US 20140129686A1

(19) United States

(12) Patent Application Publication

(10) Pub. No.: US 2014/0129686 A1

(43) Pub. Date: May 8, 2014

Haemel et al.

(54) **MOBILE COMPUTING DEVICE CONFIGURED TO FILTER AND DETECT APPLICATION PROFILES, A METHOD OF MANUFACTURING THE SAME AND AN EXTERNAL SOURCE FOR DELIVERING HIERARCHICAL FILTERED APPLICATION PROFILES TO MOBILE COMPUTING DEVICES**

(71) Applicant: **Nvidia Corporation**, Santa Clara, CA (US)

(72) Inventors: **Nicholas Haemel**, Santa Clara, CA (US); **Cathy Donovan**, Santa Clara, CA (US); **David Chait**, Bedford, MA (US)

(73) Assignee: **Nvidia Corporation**, Santa Clara, CA (US)

(21) Appl. No.: **13/894,869**

(22) Filed: **May 15, 2013**

Related U.S. Application Data

(60) Provisional application No. 61/724,088, filed on Nov. 8, 2012.

Publication Classification

(51) **Int. Cl.**
G06F 15/173 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 15/173*** (2013.01)
USPC **709/220**; 709/223

(57) **ABSTRACT**

A mobile computing device, a method of operating thereof, a method of manufacturing and an external source for dynamic profile settings for mobile computing devices. In one embodiment, the mobile computing device includes: (1) a settings reservoir configured to store dynamic sets of profile settings and static set of profile settings for the computing device and (2) a profile generator configured to generate coalesced sets of profile settings for applications on the computing device based on the dynamic sets of profiles and the static set of profiles.

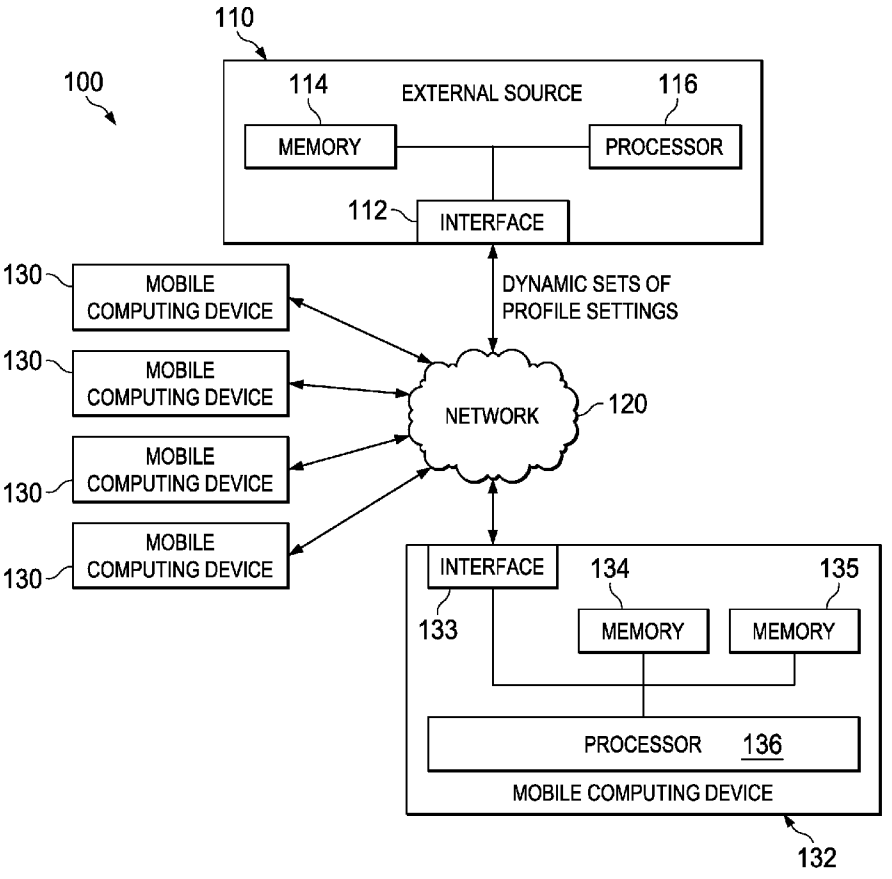
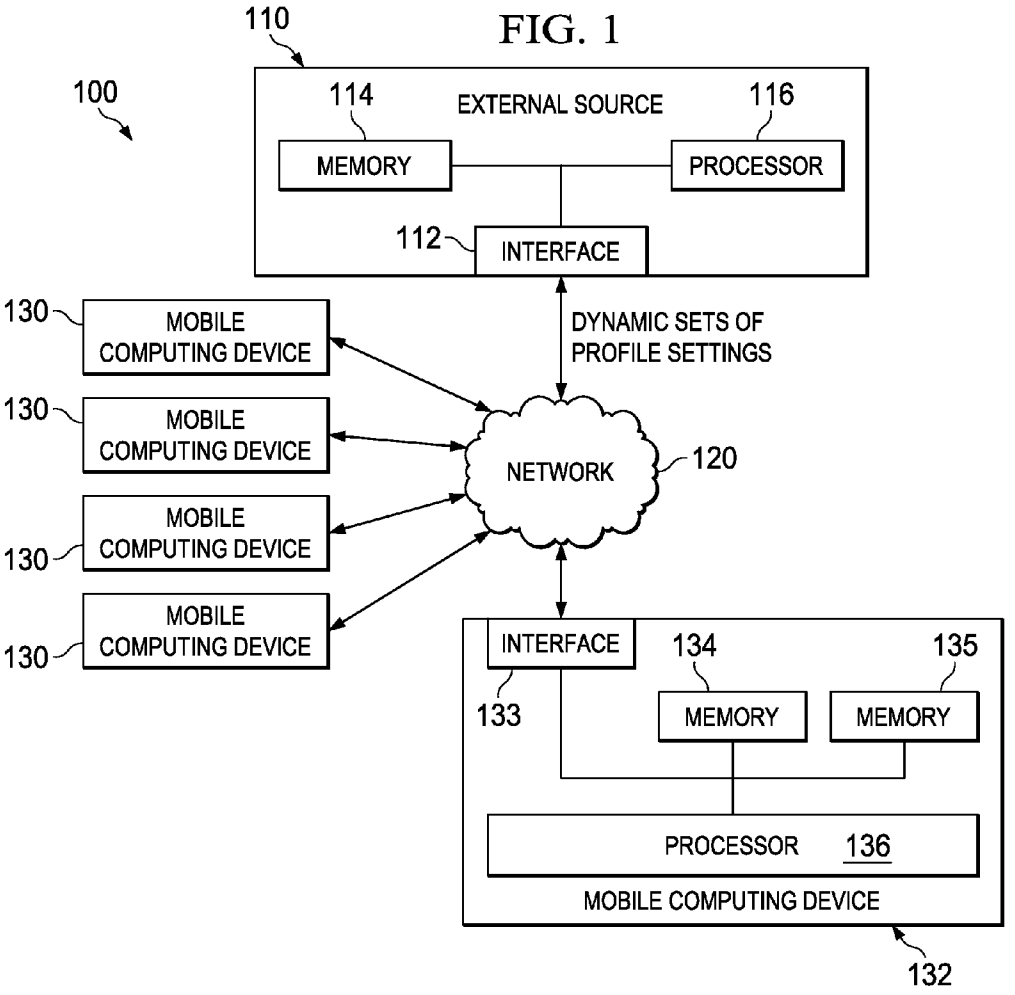


