



(22) Date de dépôt/Filing Date: 2005/01/04

(41) Mise à la disp. pub./Open to Public Insp.: 2005/07/05

(30) Priorité/Priority: 2004/01/05 (10/751,694) US

(51) Cl.Int.⁷/Int.Cl.⁷ G06F 3/00, G06F 9/44, H04N 7/173

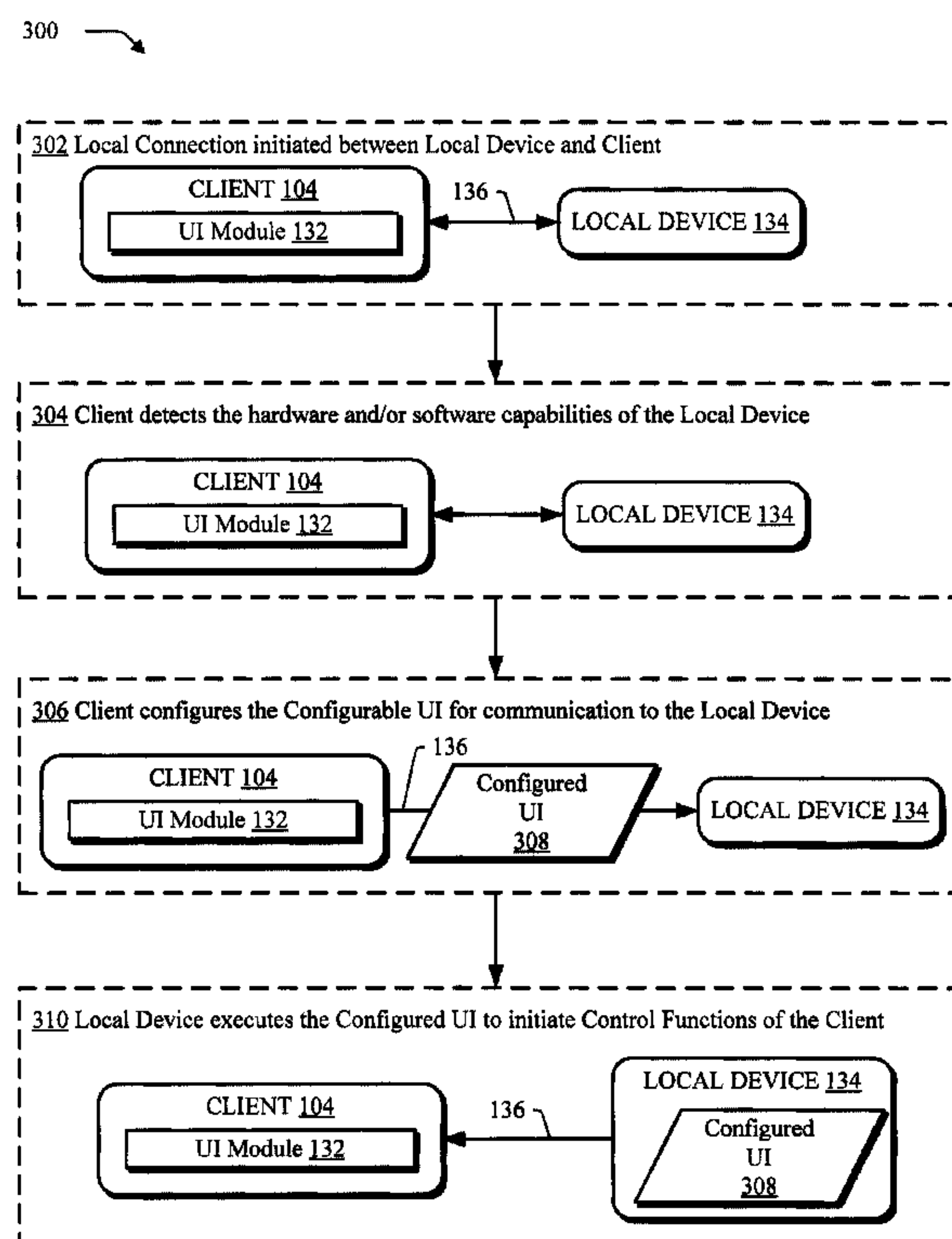
(71) Demandeur/Applicant:
MICROSOFT CORPORATION, US

