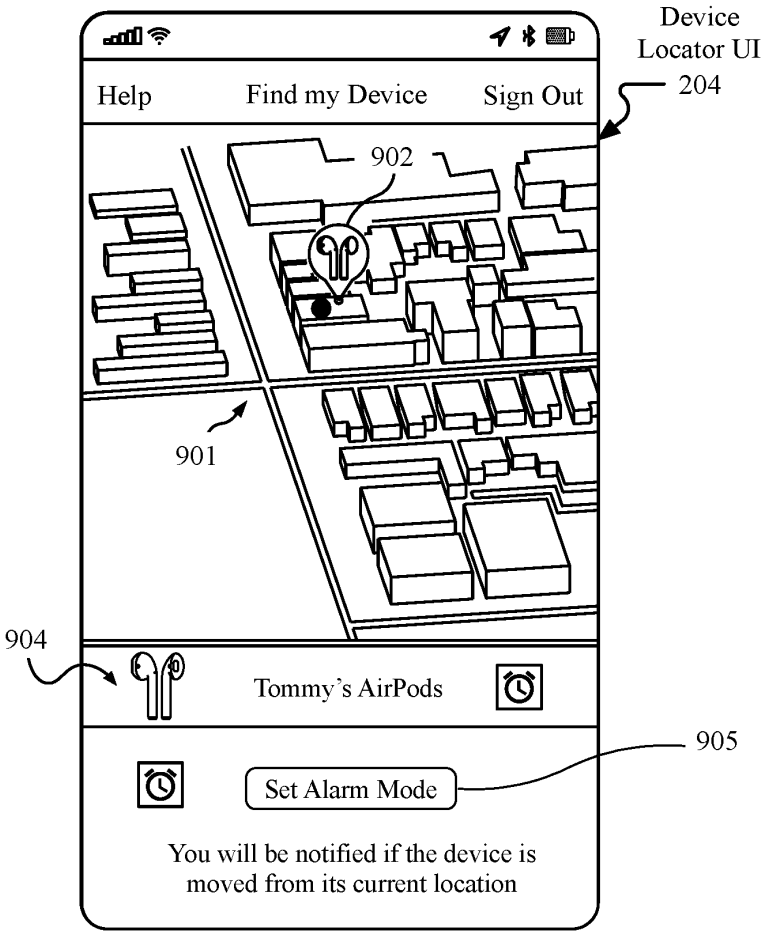

Device
Locator UI
204
Help
Find my Device
Sign Out
902
901
904
Tommy's AirPods
905
Set Alarm Mode
You will be notified if the device is
moved from its current location

FIG. 9B

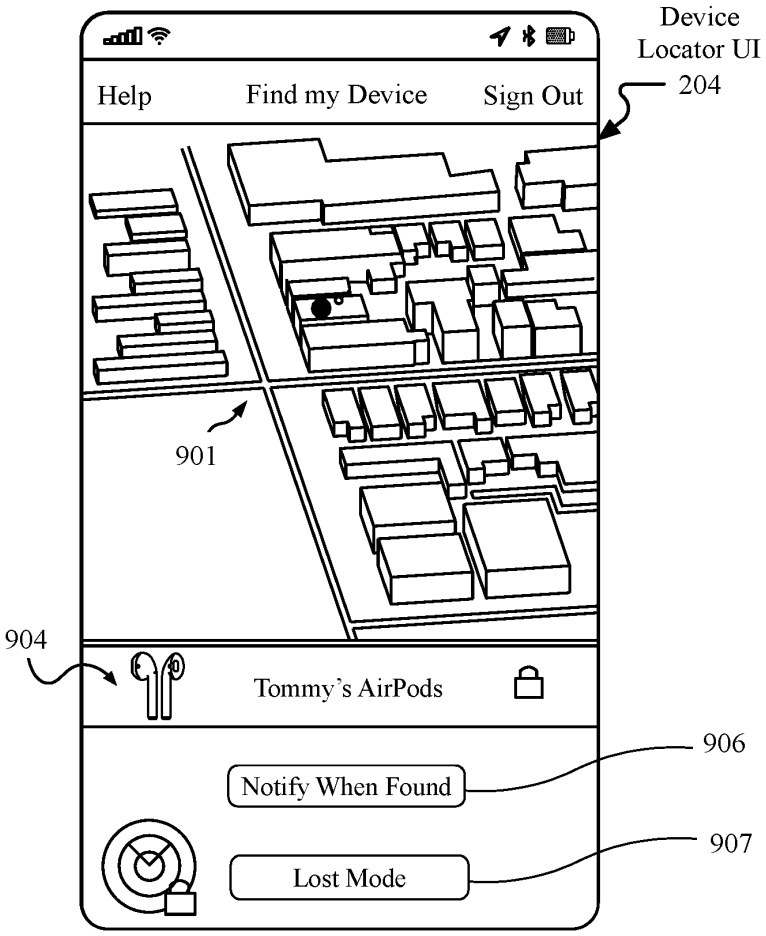

Device
Locator UI
204
Help
Find my Device
Sign Out
901
904
Tommy's AirPods
906
Notify When Found
907
Lost Mode

FIG. 9C

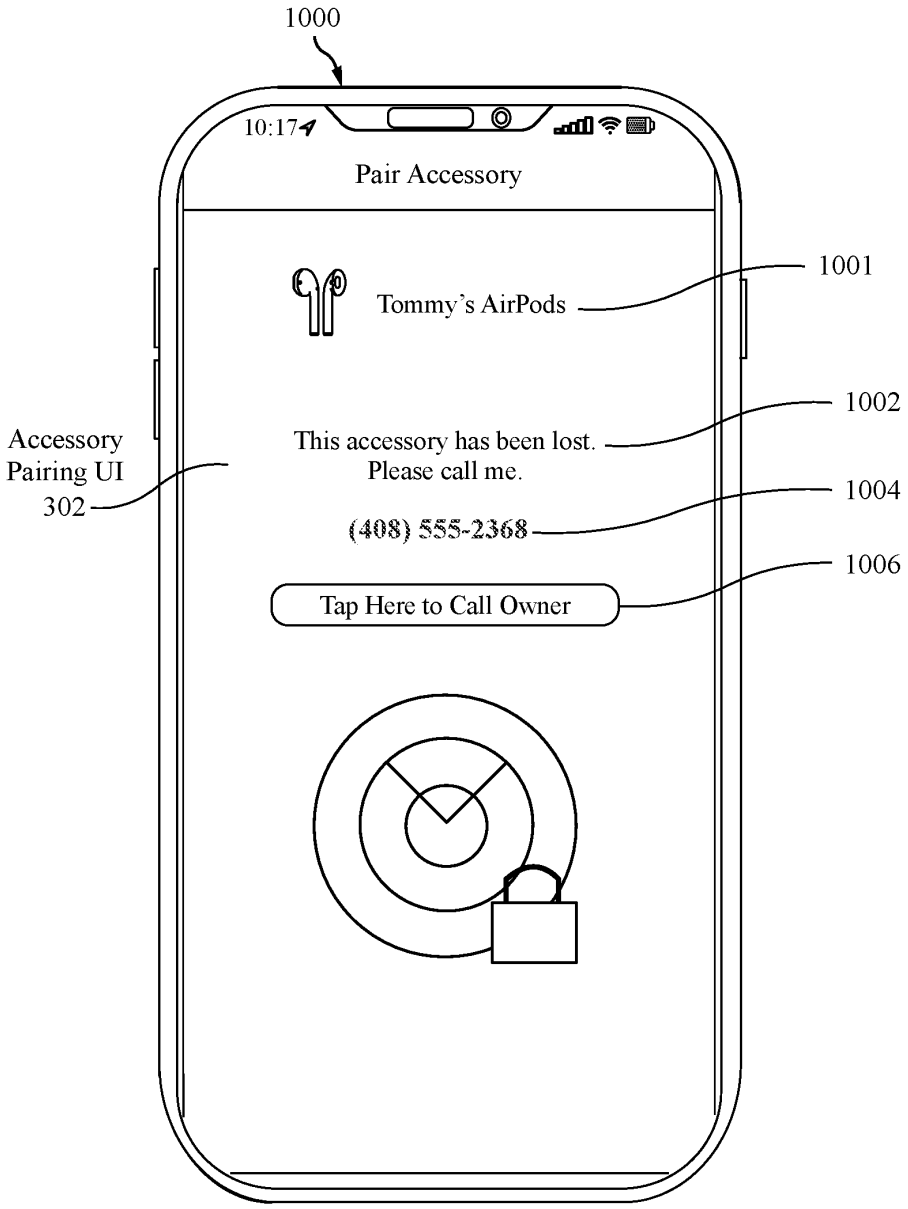

1000
10:17
Pair Accessory
Tommy's AirPods
1001
1002
Accessory
Pairing UI
302
This accessory has been lost.
Please call me.
1004
(408) 555-2368
1006
Tap Here to Call Owner

FIG. 10

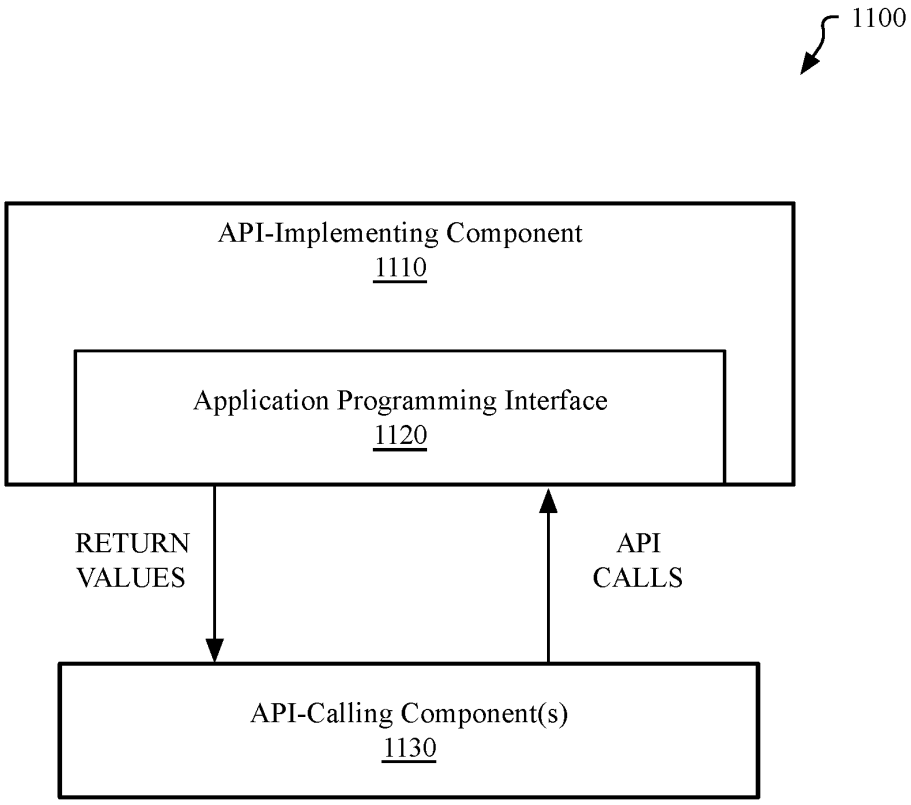

1100
API-Implementing Component
1110
Application Programming Interface
1120
RETURN
VALUES
API
CALLS
API-Calling Component(s)
1130

