A METHOD FOR REAL-TIME SYNCHRONIZATION BETWEEN A DEVICE AND HOST SERVERS

Background of the Invention

Field of the Invention

This invention relates to a method for autonomous synchronization in a device, and more particularly to a method for enabling a device to be self-aware, and independently initiating connection with host servers for synchronization.

Description of Related Arts

Traditionally, networks operate on the principle that servers discover the properties of devices connected to them via scans or polls of the networks. Such scans or polls are typically server-initiated or determined by a central network controller operating within the network's scope of operations. This can result in unnecessary constant network loads and in certain cases, even fail to detect the presence of some of the devices.

Today, a typical user may own a plurality of devices which are interconnected through a network, for example, a local area network (LAN) or wireless LAN or via telecommunications networks such as GPRS and 3G. It can be difficult for the user to keep track of all data and applications on these devices especially when each of the devices has its own particular operating system, device functionalities and capabilities.

US Patent Application Publication No. 20060190569 A1 disclosed a method for facilitating mobile device awareness of new or updated server side applications, in response to a new application or an updated version of an application made available to the server. A message is transmitted over a wireless connection to the devices indicating the availability of the new applications or updated applications. Notification is displayed on each of the devices and the user may choose to register for the new or updated applications. This method relies on a system

administrator at a middleware server to trigger the notification message to the mobile devices. Additional server is required and there may be failure to detect the presence of some devices.

US Patent No. 8019863 B2 disclosed a method for synchronizing events between mobile devices and servers. The events to be synchronized between the mobile devices and servers are detected by deducing activities that occurred, followed by reading data records, and creating packets of operation for recreating the activities on the target, wherein the target is either of the mobile device or servers. Detection of the activities is conducted by a listener which polls the server and the device. Similarly, this method requires scanning or polling to be performed on the servers and the mobile devices which require additional resources.

With the rapid proliferation of mobile devices, it can be seen in the prior arts there exists a need to provide a method that permits devices to autonomously initiate self-check of their properties and establish connection for synchronization with the servers.

Summary of Invention

It is an objective of the present invention to provide a method for enabling devices to be autonomously self-aware in relation to content of host servers.

It is also an objective of the present invention to provide a method for enabling devices to independently present or update device capabilities to host servers and receive appropriate responses from the host servers that are connected.

It is yet another objective of the present invention to provide a method for real-time synchronization between a device and host servers.

It is a further objective of the present invention to provide a method for enabling the device properties and status to a host server for synchronizing applications, content, and data, according to the device's unique capabilities.

Accordingly, these objectives may be achieved by following the teachings of the present invention. The present invention relates to a method for real-time synchronization between a device and host servers, characterized by: providing an embedded program to the device to enable self discovery of the device; executing a decision controller process of the embedded program on the device to detect a tag identity previously generated and embedded in the device, by the host servers; in absence of the tag identity, scanning the device to obtain a device profile, allocating a tag identity to the device profile, and storing the tag identity in the device; performing synchronization between the device and the host servers according to the tag identity.

Brief Description of the Drawings

The features of the invention will be more readily understood and appreciated from the following detailed description when read in conjunction with the accompanying drawings of the preferred embodiment of the present invention, in which:

Fig. 1 is a block diagram illustrating an example in which a method of synchronization between devices and hot servers is implemented.

Detailed Description of the Invention

As required, detailed embodiments of the present invention are disclosed herein; however, it is to be understood that the disclosed embodiments are merely exemplary of the invention, which may be embodied in various forms. Therefore, specific structural and functional details disclosed herein are not to be interpreted as limiting but merely as a basis for claims. It should be understood that the drawings and detailed description thereto are not intended to limit the invention to the particular form disclosed, but on the contrary, the invention is to cover all modification, equivalents and alternatives falling within the scope of the present invention as defined by the appended claims. As used throughout this application, the word "may" is used in a permissive sense (i.e., meaning having the potential to), rather than the mandatory sense (i.e., meaning must). Similarly, the words

"include," "including," and "includes" mean including, but not limited to. Further, the words "a" or "an" mean "at least one" and the word "plurality" means one or more, unless otherwise mentioned. Where the abbreviations or technical terms are used, these indicate the commonly accepted meanings as known in the technical field. The present invention will now be described with reference to Fig. 1.

The present invention relates to a method for real-time synchronization between a device and host servers, characterized by:

providing an embedded program to the device to enable self discovery of the device;

executing a decision controller process of the embedded program on the device to detect a tag identity previously generated and embedded in the device, by the host servers;

in absence of the tag identity, scanning the device to obtain a device profile, allocating a tag identity to the device profile, and storing the tag identity in the device;

performing synchronization between the device and the host servers according to the tag identity.

In a preferred embodiment of the method for real-time synchronization between a device and host servers, the decision controller process is handled by critical binary functions.