(72) Inventeur/Inventor:
BARRETT, PATER T., US

(74) Agent: SMART & BIGGAR

(54) Titre : CONFIGURATION D'INTERFACES UTILISATEUR

(54) Title: CONFIGURATION OF USER INTERFACES



(57) Abrégé/Abstract:

Configuration of user interfaces is described. In an implementation, a method includes receiving a query from a device. The query is for a user interface (UI) from a client. The UI, when executed by the device, is configured to form a request for execution of a control function by the client. A processing of the request determines whether the execution by the client of the control function is permitted. The processing is performed at a head end by using client state data. The hardware and/or software capabilities of the device to execute the UI are detected. The UI is configured based on the hardware and/or software capabilities of the device.

ABSTRACT

Configuration of user interfaces is described. In an implementation, a method includes receiving a query from a device. The query is for a user interface (UI) from a client. The UI, when executed by the device, is configured to form a request for execution of a control function by the client. A processing of the request determines whether the execution by the client of the control function is permitted. The processing is performed at a head end by using client state data. The hardware and/or software capabilities of the device to execute the UI are detected. The UI is configured based on the hardware and/or software capabilities of the device.

CONFIGURATION OF USER INTERFACES

TECHNICAL FIELD

[0001] The present invention generally relates to the field of user interfaces and in particular to configuration of user interfaces.

BACKGROUND

[0002] Users of clients, such as set-top boxes, wireless phones, computers, and so on, are continually exposed to an ever increasing variety of content. A user of a set-top box, for instance, may view traditional television programming obtained from a broadcast network for display on a television, as well as pay-per-view movies, video-on-demand (VOD), interactive video games, and so on. Likewise, a user of a wireless phone may place and receive traditional telephone calls, as well as read email, schedule appointments, play digital music, and so on.

[0003] As shown in the previous examples, the capabilities provided by clients have also continued to increase to address the ever increasing varieties of content. For example, a user may be able to receive email on a computer as well as a set-top box, a personal digital assistant, and a wireless phone. Even though the capabilities of clients and networks that communicatively couple the clients have improved, a variety of applications are still not implemented due to failure of applications to address the capabilities and limitations of the clients and the networks. In other words, an application that may be configured for use on a particular type of client, such as a client having a particular combination of hardware and/or software capabilities, may have limited utility on another type of client that does not have the particular combination. Data-entry

intensive applications, for instance, when implemented in a traditional television environment may be limited due to limitations of input devices available to the user and because of limited resolution of traditional televisions. For example, a traditional remote control may have buttons to directly enter television channels, offer four-way navigation, and manipulate power and volume. Functions such as text entry and searching through a large list of options, which may be readily performed when using a keyboard, may become onerous when using the traditional remote control.

[0004] In addition, limited resolution that is available on a standard definition television may limit the effective resolution of data that may be displayed to the user at any one particular time. For example, limitations in the display of readable text on a traditional National Television Systems Committee (NTSC) television may limit the amount of text that may be displayed on the television at any one time. Additionally, the distance at which the display is to be viewed may limit the effective resolution of the display. For instance, to display an electronic program guide on a standard definition television such that the display may be viewed when the user is positioned at a typical distance from the television, the text of the electronic program guide may be enlarged such that it is readable from the desired distance. Therefore, even if the user is provided with a keyboard, the television experience may not be able to provide the functionality that is provided when the user interacts with the application using a computer.

[0005] Accordingly, there is a continuing need for user interfaces that address device capabilities and limitations to improve user interaction.

