(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
30 December 2009 (30.12.2009)

PCT

(10) International Publication Number
WO 2009/158335 A1

(51) International Patent Classification:
H04L 27/04 (2006.01)

(21) International Application Number:
PCT/US2009/048254

(22) International Filing Date:
23 June 2009 (23.06.2009)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
12/147,770 27 June 2008 (27.06.2008) US

(71) Applicant *(for all designated States except US)*: **UNIVERSAL ELECTRONICS INC.** [US/US]; 6101 Gateway Drive, Cypress, CA 90630-4841 (US).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: **BLACK, Jeremy** [US/US]; 13132 Lassen Way, Santa Ana, CA 92705 (US). **HAYES, Patrick, H.** [US/US]; 22981 Arija, Mission Viejo, CA 92691 (US).

(74) Agents: **JAROSIK, Gary, R.** et al.; Greenberg Traurig, LLP, Suite 3100, 77 W. Wacker Drive, Chicago, IL 60601-1732 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

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

(54) Title: SYSTEM AND METHOD FOR UBIQUITOUS APPLIANCE CONTROL

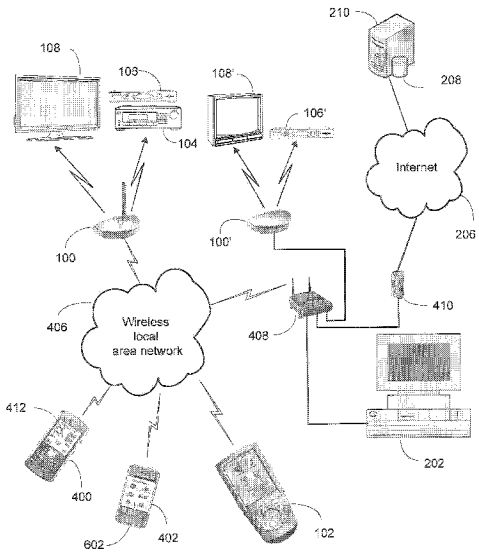


Figure 4a

(57) Abstract: A slave relay station is adapted to serve and/or host pages comprising a simplified graphic user interface (GUI) encoded in a widely recognized format such as, for example, HTML or WML. The GUI embodies activatable links corresponding to control functions for configured appliances. A wireless phone or other device with network access and the capability to process and present such pages, for example via a Web browser, may then be utilized to effect control of such appliances by simply navigating to the network address of the slave relay station, obtaining an appropriate GUI page, and interacting with the links.

WO 2009/158335 A1

SYSTEM AND METHOD FOR UBIQUITOUS APPLIANCE CONTROL.

BACKGROUND

Portable controlling devices, such as for example universal remote controls, and the features and functions offered by such devices are well known in the art. Increasingly sophisticated implementations of these devices incorporate technologies such as color touch screens, wireless home network compatibility, user configurable graphical user interfaces, slave relay stations positioned to control appliances not situated in line of sight of the controlling device, etc. Contemporaneously, personal communication, productivity, and entertainment devices such as cellular phones, portable email devices, hand-held games, etc. have begun to offer features such as graphical user interfaces on color touch screens, wireless Internet capability, etc.

SUMMARY OF THE INVENTION

This invention relates generally to systems and methods for adapting various appliance control capabilities of a universal remote control system such that they may be ubiquitously accessed by personal communication devices within a wireless network. Specifically, one or more network-capable slave relay stations installed in conjunction with a universal remote control device may additionally be adapted to serve and/or host pages comprising a simplified graphic user interface (GUI) encoded in a widely recognized format such as for example HTML or WML, which GUI embodies activatable links corresponding to control functions for configured appliances. A wireless phone or other device with network access and the capability to process and present such pages, for example via a Web browser, may then be utilized to effect control of such appliances by simply navigating to the network address of that slave relay station. Such devices may include without limitation cellular phones, smartphones, personal productivity devices, personal gaming, audio, or video players, game controllers, PDAs, etc., all collectively referred to hereafter as personal communication devices.

In some embodiments, the GUI pages to be served by slave relay stations may be created using the same editor as provided for use in creating or modifying the universal controlling device graphical user interface. Further, in certain embodiments the GUI pages to be served may be dynamically selected, scaled, or otherwise modified by a local

or remote service based upon the known or inferred capabilities of the requesting personal communication device.

A better understanding of the objects, advantages, features, properties, and relationships of the invention will be obtained from the following detailed description and accompanying drawings which set forth illustrative embodiments and which are indicative of the various ways in which the principles of the invention may be employed.

BRIEF DESCRIPTION OF THE DRAWINGS

For a better understanding of the various aspects of the invention, reference may be had to preferred embodiments shown in the attached drawings in which:

Figure 1 illustrates an exemplary prior art universal controlling device and system;

Figure 2 illustrates an exemplary prior art system for creating graphical user interface pages for the controlling device of Figure 1;

Figure 3 illustrates a block diagram of exemplary components of an exemplary slave relay device;

Figures 4a and 4b illustrate exemplary systems in which exemplary personal communication devices may be used as controlling devices in accordance with the teachings of the instant invention;

Figure 5 illustrates in flow chart form exemplary actions of an exemplary slave relay device when serving GUI pages to a personal communication device;

Figure 6 illustrates an exemplary personal communication device functioning as a controlling device;

Figure 7 illustrates a second exemplary personal communication device functioning as a controlling device;

Figure 8 illustrates in flow chart form an exemplary method for adapting or selecting GUI pages to match the capabilities of exemplary personal communication devices;

Figure 9 illustrates an exemplary editing program which may be used to create GUI pages for use with personal communication devices;

Figure 10a illustrates an exemplary set of data files which may define a GUI page from which an appliance may be controlled;

Figure 10b illustrates an exemplary tag file comprising XML which may be executed by a slave relay device to effect control of an appliance; and

Figure 11 illustrates in flow chart form another exemplary method for adapting or selecting GUI pages to match the capabilities of an exemplary personal communication device.

DETAILED DESCRIPTION

With reference to Figures 1 and 2, which are representative of prior art, it is known to provide a universal controlling device 102 comprising a touch screen graphical user interface 120 through which various appliances, for example a TV 108, cable set top box (STB) 104 and/or AV receiver 104 may be controlled. For further detail regarding this type of controlling device, reference may be made to U.S. Patents 7,143,214, entitled "Hand Held Device Having a Browser Application," or 7,266,777, entitled "Configurable Controlling Device Having an Associated Editing Program," both of like assignee and incorporated herein by reference in their entirety.

In order to facilitate control of infrared (IR) signal responsive appliances which are not positioned in line of sight of controlling device 102, it is also known to provide a slave relay station 100 which receives RF communications 110 from controlling device 102 and outputs IR signals 112 to the various controlled devices. In the simplest form, said slave relay station may consist of nothing more than an analog RF demodulator and IR remodulator such as described, for example, in U.S. Patent 5,142,397, entitled "System for Extending the Effective Operational Range of an Infrared Remote Control System." However, in the more general case where the RF communication path 110 utilizes a standardized protocol such as for example WiFi, Bluetooth, Zwave, Zigbee, etc., the slave relay device 100 may be required to receive and decode incoming messages in one format and translate these into IR commands (or even sequences of commands) having one or more different format(s) recognizable by the intended target device(s) for said commands. In this regard, see for example the "Nevo Link" brochure NSL007-2 published by Universal Electronics Inc. which is incorporated herein by reference in its entirety. To this end, the slave relay devices contemplated by the teachings of this invention incorporate processing capabilities, as will be described in greater detail in conjunction with Figure 3.

As illustrated in Figure 2, it is also known to provide application software comprising an editor 200 for execution on a PC 202 or other computer which may be used to configure and create individualized graphical user interfaces and command libraries for downloading into a controlling device 102. Such editor software may also be used to configure/assign slave relay device(s) 100 attached to a local area network and accessible

to controlling device 102. The editor software may make use of local 204 and/or remote 208 database facilities for the retrieval of appliance command data, interface graphics, etc., as well as for storage of edited configurations. Remote database 208 may be hosted on a server 210 accessible via, for example, the Internet or a similar wide area network 206. For further detail regarding controlling device editor applications, the reader may refer to, for example, the aforementioned U.S. Patent No. 7,266,777, U.S. Patent No. 6,211,870, entitled "Computer Programmable Remote Control," and pending U.S. Patent Application No. 11/357,681, entitled "Configurable Controlling Device and Associated Configuration Distribution System and Method," all of like assignee and incorporated herein by reference in their entirety, as well as U.S. Patent No. 6,937,972, entitled "Fully Functional Remote Control Editor and Emulator," also incorporated herein by reference in its entirety.

Turning now to Figure 3, the architecture of an exemplary slave relay station is illustrated in block diagram form. For use in commanding the functional operations of one or more appliances in response to messages received via a wired or wireless network connection, a slave relay station 100 may include, as needed for a particular application, a processor 300 coupled to a ROM memory 304, a RAM memory 302, a non-volatile read/write memory 306, circuit(s) 308 for transmission of operating commands to appliances (e.g., IR and/or RF), a wireless network transceiver 310 (e.g. WiFi, Bluetooth, etc.) and/or a wired network interface 312 (e.g. Ethernet) for communication with a local network, a means 314 to provide feedback to the user (e.g., one or more LEDs, LCD display, speaker, and/or the like), and a power source 316.

As will be understood by those skilled in the art, some or all of the memories 302, 304, 306 may include executable instructions (collectively, the program memory) that are intended to be executed by the processor 300 to control the operation of the slave relay station 100, as well as data that serves to define appliance control protocols and command values to the operational software (the appliance code data). In this manner, the processor 200 may be programmed to control the various electronic components within the slave relay station 100 and process the input and output data thereof, for example, to receive and transmit data via network interfaces 308 and/or 310, to act upon commands and requests embodied in such data, to cause the transmission of appliance command signals via transmission circuits(s) 308 to appliances to be controlled, to control visual feedback device(s) 314, etc. In a contemplated embodiment, the non-volatile read/write memory 306, for example an EEPROM, battery-backed up RAM, FLASH, Smart Card, memory

stick, or the like, may additionally be used to store HTML page data for serving to requesting devices, as described in greater detail hereafter. While the memory 304 is illustrated and described as a ROM memory, memory 304 may also be comprised of any type of readable media, such as ROM, FLASH, EEPROM, or the like. Preferably, the memories 304 and 306 are non-volatile or battery-backed such that data is not required to be reloaded after power interruptions. In addition, the memories 302, 304, and 306 may take the form of a chip, a hard disk, a magnetic disk, an optical disk, and/or the like. Still further, it will be appreciated that some or all of the illustrated memory devices may be physically incorporated within the same IC chip as the microprocessor 300 (a so called "microcontroller") and, as such, they are shown separately in Fig. 3 only for the sake of clarity.

To cause the slave relay device 100 to perform an action, slave relay device 100 is adapted to be responsive to events, such as a received signal from network interface port 310 or 312. In response to an event, appropriate instructions within the program memory (hereafter the "operating program") may be executed. For example, receipt of a command message from controlling device 102 may result in the retrieval from the appliance code data the command value and control protocol appropriate for an intended target device and a resulting transmission of the requested command to the intended target appliance, e.g., the STB 106, in a format recognizable by the intended target appliance. Additionally, in keeping with the teachings of the instant invention, receipt of, for example, an HTTP page request from a browser capable, client, personal communication device may result in the retrieval of HTML formatted data and serving of a page comprised of the HTML formatted data back to the requesting client, as will be described in greater detail hereafter.