Fig. 11

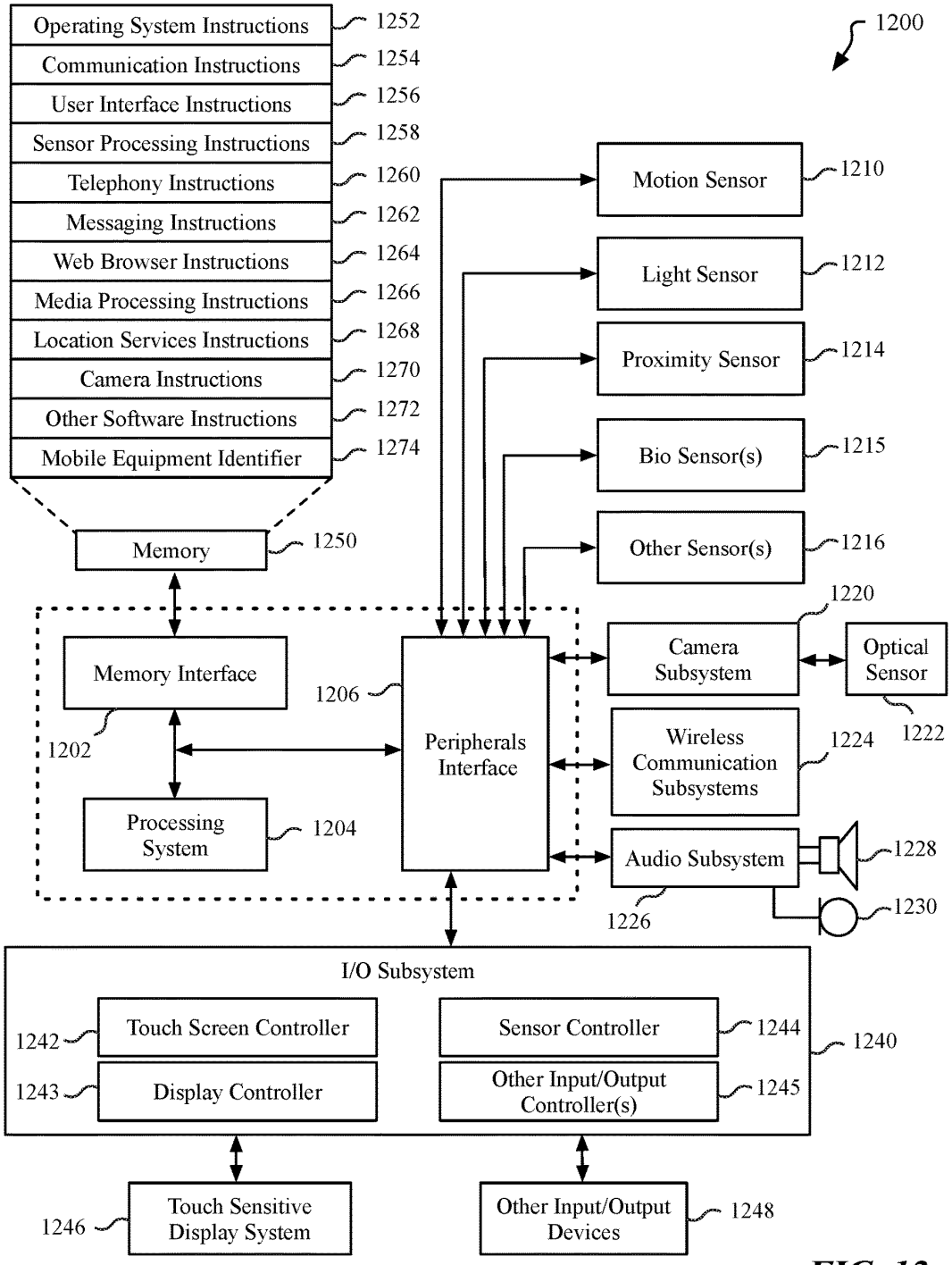

1200
Operating System Instructions
1252
Communication Instructions
1254
User Interface Instructions
1256
Sensor Processing Instructions
1258
Telephony Instructions
1260
Messaging Instructions
1262
Web Browser Instructions
1264
Media Processing Instructions
1266
Location Services Instructions
1268
Camera Instructions
1270
Other Software Instructions
1272
Mobile Equipment Identifier
1274
Memory
1250
Memory Interface
1202
Processing System
1204
1206
Peripherals Interface
Motion Sensor
1210
Light Sensor
1212
Proximity Sensor
1214
Bio Sensor(s)
1215
Other Sensor(s)
1216
1220
Camera Subsystem
Optical Sensor
1222
Wireless Communication Subsystems
1224
Audio Subsystem
1228
1230
1226
I/O Subsystem
1242
Touch Screen Controller
1243
Display Controller
Sensor Controller
1244
Other Input/Output Controller(s)
1245
1240
1246
Touch Sensitive Display System
Other Input/Output Devices
1248

FIG. 12

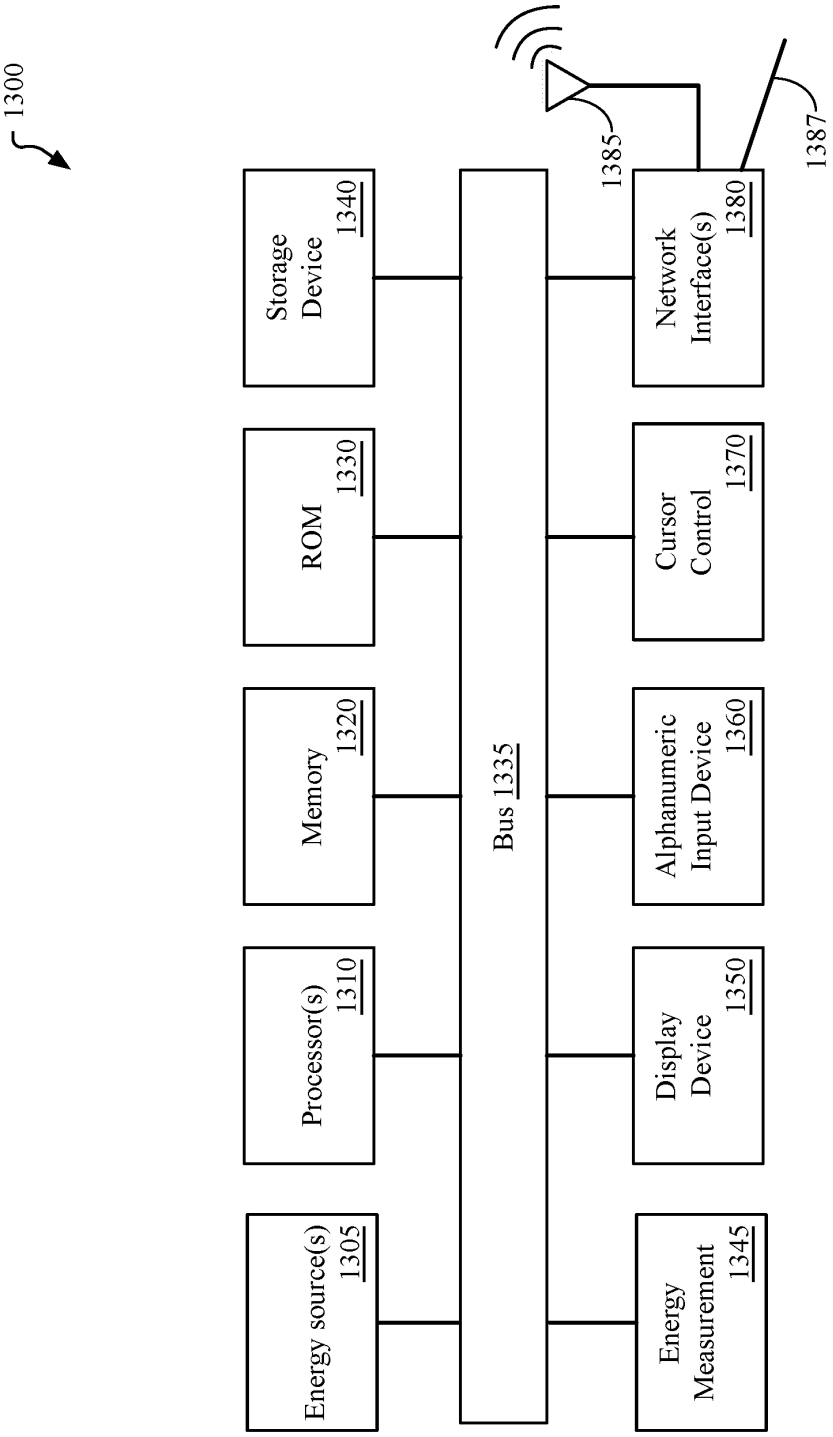

1300
Energy source(s) 1305
Processor(s) 1310
Memory 1320
ROM 1330
Storage Device 1340
Bus 1335
Energy Measurement 1345
Display Device 1350
Alphanumeric Input Device 1360
Cursor Control 1370
Network Interface(s) 1380
1385
1387

FIG. 13

SYSTEM AND METHOD FOR LOCATING WIRELESS ACCESSORIES

CROSS-REFERENCE

[0001] This application claims benefit priority to U.S. Provisional Application No. 62/738,983, filed on Sep. 28, 2018, which is hereby incorporated herein by reference in its entirety.

FIELD

[0002] Embodiments described herein relate generally to a system and method of locating wireless devices and accessories. More specifically, embodiments relate to crowdsourcing the location of wireless devices and accessories that are not connected to a wide area network.

BACKGROUND OF THE DESCRIPTION

[0003] Current security features in handheld and portable products allow the location of the product to be identified when requested by the user, such as in instances where the product is lost or stolen. If the wireless device includes positioning technology, the device can be configured to report its last location to the server computer, which is displayed by the service on a map presented to the user. Often wireless devices are used with wireless accessory devices that cannot determine their location and cannot communicate with a remote tracking services over a wide area network. These accessory devices can include, for example, wireless earbuds, headphones, headsets and other wearable devices (e.g., smartwatches, fitness bands, optical head-mounted displays) that communicate directly with the wireless device using peer-to-peer communications. For wireless accessory devices that cannot determine their location and cannot communicate with the remote tracking service, those devices cannot be tracked by the service when lost or stolen.

SUMMARY OF THE DESCRIPTION

[0004] Embodiments described herein provide for system and methods to crowdsource the location of wireless devices and accessories that lack a connection to a wide area network. One embodiment provides for a data processing system configured to perform operations comprising loading a user interface on an electronic device, the user interface to enable the determination of a location of a wireless accessory that is associated with the electronic device, generating a set of public keys included within a signal broadcast by the wireless accessory, the signal broadcast during a first period, sending the set of public keys to a server with a request to return data that corresponds with a public key in the set of public keys, decrypting the location data using a private key associated with the public key, and processing the location data to determine a probable location for the wireless accessory.

[0005] One embodiment provides for a data processing system comprising a memory to store instructions for execution and one or more processors to execute the instructions stored in memory. The instructions, when executed, cause the one or more processors to load a user interface on an electronic device that includes the data processing system, the user interface to enable the determination of a location of a wireless accessory that is associated with the electronic device, generate a set of public keys included within a signal broadcast by the wireless accessory, the signal broadcast during a first period, send the set of public keys to a server with a request to return data that corresponds with a public key in the set of public keys, decrypt the location data using a private key associated with the public key in response to receipt of location data from the server, and process the location data to determine a probable location for the wireless accessory.