SUMMARY

[0006] Configuration of user interfaces is described. User interfaces are provided by a client to devices for initiation of control functions of the client. The user interfaces, for instance, are executed on the devices to initiate control functions of the client. For example, a user interface may be executed on a local device that is communicatively coupled to the client over a local network connection, such as a local area network (LAN). The user interface may also be executed on a remote device that is communicatively coupled to the client over a remote network connection, such as a wide area network (WAN). The local and remote devices execute the respective user interfaces that are obtained from the client to initiate control functions of the client, such as to record content. The user interfaces may be configured based on detected hardware and/or software capabilities of the respective local and remote devices that execute the user interfaces. Hardware and software capabilities may include resolution, processing power, memory resources, available applications, and so on. Therefore, the local and remote devices may each be provided with respective user interfaces for initiating control function of the client that address the hardware and/or software resources of the particular device.

[0007] In an implementation, a method includes receiving a query from a device. The query is for a user interface (UI) from a client. The UI, when executed by the device, is configured to form a request for execution of a control function by the client. A processing of the request determines whether the execution by the client of the control function is permitted. The processing is performed at a head end by using client state data. The hardware and/or software capabilities of the device to execute the UI are

detected. The UI is configured based on the hardware and/or software capabilities of the device.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] FIG. 1 is an illustration of an environment in an exemplary implementation that includes a content provider that is communicatively coupled to a client over a network.

[0009] FIG. 2 is an illustration of an exemplary implementation showing a distribution server, the client, a local device, and a remote device of FIG. 1 in greater detail.

[0010] FIG. 3 is a flow diagram of a procedure in an exemplary implementation in which the client configures a user interface for execution on the local device of FIG. 2.

[0011] FIG. 4 is an illustration of an exemplary implementation showing a user interface that is configured for output by the client when the user is located close to the display device, such as the display device of FIG. 1.

[0012] FIG. 5 is an illustration of an exemplary implementation showing a user interface that is configured for output by the client when the user is located at a greater distance from the display device than the user interface of FIG. 4.

[0013] FIG. 6 is a flow diagram depicting a procedure in an exemplary implementation in which a configured user interface of FIG. 3 is employed by a user to initiate a control function of the client.

[0014] FIG. 7 is a flow diagram depicting a procedure in an exemplary implementation in which a user interface is configured for and executed by the remote device of FIG. 2 to initiate one or more control functions of the client.

[0015] The same reference numbers are utilized in instances in the discussion to reference like structures and components.

DETAILED DESCRIPTION

[0016] Overview

Configuration of user interfaces (UIs) is described. UIs are provided by a client for initiation of control functions of the client. The UIs may be executed on a variety of devices to initiate control functions of the client. In an implementation, the UI may be executed on a local device that is communicatively coupled to the client over a local network connection (local connection). For example, the client may be configured as a set-top box that provides for output of broadcast content, such as traditional television programming, and provides for interaction with additional types of content, such as applications that are executed on the client, applications that are executed remotely over a network, and so on. The local device may be configured as a personal digital assistant (PDA), e.g. a “pocket” PC, which is communicatively coupled to the set-top box over a local radio frequency (RF) connection. The PDA executes a UI that is obtained from the set-top box to initiate control functions of the set-top box, such as to record content. The set-top box may configure the UI based on detected hardware and/or software capabilities of the PDA to execute the UI, such as resolution of the PDA, processing power, colors supported, and so on. Therefore, the PDA is provided with a UI that addresses the hardware and/or software resources of the PDA.

[0017] In another implementation, the UI may be executed on a remote device that is communicatively coupled to the client over a remote network connection (remote

connection). Continuing with the previous example, the set-top box provides for the output of content that is received over a network from a head end, such as traditional broadcast television, video-on-demand (VOD), remote application processing, and so on. A remote device, such as a desktop computer, may also be communicatively coupled to the network. The remote device also executes a UI that is obtained from the set-top box to initiate control functions of the set-top box. The set-top box configures the UI based on detected hardware and/or software capabilities of the remote device to execute the UI. Therefore, the remote device may also be provided with a UI that addresses the hardware and/or software resources of the remote device. Further discussion of the configuration of UIs may be found in relation to FIGS. 3 and 7.