For selecting a set of appliance code data to be associated with an appliance to be controlled, data may be provided to the slave relay device 100 that serves to identify an intended target appliance by its type and make (and sometimes model). Such data may allow the slave relay device 100 to identify the appropriate appliance code data elements within a preprogrammed library of appliance code data, to be used to transmit recognizable commands in a format appropriate for such identified appliances. Alternatively, either in place of or in addition to a pre-stored library, appliance code data may be downloaded into slave relay device 100 via a network interface(s) 310, 312 either during an initialization phase or on an as required basis.

Referring now to Figure 4a, in an exemplary embodiment a wireless local area network 406, for example a WiFi network, provides a means for communication between slave relay devices 100 and 100' and a universal controlling device 102. Slave relay devices 100 and 100' provide a conduit for controlling device 102 to issue operational commands to appliances 108, 106, 104 and 108', 106' respectively. Various embodiments of such slave relay devices may interface directly to wireless network 406 (e.g., via a built-in wireless network transceiver 310) as illustrated by device 100, or may interface to the network via a hardwired connection to a router 408 (e.g., via a built-in wired network interface 312) as illustrated by device 100'. In either case, wireless commands originating from controlling device 102 are converted into appliance control commands of a format suitable for use with the target appliance(s), e.g. 104, 106 or 108, as previously described in conjunction with Figures 1 and 3. A local personal computer 202 or other device having Internet connectivity and an Internet connection 410 (e.g., a cable modem) are also connected to router 408 and thus accessible via the wireless local network 406.

In an exemplary embodiment, in addition to servicing requests from universal controlling device 102 as is known in the art, the illustrated slave relay devices 100 and 100' of the instant invention are also capable of serving HTML-formatted pages over local area network 406 as requested by browser-capable devices such as personal communications devices 400 or 402, thereby allowing such devices to be used as surrogate or additional universal controlling devices, as will be described hereafter in greater detail.

As will be appreciated by those of skill in the art, not all personal communication devices may be equipped to communicate directly with a local network (e.g., have a WiFi capability). As illustrated in Figure 4b, certain exemplary personal communication devices, e.g., cell phone 422, may be adapted to communicate only via a closed wide area network 424, 426 (e.g., a cellular phone network) to a contracted service provider 420. Such a personal communication device may however offer access to the Internet 206 via a service provider gateway 428. In such cases, it will be appreciated that the exemplary personal communication device may request HTML formatted GUI pages from slave relay station 100 via Internet 206 and local router 408. Accordingly in the discussions that follow, it is to be understood that the services described are not intended to be limited to only those devices which are equipped to directly access local network 406.

Turning now to Figure 5, there is illustrated in flowchart form an exemplary series of steps which may be performed by the operating program of a slave relay device upon

receipt of a communication from the local network. While for the purpose of this illustrative example certain terminology common to a WiFi-based local network and HTTP transaction protocol may be used for clarity, this is not intended to be limiting as it will be appreciated that the concepts presented may be practiced equally effectively using other appropriate network architectures and/or protocols.

Upon initial receipt of a network communication, at step 502 the operating program of the slave relay station may first determine if the communication comprises a function command from an associated universal controlling device (e.g., controller 102). If so, at steps 514, 516 and 518 the operating software may retrieve the requested command function from the appliance code data previously associated with the desired appliance, transmit that command to the appliance in the appropriate format, and then issue a completion confirmation response to the initiating device, all as is known in the prior art.

If the received communication is not a function command from a universal controlling device, the operating software may next determine if the communication is a request from a user agent, e.g., a browser application resident in a networked device such as personal communication devices 400, 402, or 422 (each alternatively referred to hereafter as the "client device"). If not, at step 520 an error message, e.g., an HTTP "Error 404 – Not Found" response is issued to the originating network device. Alternatively the operating software may be configured to issue a simplified default interface page in the event that the unknown client device is able to render rudimentary HTML or other markup language pages. If the communication is a recognized user agent request, the operating software at step 506 then determines if this is GUI page request, e.g., a request for an HTML file. If so, at step 530 the operating software resolves the request file path and determines the GUI page to be served in response to the user agent request. (If any error is encountered, e.g., the file path/name does not exist, an error response akin to step 520 may be issued to the requesting device. For the sake of clarity, this and other similar error conditions are not exhaustively illustrated in the flowchart of Figure 5.) Once the GUI page to be served has been identified, the operating software at step 534 may next optionally determine the requesting client device type and capabilities. This may include, for example, examining various fields in the received request (for example, "accept" and "user agent" data in an HTTP request header) or other pre-defined identification data, searching a local or remote cross-reference table of IP or MAC addresses to device types, or even interacting with the requesting device itself to obtain

the identification data. Alternatively, a fixed set of device capabilities for the request issuing device may have been predetermined at the time the slave relay device was configured. Once the client device capabilities are known or inferred, at step 538 the page data corresponding to the identified GUI page may be scaled or reformatted, or an alternate GUI page selected, based on the capabilities of the client device. This process will be described in greater detail later. Finally, at step 540 the requesting client device is responded to with the GUI page data.

Turning momentarily to Figure 6, by way of example only, such a GUI page may comprise HTML data 600 as illustrated. When rendered on a personal communication device 402, such an HTML page may appear as illustrated at 602. In particular such a GUI page display may comprise a series of icons representative of appliance control actions which may be initiated from the displaying client device. When one of the displayed icons or links is selected (it will be appreciated that such selection may be by touch, by cursor movement, by operation of navigation keys, etc., as appropriate to the particular client device currently rendering the GUI page) this may result in a message being transmitted back to the originating slave relay device, receipt of which causes the desired command(s) to be issued to an appliance. By way of more detailed example, the GUI page display 602 of the embodiment illustrated in Figure 6 comprises six channel selection icons, four of which are configured to cause a tunable appliance (e.g., set top box 106) to be set to particular channel, while the remaining two are configured to issue generic "channel up" and "channel down" commands to the appliance. To this end, the HTML data comprising GUI page 600 may include six HTML "tags" such as for example tag 604 corresponding to the broadcast TV channel "ABC" or tag 606 corresponding to the "channel up" function. In this implementation, each tag comprises a definition of an icon to be displayed, e.g., "abc_button.png" (in this example, a graphic file to be retrieved from the originating server), an action to be taken if and when the icon is selected (in this example, an HTTP request back to the originating server for a filename such as for example "switch_to_abc.irm" or "channel_up.irc"), and size and formatting information for positioning the displayed icon.

Returning now to Figure 5, if the operating program of the slave relay device has determined that a received communication is not a function command from an associated universal controlling device and is not a GUI page request, it next determines at step 510 if the received message is a request for an individual appliance command function (e.g., "Channel up", "Mute" or "Pause"). This may be determined, for example, by the type or

format of the request. By way of further specific example, in the HTML example described above in conjunction with Figure 6, the file extension ".irc" may have special significance to the slave relay device operating software, i.e., receipt of a request for "channel_up.irc" may cause the operating software to retrieve from the stored appliance code data the appropriate command value and control protocol and use these to transmit the requested command to the designated appliance as illustrated at steps 524 and 526. The manner in which an association is formed between a function request and the appropriate appliance command will be discussed in greater detail later. Thereafter, at step 528 the operating software may determine the next GUI page to be presented, if any, and serve this back to the requesting client device as previously described.

If the received communication is not a request for an individual appliance command function, the operating software next checks if the request is for a sequence of commands (e.g., the digits "0", "0", "7" to tune to channel seven). Once again, this may be determined, for example, by the type or format of the request. If the request is for a command sequence, the desired series of commands is determined at step 522 and then transmitted to the specified appliance(s) in similar manner to that described previously. By way of further specific example, in the HTML example described above in conjunction with figure 6, the file extension ".irm" may have special significance to the slave relay device operating software, i.e. receipt of a request for "switch-to-abc.irm" may cause the operating software to retrieve a stored series of commands operable to change a pre-determined channel tuning device to the ABC broadcast channel and transmit these to that appliance. The manner in which an association is formed between a command sequence request and a series of appliance command will be discussed in greater detail later. Upon completion of the sequence transmission, the operating software may determine the next GUI page to be presented, if any, and serve this back to the requesting client device as previously described. By way of example, on completion of a channel changing sequence, it may be desirable to automatically switch to a GUI page from which television volume may be controlled, e.g., the page 412 illustrated in Figure 4a.

Finally, at step 542 the operating program of the slave relay device may check for other valid requests (for example, requests for icon graphics as mentioned above in conjunction with the HTML data of Figure 6). If valid, other requests may be serviced as appropriate at step 544, else an error message may be returned to the requesting client device.

As mentioned above in connection with step 538 of Figure 5, GUI pages to be served may be scaled, reformatted, or selected to match the capabilities of the requesting client device. By way of specific example, again using HTTP as an exemplary transfer protocol, the HTTP request header issued by client device 402 illustrated in Figures 4a and 6 may include a user agent string such as:

User-Agent = [Mozilla/5.0 (iPhone; u; CPU like Mac OS X; en) AppleWebKit/420.1 (KHTML, like Gecko) Version/3.0 Mobile/4A102 Safari/419.3]

while the HTTP request header issued by client device 422 illustrated in Figures 4b and 7 and may include a user agent string such as:

User-Agent = [Nokia6500s-1/2.0 (04.80) Profile/MIDP-2.1 Configuration/CLDC-1.1]

Upon receipt of an HTTP "GET" command requesting a GUI page, the operating program of a slave relay device may examine the request header contents (or forward them to another system for examination) to determine an appropriate format for the GUI page to be served. Continuing the specific example, an examination of the HTML defining the "ABC" tags illustrated at 604 and 704 will reveal that the size of the button icon (i.e., the "width" and "height" parameters) is adjusted to better match the screen resolution, screen size, orientation capabilities, and rendering engines of the respective client devices 402 (e.g., an Apple iPhone) and 422 (e.g. a Nokia6500s).

Turning now to Figure 8, by way of further example a method by which GUI pages may be modified or selected to match specific requesting client devices, corresponding to steps 534 and 538 of Figure 5, is illustrated in greater detail once again using HTTP as an exemplary transfer protocol, without limitation. Having determined that a particular GUI page (for example, an HTML file) is to be served to a requesting client device, the operating software of the exemplary slave relay device 100 may, at step 802, determine if GUI page scaling/selection is enabled. If not, at step 820 a fixed GUI page is returned and no further page formatting action is taken. This may occur, for example, when an author of a group of universal remote control GUI pages has decided to support only a lowest common denominator set of client device capabilities, when a specific GUI page comprises only a single function such as a status message or on/off button, etc. In various embodiments such pages may be stored in memory 302,304,306 of slave relay device 100, may be retrieved from a local 202 or remote 210 server by the slave relay device 100, or a combination thereof, as appropriate. In addition, it will be appreciated that such pages may be stored in a local 202 or remote 210 server and merely caused by the slave relay device 100 to be forwarded to the client device via an

appropriate communications channel. Still further, the local 202 or remote 210 server may forward such pages to the client device via an appropriate communications channel upon seeing a page request being issued from the client device without requiring the slave relay device 100 itself to initiate any operations associated with such page fulfillment requests.