In a preferred embodiment of the method for real-time synchronization between a device and host servers, the device profile comprises the device physical and logical properties.

In a preferred embodiment of the method for real-time synchronization between a device and host servers, the tag identity allocated to a device profile is associated to a user's profile.

In a preferred embodiment of the method for real-time synchronization between a device and host servers, the synchronization is initiated autonomously or upon request by a user.

In a preferred embodiment of the method for real-time synchronization between a device and host servers, the synchronization is performed upon confirmation by the host servers the permissible authorizations granted to the device.

The present invention comprises an embedded program on the device which enables self discovery of the device. In a preferred embodiment, the embedded program comprises proprietary software codes for operations on the device and the servers. In a preferred embodiment, the program can be implemented in either software or hardware in one or more devices, which are capable of fetching computing instructions found therein. For example, the program is embedded in a silicon chip attached on the circuit board of the device. Alternatively, the program may be a pre-compiled software which may be downloaded and installed as part of the manufacturer's firmware upgrade. In a preferred embodiment, the embedded program initializes itself during the standard bootstrap process when the device is switched on.

The decision controller process is executed to detect a tag identity previously generated and embedded in the device. In a preferred embodiment, the tag identity is embedded in a predetermined location such that the program would be able to locate the tag identity. For example, if the tag identity is embedded in a silicon chip, the program would be written to locate the tag identity at where the chip is located at the circuit board of the device. If the tag identity is embedded in software form, the specific file storing the tag identity would be search upon by the program. In a preferred embodiment, the decision controller process is handled by critical binary functions, thereby increasing the speed of the decision controller process.

If the tag identity is not located, the decision controller process will proceed to scan

the device for discovering the device environment, and thus to obtain the device profile from the device. In a preferred embodiment, the device profile comprises physical and logical properties of said device. The device profile includes details of the device including but not limited to the device model, operating system version, and other pertinent information for reference to the compatibility and availability to various applications and applications data. With the device profile obtained, the host server allocates a tag identity to the device profile, which is then stored on a predetermined location in the device. In a preferred embodiment, the user may have multiple devices which are each allocated with a unique tag identity, but are associated under one user's profile to ensure data integrity, privacy of content, and device separation.

Once the tag identity has been identified, the device autonomously initiates an independent remote connection request to a host server through the process communication system. The host server confirms the permissible authorizations granted to the device, establishes the connection request and synchronizes the device with appropriate device-specific applications, content and data, according to the tag identity. For example, if a new updated version of mobile application has been made available at the host server, real-time synchronization allows the compatible device to be informed and have the application installed autonomously. In another example, if the user changes to another similar or dissimilar mobile device, the tag identity associated with the user profile allows the exact operational environment to be reinstated according to the data stored in the host server under the user profile, said operational environment includes application settings such as instant messaging and email setting. Additionally, if the new device has better capabilities than the previous, for example, a change in the operating system of the device, the embedded program recognizes these capabilities in real-time and automatically scales to enable new additional features to enhance the user's experience on the new mobile device. In a preferred embodiment, the synchronization is initiated autonomously or requested by a user. In a preferred embodiment, the synchronization can be independently initiated from the device to host servers or vice versa.

With reference to Fig .1, the network layer comprises a combination of wired or wireless network backbone to which a plurality of devices are connected and autonomously synchronized with a plurality of host servers. Each host server provides a plurality of specific services and products including but not limited to provision of applications, content, and data. Each user's plurality of devices is uniquely tagged with the tag identity under each user profile. The particular type of operating system running on each device and the host servers is not significant to the proper operation of the present invention.

Although the present invention has been described with reference to specific embodiments, also shown in the appended figures, it will be apparent for those skilled in the art that many variations and modifications can be done within the scope of the invention as described in the specification and defined in the following claims.

Claims

1. A method for real-time synchronization between a device and host servers, characterized by:

providing an embedded program to the device to enable self discovery of the device;

executing a decision controller process of the embedded program on the device to detect a tag identity previously generated and embedded in the device, by the host servers;

in absence of the tag identity, scanning the device to obtain a device profile, allocating a tag identity to the device profile, and storing the tag identity in the device;

performing synchronization between the device and the host servers according to the tag identity.

2. A method for real-time synchronization between a device and host servers according to claim 1, wherein the decision controller process is handled by critical binary functions.

3. A method for real-time synchronization between a device and host servers according to claim 1, wherein the device profile comprises the device physical and logical properties.

4. A method for real-time synchronization between a device and host servers according to claim 1, wherein the tag identity allocated to a device profile is associated to a user's profile.

5. A method for real-time synchronization between a device and host servers according to claim 1, wherein the synchronization is initiated autonomously or upon request by a user.

6. A method for real-time synchronization between a device and host servers according to claim 1, wherein the synchronization is performed upon confirmation by the host servers the permissible authorizations granted to the device.