[0018] In a further implementation, the head end stores client state data to process the requests for control functions received from the UIs. For example, the head end may include client state data, such as ratings limits, favorite channels, levels of service, and so on, that is accessible locally by the head end. The head end may utilize this client state data to process requests provided by the UIs that are executed locally by the local device or remotely by the remote device. After the requests are processed, the head end may then cause the client to perform the control function. By processing the requests at the head end, the head end provides an authoritative source for processing control functions to be performed by the client. This may result in a variety of increased functionality that is available to the user, such as an ability to change from an old client to a new client without manually updating client state data from the old client to the new client and remote initiation of control functions without obtaining a connection with the client.

Further discussion of the use of client state data by a head end may be found in relation of FIGS. 6 and 7.

[0019] Exemplary Systems

FIG. 1 is an illustration of an environment 100 in an exemplary implementation that includes a content provider 102 that is communicatively coupled to a client 104 over a network 106. The network 106 in the following implementations is an example of a wide area network (WAN), such as the Internet, and may also include a variety of other networks, such as a broadcast network, an intranet, a wired or wireless telephone network, and so forth.

[0020] The client 104 may be configured to receive content broadcast from the content provider 102 over the network 106. The content provider 102 includes content 108(k), where k can be any number from 1 to “K”, that is locally stored on the content provider 102. The content 108(k) may include a variety of data, such as television programming, video-on-demand, an electronic program guide (EPG), one or more results of remote application processing, and so on. The content provider 102 provides the content 108(k) over a network 110 to a head end 112. The network 110 may be the same as or different from network 106. The content 108(k) may then be stored in a database 114 as content 116(n), where n can be any number from 1 to “N”, on the head end 112 for broadcast over the network 106 to the client 104. In other words, the content 116(n) stored in the database 114 may be copies of the content 108(k) received from the content provider 102. The content 116(n) may also include additional data that is broadcast to the client 104. For example, the content 116(n) stored in the database 114 may include EPG data that is broadcast to the client 104 utilizing a carousel file system. The carousel file system

repeatedly broadcasts the EPG data over an out-of-band (OOB) channel to the client 104 over the network 106.

[0021] The head end 112 may also include a distribution server 118 to format and distribute the content 116(n) over the network 106. Distribution from the head end 112 to the client 104 may be accomplished in a number of ways, including cable, RF, microwave, and satellite. Although the head end 112 is illustrated as separate from the content provider 102, the content provider 102 may also include the head end 112.

[0022] The client 104 may be configured as a computer that is capable of communicating over the network 106, such as a desktop computer, a mobile station, an entertainment appliance, a set-top box 120 that is communicatively coupled to a display device 122 as illustrated, and so forth. The client 104 may also relate to a person and/or entity that operate the client 104. In other words, client 104 may describe a logical client that includes a user and/or a machine. Although one client 104 is illustrated, a plurality of clients may be communicatively coupled to the network 106.

[0023] The client 104 may also include a database 124 having locally stored content 126(m), where m can be any number from 1 to "M". For example, the client 104 may be configured as a personal video recorder (PVR) that includes the database 124 stored in hard disk memory. Due to the size of the memory, users are able to record content. The PVR also offers control functions, such as the ability to pause content that is currently being broadcast and allows viewers to watch the content while still in progress from the point it was paused. The PVR plays back the content from disk memory, starting at the pause event, while continuing to record the currently-broadcast content in the disk

memory. Additionally, the PVR may support other control functions, such as rewinding, fast forwarding a stored program, slow motion playback, and the like.

[0024] The client 104 is equipped with sufficient processing and storage capabilities to store and run a navigation application 128. The navigation application 128, when executed on the client 104, provides control functions for interacting with content. For example, the control functions may include the PVR control functions as previously discussed, as well as channel selection, EPG navigation, and so on.