FIG. 1


110
100
114
EXTERNAL SOURCE
116
MEMORY
PROCESSOR
112
INTERFACE
130
MOBILE COMPUTING DEVICE
DYNAMIC SETS OF PROFILE SETTINGS
NETWORK
120
INTERFACE
134
135
133
MEMORY
MEMORY
PROCESSOR 136
MOBILE COMPUTING DEVICE
132

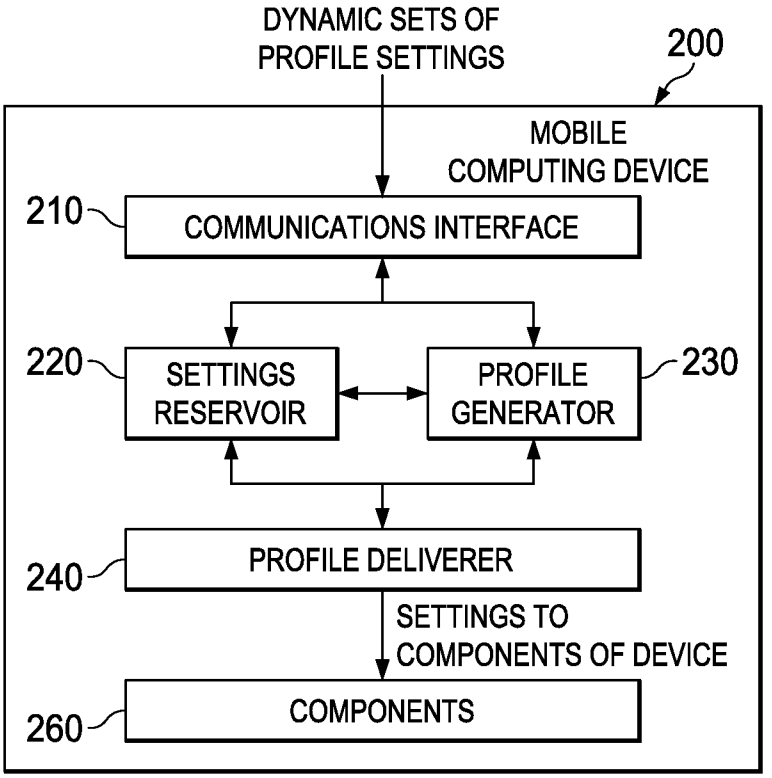

DYNAMIC SETS OF
PROFILE SETTINGS
200
MOBILE
COMPUTING DEVICE
210
COMMUNICATIONS INTERFACE
220
SETTINGS
RESERVOIR
PROFILE
GENERATOR
230
240
PROFILE DELIVERER
SETTINGS TO
COMPONENTS OF DEVICE
260
COMPONENTS