If however GUI page scaling/selection is enabled, a device adaptation service 800 may be invoked. At step 804 the user agent string may be retrieved from the client device's HTTP request header, which as previously illustrated may contain information which serves to identify the requesting client device. If the user agent string is recognized at step 806, device capability information corresponding to the requesting client device is retrieved at step 814. By way of example only, in one contemplated embodiment client devices may be categorized into several classes based on maximum supported horizontal screen resolution, e.g., less than 105 pixels, 106 to 175 pixels, 176 to 239 pixels, 240 to 319 pixels, 320 to 639 pixels, etc., etc. Once a client device has been identified, its class may be determined via a look-up table and an appropriate HTML file selected by the device adaptation service at step 818 from a library of pre-formatted versions of each GUI page, one for each device class. Alternatively, in another exemplary embodiment, a single master HTML file may be created for each GUI page based on a default resolution, and then scaled by the device adaptation service at step 818 to match the exact resolution of the target client device, once again determined from a look-up table. In this regard, it should be noted that certain client device browser implementations may be adapted to compress or shrink received graphic pages in order to emulate a browser screen of greater resolution than that of the underlying personal communication device hardware. In such instances, it will be appreciated that the parameters used by the device adaptation service in creating or scaling GUI pages should match the emulated, and not the actual, resolution of the target client device. Further, in various embodiments, other client device capabilities, e.g., color versus monochrome, touch screen selection of icons versus navigation keys, etc., may also be used to additionally refine GUI pages to match a specific target client device.

In the event a user agent string is not immediately recognized, e.g. by being found in a look-up table as described above, at step 808 a determination is made by the device adaptation service whether additional analysis is possible. When possible, at step 810 such analysis may include forwarding the user agent string data to additional search services for further processing, inspection of additional fields in the HTTP request header

(e.g., a URL pointing to profile data for the client device), direct interaction with the requesting client device, etc. By way of specific example, the HTTP request header issued by client device 422 illustrated in Figures 4b and 7 may include a reference to a device profile, e.g.:

x-wap-profile = ["http://nds1.nds.nokia.com/uaprof/N6500sr100.xml"]

which profile data, when retrieved from the indicated URL, may include for example statements such as:

<prf:ColorCapable>Yes</prf:ColorCapable>

...

<prf:ImageCapable>Yes</prf:ImageCapable>

...

<prf:ScreenSize>240x320</prf:ScreenSize>

which are indicative of the capabilities of that client device.

As illustrated at step 812, if the client device is successfully identified via these further measures, the device adaptation service proceeds with retrieval of client device capability information at step 814, else at step 816 the client device capabilities are set to a default value, for example the most common capability set, the set of capabilities used in creating a master HTML file, etc.

As will be appreciated, the steps of the exemplary method described above in conjunction with Figure 8 may be performed locally by the operating software of the slave relay device, or by a service hosted on a local (e.g. 202) or remote (e.g. 210) computer system and available to the operating software via network(s) 406, 206, or by any combination thereof, as appropriate to a particular embodiment. For example, the service performed by the device adaptation service may be performed at the slave relay station itself, may be performed at another device in communication with the slave relay station with a modified HTML page being returned to the slave relay station for transfer, and/or may be performed at a device in the communication channel between the slave relay station and the requesting client device whereupon a HTML page received or already stored by such device is modified prior to its being forwarded on to the requesting client device. Accordingly the above descriptions are not intended to be limiting as to location of the various steps of the device adaptation services described.

In order to create fully functional GUI page definitions for use in the above described systems, a means may be provided to generate graphic pages with activatable icons suitable for rendering by target client devices (e.g., HTML files) as well as means to generate an association between each activatable icon and a desired control action on the

part of the host slave relay device. While these activities may be performed as separate steps using separate tools, for example, any convenient HTML editor to generate loadable graphic pages with embedded tags and any convenient text editor to create slave relay device-recognizable XML files defining the actions to be taken for each tag, in certain embodiments a single software tool may be made available to perform both functions, thereby offering greater ease of use and consistency of output. Advantageously, this may comprise an extension to an editor provided for creation and/or modification of the graphic user interface of a universal controlling device (e.g. controlling device 102) used in conjunction with the slave relay device (e.g. device 100), examples of which may be found described in the previously referenced and incorporated U.S. Patents Nos. 7,266,777, 6,211,870, 6,937,972 and pending U.S. Patent Application No. 11/357,681.

Turning now to Figure 9, there is illustrated an exemplary editor adapted to support the creation of GUI files for use with a slave relay device as described above. Such an editor application may be resident on a local PC (e.g., PC 202) or may be located remotely and used to create edited GUI pages for later transfer to the local systems. In either case, the editor screen 900 comprises a layout area 902 in which a WYSIWYG image of the GUI page under creation may be displayed, together with a gallery of icons 904 from which images be dragged and placed into layout area 902. After placement into the layout, an icon, for example 906, may be selected and the action or series of actions to be performed upon activation of that icon may be defined in an actions panel 908. Such actions may, for example include transmission of appliance commands 910, the specification of such commands being performed by dragging specific actions from a displayed device function list 914 into action panel 908; or GUI page switching actions 912, specified by dragging pointers to other GUI pages, e.g. 918, from a page tree 916 into action panel 908. In the exemplary embodiment, once the layout is completed and all actions specified to the satisfaction of the user, the edited GUI may be saved as a group of files suitable for use by a slave relay device acting as a server to a requesting client device. It will be appreciated that in accordance with the needs of the device adaptation service(s) of the particular embodiment, the GUI definition may be saved as a single master version suitable for scaling, as a set of several selectable file groups, one for each class of client device to be supported, etc.

By way of example only, Figure 10a illustrates such a group of files 1000. As illustrated, this group may comprise an HTML file 600 to be served to the requesting client device together with the resources to support that page, for example, graphics files

1010 representing icons to be displayed and tag files 1012. Tag files 1012 may specify the actions to be performed by a slave relay device when an HTML tag is activated on the client device. In an illustrative embodiment tag files may comprise a series of XML statements to be executed by the operating software of a slave relay device, e.g., device 100. By way of further example only, Figure 10b illustrates an exemplary XML encoding of the tag file "switch_to_abc.irm," corresponding to the editor action panel 908 of Figure 9 and the HTML statement 604 of Figure 6. The illustrative tag file content includes a definition 1020 of the appliance to which commands are to be directed, a list 1022 of the functions to be transmitted, and a specification 1024 of the GUI page to be served the client device upon completion of the command transmission. In summary, when the exemplary slave relay device receives an HTTP "GET" request for "switch_to_abc.irm" from a client, this will result in appliance type "C1376" (e.g., a Motorola cable STB) being commanded to tune to channel "007" (e.g., by transmitting the digit commands "0", "0" and "7" in sequence) after which the GUI page "tv_controls_html" will be retrieved and served to the requesting client device (with appropriate scaling and/or reformatting if this is enabled).

The data files which comprise the GUI pages to be supported by the slave relay device may be downloaded directly to it for storage in memory (e.g., memories 302, 306), or may be stored on a local (e.g., computer 202) or remote (e.g., computer 210) server system and accessed on an as needed basis by slave relay devices (e.g. 100) for transfer to the client device. In the latter example, the data transfer may take place by having the requested data downloaded to and served from the slave relay station to the client device and/or by having the requested data served directed from the storage source to the client device via the network . Further, it will be appreciated that in some embodiments these data may reside in multiple locations, for example HTML files (e.g., HTML page 600) may be stored locally on a slave relay device while resources such graphics files (e.g., button images 1010) may be stored on a network-accessible server (e.g., computers 202 or 210) and retrieved on an as needed basis by the slave relay device, or even directly by a client device.

In certain embodiments, it may also be desirable to allow a user to configure appliance control functionality directly on a personal communication device without necessitating the use of a separate editor application as described above. In this regard, it will be appreciated that an HTML or other markup language generating application or "engine" (such as ActiveX software technology developed by Microsoft Corporation, or

that described in US Patent No. 7,216,298 assigned to Oracle Corporation, which is hereby incorporated by reference herein in its entirety) may be stored on or made available to the slave relay device, or stored on a network-accessible server (e.g., computers 202 or 210) and used or retrieved on an as needed basis by the slave relay device, or even directly by a client device. In this way a user may be presented with a generic interface for purposes of setting up her client device to control the desired appliance(s) upon an initial interaction with the engine (e.g., browsing to the IP or host URL of the slave relay device). As described herein, the engine may be configured to read and analyze user agent string information from the client device in order to present a generic interface appropriate for display on the particular requesting client device, and/or to customize various features and functions presented to the user during a setup process or subsequent control based operations. During the setup process a user may interact with the engine to set up individual appliances (e.g., devices 104, 106, 108, etc) using device type and model number searching, direct code entry from a library of available codes, or a variety of other known universal remote control setup techniques as may be configured in the engine. For each appliance set up via the engine a default interface may be automatically generated and presented to the user for immediate testing and/or operation of appliance commands via the slave relay device. After set up of individual devices a user may save the whole configuration for later customization either on the slave relay device, on a network-accessible server (e.g., computers 202 or 210) or even directly on the client device. It will be appreciated that such configuration actions may be completely in lieu of, or may be a prelude to further refinement of the GUI using, for example, an editor application as described previously.

By way of further example, turning to Figure 11 there is illustrated a series of steps which in certain embodiments may be performed, for example, between steps 504 and 506 of the flowchart previously presented in Figure 5. Upon receipt of a request message, the operating software of a slave relay device may at step 1102 examine the request header user agent information in order to identify the requesting client device type. If the client device is determined to be an already configured client at step 114, processing continues as previously described in conjunction with Figure 5. If not, the operating software next checks at step 1106 if the target appliances to be controlled have been identified and set up, i.e., that the appropriate appliance code data and protocols have been identified and are available for use by the operating software. If not, the operating software at step 1108 may interact with the user (e.g., via HTTP and pre-

formatted HTML pages) to obtain appliance identification information. Such information may include for example, appliance types, brand name(s), model number(s), set up codes derived from a printed list or web-based look up site, etc. without limitation. At step 1110, the obtained appliance identification(s) are submitted to a set-up service. Such a set up service may be part of the operating software of slave relay device 100, or may be an application hosted on a local 202 or remote 210 computing system, or any combination thereof. The set up service may analyze the submitted appliance identification information and identify to the operating software of slave relay device 100 which appliance code data and protocols are to be used to communicate to each of the appliances. In certain embodiments, if the necessary appliance code data and/or protocol definition are not already present in a pre-programmed library of the slave relay device, these may be downloaded into the slave relay device by the set up service. Once the appliances to be controlled have been identified, at step 1112 the user agent and appliance identity information may be submitted to a GUI selection engine, e.g., as described above, which may once again be part of the operating software of slave relay device 100, or may be an application hosted on a local 202 or remote 210 computing system, or any combination thereof. It will also be appreciated that the set up service and the GUI selection engine may be separate application programs, resident on different computing systems or co-resident on a single system; or may be combined into a single application program, as appropriate for a particular implementation. Based upon the supplied information, the GUI selection engine may select and/or generate appropriate default HTML GUI pages commensurate with the capabilities of the client device and the control functions of the target appliance (for example, a page scaled for the display of the client device providing function keys only for operations that are actually supported by the target appliance). At step 1114 these pages (or pointers thereto) are identified and stored by the operating software of slave relay device 100, where after they may be used to service HTTP requests from the identified client device. It will be appreciated that alternative embodiments providing similar functionality are also possible: for example, a slave relay device may store only the user-supplied appliance identification information and re-submit this together with the user agent identity every time a request is received, e.g., using a GUI selection engine which re-generates the HTML GUI page(s) for every instance of use.