[0006] One embodiment provides for a method implemented on an electronic device, the method comprising performing a beacon scan using a baseband processor of the electronic device. The beacon scan can be performed while an application processor of the electronic device is in a low power state. The method additionally comprises receiving beacon data via a wireless radio coupled to the baseband processor, the beacon data to identify a wireless accessory. The beacon identifier can be a public key generated by the wireless accessory. The method additionally comprises storing a timestamp and beacon identifier to a beacon scan buffer, determining a location for the electronic device that corresponds with the timestamp when an application processor of the electronic device is in an active state, encrypting the location for the electronic device using the beacon identifier for the wireless accessory, and transmitting encrypted location data, the beacon identifier, and the timestamp to a server device, the server to store the encrypted location data for retrieval by an electronic device associated with the wireless accessory.

[0007] The above summary does not include an exhaustive list of all embodiments in this disclosure. All systems and methods can be practiced from all suitable combinations of the various aspects and embodiments summarized above, and also those disclosed in the Detailed Description below.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] The present invention is illustrated by way of example and not limitation in the figures of the accompanying drawings in which like references indicate similar elements, and in which:

[0009] FIG. **1** is a block diagram of a network operating environment for mobile devices, according to an embodiment;

[0010] FIG. **2** illustrates a system to locate a wireless accessory that lacks access to a wide area network, according to an embodiment;

[0011] FIG. **3** illustrates a system for pairing and locating a wireless accessory, according to embodiments described herein;

[0012] FIG. **4**A-**4**C are flow diagrams illustrating methods for use with the device locator systems described herein;

[0013] FIG. **5** is a flow diagram illustrating a method of broadcasting a signal beacon at a wireless accessory, according to an embodiment;

[0014] FIG. **6**A-**6**B illustrate operations of a method that can be performed by a finder device, according to embodiments described herein;

[0015] FIG. **7** illustrates the gathering of signal and ranging data by a finder device, according to an embodiment;

[0016] FIG. **8** illustrates a networked system for locating devices and wireless accessories, according to an embodiment;

[0017] FIG. **9**A-**9**C illustrate a device locator user interface, according to an embodiment;

[0018] FIG. **10** illustrates an accessory pairing user interface that is displayed when attempting to pair with a lost wireless accessory, according to an embodiment;

[0019] FIG. **11** is a block diagram illustrating an exemplary API architecture, which may be used in some embodiments of the invention;

[0020] FIG. **12** is a block diagram of a device architecture for a mobile or embedded device, according to an embodiment; and

[0021] FIG. **13** is a block diagram of a computing system, according to an embodiment.

DETAILED DESCRIPTION

[0022] Embodiments described herein provide techniques to enable secure crowdsourced locator services for lost or misplaced devices that cannot communicate with a wide area network. Various embodiments and aspects will be described with reference to details discussed below, and the accompanying drawings will illustrate the various embodiments. The following description and drawings are illustrative and are not to be construed as limiting. Numerous specific details are described to provide a thorough understanding of various embodiments. However, in certain instances, well-known or conventional details are not described in order to provide a concise discussion of embodiments.

[0023] The terminology used in this description is for the purpose of describing particular embodiments only and is not intended to be limiting of the invention. As used in the description of the invention and the appended claims, the singular forms "a", "an" and "the" are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will also be understood that the term "and/or" as used herein refers to and encompasses any and all possible combinations of one or more of the associated listed items. It will be further understood that the terms "comprises" and/or "comprising," when used in this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof.

[0024] A portion of this disclosure contains material that is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure, as it appears in the Patent and Trademark Office patent files or records, but otherwise reserves all copyright rights whatsoever. Copyright 2018 Apple Inc.

[0025] In the discussion that follows, a computing device that includes a touch-sensitive display is described. It should be understood, however, that the computing device may include one or more other physical user-interface devices. The various applications that may be executed on the device may use at least one common physical user-interface device, such as the touch-sensitive surface. One or more functions of the touch-sensitive surface as well as corresponding information displayed on the device may be adjusted and/or varied from one application to the next and/or within a respective application. In this way, a common physical architecture (such as the touch-sensitive surface) of the device may support the variety of applications with user interfaces that are intuitive and transparent.

[0026] Some processes are described below in terms of some sequential operations. However, it should be appreciated that some of the operations described may be performed in a different order. Moreover, some operations may be performed in parallel rather than sequentially.

[0027] FIG. **1** is a block diagram of a network operating environment **100** for mobile devices, according to an embodiment. The network operating environment **100** includes multiple mobile devices, such as mobile device **102**A and mobile device **102**B. The mobile devices **102**A-**102**B can each be any electronic device capable of communicating with a wireless network and a wireless accessory device. Some example mobile devices include but are not limited to a smartphone, a tablet computer, a notebook computer, a wearable computer (e.g., smartwatch or other wearable computing accessory), a mobile media player, a personal digital assistant, and other similar devices. Each of mobile device **102**A and mobile device **102**B include a user interface, such as user interface **104** of mobile device **102**B. Mobile device **102**A and mobile device **102**B can communicate over one or more wired and/or wireless networks **110** to perform data communication. For example, a wireless network **112** (e.g., cellular network, Wi-Fi network) can communicate with a wide area network **114**, such as the Internet, by use of a gateway **116**. Likewise, an access device **118**, such as a mobile hotspot wireless access device, can provide communication access to the wide area network **114**. The gateway **116** and access device **118** can then communicate with the wide area network **114** over a combination of wired and/or wireless networks.

[0028] In some implementations, both voice and data communications can be established over the wireless network **112** and/or the access device **118**. For example, mobile device **102**A can place and receive phone calls (e.g., using VoIP protocols), send and receive e-mail messages (e.g., using POP**3** protocol), and retrieve electronic documents and/or streams, such as web pages, photographs, and videos, over the wireless network **112**, gateway **116**, and wide area network **114** (e.g., using TCP/IP or UDP protocols). In some implementations, mobile device **102**A can place and receive phone calls, send and receive e-mail messages, and retrieve electronic documents over the access device **118** and the wide area network **114**. In some implementations, mobile device **102**A or mobile device **102**B can be physically connected to the access device **118** using one or more cables, for example, where the access device **118** is a personal computer. In this configuration, mobile device **102**A or mobile device **102**B can be referred to as a "tethered" device. In one embodiment, mobile device **102**A can communicate with mobile device **102**B via a wireless peer-to-peer connection **120**. The wireless peer-to-peer connection **120** can be used to synchronize data between the devices.

[0029] Mobile device **102**A or mobile device **102**B can communicate with one or more services, such as a telephony service **130**, a messaging service **140**, a media service **150**, a storage service **160**, and a device locator service **170** over the one or more wired and/or wireless networks **110**. For example, the telephony service **130** can enable telephonic communication between mobile device **102**A and mobile device **102**B, or between a mobile device and a wired telephonic device. The telephony service **130** can route voice over IP (VoIP) calls over the wide area network **114** or can access a cellular voice network (e.g., wireless network **112**). The messaging service **140** can, for example, provide

e-mail and/or other messaging services. The media service **150** can, for example, provide access to media files, such as song files, audio books, movie files, video clips, and other media data. The storage service **160** can provide network storage capabilities to mobile device **102**A and mobile device **102**B to store documents and media files. The device locator service **170** can enable a user to locate a lost or misplaced device that was, at least at some point, connected to the one or more wired and/or wireless networks **110**. Other services can also be provided, including a software update service to update operating system software or client software on the mobile devices. In one embodiment, the messaging service **140**, media service **150**, storage service **160**, and device locator service **170** can each be associated with a cloud service provider, where the various services are facilitated via a cloud services account associated with the mobile devices **102**A-**102**B.

[0030] FIG. **2** illustrates a system **200** to locate a wireless accessory **201** that lacks access to a wide area network, according to an embodiment. In one embodiment, the wireless accessory **201** includes one or more wireless transceivers and can communicate, either directly or indirectly (e.g., through another device or computer) with a companion device (e.g., mobile device **102**) over a wireless network or peer-to-peer communication link. Some examples of wireless accessory devices include but are not limited to wireless earbuds, headphones, headsets and other wearable devices (e.g., smartwatches, fitness bands, optical head-mounted displays). The wireless accessory **201** can also include other wireless devices such as game controllers or remote controls. The wireless accessory **201**, in one embodiment, also includes smartphones, tablet computers, laptop computers, smart speaker devices, televisions, or television set top boxes that at least temporarily are unable to access a wide area network, such as the Internet (e.g., wide area network **114** as in FIG. **1**). The wireless accessory can also be any other wireless device, including beacons or locator tags that can be attached to other devices to enable the tracking or locating of those devices. In one embodiment, the wireless accessory **201** can be paired with the mobile device **102** using a wireless technology standard, such as but not limited to Bluetooth. The wireless accessory **201** can also communicate with the mobile device **102** over wireless technologies such as Wi-Fi direct, Zigbee, or AirPlay. While the companion device to which the wireless accessory **201** is paired is generally referred to as a mobile device **102**, companion devices are not limited to mobile devices. Companion devices, in some embodiments, can also include laptop or desktop devices and can additionally include some wearable accessories, such as but not limited to a smart watch device or a wearable display.

[0031] In one embodiment, the wireless accessory **201** can periodically transmit a wireless beacon signal. The wireless accessory **201** can transmit the beacon signal using one of a variety of wireless technologies described herein (e.g., Bluetooth, Wi-Fi, etc.) and in one embodiment can also beacon using an ultra-wide band (UWB) radio technology. The beacon signal can be transmitted using a single wireless technology, one of multiple selectable wireless technologies, or multiple simultaneous wireless technologies. The beacon signal can transmit a beacon identifier that includes information to specifically identify the wireless accessory **201**. In one embodiment, the beacon identifier is a public encryption key associated with the device.

[0032] The beacon signal can also convey information about the wireless accessory **201**, such as a beacon type, device classification, battery level. In one embodiment the beacon signal can also convey device status, such as a lost status, alarm status, or a near-owner status. The beacon signal can also include information that specifies battery life, charging status, and/or other status information. The lost status can indicate that the wireless accessory **201** has determined itself to be lost or has been placed into a lost state by the owner of the device. The alarm status can indicate that the wireless accessory **201** was placed in a state that the device should trigger an alarm if moved from a current location. The near-owner status can indicate that the wireless accessory **201** has detected the nearby presence of the mobile device **102** associated with the owner of the accessory.

[0033] The beacon signal can be detected by a finder device **202**, which is locally proximate to the wireless accessory **201**. The finder device **202** can be a similar device as the mobile device **102** and can receive and transmitting data over a wide area network **114** and receiving and transmitting using similar wireless technologies as the wireless accessory **201** (e.g., Bluetooth, etc.). Particularly, the finder device **202** can receive data using the wireless protocol over which the beacon signal is transmitted. The finder device **202** can determine a location using one or more location and/or positioning services including, but not limited to a satellite positioning service **206** or a terrestrial positioning system using RF signals received from wireless base stations **205** such as Wi-Fi access points or cell tower transmitters of a cellular telephone network. In an embodiment, the finder device **202** periodically stores its location as determined based on the one or more location and/or positioning services. The stored location can be associated with a timestamp for which the location was determined. When the finder device **202** receives a beacon signal from the wireless accessory **201**, the finder device **202** can transmit a location for the finder device over the wide area network **114** to a device locator server **203**. The timestamp for a determined location for the finder device **202** can be correlated with a timestamp for which a beacon signal was received to associate a geographic location with a received beacon signal.

[0034] Where the wireless accessory **201** provides a public key within the beacon signal, the finder device **202** can encrypt the determined location data and transmit the encrypted location data to the device locator server **203** over the wide area network **114**. In one embodiment, additional data can either be encrypted and transmitted along with the location data or transmitted unencrypted to the device locator server **203**. For example, a received signal strength indicator (RSSI) for the beacon signal can be transmitted along with the location data. The RSSI data can then be used to determine the distance of the wireless accessory **201** from the finder device **202** and assist in triangulation on the owner device. Where the RSSI data is transmitted in an unencrypted state, in one embodiment the server can use RSSI information to reduce noise by discarding very weak signals if other, stronger signals are present. In one embodiment, UWB ranging data can also be provided, where such data is available.