FIG. 2

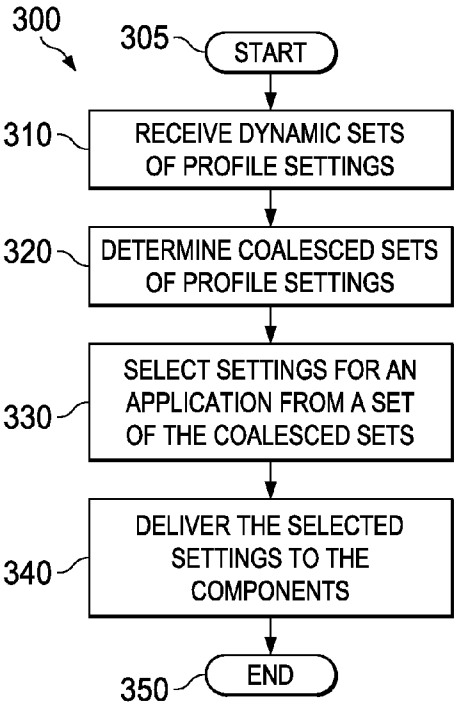

300
305
START
310
RECEIVE DYNAMIC SETS OF PROFILE SETTINGS
320
DETERMINE COALESCED SETS OF PROFILE SETTINGS
330
SELECT SETTINGS FOR AN APPLICATION FROM A SET OF THE COALESCED SETS
340
DELIVER THE SELECTED SETTINGS TO THE COMPONENTS
350
END

FIG. 3

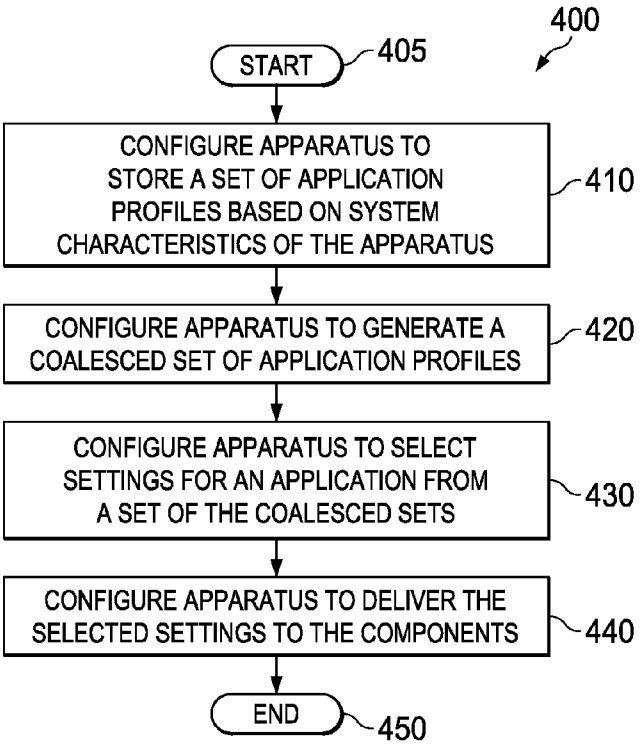

400
START
405
CONFIGURE APPARATUS TO STORE A SET OF APPLICATION PROFILES BASED ON SYSTEM CHARACTERISTICS OF THE APPARATUS
410
CONFIGURE APPARATUS TO GENERATE A COALESCED SET OF APPLICATION PROFILES
420
CONFIGURE APPARATUS TO SELECT SETTINGS FOR AN APPLICATION FROM A SET OF THE COALESCED SETS
430
CONFIGURE APPARATUS TO DELIVER THE SELECTED SETTINGS TO THE COMPONENTS
440
END
450

FIG. 4

MOBILE COMPUTING DEVICE CONFIGURED TO FILTER AND DETECT APPLICATION PROFILES, A METHOD OF MANUFACTURING THE SAME AND AN EXTERNAL SOURCE FOR DELIVERING HIERARCHICAL FILTERED APPLICATION PROFILES TO MOBILE COMPUTING DEVICES

CROSS-REFERENCE TO RELATED APPLICATION

[0001] This application claims the benefit of U.S. Provisional Application Ser. No. 61/724,088, filed by Nicholas Haemel, et al., on Nov. 8, 2012, entitled "METHOD FOR DETERMINING DRIVER LIBRARY COMPATIBILITY AND DELIVERING UPDATES," commonly assigned with this application and incorporated herein by reference.

TECHNICAL FIELD

[0002] This application is directed, in general, to computing devices and, more specifically, to profile settings for different components of mobile computing devices.

BACKGROUND

[0003] Mobile telephones typically perform many functions in addition to providing voice communication. Many mobile telephones, such as smart phones, and other mobile computing devices now also provide additional services including the ability to communicate by text, connect to the web, watch videos, make videos, take pictures, play games, etc. The various computing devices use many types of applications to perform these different functions.

SUMMARY

[0004] In one aspect, the disclosure provides a mobile computing device. In one embodiment, the mobile computing device includes: (1) a settings reservoir configured to store dynamic sets of profile settings and static set of profile settings for the computing device and (2) a profile generator configured to generate coalesced sets of profile settings for applications on the computing device based on the dynamic sets of profiles and the static set of profiles.

[0005] In another aspect, the disclosure provides a method of operating a mobile computing device. In one embodiment, the method includes: (1) receiving dynamic sets of profile settings based on system characteristics of the computing device, (2) determining a coalesced set of profile settings for the computing device based on the dynamic sets of profile settings and sets of static profile settings stored on the computing device and (3) selecting, for an application on the computing device, settings from a set of the coalesced sets of profile settings for components of the computing device.