In order to guard against unauthorized manipulation of a user's appliances, various security measures may be implemented to limit GUI page access to only authorized

clients. These may take the form of, for example, password protection of a slave relay device's master (e.g., "home") pages, restriction of access to only certain pre-defined client devices (e.g., using MAC or IP address filtering), mutual authentication, or any other suitable method, all as are well known in the art.

* * * * *

While various concepts have been described in detail, it will be appreciated by those skilled in the art that various modifications and alternatives to those concepts could be developed in light of the overall teachings of the disclosure. For example, while many of the exemplary embodiments above are described in terms of HTP and HTML methodologies, it will be appreciated that other communication and transfer protocols may be used as appropriate. Further, while an exemplary slave relay device communicates with controlled appliances via an IR signal, it will be appreciated that for the purposes of this invention various alternate embodiments of slave relay device may communicate with controlled appliances via any combination of IR, RF and/or hard wired connections. Yet further, it will be appreciated that while the exemplary slave relay devices of the illustrative embodiments are presented as stand alone units, in alternative embodiments the described slave relay device functionality may be incorporated into one or more of the controlled appliances, or accommodated in any other convenient item of furniture or equipment, as appropriate for a particular implementation.

Further, while described in the context of functional modules and illustrated using block diagram format, it is to be understood that, unless otherwise stated to the contrary, one or more of the described functions and/or features may be integrated in a single physical device and/or a software module, or one or more functions and/or features may be implemented in separate physical devices or software modules. It will also be appreciated that a detailed discussion of the actual implementation of each module is not necessary for an enabling understanding of the invention. Rather, the actual implementation of such modules would be well within the routine skill of an engineer, given the disclosure herein of the attributes, functionality, and inter-relationship of the various functional modules in the system. Therefore, a person skilled in the art, applying ordinary skill, will be able to practice the invention set forth in the claims without undue experimentation. It will be additionally appreciated that the particular concepts disclosed are meant to be illustrative only and not limiting as to the scope of the invention which is to be given the full breadth of the appended claims and any equivalents thereof.

All patents and published documents cited within this application are hereby incorporated by reference in their entirety.

CLAIMS

What is claimed is:

1. A method for using a personal communication device having a display and a slave relay station in communication with the personal communication device via a network to control operations of one or more controllable devices; comprising:

transmitting from the personal communication device to the slave relay station via the network a first request;

receiving at the personal communication device a graphical user interface page having activatable links corresponding to controllable operations of one or more controllable devices, the graphical user interface page being provided to the personal communication device in response to the first request;

causing the received graphical user interface page to be displayed in the display of the personal communication device;

receiving at the personal communication device a selection of one or more of the activatable links of the displayed, received graphical user interface page;

transmitting from the personal communication device to the slave relay station via the network a second request in response to the selection of one or more of the activatable links of the displayed, received graphical user interface page; and

causing the slave relay station to transmit to the one or more controllable devices one or more commands to thereby control one or more operations of the one or more controllable device, the one or more commands being transmitted by the slave relay station in response to the transmitted second request.

2. The method as recited in claim 1, wherein the graphical user interface page is stored in a memory of the slave relay station and provided to the personal communication device via the network in response to the first request.

3. The method as recited in claim 1, wherein the graphical user interface page is stored in a memory of a server device in communication with the slave relay station and the personal communication device and is provided to the personal communication device from the server device via the network in response to the first request.

4. The method as recited in claim 1, wherein the graphical user interface page is stored in a memory of a server device in communication with the slave relay station and is

provided to the personal communication device from the server device in response to the first request.

5. The method as recited in claim 4, wherein the server device is in communication with the personal communication device via a further network and the graphical user interface page is provided to the personal communication device via the further network.

6. The method as recited in claim 1, wherein the graphical user interface page is stored in a memory of a server device in communication with the slave relay station and is retrieved from the memory of the server device by the slave relay station and provided to the personal communication device from the slave relay station in response to the first request.

7. The method as recited in claim 1, comprising placing the graphical user interface page into a form determined to be appropriate for capabilities of the personal communication device.

8. The method as recited in claim 7, wherein the first request comprises data indicative of the capabilities of the personal communication device.

9. The method as recited in claim 8, wherein the data indicative of the capabilities of the personal communication device comprises a make and model of the personal communication device.

10. The method as recited in claim 8, wherein the data indicative of the capabilities of the personal communication device comprises an identity and revision of software used in connection with the display.

11. The method as recited in claim 1, wherein the graphical user interface page is selected from a plurality of preformatted graphical user interface pages as a function of capabilities of the personal communication device.

12. The method as recited in claim 11, wherein the first request comprises data indicative of the capabilities of the personal communication device.

13. The method as recited in claim 12, wherein the data indicative of the capabilities of the personal communication device comprises a make and model of the personal communication device.

14. The method as recited in claim 12, wherein the data indicative of the capabilities of the personal communication device comprises an identity and revision of software used in connection with the display.

15. The method as recited in claim 1, wherein the slave relay station is preconfigured with the one or more commands to control the one or more operations of the one or more controllable devices.

16. The method as recited in claim 1, wherein the slave relay station retrieves from a remote repository the one or more commands to control the one or more operations of the one or more controllable devices in response to the transmitted second request.

17. The method as recited in claim 1, comprising providing to the slave relay station data indicative of the one or more controllable devices and using the data to cause the slave relay station to configure itself to transmit commands to the one or more controllable devices.

18. The method as recited in claim 17, wherein the slave relay station uses the data to select from a library of commands stored in a memory of the slave relay station those commands appropriate for the one or more controllable devices.

19. The method as recited in claim 17, wherein the slave relay station uses the data to retrieve from a library of commands stored remotely from the slave relay station those commands appropriate for the one or more controllable devices.

20. The method as recited in claim 17, wherein the slave relay station uses the data to retrieve from a library of graphical user interface pages stored remotely from the slave relay station those graphical user interface pages appropriate for the one or more controllable devices to be thereafter provided to the personal communications device in response to corresponding requests.

21. The method as recited in claim 17, wherein the data indicative of the one or more controllable devices is provided to the slave relay station via user interaction with a configuration graphical user interface page displayed in the display of the personal communication device.

22. The method as recited in claim 1, comprising providing to a server device in communication with the slave relay station data indicative of the one or more controllable devices and using the data at the server device to cause the slave relay station to be configured to transmit commands to the one or more controllable devices.

23. The method as recited in claim 22, wherein the slave relay station is configured by using the server device to cause the slave relay station to select from a library of commands stored in a memory of the slave relay station those commands appropriate for the one or more controllable devices.

24. The method as recited in claim 22, wherein the server device uses the data to download to the slave relay station commands appropriate for the one or more controllable devices.

25. The method as recited in claim 22, wherein the server device uses the data to download to the slave relay station graphical user interface pages appropriate for the one or more controllable devices to be thereafter provided to the personal communications device in response to corresponding requests.

26. The method as recited in claim 1, wherein the graphical user interface page comprises a markup language page.

27. The method as recited in claim 1, wherein at least a portion of communications between the personal communication device and the slave relay station are secured.

28. A method for using a computer-based editor program to create a graphical user interface page having activatable links corresponding to controllable operations of one or more controllable devices to provide a means for using a personal communication device having a display and a slave relay station in communication with the personal

communication device via a network to control operations of the one or more controllable devices; comprising:

displaying in the editor an image of a graphical user interface page under creation and a plurality of activatable links;

accepting input into the computer to select and place an activitable link from the plurality of activatable links into the graphical user interface page under creation;

accepting input into the computer to select and associate with the activatable link placed into the graphical user interface page being created at least one action to be performed upon activation of the activatable link when displayed in the display of the personal communication device, wherein the action is defined by a tag to be transmitted by the personal communication device to the slave relay device in response to activation of the activatable link when displayed in the display of the personal communication device and one or more actions to be executed by operating software of the slave relay station upon receipt of the transmitted tag;

upon receiving input into the computer that indicates completion of editing of the graphical user interface page, placing data representative of the graphical user interface page and actions associated with activatable links of the graphical user interface page into a file; and

associating the file with a network address whereupon the file is available to be provisioned to the slave relay device to facilitate the control of operations of the one or more controllable devices by means of the personal communication device in communication with slave relay device via the network.

29. The method as recited in claim 28, wherein the graphical user interface page comprises a markup language page and the one or more statements comprise markup language statements.

30. The method as recited in claim 29, wherein the one or more actions to be executed by operating software of the slave relay stations comprise statements that identify a device to be controlled and commands to be transmitted to the identified device.

31. The method as recited in claim 28, wherein the activatable links comprise an image.

32. The method as recited in claim 30, wherein the image is stored on a server device remote from the slave relay device and the file includes a pointer to the image.

33. The method as recited in claim 28, wherein the file comprises data adapted to be modifiable to place the graphical user interface page into a form determined to be appropriate for capabilities of the personal communication device.

34. The method as recited in claim 28, wherein the file comprises data adapted to present the graphical user interface page in a form determined to be appropriate for capabilities of a predetermined type of personal communication device.

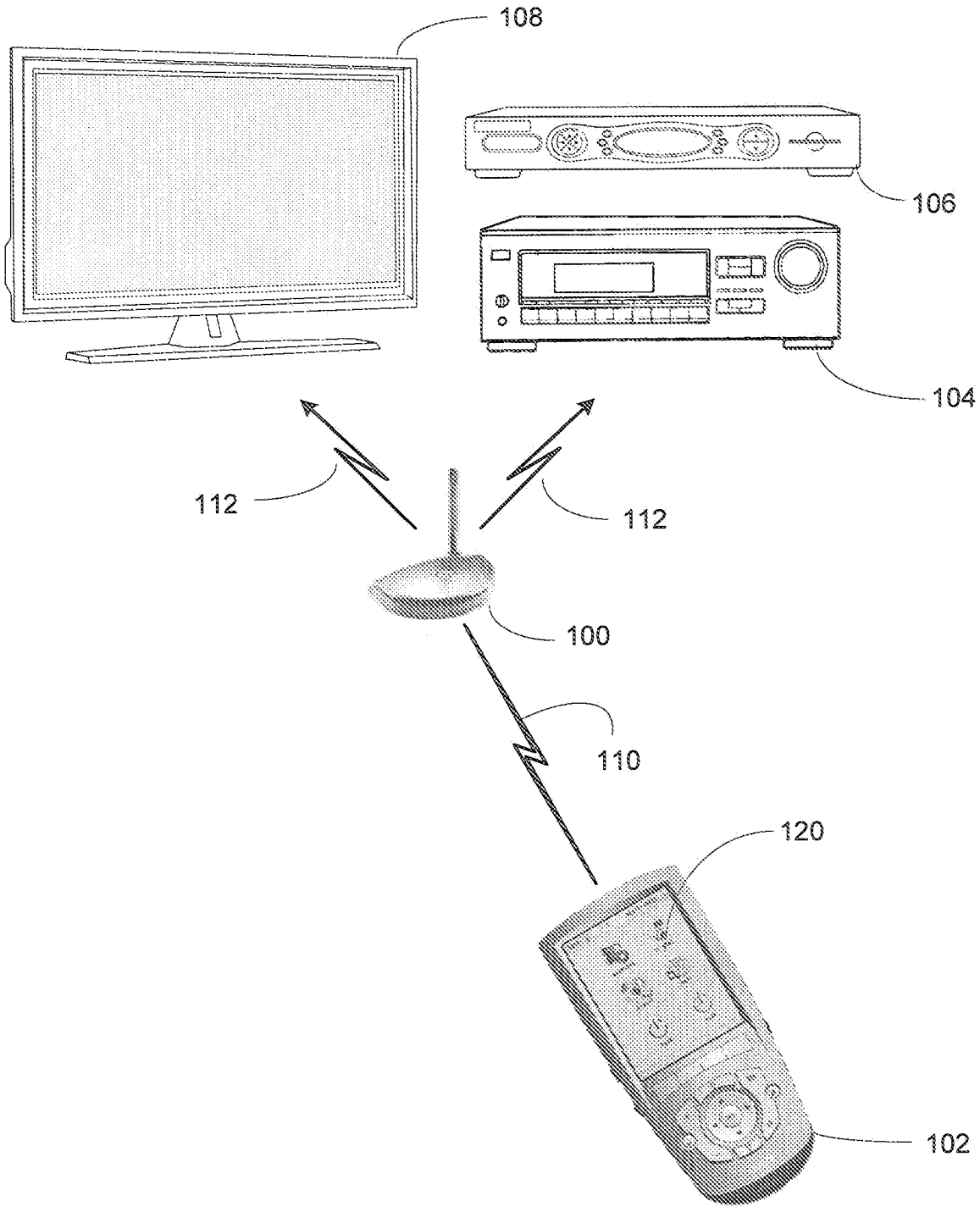


Figure 1
(Prior art)