[0025] The client 104 also includes a configurable UI 130 and a UI module 132. The configurable UI 130 provides an interface for execution on a device such that a user of the device may initiate one or more control functions of the navigation application 128. The UI module 132 is executed on the client 104 to configure the configurable UI 130 based on detected hardware and/or software resources of a device that is to execute the configurable UI 130. For example, a local device 134, such as a PDA as illustrated, may be communicatively coupled to the client 104 over a local connection 136. The UI module 132, when executed, detects the capabilities of the local device 134 to execute the configurable UI 130 and then configures the configurable UI 130 accordingly. The client 104 may also be communicatively coupled to a remote device 138, e.g. a desktop computer as illustrated, over the network 106. The UI module 132, when executed on the client 104, may also configured the configurable UI 130 based on the detected hardware and/or software capabilities of the remote device 138 to execute the configurable UI 130. The UI module 132 may be implemented in a variety of ways, such as an HTTP server that provides UIs in an HTML format to the local and remote devices 134, 138. The client 104, through execution of the UI module 132, may provide UIs for execution on

the local and remote device 134, 138 based on the capabilities of the respective local and remote device 134, 138 to execute the UIs.

[0026] The head end 112 includes a database 140 for storage of a plurality of client state data 142(j), where j can be any number from 1 to “J”, to process requests that were initiated by UIs that are executed on the local and remote device 134, 138. For example, the distribution server 118 may include a remoting module 144 that is executed on the distribution server 118 to process requests to cause the navigation application 128 of the client 104 to perform one or more control functions. The remoting module 144, when executed, processes the requests using the client state data 142(j) that corresponds to the client 104 that is to perform the control function. If the client 104 is permitted to perform the control function, the distribution server 118 causes the client 104, and specifically the navigation application 128, to perform the control function. In this way, the head end 112 provides an authoritative source for client state data 142(j) in the environment 100.

[0027] Although the set-top box 120 is shown separately from the display device 122, the set-top box 120 may be built into the display device 122 to form an integral unit. Furthermore, in other implementations, the client 104 may be embodied as other computers capable of executing the navigation application 128 and UI module 132, such as a broadcast-enabled computer, an information appliance, a laptop computer, a personal video recorder (PVR), or the like.

[0028] FIG. 2 is an illustration of an exemplary implementation 200 showing the distribution server 118, the client 104, the local device 134 and the remote device 138 of FIG. 1 in greater detail. The client 104 includes a processor 202 and memory 204. The UI module 132 and the navigation application 128 are illustrated as being executed on the

processor 202 and are storable in memory 204. The configurable UI 130 is illustrated as being stored in the memory 204.

[0029] The client 104 may include a remote network interface (remote interface) 206 that may be utilized to receive the content 116(n) of FIG. 1 that is broadcast over the network 106. For example, the remote interface 206 may be configured as a tuner that receives broadcast content from over the network 106, may be configured as a transmitter/receiver (transceiver) that is suitable for two-way communication over the network 106, and so on. The content 116(n) may be stored in the database 124 for later output by the client 104 and/or provide for immediate output of the content 116(n). The database 124 is illustrated as being included in the memory 204, but may also be included in a separate storage device. For example, the storage device for the database 124 may be configured as a hard disk drive and the memory 204 may be configured as RAM, both the memory 204 and the storage device may be configured as RAM, both the memory 204 and the storage device may be configured as removable memory, and so forth. The client 104 executes the navigation application 128 to retrieve the content 126(m) of FIG. 1 from the database 124 and output the content 126(m) through an output interface 208 for rendering on the display device 122. Thus, in this implementation, the client 104 is capable of operating as a PVR that stores and plays back the content 126(m) of FIG. 1 in a manner akin to a video cassette recorder.