[0006] In yet another aspect, the disclosure provides a method of configuring an apparatus to operate employing profile settings for components thereof that are tailored to particular applications on the apparatus and to characteristics of the components. In one embodiment, the method of manufacturing includes: (1) configuring the apparatus to store a set of application profiles based on system characteristics of the computing device, (2) configuring the apparatus to generate a coalesced set of application profiles by comparing the set of application profiles with a second set of application profiles from another source and (3) configuring the apparatus to deliver profile settings to components of the apparatus for an application when the application is launched, wherein the profile settings are from a set of the coalesced sets of profile settings.

[0007] In still yet another aspect, the disclosure provides an external source for dynamic profile settings for mobile computing devices. In one embodiment, the external source includes: (1) a memory configured to store profile settings for applications that execute on mobile computing devices and (2) a processor configured to determine a dynamic set of profile settings for executing an application on a specific one of the mobile computing devices by hierarchically filtering the profile settings according to characteristics of the specific one.

BRIEF DESCRIPTION

[0008] Reference is now made to the following descriptions taken in conjunction with the accompanying drawings, in which:

[0009] FIG. **1** illustrates a diagram of an embodiment of a distribution system employed by a mobile computing device to receive dynamic sets of application profiles according to the principles of the disclosure;

[0010] FIG. **2** illustrates a block diagram of an embodiment of a mobile computing device constructed according to the principles of the disclosure;

[0011] FIG. **3** illustrates a flow chart of an embodiment of a method for operating a mobile computing device carried out according to the principles of the disclosure; and

[0012] FIG. **4** illustrates a flow diagram of an embodiment of a method of configuring an apparatus for operation carried out according to the principles of the disclosure.

DETAILED DESCRIPTION

[0013] Mobile computing devices have the ability to perform multiple services or functions under the direction of different applications loaded thereon. An application is a computer program or programs designed for use by an end user. In some embodiments, an application includes both systems programs or software and end-user programs or software. For optimal enjoyment or execution of an application from a user's perspective or from the perspective of a mobile computing device itself, the settings for the various components of a mobile computing device can vary for the different applications. For example, one application may have different power requirements and perform optimally according to particular power management guidelines. Another application can ideally operate with particular settings for a display of the mobile computing device compared to other applications. Thus, the optimal settings or profile settings for the various components of a mobile computing device for different applications can change. Profile settings as used herein are a group or set of states and/or values of parameters for the components or modules of a mobile computing device. A component or module as used herein is a software library or module providing specific system functionality on the mobile computing device such as CPU governor strategy, OpenGL rendering, display composition, etc.

[0014] Accordingly, the disclosure provides a scheme that allows a mobile computing device to select a set of coalesced profile settings that correspond to a particular application on a mobile computing device. A mobile computing device is a computing device constructed specifically for a mobile archi-

tecture that enables portability. A mobile computing device includes the mobile hardware and software that is configured to perform various functions including communicating over a network. Mobile computing devices can be mobile telephones including smart phones, tablets, clamshells, game devices, etc., that include at least one processor and memory. One skilled in the art will also understand that each of the mobile computing devices disclosed herein include other components that are typically included in such devices including a display, user interface, power supply, communications interface, etc.

[0015] The coalesced profile settings for an application are generated from dynamic sets of profile settings and at least one static set of profile settings. The dynamic sets of profile settings are provided from a source that is external to the mobile computing device, i.e., an external source. The external source can be a server that the mobile computing device connects to via, for example, a conventional connection that is wired, wireless or a combination thereof. The external source provides a scalable solution to providing profile settings to mobile computing devices.

[0016] The dynamic sets of profile settings include sets of profile settings wherein each of the sets are specifically tailored for the characteristics of the components of the mobile computing device with respect to a particular application on the mobile computing device. For example, the dynamic sets of profile settings can include a set of profile settings for a particular game, a set of profile settings for a voice application, a set of profile settings for a texting application, a system application, etc., wherein each of the sets is tailored for the components or modules of the mobile computing device. As such, a set of the dynamic sets of profile settings correspond to the specific characteristics of the components of a mobile computing device for each application. Thus, each set of the dynamic sets of profile settings can be for a particular application and directed to characteristics of a device including, for example, a specific manufacturer, a model, a carrier, an operating system, a version of an operating system, a chip type, a chip family or any combination thereof. In some embodiments, the dynamic sets of profile settings are hierarchically filtered application profiles that are filtered based on the characteristic data.

[0017] Thus, in some embodiments the external source can generate, store and deliver hierarchically filtered application profiles to mobile computing devices. As such, the external source is configured to use a hierarchical filtering approach to determine which profile settings to return to a mobile computing device. The hierarchical categories for filtering the profile settings can range from general (e.g., operating system, chip family) to specific (e.g., device name, device manufacturer) characteristic data of the mobile computing device. In one embodiment the external source is configured to store many profile settings associated with a mobile computing device and only select the correct profile setting to send to the mobile computing device. The correct profile setting is the most specific, hierarchically filtered profile setting for a particular mobile computing device. For example, game XYZ may have a general profile setting containing stereo settings stored in the database of the external source. Additionally, a device profile for a specific type of mobile computing device is stored on the external source that describes the stereo settings and power biases for the game XYZ. In this example, the correct profile returned to the mobile computing device would be the device profile for the specific type of mobile computing device instead of the general profile for the game XYZ.