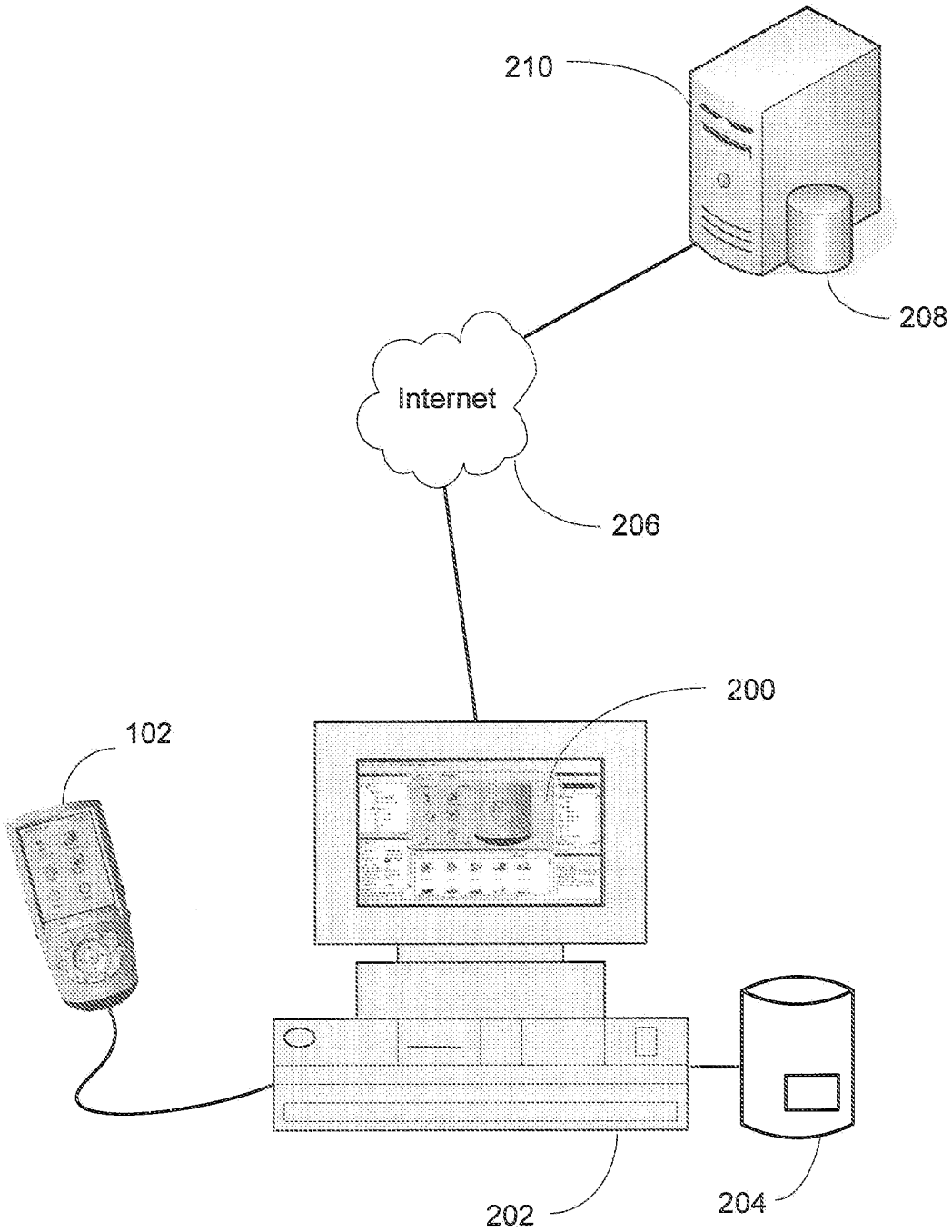


Figure 2
(Prior art)

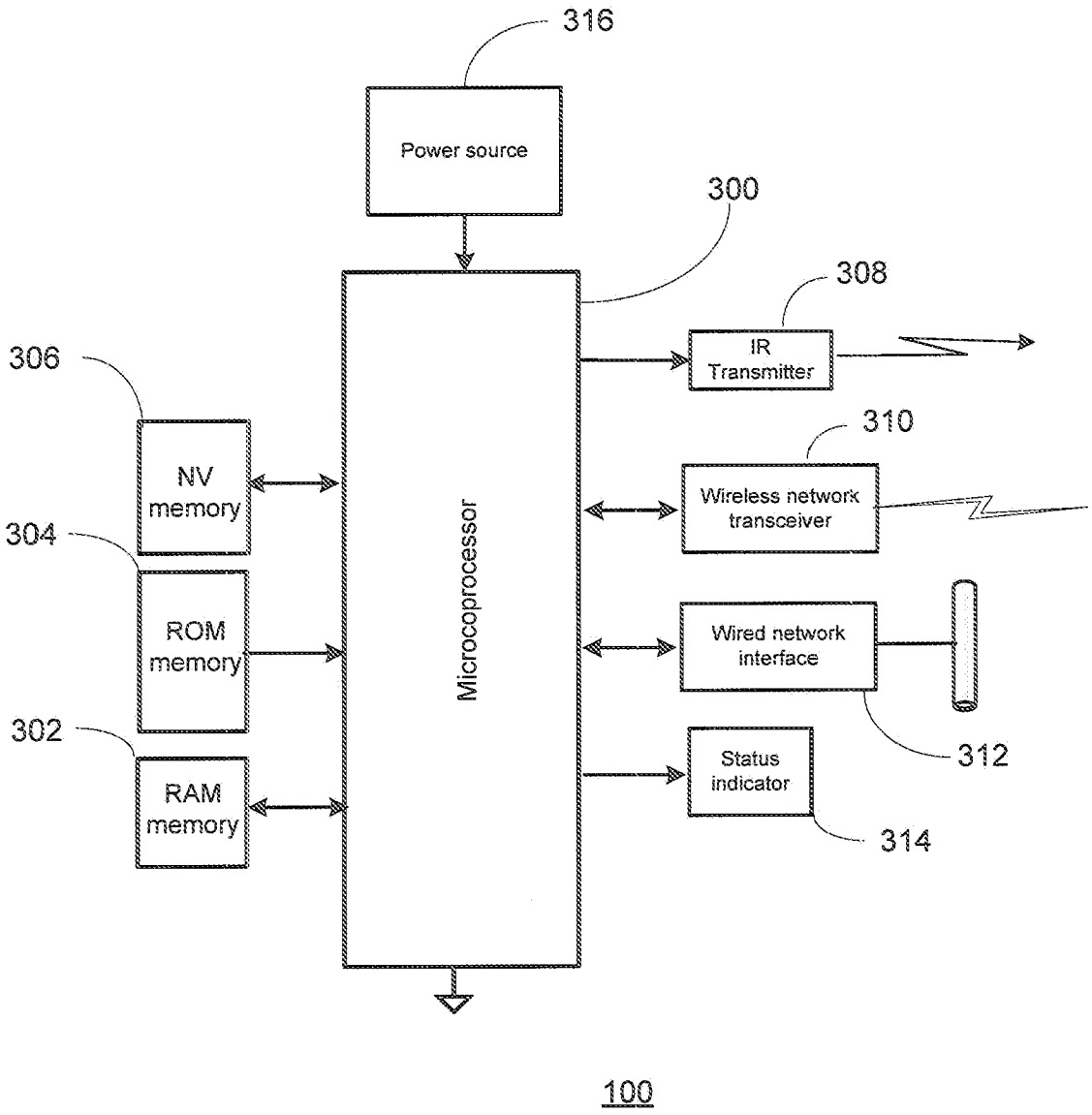

316
Power source
300
308
IR Transmitter
306
NV memory
310
Wireless network transceiver
304
ROM memory
Microcoprocessor
Wired network interface
302
RAM memory
Status indicator
312
314
100

Figure 3

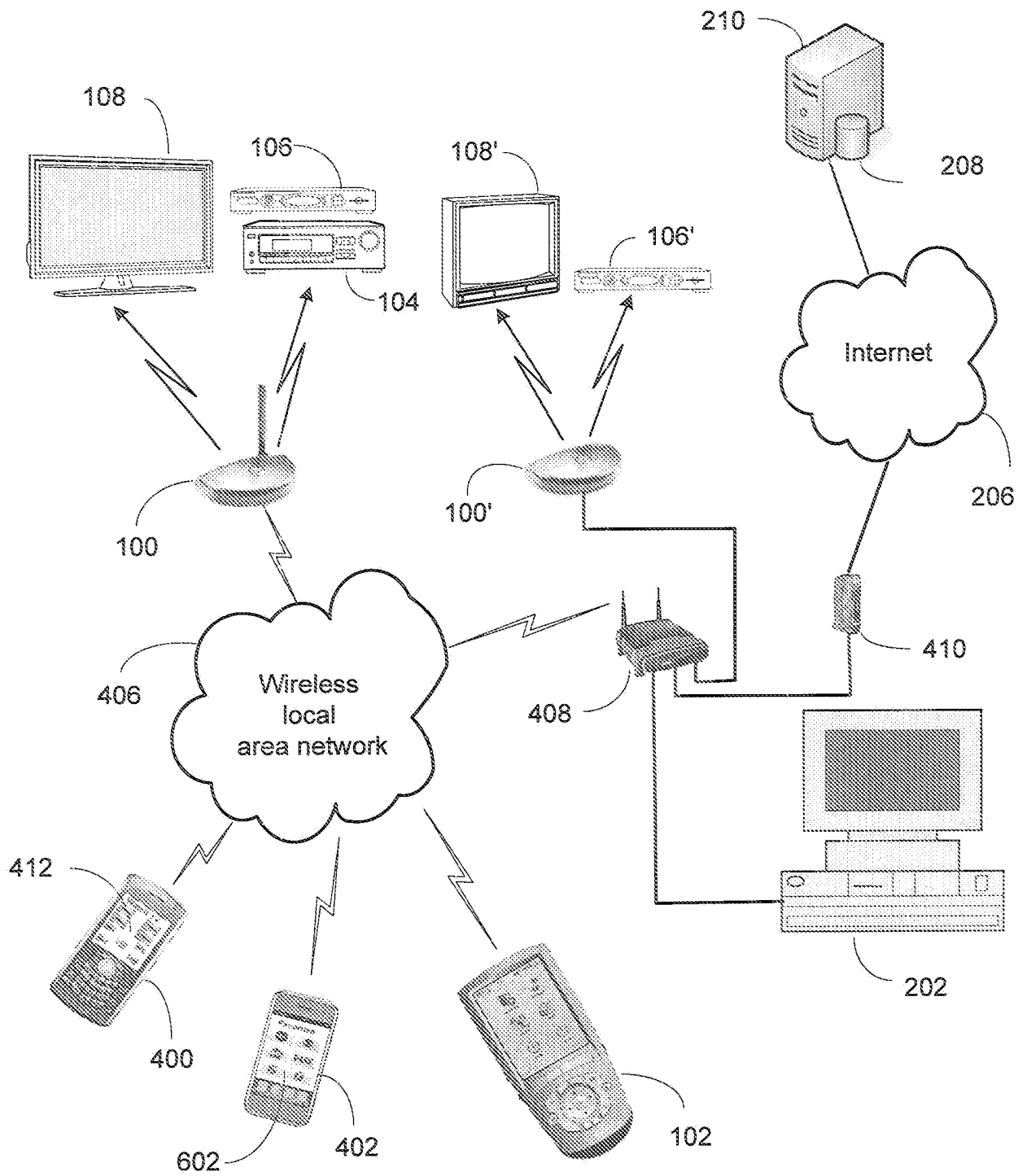

210
208
108
106
108'
106'
104
Internet
206
100
100'
406
Wireless
local
area network
408
410
412
400
602
402
102
202