[0035] In one embodiment, the finder device **202** can behave differently upon receiving a beacon signal from a wireless accessory **201** depending upon a device status

conveyed by the wireless accessory **201**. For standard beacon signals, the finder device **202** can place encrypted location data into a queue and transmit the location data to the device locator server **203** during a periodic transmission window. However, if the wireless accessory **201** is indicating an alarm state, the finder device **202** can transmit the location data to the device locator server **203** immediately. Additionally, the finder device **202** may not transmit the location data to the device locator server **203** if the beacon signal of the wireless accessory **201** indicates that the accessory is near the owner of the accessory. Alternatively, the finder device **202** may delay transmission of encrypted location data.

[0036] If the owner of the wireless accessory **201** wishes to locate the wireless accessory, the owner can access a device locator user interface **204** on the mobile device **102**. The device locator user interface **204** can be associated with a device locator application that is used to locate electronic devices and accessories that are registered with an online account of the user, such as a cloud services account or another type of online account. The device owner, using the device locator UI **204**, can query the device locator server **203** for location data that may have been transmitted to the device locator server by a finder device **202** of the wireless accessory **201**. In one embodiment, the mobile device **102** can transmit the public encryption key associated with the wireless accessory **201** to the device locator server **203**. The device locator server **203** can then return any stored location data that corresponds with the public encryption key. The location data returned to the mobile device **102** can be encrypted data that is encrypted by the finder device **202** using the public encryption key. The mobile device **102** can use an associated private key to decrypt the encrypted location data. The decrypted location data can then be processed by the mobile device **102** to determine a most probable location for the wireless accessory **201**. In various embodiments, the most probable location for the wireless accessory **201** can be determined by triangulation from multiple received locations and using other data, such as a beacon signal RSSI associated with each location and timestamp or UWB ranging data included within the location data.

[0037] FIG. **3** illustrates a system **300** for pairing and locating a wireless accessory, according to embodiments described herein. In one embodiment a mobile device **102** of a user of the wireless accessory **201** can present an accessory pairing UI **302** by which the user can pair the mobile device **102** with the wireless accessory **201**. During an initial pairing (**305**) between the mobile device **102** and the wireless accessory, a public key exchange (**310**) can be performed between the mobile device and the wireless accessory. In one embodiment, during the public key exchange (**310**) the mobile device **102** and the wireless accessory **201** exchange public keys of public key pairs generated by the device and the accessory. In one embodiment the public key exchange (**310**) is a one-way transfer, in which the mobile device **102** transmits a public key of a public/private key pair to the wireless accessory **201**. Alternatively, or additionally, the public key exchange (**310**) may be a Diffie-Hellman key exchange in which the device and the accessory establish a shared secret between two parties. In one embodiment, the public key exchange (**310**) additionally uses elliptic curve cryptography to establish the shared secret. For example, Elliptic-curve Diffie-Hellman (ECDH) can be used to enable the establishment of a public key pair and one or more shared secrets. In one embodiment, the one or more shared secrets include an anti-tracking secret, which can the wireless accessory **201** to periodically derive additional public keys.

[0038] After the wireless accessory **201** has been paired with the mobile device **102**, the wireless accessory **201** can periodically broadcast a beacon signal **301** that includes device status information and a beacon identifier. In one embodiment the beacon identifier is a public key derived from a shared secret that is established during the public key exchange (**310**). Additionally, the wireless accessory **201** can periodically perform a public key derivation (**315**) to generate a new public key and begin broadcasting the new public key as the beacon identifier. The public key is a K-byte key, with a new K-byte key generated every M minutes. The value K and M can vary between embodiments. In one embodiment, a K value of 28 bytes is used. In one embodiment, a K value of 27 bytes is used. The value K can be determined at least in part based on the beacon length associated with the wireless protocol used to transmit the beacon signal **301**. In one embodiment, the beacon signal can transmit a variant of beacon advertisement packet associated with a low-energy radio protocol, such as Bluetooth Low Energy.

[0039] The value M, in one embodiment, is 15 minutes, such that a new K-byte key is generated every 15 minutes. The public key can be derived deterministically based on a timestamp and an anti-tracking secret generated during the public key exchange **310**. The public key derivation (**315**) process enables the wireless accessory **201** to use different keys over time, preventing the long-term association with a specific key with a specific device. The key can be derived based on an anti-tracking secret known only to the mobile device **102** and the wireless accessory **201**, allowing the mobile device **102**, and only the mobile device, to determine which public key will be broadcast by the wireless accessory **201** at any given timestamp. The anti-tracking secret can be generated along with an ECDH public key and transferred to the wireless accessory **201**. The anti-tracking secret can then be used to enable the wireless accessory **201** to generate a sequence of public keys P_i. In one embodiment, the sequence of public keys $P_i=X_i \cdot P$, which defines a group operation between a scalar or exponent value λ_i and group elements, such as, for example, Elliptic Curve points P. The scalar or exponent value λ=KDF(AT, i), where KDF is a key derivation function, AT is the anti-tracking secret, and i is a counter or timestamp.

[0040] In one embodiment, backtracking resistance can be enabled to protect the anti-tracking secret in the event the wireless accessory **201** is compromised. When backtracking resistance is enabled, the anti-tracking secret is transferred to the wireless accessory **201** but is not retained by the wireless accessory. Instead, the accessory computes a value $\lambda_{i+1}=H(\lambda_i \parallel \text{time})$, with λ_0=AT and H being a cryptographic hash function. The wireless accessory **201** then stores λ_i for a given time period i. If the wireless accessory **201** is compromised, only λ_i for current and future values of i is exposed, without exposing the anti-tracking secret AT. In one embodiment, backtracking resistance is performed by periodically writing λ_i to non-volatile memory of the wireless accessory **201**.

[0041] In one embodiment the wireless accessory **201** can transmit the beacon signal **301** every two seconds, although

other beacon rates can be used, and the beacon rate can vary under certain circumstances. For example, the wireless accessory **201** can decrease a beacon rate when in a near-owner state. Beacon rate can also vary based on accelerometer triggered events. For example, the wireless accessory **201** can increase the beacon rate when in an alarm state, which can be triggered by the accelerometer on the wireless accessory **201**.

[0042] The wireless accessory **201** can enter the near-owner state if, after transmitting the beacon signal **301**, the wireless accessory **201** receives a reply from the mobile device **102** associated with the user of the accessory, which indicates that the mobile device **102** is within range of the wireless accessory. Additionally, while the wireless accessory is in the near-owner state, the amount of data transmitted by the beacon signal **301** may be reduced. In one embodiment, the rate at which new public keys are generated can also be reduced while the wireless accessory is in the near-owner state.

[0043] The wireless accessory **201** can enter an alarm state upon receiving a message from the mobile device **102** that indicates that the wireless accessory **201** should enter the alarm state. When in the alarm state, the wireless accessory can initially enter an armed state in which the wireless accessory **201** can reduce or cease the transmission of locator beacon signals, although other types of wireless signaling can persist. The wireless accessory **201** can remain in the armed state until the state is deactivated by the mobile device **102** or alarm is triggered. The alarm can be triggered, in one embodiment, upon detection of movement, for example, via an accelerometer within the wireless accessory **201**. The alarm can also be triggered, in one embodiment, upon detection that the wireless accessory has moved out of range of the mobile device and is no longer in the near-owner state. When the alarm is triggered, the rate at which the beacon signal **301** can be increased, to increase the speed by which the wireless accessory **201** can be located.

[0044] The beacon signal **301** transmitted by the wireless accessory **201** can be detected by a set of finder devices **303**, which are other electronic devices that can receive the beacon signal transmitted by the wireless accessory and are transmit location and other data associated with the beacon signal **301** to the device locator server **203** via the wide area network **114**. In one embodiment the set of finder devices **303** include variants of the mobile device **102** or can be other types of electronic devices. The set of finder devices **303** can include a variant of the finder device **202** of FIG. **2** and can determine similar location determination techniques. For example, the set of finder devices can perform operations (**320**) to correlate the beacon signal **301** received from the wireless accessory **201** with a device location associated with the finder device. As described with respect to FIG. **2**, the device location can be determined via a satellite positioning service or a terrestrial positioning system that uses RF signals received from wireless base stations (e.g., Wi-Fi access points or cell tower transmitters). In one embodiment the set of finder devices **303** can also include stationary devices such as smart speaker devices, televisions, or television set top boxes that can receive the beacon signal **301**.

[0045] The set of finder devices **303** can encrypt the location data with the beacon identifier (e.g., public key) received within the beacon signal **301** and send the location data (**325**) to the device locator server **203**. The data sent by the set of finder devices **303** is send anonymously and no identifying information for the finder devices is stored with the data sent by the finder devices.

[0046] The device locator server **203** can store encrypted location data in a data store **304**, which in one embodiment can be a distributed database having multiple nodes. Hashes of the beacon identifier/public key of an accessory can be sent along with encrypted location data. The encrypted location data can be stored to a database node based on a hash of the beacon identifier. The encrypted location data can be indexed by the device locator server **203** using the hash of the beacon identifier. Sending the hash of the beacon identifier instead of the full beacon identifier prevents the storage of the full beacon identifier to the server. Other information can also be sent and stored with the location data, either in an encrypted or unencrypted state. The other information can include timestamps for when the beacon signal **301** was received, RSSI information for the received beacon, and/or ranging information determined, for example, via UWB ranging.

[0047] When the user or owner of the wireless accessory **201** wishes to locate the accessory, the user or owner can access the device locator UI **204** on the mobile device **102**. The device locator UI **204** can be associated with a device locator application or feature of the mobile device **102**. The device locator UI **204** may also have a web-based interface that can be accessed from the mobile device **102** or another type of electronic device, such as a laptop or desktop device. The mobile device **102**, upon loading the device locator UI **204**, can send a request (**330**) for location data to the device locator server **203**. The request **330** can include a set of public keys or public key hashes, which can serve as beacon identifiers for the beacon data. The mobile device **102** can generate the set of public keys based on the secret information held by the mobile device **102** and the wireless accessory **201** and the timestamps over which the mobile device **102** wishes to receive location data. In one embodiment the set of public keys is the sequence of public keys P_i that are generated based on the anti-tracking secret. The sequence of public keys P_i corresponds to a matching sequence of private keys d_i. The mobile device **102** can generate the sequence of public keys, as well as the corresponding sequence of public keys d_i, where i is a counter or timestamp. In one embodiment, the mobile device **102** can generate and send the previous 24 hours of public keys (or hashes of the 24 hours of public keys) within the request **330**. If no data is found for 24 hours of public keys, the mobile device **102** can send generate keys for an earlier period, back to a pre-determined location data retention limit.

[0048] In one embodiment the encrypted location data is stored and indexed based on a hash of the public key instead of the public key to prevent the provider of the location service data from storing data that can be used to tie the encrypted location data to a specific device, and thus a specific user or user account. The finder device can send the hash of the public key that is broadcast within the beacon signal **301** associated with an observation location. The owner of the device can query the device locator server **203** using a hash of the public key that is determined for a query period.