[0018] Accordingly, the hierarchical filtered application profiles define a set of system states (e.g., CPU/GPU governor biases, CPU caps, screen resolution, screen size, stereo settings, etc.) to be set on a mobile computing device for each desired application. The values for a profile set can be determined experimentally or algorithmically and stored in a profile database on an external source (e.g., a server). As such, in one embodiment, the external source is configured to hierarchically filter a database of profile settings based on the characteristic data from a mobile computing device to determine a dynamic set of profile settings to be returned to the mobile computing device. The characteristic data can be provided to the external source via a query from the mobile computing device. In some embodiments, each mobile computing device can periodically (e.g., every forty eight hours) provide the characteristic data to the external source and receive any updates to its set of application profiles. Additionally, the external source can detect which mobile computing devices are older and control how often they query for new profile settings. Profile settings for older devices change less frequently and this mechanism provides a way to reduce traffic on mobile computing devices nearing end-of-life.

[0019] By employing the hierarchical filtered application profiles, the external source provides profile settings that vary for different reasons such as different versions of the operating system, different chipsets, etc. Thus, multiple profile settings can be provided for a single application.

[0020] The static set of profile settings is stored on the mobile computing device. Unlike the dynamic sets of profile settings, the static set of profile settings do not change. In one embodiment, the static set of profile settings includes original equipment manufacturer (OEM) settings for components of the mobile computing device. In some embodiments, multiple static sets of profile settings are stored and employed to generate the coalesced profile settings. In such embodiments, the discussion herein regarding a single static set of profile settings applies to the multiple static sets of profile settings.

[0021] In one embodiment, the mobile computing device generates the coalesced profile settings by receiving and loading dynamic sets of profile settings. As noted above, the dynamic sets can be downloaded from a server over a communication network. The mobile computing device also loads a static set of profile settings contained on the device and generates the coalesced sets of profile settings based on both the dynamic sets of profile settings and the static set of profile settings. In one embodiment, the mobile computing device is configured to replace any overlapping profile settings in the dynamic sets with settings from the static profile settings. This can be based on if the settings from the static set of profile settings have override capabilities. If both a static profile and dynamic profile exist for an application, the static set will be chosen. For example, if the dynamic profile limits total device power to 3.5 watts for a given application and the static profile limits total device power to 3.0 watts, the mobile computing device will limit to the static setting of 3.0 watts. Settings do not always overlap between the two profile types. In this case the profile settings are used in the only profile type in which they exist. As such, settings from the static set of profile settings override settings from dynamic sets when a conflict exists. The OEM, therefore, can have final say on how components operate on their device. This creates a single set

of profiles that persist on the mobile computing device referred to herein as the coalesced sets of profile settings. A profile generator on the mobile computing device is used to generate the coalesced sets of profile settings.

[0022] When an application starts on the mobile computing device, any of the system modules of the mobile computing device, such as OpenGL, Power management, 2D, etc., query a profile deliverer of the mobile computing device for the profile settings for that particular application. The profile deliverer determines which set of the coalesced sets of profile settings are relevant for the application and deliver the appropriate settings from the set of coalesced sets of profile settings to correctly set up the systems of the mobile computing device to optimally execute the application.

[0023] In one embodiment, the profile deliverer determines which set of the coalesced sets of profile settings are to be used when an application is launched, by examining the name of the application, the version of the application, a hash or checksum of the application, or a combination of any of the above. For example, the profile deliverer can examine the name of the application and a hash or checksum of the binary of the application. These are then compared against the database of coalesced sets of profile settings and both must match for an application profile to be used. This prevents applications on the mobile computing device that have been tampered with from loading application profiles. Additionally, this allows developers to be unaffected by existing profile settings while developing updates to their applications which might have profiles for older versions of the applications.

[0024] FIG. **1** illustrates a diagram of an embodiment of a distribution system **100** employed by mobile computing devices to receive dynamic sets of application profiles according to the principles of the disclosure. The distribution system **100** includes an external source **110**, a communications network **120** and multiple mobile computing devices **130**. The external source **110** includes a communications interface **112**, a processor **116** and a memory **114**. The processor **116** can be configured to perform and direct the various operations of the external source **110** disclosed herein. Accordingly, the processor **116** includes the necessary circuitry and logic to perform the disclosed features of the external source **110**. The memory **114** can be configured to store the dynamic profile settings. Each of the components of the external source **110** is connected via conventional means. In one embodiment, the external source **110** is a server.

[0025] The external source **110** is configured to store dynamic sets of profile settings for the mobile computing devices **130**. Additionally, the external source **110** is configured to determine a set of profile settings for different types of mobile computing devices according to the characteristics of the mobile computing device. The external source **110** is also configured to send the dynamic sets of profile settings to the mobile computing devices **130**. The external source **110** includes a communications interface **112**, a memory **114** and a processor **116**.

[0026] In some embodiments, the external source **110** is configured to periodically send the dynamic sets of profile settings to the mobile computing devices **130**. The queries and downloads can be communicated according to conventional networking protocols over the network **120**. The communications interface **112** is configured to receive and send the queries and downloads. In one embodiment, the communications interface **112** includes conventional communications circuitry.

[0027] In one embodiment the queries include characteristic data about the mobile computing devices **130** for filtering by device-specific characteristics. In some embodiments, a query includes a shim that includes multiple data fields pertaining to the specific one of the mobile computing devices **130** that generated the query. For example, a query includes data for "type" that lists the specific type of mobile computing device and components thereof. Additionally, the query includes multiple data sections that further describe the device and components. Furthermore, the query includes a "notes" section wherein additional information or options can be provided for a specific one of the multiple data fields.