Figure 4a

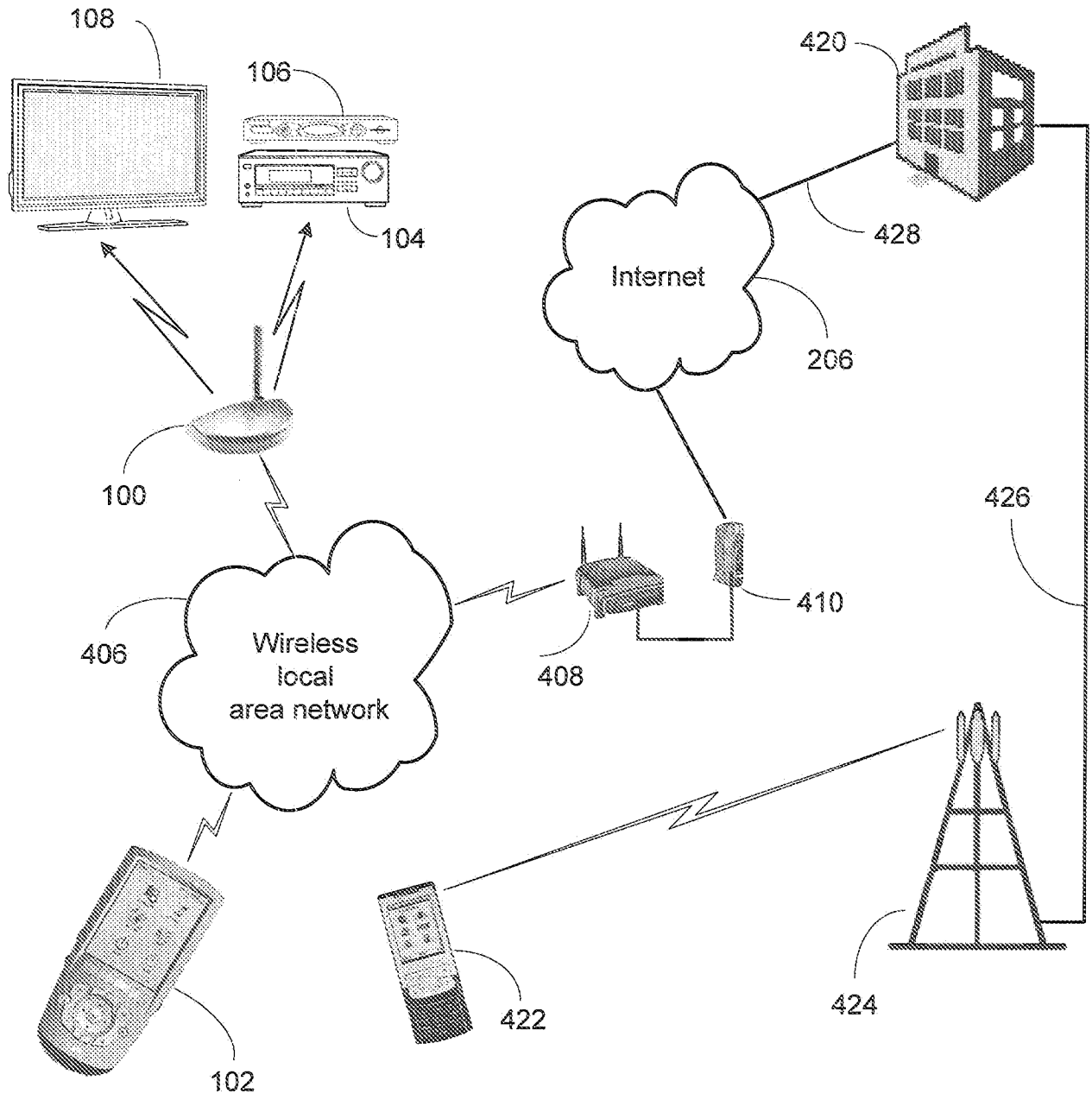

108
106
104
420
428
Internet
206
100
426
406
Wireless
local
area network
408
410
422
424
102

Figure 4b

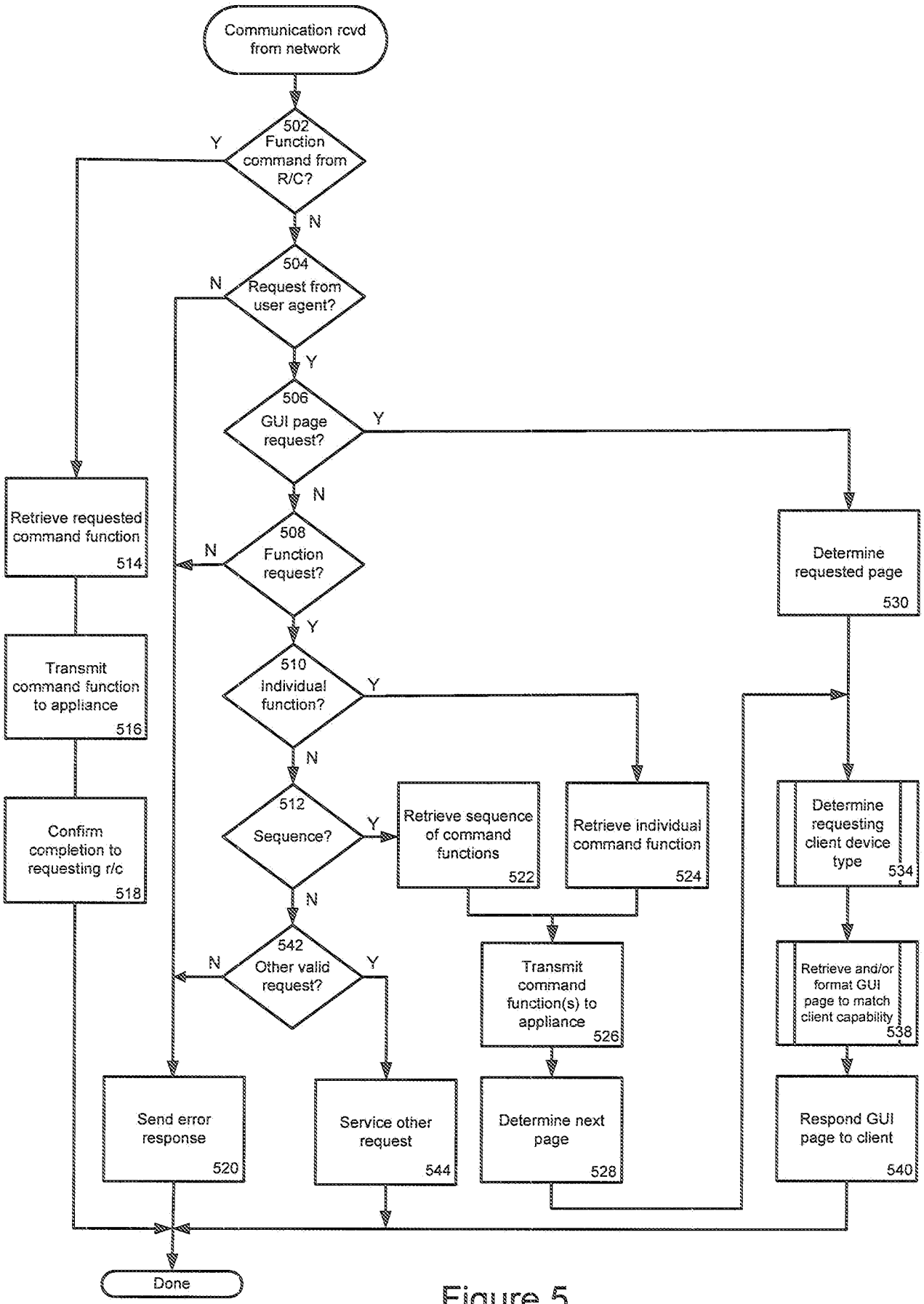
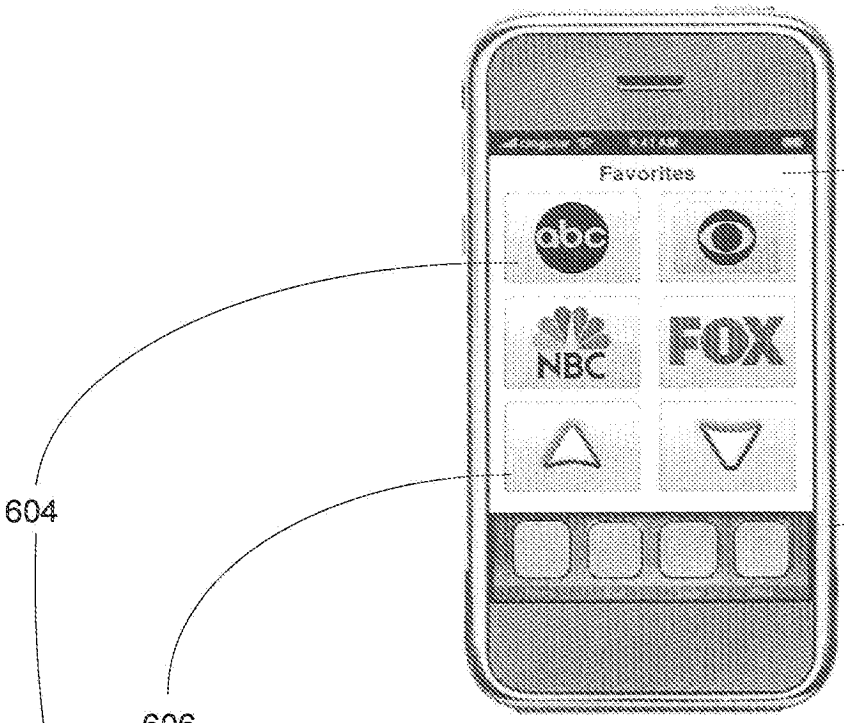