[0049] In some embodiments, if a location query is to be performed via the web-based interface from an electronic device, such as a laptop or desktop device, keys to enable the decryption of the location data may be required to be sent to the electronic device. In one embodiment, decryption keys

for the location data may be sent to the server that provides the web-based interface to enable the server to decrypt location data, at least while the location data is being viewed through the web-based interface. Before location data is displayed via the web-based interface, a notice may be presented to inform the user that location decryption keys are being temporarily shared with the web-based interface server to enable location data to be decrypted and presented. In one embodiment, the sharing of the location decryption keys can be performed via an automatic and temporarily delegation of location query rights with a proxy account associated with the web-based interface.

[0050] In one embodiment, the wireless accessory **201** can be placed in a light lost mode. In the light lost mode, a set of future public keys can be generated for the wireless accessory and transmitted to the device locator server **203**. The device locator server **203** can then notify the mobile device **102** if any location data is received that correspond with a key in the set of future public keys. In one embodiment, a finder device that sends a location for a wireless accessory that is in the light lost mode can be directed by the device locator server **203** to relay a message to the wireless accessory **201** that notifies the wireless accessory that it is in the light lost mode. A similar mechanism can be used to relay a message to the wireless accessory **201** that places the accessory in an explicit lost mode. The explicit lost mode can be enabled by the user via the device locator UI **204**. In the explicit lost mode, the wireless accessory **201** cannot be paired with another device unless unlocked by the owner.

[0051] FIG. **4**A-**4**C are flow diagrams illustrating methods for use with the device locator systems described herein. FIG. **4**A illustrates a method **400** to pair a mobile device with a wireless accessory. FIG. **4**B illustrates a method **410** to determine a location for a wireless accessory via a device locator server. FIG. **4**C illustrates an additional method **420** to determine a location for a wireless accessory via a device locator server. Aspects of method **400**, **410**, and **420** are also illustrated in FIG. **2** and FIG. **3**, as described above. For example, the description of the operations below refers to the mobile device **102**, wireless accessory **201** and device locator server **203**.

[0052] As shown in FIG. **4**A, method **400** includes an operation (block **401**) that performs an initial pairing with a wireless accessory. The initial pairing can be a Bluetooth pairing or another type of pairing using other wireless radio technologies. During the initial pairing, the mobile device and the wireless accessory can exchange identifiers, pass-keys, or other credentials that enables a wireless data exchange to be performed between a mobile or another electronic device and the wireless accessory. On one embodiment the initial paring with the wireless accessory can include the exchange of credentials associated with the wireless protocol for which the pairing is performed, allowing all data exchanged wirelessly to have at least a first layer of encryption.

[0053] The mobile device can then generate a public/private key pair and one or more additional shared secrets (block **402**). The device can then send the public key and one or more additional shared secrets to the wireless accessory (block **403**). A variety of key generation techniques can be used. In one embodiment, a variant of ECDH is used to generate a public key pair for encryption. In one embodiment, the one or more additional shared secrets can include an anti-tracking secret that enables the wireless accessory to derive a new public key based on an existing public key.

[0054] After generating the public/private keypair and one or more additional shared secrets, the mobile device can store public/private key pair to keystore (block **404**). In one embodiment the keystore is a cloud-based keystore that can be synchronized with other devices associated with the same cloud services account, or family of cloud services accounts, to which the mobile device and wireless accessory are associated. The cloud-based keystore allows the wireless accessory to be located by other synchronized devices. The mobile device can then register the wireless accessory with a device management server (block **405**). Registering the wireless accessory with the device management server can form an association between the wireless accessory and the cloud services account to which the mobile device is associated. The device management server can be associated with other cloud-based servers that are used to facilitate cloud-based services accessible to the mobile device, such as the device locator server **203** of FIG. **2** and FIG. **3**.

[0055] As shown in FIG. **4**B, method **410** includes an operation in which an electronic device launches a device locator UI (block **411**). In response to launching the device locator UI, the electronic device, which can be a mobile device as described herein, or another electronic device associated with the same cloud services account as the mobile electronic device, can perform an operation to generate a set of public keys that were included within a beacon signal broadcast by a wireless accessory during a first period (block **412**). The first period can be, for example, a previous 24 hours. The electronic device is aware of the frequency in which the wireless accessory is to generate new public keys and, using a shared secret generated with the wireless accessory, can generate a set of public keys that correspond with the keys that were generated by the wireless accessory over the first period. The electronic device can then send the set of public keys within a request for the device locator server to send location data that corresponds with the set of public keys (block **413**). In one embodiment, location data sent by the server in response to the request will be encrypted using the public key transmitted as the beacon identifier of the wireless accessory. The electronic device can decrypt the encrypted location data received by the server using the private key generated during the initial pairing with the wireless accessory (block **414**). The electronic device can then process the location data to determine the highest probability location for the wireless accessory (block **415**).

[0056] Processing the location data can include a variety of different operations. In one embodiment the location data includes latitude and longitude information along with a timestamp for which the location was determined. The electronic device can triangulate based on the timestamps and remove noise or outlier locations. In one embodiment the location data specifies the location of the finder device that detected the beacon. The location data can additionally include UWB ranging information and/or RSSI information for the beacon detected by the finder device. The electronic device can analyze the UWB ranging information and/or RSSI information in context with the device locations to develop a more accurate location for the wireless accessory. Data that can be transmitted by a finder device and used for location processing is shown in FIG. **10** and described below.

[0057] As shown in FIG. **4**C, method **420** includes operations that can be performed if the device locator server does not have location data to provide to the electronic device in response to a request. The electronic device can generate a first set of public keys that were included within a beacon signal broadcast by wireless accessory during a first period (block **421**). The first period can be, for example, 24 hours, although other initial search periods can be used. The electronic device can perform a subsequent operation to request the device locator server to send location data that corresponds with first set of public keys (block **422**). If the data is returned by the server (block **423**, "yes"), the electronic device can decrypt the location data received from the server using the private key that corresponds with the set of public keys (block **429**).

[0058] If data is not returned by the server (block **423**, "no") the electronic device can generate a second set of public keys that were included within a beacon signal broadcast by the wireless accessory during a second period (block **424**). The second period can be the **24**, **48**, or another number of hours before the first period. The electronic device can then request for the device locator server to send data that corresponds with the second set of public keys (block **425**). If, in response to the request, data is returned by the server (block **426**, "yes"), method **420** can proceed to block **429**, in which the electronic device decrypts the received data. If data is not returned by the server (block **426**, "no"), or the server sends a reply that indicates data is not available, method **420** includes for the electronic device can widen the search time by requesting successively older time periods until the max period is reached (block **427**).

[0059] FIG. **5** is a flow diagram illustrating a method **500** of broadcasting a signal beacon at a wireless accessory, according to an embodiment. Aspects of method **500** are also illustrated in FIG. **2** and FIG. **3**. Method **500** includes for the wireless accessory to derive a public key (block **502**). The public key can be derived based on a shared secret and a timestamp determined based on a clock or time keeping device of the wireless accessory. The wireless accessory can then transmit a beacon signal at a first frequency, where the beacon signal includes the public key (block **503**). The first frequency can vary, and in one embodiment is one beacon every two seconds.

[0060] After transmitting a beacon signal, the wireless accessory can listen for a response from the owner device. If the wireless signal receives a response from the owner device (block **504**, "yes"), the wireless accessory can enter a near-owner state (block **505**) and begin to transmit the beacon signal at a second, lower frequency (block **507**). If the wireless accessory does not receive a response from the owner device (block **504**, "no"), the wireless accessory can continue beaconing at the first frequency (block **506**).

[0061] Method **500** additionally includes for the wireless device, while beaconing, to rotate the public key every M minutes, where the value of M can vary across embodiments and/or based on the device state. Based on a timer expiration, counter, or another mechanism, the wireless accessory can determine whether the accessory has entered a new key period (block **508**). While the wireless accessory has not entered a new key period (block **508**, "no"), the accessory can continue beaconing using the current public key (block **510**). When the wireless accessory detects that it has entered a new key period (block **508**, "yes") the accessory can derive a new public key using the current timestamp (block **509**). In one embodiment the new public key can be derived using an existing public key, a timestamp, and an anti-tracking secret.

[0062] FIG. **6**A-**6**B illustrate operations of a method **600** that can be performed by a finder device, according to embodiments described herein. Aspects of method **600** are also illustrated in FIG. **2** and FIG. **3**.

[0063] As shown in FIG. **6**A, method **600** includes for the finder device to perform a periodic beacon scan using a wireless baseband processor while an application processor of the finder device is in a low power mode (block **601**). While the beacon scan can also be performed when the application processor is active, beacon scans can be performed by the wireless processor and a wireless radio receiver as a low power operation while the finder device is idle, inactive, or otherwise in a low power state. The finder device can store a timestamp and a beacon identifier to a beacon scan buffer for any beacon data received by the finder device (block **602**). The beacon identifier, in one embodiment, is a public key that is generated by the wireless device based on a timestamp and a shared secret generated with the mobile device of the owner.

[0064] Method **600** additionally includes for the finder device to perform periodic Wi-Fi scans using the wireless processor while application processor is in a low power mode (block **603**). While the Wi-Fi scans can also be performed when the application processor is active, Wi-Fi scans can be performed by the wireless processor and a wireless radio receiver as a low power operation while the finder device is idle, inactive, or otherwise in a low power state. The finder device can then store Wi-Fi service set identifiers (SSIDs) and scan timestamps to a Wi-Fi scan buffer on the finder device (block **604**).

[0065] In one embodiment, the Wi-Fi scan buffer is a rolling buffer that stores the most recently detected SSIDs, while overwriting older detected SSIDs. In one embodiment the beacon scan buffer can be a fixed-size buffer having space for a pre-determined number of entries. The finder device can wake the application processor when the beacon scan buffer becomes full (block **605**) and correlate those beacon scan with the most recently detected SSIDs in the Wi-Fi scan buffer. That correlation can enable the finder device to determine a set of device locations that correspond with received beacons based on Wi-Fi scan buffer data (block **606**).

[0066] Method **600** continues in FIG. **6**B and includes for the finder device to correlate device locations from the Wi-Fi scan buffer data with other location data if other location data is available (block **607**), to generate refined device locations. If refined device locations are generated, the finder device can optionally combine the beacon data with refined device locations (block **608**). The finder device can also add signal strength (RSSI) or ranging data to the location data (block **609**). The signal strength and ranging data (e.g., UWB ranging data) can be gathered when the beacon signal is received by the finder device. The finder device can then encrypt the location data with one or more public keys received within the beacon data (block **610**). The signal and ranging data may be encrypted along with the location data or can be send unencrypted along with the encrypted location data. The finder device can enqueue encrypted location data for transmission to the device locator server (block **611**). The device locator server can be one of multiple cloud services servers to which communication

is generally performed in a batched and throttled manner. A batch of encrypted data can be gathered and placed in the transmission queue until a transmit interval arrives, during which the finder device can transmit data to the cloud services servers (block **612**).

[0067] FIG. **7** illustrates the gathering of signal and ranging data by a finder device, according to an embodiment. In one embodiment, the finder device **202** can gather signal strength information (e.g., RSSI **704**A-**704**N) for a beacon signal **301** received from the wireless accessory **201** across multiple locations **702**A-**702**N. The finder device **202** can also represent multiple finder devices, such as the set of finder devices **303** in FIG. **3**, where each finder device detects the beacon signal at a different location. Each finder device **202** can send different locations and signal strengths and the location and signal strength data received from the multiple finder devices will be aggregated by the device locator server. In one embodiment, where a finder device and the wireless device each include UWB radios, UWB ranging **706** can be performed if the finder device and the wireless device are within range of UWB transmissions. UWB ranging and signal strength data can be transmitted along with location data for the finder devices to the device locator server.