[0028] For example, the type column can include device, system, system software, display, features, applications and miscellaneous. For device, the data sections can include manufacturer, model, serial number, security ID, carrier and operating system version. For system, the data sections can include chip type, memory, memory clock, GPU clock max, CPU clock max, power management mode and network connection. The notes for the power management mode can include max performance, balanced, max battery and none. For the network connection, the notes section can include type and speed. For the applications, the data sections can include not only the application name but also the version of the application. Accordingly, profile settings, can be matched to specific versions of applications. With the specific data from each of the mobile computing devices **130**, the correct hierarchical filtered profile settings can be selected by the external source **110** for delivery to the appropriate one of the mobile computing devices **130**. The processor **116** can be configured to hierarchically filter the profile settings stored on the memory **114** according to the characteristic data provided by the query.

[0029] The external source **110** can log the data provided by the queries, traverse the dynamic sets of profile settings stored thereon based on the data from the mobile computing devices **130** and return one of the dynamic sets of profile settings to the appropriate one of the mobile computing devices **130**. In some embodiment, the external source **110** is configured to store already determined hierarchical filtered application profiles for possible future downloads. The external source **110** communicates with the mobile computing devices **130** via the communications network **120**. The communications network **120** is a conventional network that is used by the mobile computing devices to transmit and receive data to and from the external source **110**. The external source **110** and the mobile computing devices **130** include the necessary circuitry to communicate over the communications network employing the proper networking protocol.

[0030] One of the mobile computing devices **130** is selected as a representative and is denoted as mobile computing device **132**. The mobile computing device **132** includes an interface **133**, a memory **134**, a memory **135** and a processor **136**. One skilled in the art will understand that the mobile computing device **132** also includes additional components such as a display, a user interface, a microphone, a speaker, etc. that are not illustrated herein.

[0031] The interface **133** is a communication interface that includes the necessary circuitry to transmit and receive data via, for example, the communications network **120**. As such, the interface **133** is configured to communicate via communication protocols including, for example, networking proto-

cols such as Ethernet, Wi-Fi or Internet protocol. The interface **133** is configured to communicate via wireless or wired connections.

[0032] The interface **133** receives the dynamic sets of profile settings from the external source **110**. The dynamic sets are then stored in the memory **134**. Receiving, encoding and storing the dynamic sets can be performed in a conventional manner.

[0033] Also stored within the mobile computing device **132** are static set of profile settings. The static set of profile settings is stored in the memory **135**. In one embodiment, the memory **134** is a dynamic memory and the memory **135** is a static memory. In some embodiments, the memories **134**, **135** can be partitioned portions of a single memory.

[0034] The processor **136** is configured to generate coalesced sets of profile settings from the dynamic sets and the static set of profile settings. Accordingly, the processor **136** includes the necessary circuitry and logic to load the dynamic sets and the static set of profile settings and generate therefrom the coalesced sets of profile settings. In one embodiment, the coalesced sets of profile settings are then stored in the memory **134**.

[0035] The processor **136** is also configured to deliver settings from a set of the coalesced sets of profile settings to the various components of the mobile computing device **132** for particular applications operating thereon. The processor **136** is configured to verify the particular set before selecting and delivering the settings. This can be done by comparing the name of a particular application to the database of coalesced sets of profile settings. In some embodiments, additional security is provided by also comparing a hash or checksum value of an application on the mobile computing device that is, for example, being launched, to a hash or checksum value stored under the name of the application with the coalesced sets of profile settings. Communication between the interface **133**, the memory **134**, the memory **135** and the processor **136** and with the various components of the mobile computing device **132** can be via conventional means employing typical protocols known in the art.

[0036] FIG. **2** illustrates a block diagram of an embodiment of a mobile computing device **200** constructed according to the principles of the disclosure. The mobile computing device **200** can be one of the mobile computing devices **130** of FIG. **1**. The mobile computing device **200** is an apparatus that a user employs for various services or for operating various applications. In one embodiment, the mobile computing device **200** is a computing device designed for voice communication, such as a smart telephone. In another embodiment, the mobile computing device **200** is not designed for voice communication over a cellular communication network.

[0037] The mobile computing device **200** includes a communications interface **210**, a settings reservoir **220**, a profile generator **230** and a profile deliverer **240**. One skilled in the art will understand that the mobile computing device **200** also includes additional components, represented by components **260**, that are not illustrated such as, a display, a user interface, a power supply, etc.

[0038] The communications interface **210** can be a conventional interface on mobile computing devices that is configured to transmit and receive data. The communications interface **210** can include an antenna for wireless communication. In one embodiment, the communications interface **210** can include a port or connector for wired communication.

[0039] The communications interface **210** is configured to receive the dynamic sets of profile settings. In one embodiment, the communications interface **210** is configured to receive the dynamic sets of profile settings via a wireless connection. Thus, the communications interface **210** may employ an antenna to communicate with a server, such as external source **110**, to receive the dynamic sets. In such embodiments, the dynamic sets of profile settings may be received on a periodic basis. For example, the dynamic sets may be downloaded every **48** hours or another selected time period. In some embodiment, the dynamic sets of profile settings may only be downloaded when a change has been made thereto. A query can be used to inquire if a change has been made. As discussed above, the query can include characteristic data of the mobile computing device **200** that details the characteristics and features of the mobile computing device **200**.