[0030] The client 104 may be controlled by a user via inputs provided by an input device 210. The inputs are received by the client 104 from an input interface 212 over a local connection 214. The local connection 214 may be the same as or different from the local connection 136 that communicatively couples the local device 134 with the local

interface 222. The input interface 212, local connection 214 and input device 210 may be configured in a variety of ways. For example, the input interface 212 may be configured as a wireless port, such as an infrared (IR) or Bluetooth wireless port, for receiving wireless communications from input device 210, such as a remote control device, a handheld input device, or any other wireless device, such as a wireless keyboard. In alternate embodiments, the input interface 212 may use an RF communication link or other mode of transmission to communicate with client 104, such as a wired connection which may include a universal serial bus (USB) connection, and so on.

[0031] When output of content is requested, the navigation application 128 is executed on the processor 202 to obtain content, such as from content that is streamed from the distribution server 118 over the network, content 126(m) that is stored locally on the database 124, and so on. The navigation application 128 may also restore the content to the original encoded format as provided by the content provider 102 of FIG. 1. For example, content 116(n) of FIG. 1 may be compressed and then streamed from the distribution server 118 to the client 104. Therefore, when the navigation application 128 receives the content, the content may be decompressed for rendering by the display device 122.

[0032] The client 104 may also be controlled through use of a UI that is executed by a device. For example, the local device 134 may include a processor 216 and memory 218. The local device 134 includes a UI 220 that was provided by the client 104 through execution of the UI module 132 and the configurable UI 130. When executed on the processor 216, the UI 220 enables a user to initiate one or more control functions of the navigation application 128 on the client 104. The local device 134 is communicatively

coupled over the local connection 136 to the client using a local interface 222. The local device 134 may also include a navigation application 224 to provide for content interaction on the local device 134 itself. The navigation application 224, for instance, may be configured as a browser, a media player, and so on.

[0033] The remote device 138 may also include a processor 226 and memory 228. Like the local device 134, the remote device 138 includes a UI 230 that is illustrated as being executed on the processor 226 and is storable in memory 228. The remote device 138 may also include a navigation application 232 that may be similar to or different from the navigation applications 128, 224 of the client 104 and/or local device 134. When executed on the processor 226, the UI 230 enables a user to initiate one or more control functions of the navigation application 128 on the client 104. By interacting with the UI 230 on the remote device 138, the user of the remote device 138 is able to initiate control functions of the client 104 from over the network 106. Communication between the remote device 138 and the client 104 may also be provided using the remote interface 206.

[0034] By initiating the control functions by the UIs 220, 230, the user can request a variety of control functions. Examples of control functions include recordation of content 116(n) of FIG. 1 that is streamed over the network 106 to the client 104, navigation through content, such as to fast forward, rewind or pause the output of the content, and so on. The control functions, for example, may provide non-linear playback of the content (i.e., time shift the playback of the content) such as pause, rewind, fast forward, slow motion playback, and the like. For instance, during a pause, the client 104 may continue to record the content in the database 124. The client 104, through

execution of the navigation application 128, may then playback the content 126(m) of FIG. 1 from the database 124, starting at the point in time the content 126(m) was paused, while continuing to record the currently-broadcast content 126(m) in the database 124.

[0035] The distribution server 118 included in the head end 112 of FIG. 1 also includes a processor 234 and memory 236. The remoting module 144 is illustrated as being executed on the processor 234 and is storable in memory 236. Client state data 142(j) is illustrated as being stored in memory 236. The database 114 is illustrated as included in the memory 236, but may also be included in a separate storage device. For example, the storage device for the database 114, like the database 124 of the client 104, may be configured as a hard disk drive and the memory 236 may be configured as RAM, both the memory 236 and the storage device may be configured as RAM, both the memory 236 and the storage device may be configured as removable memory, and so forth. Thus, the distribution server 118 may locally access the client state data 142(j).