Communication rcvd from network
502 Function command from R/C?
Y
N
504 Request from user agent?
N
Y
506 GUI page request?
Y
N
508 Function request?
N
Y
510 individual function?
Y
N
512 Sequence?
Y
N
542 Other valid request?
N
Y
Retrieve requested command function 514
Transmit command function to appliance 516
Confirm completion to requesting r/c 518
Send error response 520
Retrieve sequence of command functions 522
Retrieve individual command function 524
Transmit command function(s) to appliance 526
Service other request 544
Determine next page 528
Determine requested page 530
Determine requesting client device type 534
Retrieve and/or format GUI page to match client capability 538
Respond GUI page to client 540
Done

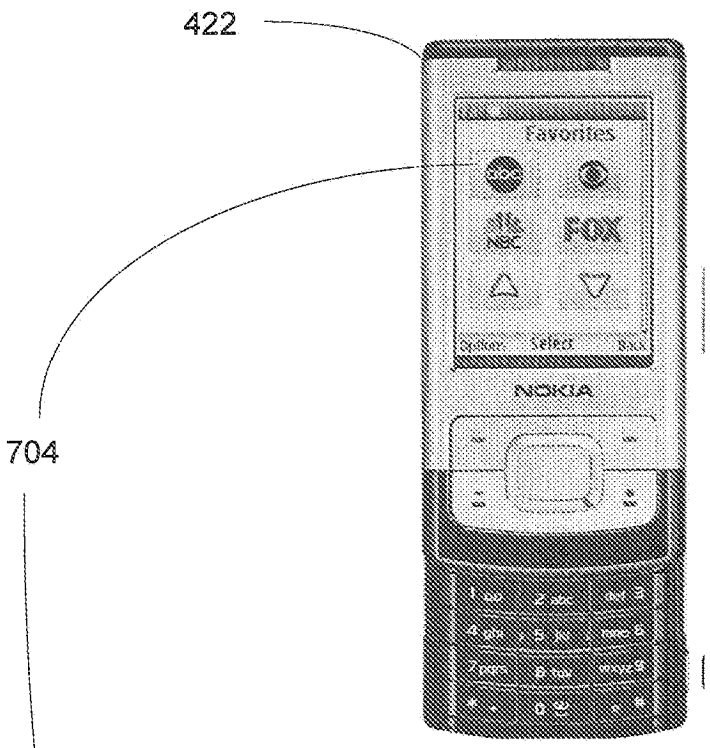
Figure 5




```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Strict//EN">
<html>
<head>
<meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
<title>Favorites</title>
</head>
<body style="color: rgb(0, 0, 0); background-color: rgb(255, 255, 204);">
<div style="text-align: center;"><span style="font-family: Helvetica,Arial,sans-
serif;"> <big style="font-weight: bold;"><big><span style="color: rgb(51, 51,
153);">        Favorites</span></big></big></span></div>
<div>
<a href="switch_to_abc.irm"><img style="border: 0px solid ; width: 445px; height:
240px;" alt="ABC" src="abc_button.png" hspace="15" vspace="5"></a>
<a href="switch_to_cbs.irm"><img style="border: 0px solid ; width: 445px; height:
240px;" alt="CBS" src="cbs_button.png" hspace="5" vspace="5"></a>
</div>
<div>
<a href="switch_to_nbc.irm"><img style="border: 0px solid ; width: 445px; height:
240px;" alt="NBC" src="nbc_button.png" hspace="15" vspace="5"></a>
<a href="switch_to_fox.irm"><img style="border: 0px solid ; width: 445px; height:
240px;" alt="Fox" src="fox_button.png" hspace="5" vspace="5"></a>
</div>
<div>
<a href="channel_up.irc"><img style="border: 0px solid ; width: 445px; height: 240px;"
alt="Channel up" src="button_channel_up.png" hspace="15" vspace="5"></a>
<a href="channel_down.irc"><img src="button_channel_down.png" alt="Channel
down" style="border: 0px solid ; width: 445px; height: 240px;" hspace="5"
vspace="5"></a>
</div>
</body>
</html>
```

600

Figure 6




```
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Strict//EN">
<html>
<head>
<meta content="text/html; charset=ISO-8859-1" http-equiv="content-type">
<title>Favorites</title>
</head>
<body style="color: rgb(0, 0, 0); background-color: rgb(255, 255, 204);">
<div style="text-align: center;"><span style="font-family: Helvetica,Arial,sans-
serif;"> <big style="font-weight: bold;"><big><span style="color: rgb(51, 51,
153);">        Favorites</span></big></big></span></div>
<div>
<a href="switch_to_abc.irm"><img style="border: 0px solid ; width: 88px;
height: 61px;" alt="ABC" src="abc_button.png" hspace="15" vspace="5"></a>
<a href="switch_to_cbs.irm"><img style="border: 0px solid ; width: 88px;
height: 61px;" alt="CBS" src="cbs_button.png" hspace="5" vspace="5"></a>
</div>
<div>
<a href="switch_to_nbc.irm"><img style="border: 0px solid ; width: 88px;
height: 61px;" alt="NBC" src="nbc_button.png" hspace="15" vspace="5"></a>
<a href="switch_to_fox.irm"><img style="border: 0px solid ; width: 88px;
height: 61px;" alt="Fox" src="fox_button.png" hspace="5" vspace="5"></a>
</div>
<div>
<a href="channel_up.irc"><img style="border: 0px solid ; width: 88px; height: 61px;"
alt="Channel up" src="button_channel_up.png" hspace="15" vspace="5"></a>
<a href="channel_down.irc"><img style="border: 0px solid img width: 88px; height:
61px;" alt="Channel down" src="button_channel_down.png"  hspace="5"
vspace="5"></a>
</div>
</body>
</html>
```

700

Figure 7

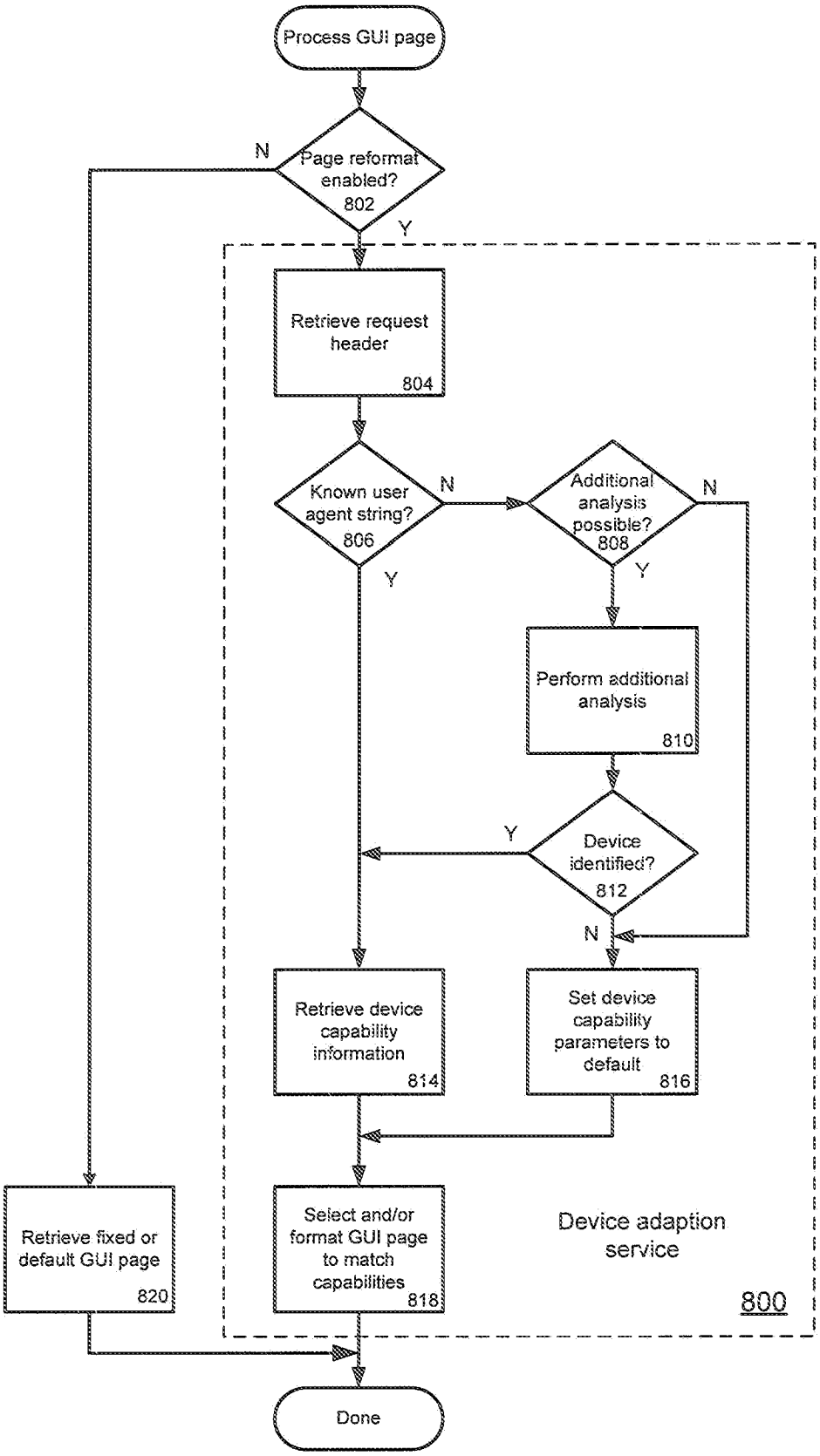

Process GUI page
Page reformat enabled? 802
N
Y
Retrieve request header 804
Known user agent string? 806
N
Y
Additional analysis possible? 808
N
Y
Perform additional analysis 810
Device identified? 812
Y
N
Retrieve device capability information 814
Set device capability parameters to default 816
Select and/or format GUI page to match capabilities 818
Retrieve fixed or default GUI page 820
Device adaption service
800
Done