[0068] The owner device can retrieve the RSSI or UWB information from the device locator server along with location data, which in one embodiment is provided the form of latitude and longitude information, along with timestamps for which the locations were determined. The owner device can then use the location data, timestamps, and signal information to triangulate a most probable location for the wireless accessory **201**.

[0069] FIG. **8** illustrates a networked system **800** for locating devices and wireless accessories, according to an embodiment. The system **800** also illustrates an exemplary server architecture for the device locator server **203**, according to an embodiment. In one embodiment the device locator server **203** is a cluster of interconnected server devices, which may be physical or virtual servers within a single datacenter or distributed across multiple datacenters and/or geographic locations. As described above, the device locator server **203** can communicate with a mobile device **102** of an accessory owner or user and the set of finder devices **303** over a wide area network **114**. The mobile device **102** includes a UI provided by a local or web application that enables the location of a wireless accessory and the finder devices **303** receive beacon signals from wireless accessories and transmits location data associated with the received signals to the device locator server **203**.

[0070] In one embodiment the device locator server **203** includes a locator server front-end **803**, an account database **825**, a database cluster manager **813**, and a set of database cluster nodes **823**A-**823**C. The locator server front-end **803** is a front-end interface to which the mobile device **102** and the set of finder devices **303** can communicate. The account database **825** stores account profile data for accounts of a cloud service provider to which the mobile device **102** and the finder devices **303** are associated. The database cluster manager **813** can configure the database cluster nodes **823**A-**823**C as a distributed location database that can store location, signal, and ranging data in association with beacon identifiers for signal beacons received by the set of finder devices **303**.

[0071] In one embodiment, the account database **825** can contain a list of devices that are associated with each cloud services account. In response to a request to locate a given device, including a wireless accessory as described herein, the account database **825** can verify that the request is coming from a device that is authorized to request the location of the given device. In one embodiment, when a user launches a device locator UI and communicates with the locator service front-end **803**, the locator service front-end can communicate with the account database **825** and provide a current or last known location for each device that is associated with a requesting user, including devices and/or wireless accessories associated with other users that are in a family of accounts associated with the requesting user.

[0072] In one embodiment, the database cluster manager **813** can select a database cluster node **823**A-**823**C to which beacon data is to be stored by hashing the beacon ID associated with a set of location data. Each database cluster node **823**A-**823**C can be associated with a range of hash values. The database cluster manager can then store location data to the cluster node that corresponds with the range of hash values associated with the hash of a given beacon ID. Although three database cluster nodes are illustrated, embodiments are not limited to any specific number of nodes and greater or fewer nodes may be used.

[0073] FIG. **9**A-**9**C illustrate a device locator UI **204**, according to an embodiment. FIG. **9**A shows a first graphical user interface of the device locator UI **204**, according to an embodiment, which shows a location for various electronic devices and wireless accessories of a user. FIG. **9**B shows a second graphical user interface of the device locator UI **204**, according to an embodiment, which enables a wireless accessory to be set to an alarm mode. FIG. **9**C shows a third graphical user interface of the device locator UI **204**, according to an embodiment, which enables a wireless accessory to be set to a lost mode.

[0074] As shown in FIG. **9**A, the device locator UI **204** can be displayed on an electronic device **900**, which can be a mobile device, or any other type of electronic device described herein. The device locator UI **204** can present a unified graphical interface through which multiple different types of devices and accessories can be located, including wireless devices with network or cellular access and wireless accessories without native network access. The device locator UI **204** can include a map **901** with a marker **902** that shows the current or last known location of a wireless device or accessory. The marker **902** can be an icon, image, graphic or any other user interface element that identifies the accessory and conveys a location for the accessory. A selectable element **903** in the device locator UI can present a description or name of the wireless device or accessory and can show an estimated distance between the wireless device or accessory and the current location of the electronic device **900**.

[0075] As shown in FIG. **9**B, the device locator UI **204** can present a second user interface that enables a wireless accessory to be set to an alarm mode. The second user interface can be displayed, in one embodiment, in response to the selection of the selectable element **903** shown in FIG. **9**A. The second user interface can present a user interface element **904** that represents and/or describes the wireless accessory in question, as well as the map **901** and marker **902** that show the current or last known location of the wireless accessory. In one embodiment, the device locator

UI **204** can present a selectable element **905**, such as a button or another user interface element, that allows a user of the device locator UI **204** to place a selected wireless accessory into an alarm mode. While in the alarm mode, the wireless accessory can be configured to trigger a notification to the user if the wireless accessory is moved from its current location.

[0076] In one embodiment the wireless accessory can detect movement via an accelerometer or another type of motion sensor within the wireless accessory. The notification can be initiated by the wireless accessory by setting a flag in the data packet transmitted by the beacon signal of the wireless accessory that indicates the wireless accessory alarm has been triggered. In various embodiments, other trigger or notification modes can be used. In one embodiment, the alarm can optionally be triggered by the mobile device upon detection that the wireless accessory has moved out of range of the mobile device and is no longer in the near-owner state. In one embodiment, the alarm can optionally be triggered when the wireless accessory is out of range of, or otherwise cannot be located by, any of the devices associated with the account or family of user accounts to which the wireless accessory is associated.

[0077] As shown in FIG. **9**C, the device locator UI **204** can present a third graphical user interface that enables a wireless accessory to be set to a lost mode. In one embodiment, when a wireless accessory cannot be located via the device locator UI **204**, the map **901** will not display a marker that indicates a location for the accessory. The device locator UI **204** can present the user interface element **904** that represents and/or describes the wireless accessory in question and a set of selectable user interface elements. One selectable user interface element **906** can present the option to notify the user when the accessory is found. When notify when found is enabled, in one embodiment the wireless accessory can be placed into a light lost mode. The electronic device associated with the device locator UI **204** can generate a set of public keys that the wireless accessory will broadcast with the beacon signal during a future time period (e.g., next **24** hours, next **48** hours, etc.). If a signal is detected by a finder device using one of the future keys, the device locator server can notify one or more electronic devices associated with the user.

[0078] Another selectable user interface element **907** can place the wireless accessory into an explicit lost mode. When explicitly placed into lost mode, the wireless accessory will be unable to be paired with other devices until the accessory is unlocked by the user or owner that places the device into lost mode. When sending a request to place a wireless accessory into lost mode, the requesting user can be required to enter authenticating information to ensure that the requesting user is authorized to request that lost mode be initiated on the lost accessory. The authenticating information can include a username or password associated with an account of a user, such as a cloud services account to which the user, electronic device, and wireless accessory are associated. The authenticating information can also include biometric information, such as a fingerprint or facial recognition data.

[0079] In one embodiment, a message and contact information provided by the requesting user can be displayed on the user device to alert a person who finds the lost wireless accessory on how to contact the requesting user. In one embodiment, the message and contact information can be displayed when another user attempts to pair another electronic device with the lost accessory.

[0080] FIG. **10** illustrates an accessory pairing UI **302** that is displayed when attempting to pair with a lost wireless accessory, according to an embodiment. In one embodiment, when an electronic device **1000** that is different from the electronic device **9000** of FIG. **9** and is not associated with the registered user or owner of a wireless accessory attempts to pair with a lost wireless accessory, the accessory pairing UI of the electronic device can be displayed as shown in FIG. **10**. In one embodiment, the accessory pairing UI **302** can display a name or description **1001** associated with the wireless accessory, as well as a message **1002** entered by the user of the accessory upon placing the accessory into lost mode. Contact information **1004** can also be displayed, along with a user interface element **1006**, such as a button, that enables the electronic device **1000** to contact the requesting user by using the provided contact information **1004**.

[0081] Embodiments described herein include one or more application programming interfaces (APIs) in an environment in which calling program code interacts with other program code that is called through one or more programming interfaces. Various function calls, messages, or other types of invocations, which further may include various kinds of parameters, can be transferred via the APIs between the calling program and the code being called. In addition, an API may provide the calling program code the ability to use data types or classes defined in the API and implemented in the called program code.

[0082] An API allows a developer of an API-calling component (which may be a third-party developer) to leverage specified features provided by an API-implementing component. There may be one API-calling component or there may be more than one such component. An API can be a source code interface that a computer system or program library provides to support requests for services from an application. An operating system (OS) can have multiple APIs to allow applications running on the OS to call one or more of those APIs, and a service (such as a program library) can have multiple APIs to allow an application that uses the service to call one or more of those APIs. An API can be specified in terms of a programming language that can be interpreted or compiled when an application is built.

[0083] In some embodiments, the API-implementing component may provide more than one API, each providing a different view of or with different aspects that access different aspects of the functionality implemented by the API-implementing component. For example, one API of an API-implementing component can provide a first set of functions and can be exposed to third party developers, and another API of the API-implementing component can be hidden (not exposed) and provide a subset of the first set of functions and also provide another set of functions, such as testing or debugging functions which are not in the first set of functions. In other embodiments, the API-implementing component may itself call one or more other components via an underlying API and thus be both an API-calling component and an API-implementing component.

[0084] An API defines the language and parameters that API-calling components use when accessing and using specified features of the API-implementing component. For example, an API-calling component accesses the specified features of the API-implementing component through one or

more API calls or invocations (embodied for example by function or method calls) exposed by the API and passes data and control information using parameters via the API calls or invocations. The API-implementing component may return a value through the API in response to an API call from an API-calling component. While the API defines the syntax and result of an API call (e.g., how to invoke the API call and what the API call does), the API may not reveal how the API call accomplishes the function specified by the API call. Various API calls are transferred via the one or more application programming interfaces between the calling (API-calling component) and an API-implementing component. Transferring the API calls may include issuing, initiating, invoking, calling, receiving, returning, or responding to the function calls or messages; in other words, transferring can describe actions by either of the API-calling component or the API-implementing component. The function calls or other invocations of the API may send or receive one or more parameters through a parameter list or other structure. A parameter can be a constant, key, data structure, object, object class, variable, data type, pointer, array, list or a pointer to a function or method or another way to reference a data or other item to be passed via the API.

[0085] Furthermore, data types or classes may be provided by the API and implemented by the API-implementing component. Thus, the API-calling component may declare variables, use pointers to, use or instantiate constant values of such types or classes by using definitions provided in the API.

[0086] Generally, an API can be used to access a service or data provided by the API-implementing component or to initiate performance of an operation or computation provided by the API-implementing component. By way of example, the API-implementing component and the API-calling component may each be any one of an operating system, a library, a device driver, an API, an application program, or other module (it should be understood that the API-implementing component and the API-calling component may be the same or different type of module from each other). API-implementing components may in some cases be embodied at least in part in firmware, microcode, or other hardware logic. In some embodiments, an API may allow a client program to use the services provided by a Software Development Kit (SDK) library. In other embodiments, an application or other client program may use an API provided by an Application Framework. In these embodiments, the application or client program may incorporate calls to functions or methods provided by the SDK and provided by the API or use data types or objects defined in the SDK and provided by the API. An Application Framework may in these embodiments provide a main event loop for a program that responds to various events defined by the Framework. The API allows the application to specify the events and the responses to the events using the Application Framework. In some implementations, an API call can report to an application the capabilities or state of a hardware device, including those related to aspects such as input capabilities and state, output capabilities and state, processing capability, power state, storage capacity and state, communications capability, etc., and the API may be implemented in part by firmware, microcode, or other low-level logic that executes in part on the hardware component.

[0087] The API-calling component may be a local component (i.e., on the same data processing system as the API-implementing component) or a remote component (i.e., on a different data processing system from the API-implementing component) that communicates with the API-implementing component through the API over a network. It should be understood that an API-implementing component may also act as an API-calling component (i.e., it may make API calls to an API exposed by a different API-implementing component) and an API-calling component may also act as an API-implementing component by implementing an API that is exposed to a different API-calling component.