[0036] The distribution server 118 executes the remoting module 144 on the processor 234 to process requests to initiate control functions of the navigation application 128 on the client 104. The requests may originate from the UIs 220, 230 that are executed on the respective local and remote devices 134, 138. The remoting module 144, when executed on the processor 234, processes the requests according to client state data 142(j) that is stored in the memory 236. For example, a plurality of client state data 142(j) may be stored in the memory 236 that corresponds to a plurality of clients that receive content from the head end 112 of FIG. 1. Each collection of client state data 142(j) for a particular client may include general operations data utilized by the particular client, such as rating limits, favorite channels, level of service provisioned, and so on. The network

106 connecting the head end 112 to the client 104 and the remote device 138 may have sufficient bandwidth and reliability such that the head end 112 is the authoritative source for this information. In other words, the head end 112 may be considered the primary source for the client state data 142(j) for a particular client, even over the client 104 itself. For example, by storing the client state data 142(j) on the distribution server 118, a user may switch set-top boxes without transferring client state data between the set-top boxes.

[0037] Additionally, by storing client state data 142(j) for local access by the distribution server 118, low resource clients may be utilized to output content yet high-resource functionality may still be provided to devices which execute the UI to control the client 104. For example, a low resource client, such as a client having limited hardware and/or software resources, may be configured to decrypt and output content for rendering on a display device. The low resource client may also provide local storage of content, such as on a hard disk drive. A feature-rich remote device, such as a desktop computer having significant processing resources, may execute a UI that would not be able to be executed on the low resource client. The UI, for instance, may have a higher resolution (e.g. more dots per inch), support a greater number of colors for output, provide macros to initiate control functions (e.g. recording of all episodes of a particular television program), and so on. In this way, the UI may provide additional functionality to a user of the client. Examples of UIs configured for output by devices having different capabilities may be found in relation to FIGS. 4 and 5.

[0038] **Exemplary Procedures**

FIG. 3 is a flow diagram of a procedure 300 in an exemplary implementation in which the client 104 configures a UI for execution on the local device 134 of FIG. 2. At

block 302 a local connection is initiated between the local device 134 and the client 104. A variety of local connections may be initiated, such as wired and wireless connections. An example of a wired connection is an Ethernet connection, which may be utilized to connect a plurality of devices to each other locally over an Ethernet network. A variety of wireless connections are also contemplated, such as wireless connections that are compliant with the Infrared Data Association (IrDA) standard, the Shared Wireless Access Protocol (SWAP) standard developed by the HomeRF Working Group, IEEE 802.11b and 802.11g developed by the Wireless Ethernet Compatibility Alliance (WECA), Bluetooth, and so on.

[0039] At block 304, the client 104 detects the hardware and/or software capabilities of the local device 134 through execution of the UI module 132. For example, the client 104 may detect hardware capabilities such as processor speed, data bus bandwidth, data throughput speed of one or more buses of the local device 134, amount of memory space available on the local device 134, supported resolutions and color palettes of an included display device, fonts supported, availability of input devices (e.g., touch screens, cursor control devices, buttons, keyboards, and so on), availability of audio input/output devices, and so forth. The client 104 may also detect software capabilities of the local device 134, such as computer languages supported, availability of applications on the local device 134, (e.g., browsers, media players, and so on), drivers implemented on the local device 134, and so forth.

[0040] At block 306, the client 104 configures a configurable UI for communication to the local device 134 based on the detection. The client 104, for instance, may execute the UI module 132 to configure the configurable UI 130 of FIG. 2 based on the detected

capabilities of the local device 134. A variety of configurations may be provided, such as through different arrangements of functions and display characteristics of the UI. Further description of configuration of UIs may be found in relation to FIGS. 4 and 5. The client 104 may then communicate the configured UI 308 to the local device 134 over the local connection 136.

[0041] At block 310, the local device 134 executes the configured UI 308 to initiate control functions of the client 104. For example, the user may utilize an input device, such as a touch screen of a PDA, to initiate a control function by “pressing” a button of the configured UI 308. The configured UI 308 may then form a request that indicates the desired control function and communicate the request directly to the client 104 over the local connection 136. Thus, in this implementation, the control function may be provided directly to the client 104 without communicating the request over the network 106. Additional discussion of processing requests may be found in relation to FIG. 6.