[0040] In other embodiments, the communications interface **210** may receive the dynamic sets of profile settings from a wired connection such as when coupled to a desk top computer. In some embodiments, the communication interface **210** receives a memory stick that has stored thereon the dynamic sets of profile settings.

[0041] The settings reservoir **220** is configured to store dynamic sets of profile settings and static set of profile settings for the mobile computing device **200**. As such, the settings reservoir **220** is configured to store two distinct sets of profile settings. In one embodiment, the settings reservoir **220** is implemented as a memory or memories. In some embodiments, the dynamic sets of profile settings are stored in a first memory and the static set of profile settings is stored in a second memory distinct from the first memory. The first memory can be a RAM and the second memory can be a ROM.

[0042] The profile generator **230** is configured to generate coalesced sets of profile settings for applications on the mobile computing device **200** based on the dynamic sets of profiles and the static set of profiles.

[0043] The profile deliverer **240** is configured to deliver settings from a set of the coalesced profile settings to components of the mobile computing device **200** for the application. In one embodiment, the profile deliverer **240** is configured to determine the set of the coalesced profile settings by verifying both a name of the application and a hash or checksum of a binary of the application with a database of the coalesced profile settings stored on the settings reservoir. The database can include the names of applications, hash or checksum values and other identifying features for security, such as hash or checksum values, version, etc. Additionally, the database includes the settings for each of the different applications.

[0044] The profile deliverer **240** can be prompted to select and deliver the settings in response to the application being launched on the computing device. Thus, a user can initiate or launch an application employing a user interface. Thereafter, the settings for the various components of the mobile computing device **200** are retrieved for executing the application. The profile deliverer **240** identifies the relevant set of coalesced sets of profile settings that correspond to the launched application. As noted above, the application can be verified through various criteria. The profile deliverer **240** then delivers the settings of the selected set of coalesced sets of profile settings to the components of the mobile computing device **200** for execution of the application. In one embodiment, the

profile deliverer **240** is configured to deliver the settings in response to a query from the components of the mobile computing device **200**. In this embodiment, each individual component can individually request settings for executing a particular launched application. The profile deliverer **240** still verifies the set of coalesced sets of profile settings for the application and then selects the particular settings for the requesting component from the set and delivers those settings.

[0045] FIG. **3** illustrates a flow chart of an embodiment of a method **300** for operating a mobile computing device carried out according to the principles of the disclosure. The method **300** or at least a portion thereof can be implemented as a series of operating instructions stored on a non-transitory computer readable medium that directs the operation of a processor. As such, the method **300** may be implemented as a computer program product. The method **300** begins in a step **305**.

[0046] In a step **310**, dynamic sets of profile settings based on system characteristics of the computing device are received. The dynamic sets can be received via a server over a communications network.

[0047] Coalesced sets of profile settings are determined in a step **320** for the computing device. In one embodiment, the coalesced sets of profile settings are generated from the dynamic sets of profile settings and a set of static profile settings stored on the mobile computing device. In some embodiments, generating the coalesced sets of profile settings includes replacing overlapping profile settings in the dynamic sets of profile settings with profile settings from the static set of profile settings. This can be based on if the particular static settings have override capability.

[0048] In a step **330**, settings are selected, for an application on the mobile computing device, from a set of the coalesced sets of profile settings for components of the computing device. In one embodiment, the method **300** includes verifying the application with the relevant settings. Verification can be based on comparing the name of the application with a database of the coalesced sets of profile settings. Other verification procedures that provide additional security can also be used.

[0049] In a step **340**, the selected settings are delivered to the components for execution during operation of the application. The selected settings can be delivered based on receiving queries from the components in response to launching the application and selecting the set in response to the queries. The method **340** ends in a step **350**.

[0050] FIG. **4** illustrates a flow diagram of an embodiment of a method **400** of configuring an apparatus for operation carried out according to the principles of the disclosure. The apparatus can be a tablet, a clamshell, a smart phone, a gaming device or another type of mobile computing device, such as a portable device, that executes and operates applications. The method **400** may occur at manufacturing of the apparatus. In other embodiments, the method **400** could occur when upgrading the apparatus. Configuring as used herein refers to providing the apparatus with the necessary logic, circuitry, instructions or combination thereof to perform the particular functions. The method **400** begins in a step **405**.

[0051] In a step **410**, the apparatus is configured to store a set of application profile settings based on system characteristics of the apparatus.

[0052] In a step **420**, the apparatus is configured to generate a coalesced set of application profile settings by comparing the set of application profiles with a second set of application profiles from a different source. In one embodiment, the apparatus is configured to generate the coalesced set of application profile settings by replacing settings for a component of the apparatus with settings for the component from the second set of application profiles. At least a portion of the second set of application profiles can be established during manufacturing of the apparatus.

[0053] In a step **430**, the apparatus is configured to select settings for an application from a set of the coalesced sets of profile settings.

[0054] In a step **440**, the apparatus is configured to deliver profile settings to components of the apparatus for an application when the application is launched. The profile settings are from a set of the coalesced sets of profile settings. The method **400** ends in a step **450**.