[0088] The API may allow multiple API-calling components written in different programming languages to communicate with the API-implementing component (thus the API may include features for translating calls and returns between the API-implementing component and the API-calling component); however, the API may be implemented in terms of a specific programming language. An API-calling component can, in one embedment, call APIs from different providers such as a set of APIs from an OS provider and another set of APIs from a plug-in provider and another set of APIs from another provider (e.g., the provider of a software library) or creator of another set of APIs.

[0089] FIG. **11** is a block diagram illustrating an exemplary API architecture, which may be used in some embodiments of the invention. As shown in FIG. **11**, the API architecture **1100** includes the API-implementing component **1110** (e.g., an operating system, a library, a device driver, an API, an application program, software or other module) that implements the API **1120**. The API **1120** specifies one or more functions, methods, classes, objects, protocols, data structures, formats and/or other features of the API-implementing component that may be used by the API-calling component **1130**. The API **1120** can specify at least one calling convention that specifies how a function in the API-implementing component receives parameters from the API-calling component and how the function returns a result to the API-calling component. The API-calling component **1130** (e.g., an operating system, a library, a device driver, an API, an application program, software or other module), makes API calls through the API **1120** to access and use the features of the API-implementing component **1110** that are specified by the API **1120**. The API-implementing component **1110** may return a value through the API **1120** to the API-calling component **1130** in response to an API call.

[0090] It will be appreciated that the API-implementing component **1110** may include additional functions, methods, classes, data structures, and/or other features that are not specified through the API **1120** and are not available to the API-calling component **1130**. It should be understood that the API-calling component **1130** may be on the same system as the API-implementing component **1110** or may be located remotely and accesses the API-implementing component **1110** using the API **1120** over a network. While FIG. **11** illustrates a single API-calling component **1130** interacting with the API **1120**, it should be understood that other API-calling components, which may be written in different languages (or the same language) than the API-calling component **1130**, may use the API **1120**.

[0091] The API-implementing component **1110**, the API **1120**, and the API-calling component **1130** may be stored in a machine-readable medium, which includes any mechanism for storing information in a form readable by a machine (e.g., a computer or other data processing system). For

example, a machine-readable medium includes magnetic disks, optical disks, random-access memory; read only memory, flash memory devices, etc.

[0092] FIG. **12** is a block diagram of a device architecture **1200** for a mobile or embedded device, according to an embodiment. The device architecture **1200** includes a memory interface **1202**, a processing system **1204** including one or more data processors, image processors and/or graphics processing units, and a peripherals interface **1206**. The various components can be coupled by one or more communication buses or signal lines. The various components can be separate logical components or devices or can be integrated in one or more integrated circuits, such as in a system on a chip integrated circuit.

[0093] The memory interface **1202** can be coupled to memory **1250**, which can include high-speed random-access memory such as static random-access memory (SRAM) or dynamic random-access memory (DRAM) and/or non-volatile memory, such as but not limited to flash memory (e.g., NAND flash, NOR flash, etc.).

[0094] Sensors, devices, and subsystems can be coupled to the peripherals interface **1206** to facilitate multiple functionalities. For example, a motion sensor **1210**, a light sensor **1212**, and a proximity sensor **1214** can be coupled to the peripherals interface **1206** to facilitate the mobile device functionality. One or more biometric sensor(s) **1215** may also be present, such as a fingerprint scanner for fingerprint recognition or an image sensor for facial recognition. Other sensors **1216** can also be connected to the peripherals interface **1206**, such as a positioning system (e.g., GPS receiver), a temperature sensor, or other sensing device, to facilitate related functionalities. A camera subsystem **1220** and an optical sensor **1222**, e.g., a charged coupled device (CCD) or a complementary metal-oxide semiconductor (CMOS) optical sensor, can be utilized to facilitate camera functions, such as recording photographs and video clips.

[0095] Communication functions can be facilitated through one or more wireless communication subsystems **1224**, which can include radio frequency receivers and transmitters and/or optical (e.g., infrared) receivers and transmitters. The specific design and implementation of the wireless communication subsystems **1224** can depend on the communication network(s) over which a mobile device is intended to operate. For example, a mobile device including the illustrated device architecture **1200** can include wireless communication subsystems **1224** designed to operate over a GSM network, a CDMA network, an LTE network, a Wi-Fi network, a Bluetooth network, or any other wireless network. In particular, the wireless communication subsystems **1224** can provide a communications mechanism over which a media playback application can retrieve resources from a remote media server or scheduled events from a remote calendar or event server.

[0096] An audio subsystem **1226** can be coupled to a speaker **1228** and a microphone **1230** to facilitate voice-enabled functions, such as voice recognition, voice replication, digital recording, and telephony functions. In smart media devices described herein, the audio subsystem **1226** can be a high-quality audio system including support for virtual surround sound.

[0097] The I/O subsystem **1240** can include a touch screen controller **1242** and/or other input controller(s) **1245**. For computing devices including a display device, the touch screen controller **1242** can be coupled to a touch sensitive display system **1246** (e.g., touch-screen). The touch sensitive display system **1246** and touch screen controller **1242** can, for example, detect contact and movement and/or pressure using any of a plurality of touch and pressure sensing technologies, including but not limited to capacitive, resistive, infrared, and surface acoustic wave technologies, as well as other proximity sensor arrays or other elements for determining one or more points of contact with a touch sensitive display system **1246**. Display output for the touch sensitive display system **1246** can be generated by a display controller **1243**. In one embodiment, the display controller **1243** can provide frame data to the touch sensitive display system **1246** at a variable frame rate.

[0098] In one embodiment, a sensor controller **1244** is included to monitor, control, and/or processes data received from one or more of the motion sensor **1210**, light sensor **1212**, proximity sensor **1214**, or other sensors **1216**. The sensor controller **1244** can include logic to interpret sensor data to determine the occurrence of one of more motion events or activities by analysis of the sensor data from the sensors.

[0099] In one embodiment, the I/O subsystem **1240** includes other input controller(s) **1245** that can be coupled to other input/control devices **1248**, such as one or more buttons, rocker switches, thumb-wheel, infrared port, USB port, and/or a pointer device such as a stylus, or control devices such as an up/down button for volume control of the speaker **1228** and/or the microphone **1230**.

[0100] In one embodiment, the memory **1250** coupled to the memory interface **1202** can store instructions for an operating system **1252**, including portable operating system interface (POSIX) compliant and non-compliant operating system or an embedded operating system. The operating system **1252** may include instructions for handling basic system services and for performing hardware dependent tasks. In some implementations, the operating system **1252** can be a kernel.

[0101] The memory **1250** can also store communication instructions **1254** to facilitate communicating with one or more additional devices, one or more computers and/or one or more servers, for example, to retrieve web resources from remote web servers. The memory **1250** can also include user interface instructions **1256**, including graphical user interface instructions to facilitate graphic user interface processing.

[0102] Additionally, the memory **1250** can store sensor processing instructions **1258** to facilitate sensor-related processing and functions; telephony instructions **1260** to facilitate telephone-related processes and functions; messaging instructions **1262** to facilitate electronic-messaging related processes and functions; web browser instructions **1264** to facilitate web browsing-related processes and functions; media processing instructions **1266** to facilitate media processing-related processes and functions; location services instructions including GPS and/or navigation instructions **1268** and Wi-Fi based location instructions to facilitate location based functionality; camera instructions **1270** to facilitate camera-related processes and functions; and/or other software instructions **1272** to facilitate other processes and functions, e.g., security processes and functions, and processes and functions related to the systems. The memory **1250** may also store other software instructions such as web video instructions to facilitate web video-related processes and functions; and/or web shopping instructions to facilitate

web shopping-related processes and functions. In some implementations, the media processing instructions **1266** are divided into audio processing instructions and video processing instructions to facilitate audio processing-related processes and functions and video processing-related processes and functions, respectively. A mobile equipment identifier, such as an International Mobile Equipment Identity (IMEI) **1274** or a similar hardware identifier can also be stored in memory **1250**.

[0103] Each of the above identified instructions and applications can correspond to a set of instructions for performing one or more functions described above. These instructions need not be implemented as separate software programs, procedures, or modules. The memory **1250** can include additional instructions or fewer instructions. Furthermore, various functions may be implemented in hardware and/or in software, including in one or more signal processing and/or application specific integrated circuits.

[0104] FIG. **13** is a block diagram of a computing system **1300**, according to an embodiment. The illustrated computing system **1300** is intended to represent a range of computing systems (either wired or wireless) including, for example, desktop computer systems, laptop computer systems, tablet computer systems, cellular telephones, personal digital assistants (PDAs) including cellular-enabled PDAs, set top boxes, entertainment systems or other consumer electronic devices, smart appliance devices, or one or more implementations of a smart media playback device. Alternative computing systems may include more, fewer and/or different components. The computing system **1300** can be used to provide the computing device and/or a server device to which the computing device may connect.

[0105] The computing system **1300** includes bus **1335** or other communication device to communicate information, and processor(s) **1310** coupled to bus **1335** that may process information. While the computing system **1300** is illustrated with a single processor, the computing system **1300** may include multiple processors and/or co-processors. The computing system **1300** further may include memory **1320** in the form of random access memory (RAM) or other dynamic storage device coupled to the bus **1335**. The memory **1320** may store information and instructions that may be executed by processor(s) **1310**. The memory **1320** may also be main memory that is used to store temporary variables or other intermediate information during execution of instructions by the processor(s) **1310**.

[0106] The computing system **1300** may also include read only memory (ROM) **1330** and/or another data storage device **1340** coupled to the bus **1335** that may store information and instructions for the processor(s) **1310**. The data storage device **1340** can be or include a variety of storage devices, such as a flash memory device, a magnetic disk, or an optical disc and may be coupled to computing system **1300** via the bus **1335** or via a remote peripheral interface.

[0107] The computing system **1300** may also be coupled, via the bus **1335**, to a display device **1350** to display information to a user. The computing system **1300** can also include an alphanumeric input device **1360**, including alphanumeric and other keys, which may be coupled to bus **1335** to communicate information and command selections to processor(s) **1310**. Another type of user input device includes a cursor control **1370** device, such as a touchpad, a mouse, a trackball, or cursor direction keys to communicate direction information and command selections to processor (s) **1310** and to control cursor movement on the display device **1350**. The computing system **1300** may also receive user input from a remote device that is communicatively coupled via one or more network interface(s) **1380**.

[0108] The computing system **1300** further may include one or more network interface(s) **1380** to provide access to a network, such as a local area network. The network interface(s) **1380** may include, for example, a wireless network interface having antenna **1385**, which may represent one or more antenna(e). The computing system **1300** can include multiple wireless network interfaces such as a combination of Wi-Fi, Bluetooth®, near field communication (NFC), and/or cellular telephony interfaces. The network interface(s) **1380** may also include, for example, a wired network interface to communicate with remote devices via network cable **1387**, which may be, for example, an Ethernet cable, a coaxial cable, a fiber optic cable, a serial cable, or a parallel cable.

[0109] In one embodiment, the network interface(s) **1380** may provide access to a local area network, for example, by conforming to IEEE **802.11** wireless standards and/or the wireless network interface may provide access to a personal area network, for example, by conforming to Bluetooth standards. Other wireless network interfaces and/or protocols can also be supported. In addition to, or instead of, communication via wireless LAN standards, network interface(s) **1380** may provide wireless communications using, for example, Time Division, Multiple Access (TDMA) protocols, Global System for Mobile Communications (GSM) protocols, Code Division, Multiple Access (CDMA) protocols, Long Term Evolution (LTE) protocols, and/or any other type of wireless communications protocol.