Figure 8

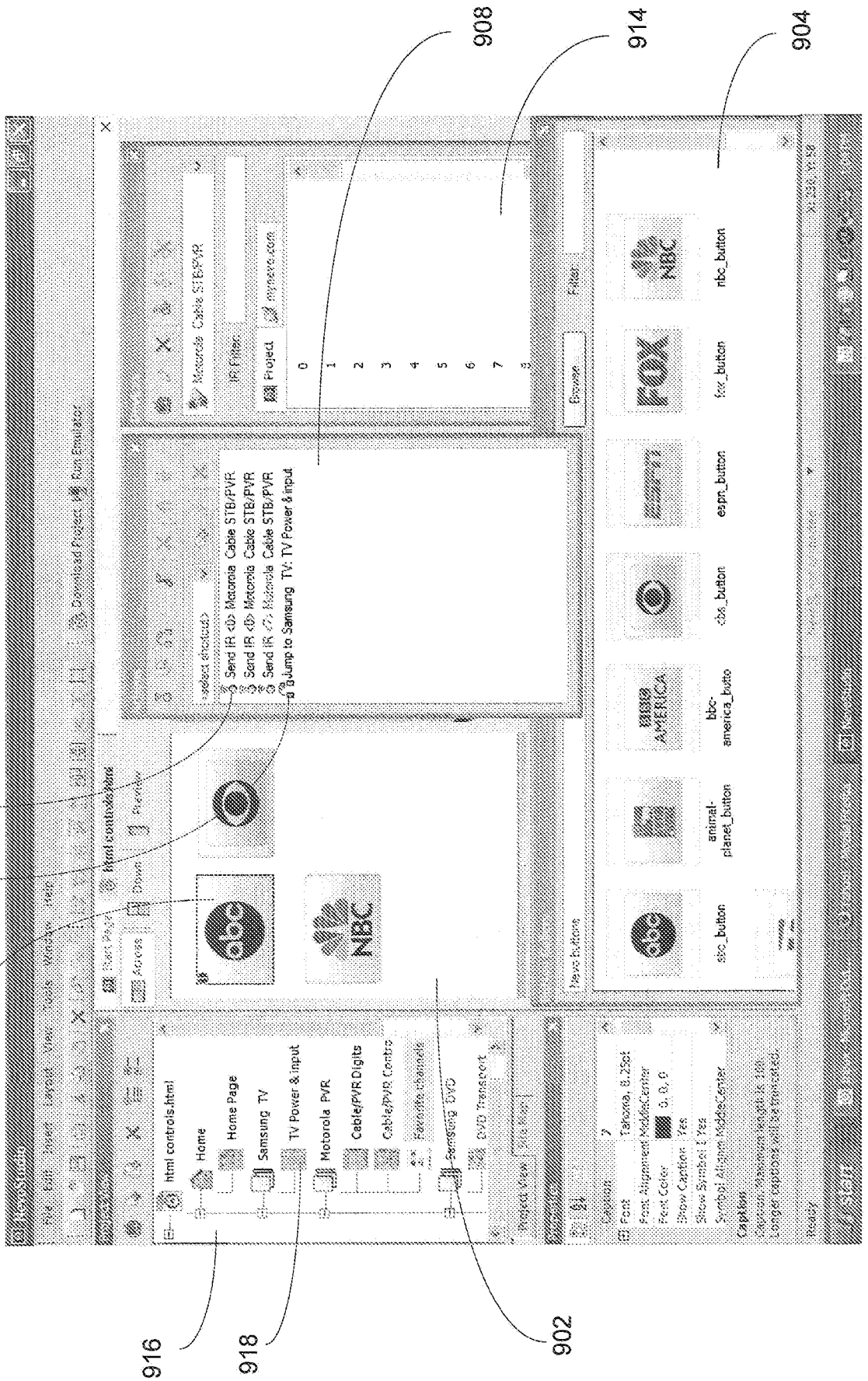

908
914
904
910
912
906
916
918
902
900
html controls.html
Home
Home Page
Samsung TV
TV Power & input
Motorola PVR
Cable/PVR Digits
Cable/PVR Centre
Favorite channels
Samsung DVD
DVD Transport
Send IR <b> Motorola Cable STB/PVR
Send IR <b> Motorola Cable STB/PVR
Send IR <> Motorola Cable STB/PVR
Jump to Samsung TV: TV Power & input
Motorola Cable STB/PVR
IR Filter:
Project
nbc_button
fox_button
espn_button
cbs_button
bbc-america_button
animal-planet_button
abc_button
NBC
FOX
AMERICA
Browse
Filter

Figure 9

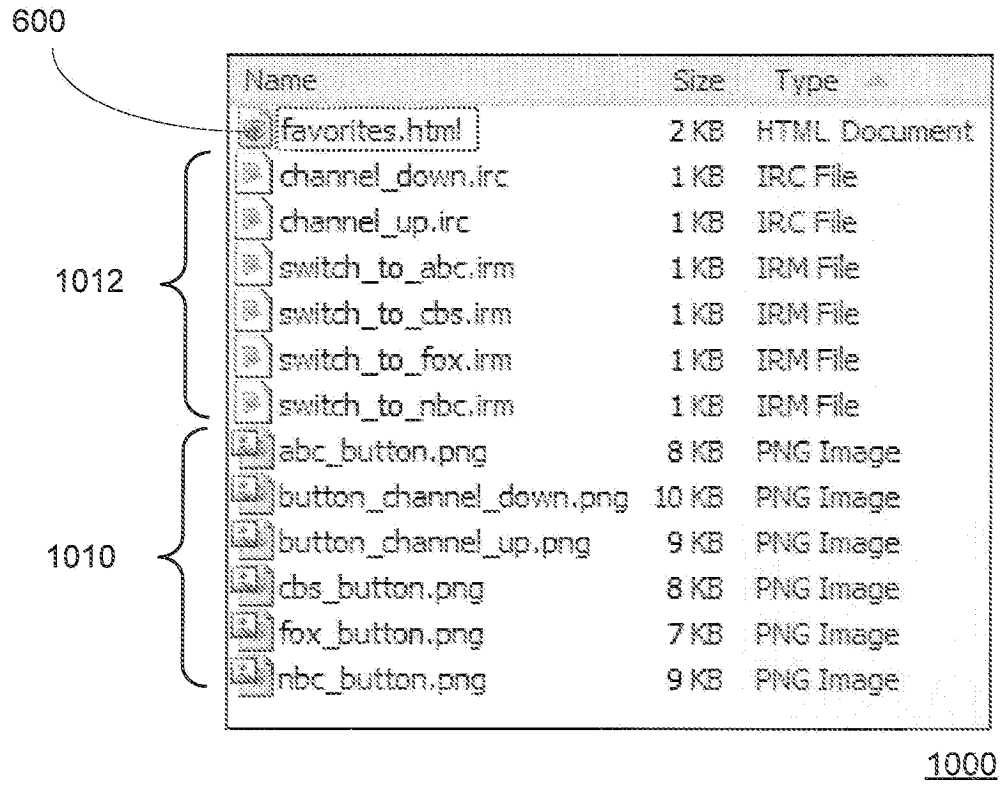
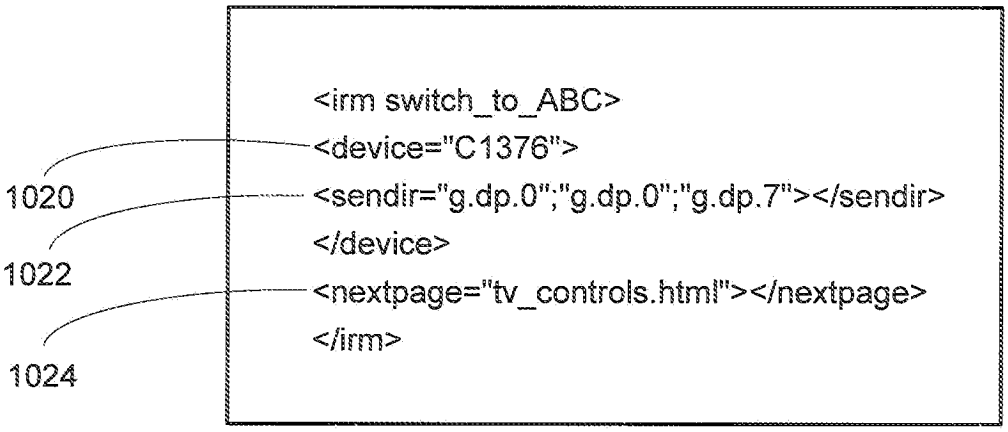


Figure 10a



File "switch_to_abc.irm

Figure 10b

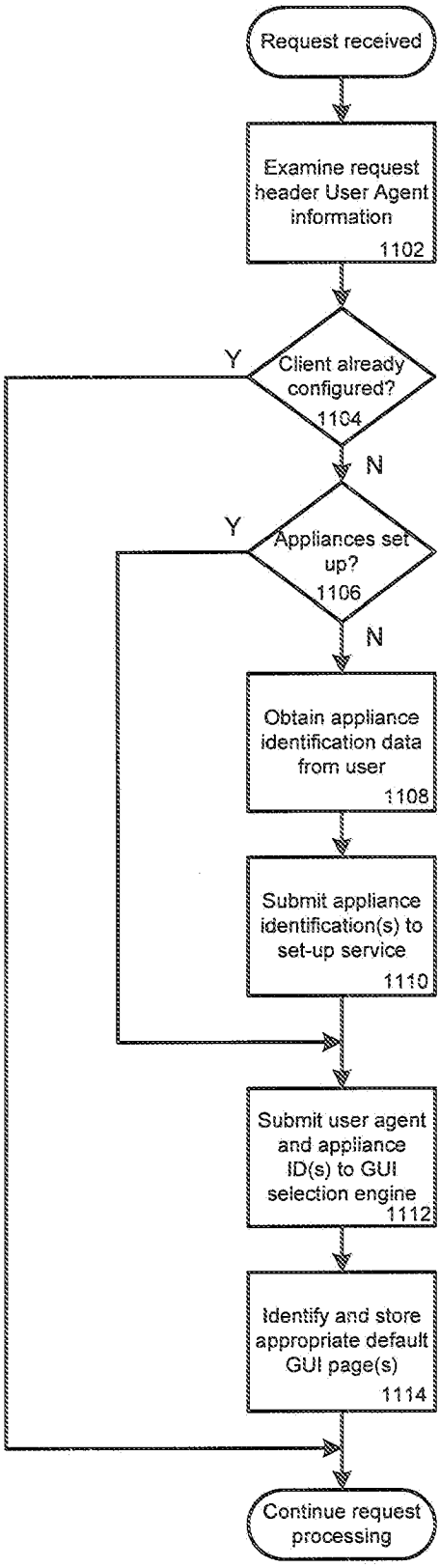

Request received
Examine request header User Agent information
1102
Client already configured?
1104
Y
N
Appliances set up?
1106
Y
N
Obtain appliance identification data from user
1108
Submit appliance identification(s) to set-up service
1110
Submit user agent and appliance ID(s) to GUI selection engine
1112
Identify and store appropriate default GUI page(s)
1114
Continue request processing

Figure 11

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US 09/48254

A. CLASSIFICATION OF SUBJECT MATTER
IPC(8) - H04L 27/04 (2009.01)
USPC - 375/310
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
USPC - 375/310

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
USPC - 375/310; 340/425.1; 345/650 (text search)

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
PubWest (PGPB, USPT, EPAB, JPAB), Google,
Search terms used: remote, control, device, appliance, item, relay, universal, user, interface, link, station, slave, code, server, format, compatib$6

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X --- Y	US 2005/0080496 A1 (Hayes et al.) 14 April 2005 (14.04.2005), Fig. 14, para [0094], [0106], [0113], [0122], [0131], [0137], [0138], [0150]-[0161]], [0184]	1-26 --- 27, 29, 30, 32
X --- Y	US 2006/0050142 A1 (Scott et al.) 09 March 2006 (09.03.2006), Fig. 23, para [0006], [0034]-[0043], [0049]	28, 31, 33, 34 --- 29, 30, 32
Y	US 2006/0288300 A1 (Chambers et al.) 21 December 2006 (21.12.2006), para [0007]	27
A	US 2004/0260427 A1 (Wimsatt) 23 December 2004 (23.12.2004), entire document	1 - 34
A	US 2005/0097478 A1 (Killian et al.) 05 May 2005 (05.05.2005), entire document	1 - 34

☐ Further documents are listed in the continuation of Box C. ☐

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
30 July 2009 (30.07.2009)	06 AUG 2009
Name and mailing address of the ISA/US Mail Stop PCT, Attn: ISA/US, Commissioner for Patents P.O. Box 1450, Alexandria, Virginia 22313-1450 Facsimile No. 571-273-3201	Authorized officer: Lee W. Young PCT Helpdesk: 571-272-4300 PCT OSP: 571-272-7774

Form PCT/ISA/210 (second sheet) (April 2007)