[0055] A portion of the above-described apparatuses, systems or methods may be embodied in or performed by various processors or computers, wherein the computers are programmed or store executable programs of sequences of software instructions to perform one or more of the steps of the methods. The software instructions of such programs may represent algorithms and be encoded in machine-executable form on non-transitory digital data storage media, e.g., magnetic or optical disks, random-access memory (RAM), magnetic hard disks, flash memories, and/or read-only memory (ROM), to enable various types of digital data processors or computers to perform one, multiple or all of the steps of one or more of the above-described methods, or functions of the apparatuses or systems described herein.

[0056] Portions of disclosed embodiments may relate to computer storage products with a non-transitory computer-readable medium that have program code thereon for performing various computer-implemented operations that embody a part of an apparatus, system or carry out the steps of a method set forth herein. Non-transitory used herein refers to all computer-readable media except for transitory, propagating signals. Examples of non-transitory computer-readable media include, but are not limited to: magnetic media such as hard disks, floppy disks, and magnetic tape; optical media such as CD-ROM disks; magneto-optical media such as floptical disks; and hardware devices that are specially configured to store and execute program code, such as ROM and RAM devices. Examples of program code include both machine code, such as produced by a compiler, and files containing higher level code that may be executed by the computer using an interpreter.

[0057] Those skilled in the art to which this application relates will appreciate that other and further additions, deletions, substitutions and modifications may be made to the described embodiments.

What is claimed is:

1. A mobile computing device, comprising:

a settings reservoir configured to store dynamic sets of profile settings and static set of profile settings for said computing device; and

a profile generator configured to generate coalesced sets of profile settings for applications on said computing device based on said dynamic sets of profiles and said static set of profiles.

2. The mobile computing device as recited in claim **1** further comprising a profile deliverer configured to deliver settings from a set of said coalesced profile settings to components of said mobile computing device for said application.

3. The mobile computing device as recited in claim 2 wherein said profile deliverer is configured to deliver said settings in response to components of said mobile computing device requests on said application being launched on said mobile computing device.

4. The mobile computing device as recited in claim 2 wherein said profile deliverer is further configured to determine said set of said coalesced profile settings by verifying both a name of said application and a hash or checksum of a binary of said application with a database of said coalesced profile settings stored on said settings reservoir.

5. The mobile computing device as recited in claim 1 further comprising a communications interface configured to wirelessly receive said dynamic sets of profile settings.

6. The mobile computing device as recited in claim 2 wherein said profile deliverer is further configured to deliver said settings in response to a query from said components.

7. The mobile computing device as recited in claim 1 wherein said static set of profiles are original equipment manufacturer profiles.

8. The mobile computing device as recited in claim 1 wherein said dynamic sets of profile settings are based on characteristics of said mobile computing device.

9. A method of operating a mobile computing device, comprising:

receiving dynamic sets of profile settings based on system characteristics of said computing device;

determining a coalesced set of profile settings for said computing device based on said dynamic sets of profile settings and sets of static profile settings stored on said computing device; and

selecting, for an application on said computing device, settings from a set of said coalesced sets of profile settings for components of said computing device.

10. The method as recited in claim 9 further comprising delivering said settings to said components for execution during operation of said application.

11. The method as recited in claim 9 wherein said determining includes replacing overlapping profile settings in said dynamic sets of profile settings with profile settings from said static set of profile settings.

12. The method as recited in claim 9 further comprising verifying said set of coalesced sets of profile settings correspond to said application based on a name of said application and an additional security check derived from a binary of said application.

13. The method as recited in claim 9 further comprising receiving queries from said components in response to launching said application and selecting said set in response to said queries.

14. The method as recited in claim 9 wherein said dynamic sets of profile settings are based on characteristics of said computing device.

15. The method as recited in claim 9 wherein said sets of static profile settings include original equipment manufacturer settings.

16. A method of configuring an apparatus to operate employing profile settings for components thereof that are tailored to particular applications on said apparatus and to characteristics of the components, the method comprising:

configuring said apparatus to store a set of application profiles based on system characteristics of said computing device;

configuring said apparatus to generate a coalesced set of application profiles by comparing said set of application profiles with a second set of application profiles from another source; and

configuring said apparatus to deliver profile settings to components of said apparatus for an application when said application is launched, wherein said profile settings are from a set of said coalesced sets of profile settings.

17. The method as recited in claim 16 wherein said method occurs during manufacturing of said apparatus.

18. The method as recited in claim 16 wherein apparatus is a mobile computing device.

19. The method as recited in claim 16 further comprising configuring said apparatus to download said set of application profiles from a server including hierarchical filtered profile settings.

20. The method as recited in claim 16 further comprising configuring said apparatus to generate said coalesced set of application profiles by replacing settings for a component of said apparatus with settings for said component from said second set of application profiles, wherein at least a portion of said second set of application profiles are established during manufacturing of said apparatus.

21. An external source for dynamic profile settings for mobile computing devices, comprising:

a memory configured to store profile settings for applications that execute on mobile computing devices; and

a processor configured to determine a dynamic set of profile settings for executing an application on a specific one of said mobile computing devices by hierarchically filtering said profile settings according to characteristics of said specific one.

22. The external source as recited in claim 21 further comprising a communications interface configured to receive a query from said specific one that includes characteristic data thereof, said processor configured to hierarchically filter said profile settings based on said characteristic data.

* * * * *