[0110] The computing system **1300** can further include one or more energy sources **1305** and one or more energy measurement systems **1345**. Energy sources **1305** can include an AC/DC adapter coupled to an external power source, one or more batteries, one or more charge storage devices, a USB charger, or other energy source. Energy measurement systems include at least one voltage or amperage measuring device that can measure energy consumed by the computing system **1300** during a predetermined period of time. Additionally, one or more energy measurement systems can be included that measure, e.g., energy consumed by a display device, cooling subsystem, Wi-Fi subsystem, or other frequently used or high-energy consumption subsystem.

[0111] In some embodiments, the hash functions described herein can utilize specialized hardware circuitry (or firmware) of the system (client device or server). For example, the function can be a hardware-accelerated function. In addition, in some embodiments, the system can use a function that is part of a specialized instruction set. For example, the can use an instruction set which may be an extension to an instruction set architecture for a particular type of microprocessor. Accordingly, in an embodiment, the system can provide a hardware-accelerated mechanism for performing cryptographic operations to improve the speed of performing the functions described herein using these instruction sets.

[0112] Reference herein to "one embodiment" or "an embodiment" means that a particular feature, structure, or characteristic described in conjunction with the embodiment can be included in at least one embodiment of the invention. The appearances of the phrase "in one embodiment" in

various places in the specification do not necessarily all refer to the same embodiment. The processes depicted in the figures that follow are performed by processing logic that comprises hardware (e.g. circuitry, dedicated logic, etc.), software (as instructions on a non-transitory machine-readable storage medium), or a combination of both hardware and software. Reference will be made in detail to various embodiments, examples of which are illustrated in the accompanying drawings. In the following detailed description, numerous specific details are set forth in order to provide a thorough understanding of the present invention. However, it will be apparent to one of ordinary skill in the art that the present invention may be practiced without these specific details. In other instances, well-known methods, procedures, components, circuits, and networks have not been described in detail so as not to unnecessarily obscure aspects of the embodiments.

[0113] It will also be understood that, although the terms first, second, etc. may be used herein to describe various elements, these elements should not be limited by these terms. These terms are only used to distinguish one element from another. For example, a first contact could be termed a second contact, and, similarly, a second contact could be termed a first contact, without departing from the scope of the present invention. The first contact and the second contact are both contacts, but they are not the same contact.

[0114] The terminology used herein is for the purpose of describing particular embodiments only and is not intended to be limiting as to all embodiments. As used in the description of the invention and the appended claims, the singular forms "a", "an" and "the" are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will also be understood that the term "and/or" as used herein refers to and encompasses any and all possible combinations of one or more of the associated listed items. It will be further understood that the terms "comprises" and/or "comprising," when used in this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof.

[0115] As used herein, the term "if" may be construed to mean "when" or "upon" or "in response to determining" or "in response to detecting," depending on the context. Similarly, the phrase "if it is determined" or "if [a stated condition or event] is detected" may be construed to mean "upon determining" or "in response to determining" or "upon detecting [the stated condition or event]" or "in response to detecting [the stated condition or event]," depending on the context.

[0116] Embodiments of computing devices, user interfaces for such devices, and associated processes for using such devices are described. In some embodiments, the computing device is a portable communications device such as a mobile telephone that also contains other functions, such as PDA and/or music player functions. Exemplary embodiments of portable multifunction devices include, without limitation, the iPhone®, iPad®, and iPod touch® devices from Apple Computer, Inc. of Cupertino, California.

[0117] In the foregoing description, example embodiments of the disclosure have been described. It will be evident that various modifications can be made thereto without departing from the broader spirit and scope of the disclosure. The specification and drawings are, accordingly, to be regarded in an illustrative sense rather than a restrictive sense. The specifics in the descriptions and examples provided may be used anywhere in one or more embodiments. The various features of the different embodiments or examples may be variously combined with some features included and others excluded to suit a variety of different applications. Examples may include subject matter such as a method, means for performing acts of the method, at least one machine-readable medium including instructions that, when performed by a machine cause the machine to perform acts of the method, or of an apparatus or system according to embodiments and examples described herein. Additionally, various components described herein can be a means for performing the operations or functions described herein.

[0118] Embodiments described herein provide for system and methods to crowdsource the location of wireless devices and accessories that lack a connection to a wide area network. One embodiment provides for a data processing system configured to perform operations comprising launching a user interface on an electronic device, the user interface to enable the determination of a location of a wireless accessory that is associated with the electronic device, generating a set of public keys included within a beacon signal broadcast by the wireless accessory, the beacon signal broadcast during a first period, sending the set of public keys to a server with a request to return data that corresponds with a public key in the set of public keys, decrypting the location data using a private key associated with the public key, processing the location data to determine a probable location for the wireless accessory, and displaying the probable location for the wireless accessory via the user interface..

[0119] One embodiment provides for a data processing system comprising a memory to store instructions for execution and one or more processors to execute the instructions stored in memory. The instructions, when executed, cause the one or more processors to launch a user interface on an electronic device that includes the data processing system, the user interface to enable the determination of a location of a wireless accessory that is associated with the electronic device, generate a set of public keys included within a beacon signal broadcast by the wireless accessory, the beacon signal broadcast during a first period, send the set of public keys to a server with a request to return data that corresponds with a public key in the set of public keys, decrypt the location data using a private key associated with the public key in response to receipt of location data from the server, process the location data to determine a probable location for the wireless accessory, and display the probable location for the wireless accessory via the user interface.

[0120] One embodiment provides for a method implemented on an electronic device, the method comprising performing a beacon scan using a baseband processor of the electronic device. The beacon scan can be performed while an application processor of the electronic device is in a low power state. The method additionally comprises receiving beacon data via a wireless radio coupled to the baseband processor, the beacon data to identify a wireless accessory. The beacon identifier can be a public key generated by the wireless accessory. The method additionally comprises storing a timestamp and beacon identifier to a beacon scan buffer, determining a location for the electronic device that corresponds with the timestamp when an application processor of the electronic device is in an active state, encrypt-

ing the location for the electronic device using the beacon identifier for the wireless accessory, and transmitting encrypted location data, the beacon identifier, and the timestamp to a server device, the server to store the encrypted location data for retrieval by an electronic device associated with the wireless accessory.

[0121] Those skilled in the art will appreciate from the foregoing description that the broad techniques of the embodiments can be implemented in a variety of forms. Therefore, while the embodiments have been described in connection with particular examples thereof, the true scope of the embodiments should not be so limited since other modifications will become apparent to the skilled practitioner upon a study of the drawings, specification, and following claims.

What is claimed is:

1. A non-transitory machine-readable medium storing instructions which, when executed, cause one or more processors of a data processing system to perform operations comprising:

launching a user interface on an electronic device, the user interface to enable determination of a location of a wireless accessory that is associated with the electronic device;

generating a set of public keys included within a signal broadcast by the wireless accessory, the signal broadcast during a first period;

sending hashes of the set of public keys to a server with a request to return data that corresponds with the set of public keys;

in response to receipt of location data from the server, decrypting the location data using a private key associated with the public key;

processing the location data to determine a probable location for the wireless accessory; and

displaying the probable location for the wireless accessory via the user interface.

2. The non-transitory machine-readable medium as in claim **1**, wherein the wireless accessory is paired with the electronic device and the set of public keys is generated based on a shared secret established between the wireless accessory and the electronic device during pairing.

3. The non-transitory machine-readable medium as in claim **1**, wherein the wireless accessory and the electronic device are registered with an online account and the private key and at least one of the public keys are stored in a keystore associated with the online account.

4. The non-transitory machine-readable medium as in claim **1**, wherein processing the location data to determine a probable location for the wireless accessory includes triangulating the probable location from multiple detected locations for the wireless accessory.

5. The non-transitory machine-readable medium as in claim **4**, wherein processing the location data to determine a probable location for the wireless accessory additionally includes processing signal strength or wireless ranging data associated with the multiple detected locations.

6. The non-transitory machine-readable medium as in claim **1**, wherein the set of public keys included within the signal broadcast during the first period is a first set of public keys and in response to receiving an indication from the server that location data for the wireless accessory is not available for the first set of public keys, generating a second set of public keys included within the signal during a second period and requesting location data from the server using hashes of the second set of public keys.

7. The non-transitory machine-readable medium as in claim **6**, wherein the second period is before the first period in time.

8. The non-transitory machine-readable medium as in claim **6**, wherein the second period is after the first period in time.

9. The non-transitory machine-readable medium as in claim **8**, the operations additionally comprising receiving a notification from the server indicating that a wireless beacon including a public key in the second set of public keys has been detected.

10. A data processing system comprising:

a memory to store instructions for execution;

one or more processors to execute the instructions stored in memory, wherein the instructions, when executed cause the one or more processors to:

launch a user interface on an electronic device that includes the data processing system, the user interface to enable determination of a location of a wireless accessory that is associated with the electronic device;

generate a set of public keys included within a signal broadcast by the wireless accessory, the signal broadcast during a first period;

send the set of public keys to a server with a request to return data that corresponds with the set of public keys;

decrypt location data using a private key associated with the public key in response to receipt of encrypted location data from the server;

process the location data to determine a probable location for the wireless accessory; and

display the probable location for the wireless accessory via the user interface.

11. The data processing system as in claim **10**, wherein the wireless accessory is paired with the electronic device, a shared secret was established between the wireless accessory and the electronic device while pairing, and the set of public keys is generated based on the shared secret.

12. The data processing system as in claim **10**, wherein the one or more processors are to send hashes of the set of public keys to the server with the request to return data that corresponds with the set of public keys

13. The data processing system as in claim **10**, wherein the wireless accessory and the electronic device are registered with an online account and the private key and at least one of the public keys are stored in a keystore associated with the online account.

14. A method implemented on an electronic device, the method comprising:

performing a beacon scan using a baseband processor of the electronic device;

receiving beacon data via a wireless radio coupled to the baseband processor, the beacon data to identify a wireless accessory and the beacon identifier is a public key generated by the wireless accessory;

storing a timestamp and beacon identifier associated with the beacon data to a beacon scan buffer;

determining a location for the electronic device that corresponds with the timestamp when an application processor of the electronic device is in an active state;

encrypting the location for the electronic device using the beacon identifier for the wireless accessory; and

transmitting encrypted location data, the beacon identifier, and the timestamp to a server device, the server to store the encrypted location data for retrieval by an electronic device associated with the wireless accessory.

15. The method as in claim **14**, additionally comprising determining a signal strength measurement for a signal associated with the beacon data and transmitting the signal strength measurement with the encrypted location data.

16. The method as in claim **14**, additionally comprising determining a distance to the wireless accessory using an ultra-wide band radio.

17. The method as in claim **14**, additionally comprising placing the application processor of the electronic device in the active state in response to a determination that the beacon scan buffer is full.

18. The method as in claim **14**, wherein determining the location for the electronic device includes:

performing a periodic Wi-Fi scan to gather a list of service set identifiers within range of the electronic device; and

determining the location for the electronic device, the location determined at least in part based on the list of service set identifiers.

19. The method as in claim **14**, additionally comprising:

storing the encrypted location data to a transmission queue in memory of the electronic device; and

transmitting the encrypted location data to the server during a transmit interval.

20. The method as in claim **14**, additionally comprising performing the beacon scan using a baseband processor of the electronic device while an application processor of the electronic device is in a low power state.

* * * * *