[0042] Although the implementation of FIG. 3 described configuration of the UI by the client 104, the UI may be configured in a variety of ways. In another implementation, the configurable UI 130 of FIG. 2 is implemented as a self-contained configurable UI that includes software to configure the presentation and implementation of the UI when received and/or executed on a device. For instance, the configurable UI 130 of FIG. 2 may be received by the local device 134, and when executed on the local device, detects hardware and/or software capabilities of the local device 134 to execute the UI. In a further implementation, the UI is configured by a distribution server, an example of which is described in relation to FIG. 7.

[0043] **Exemplary User Interfaces**

UIs may be configured in a variety of ways based on the capabilities of the device that is to execute the UI. FIG. 4 is an illustration of an exemplary implementation showing a UI 300 that is configured for output by the client when the user is located close to the display device, such as the display device 122 of FIG. 1. As previously stated, because devices utilized to execute the UIs may have different capabilities, and because the distance of a user from a display device may change, UIs may be configured to address the different capabilities and distances. For example, a user located close to a display device may be able to perceive a greater amount of detail than if the user is located at a further distance from the display device. Additionally, input devices of the local and remote devices 134, 138 of FIG. 2 may provide different respective functionality. For example, a local device may be configured for pen-based navigation, and a remote device having a mouse and keyboard may be configured for data-intensive input and navigation. Therefore, different UIs may be provided which are configured for the different respective functionality of the input devices.

[0044] UI 400 is configured for use when the user is located close to the display device. The UI 400 includes a channel guide 402 that lists a plurality of channels 404(1)-404(6) that is available from a broadcast. The UI 400 also includes a detailed description 406 of one of the plurality of programs that may provide information related to the program, such as actor, short description of the plot of the program, and so on. The UI 400 also includes a menu bar 408 to enable a user to navigate between functionality utilizing a cursor control device, such as a mouse. The illustrated menu bar 408 has functionality including current program title 410, audio functionality 412 such as closed captioning

and stereo, recent channels 414, favorite channels 416, and tools 418, such as options for arrangement, font selection, and so on.

[0045] FIG. 5 is an illustration of an exemplary implementation showing a UI 500 that is configured for output by the client when the user is located at a greater distance from the display device than the UI 400 of FIG. 4. The UI 500 may include a portion of the functionality provided in the UI 500 of FIG. 5. For example, the UI 500 may also include a channel guide 502 that lists a plurality of channels 404(1)-404(4) that is available from a broadcast. However, the channel guide 502 is configured to be viewed from a greater distance than the channel guide 402 of FIG. 4 and to be output by a display device having a lower resolution than the display device for outputting the UI 400 of FIG. 4. Therefore, the channel guide 502 includes fewer channels and programs in an enlarged format than the channel guide 402 of FIG. 4.

[0046] Additionally, the arrangement of functionality of the UI 500 may be different from the UI 400 of FIG. 4. UI 500, for example, may also include the detailed description 406 but have the detailed description 406 arranged below the channel guide 502, as opposed to above the channel guide 402 in the UI 400 of FIG. 4. Further, the UI 500 does not include the menu bar 408 of the UI 400 of FIG. 4. For example, the UI 500 may be provided at a lower resolution, i.e. fewer dots-per-inch, than the UI 400 of FIG. 4 so that the UI 500 is viewable from a greater distance. Therefore, although the UI 400 of FIG. 4 includes functionality that is not included in the UI 500 of FIG. 5, the UI 500 is viewable from a greater distance. Although arrangement of functions, display of functions and resolution have been described as differences between the UIs 400, 500 of FIGS. 4 and 5, respectively, a variety of other differences are also contemplated. For

example, the color scheme of the UI 500 may be selected such that the UI 500 is viewable from a greater distance, such as through selection of colors having greater contrast than those utilized in the UI 400 of FIG. 4. Further, the visibility and typeface of the functions may also be altered for each UI to provide an optimized viewing experience. For instance, the display device for outputting the UI 400 of FIG. 4 may support a greater number of colors than the display device for outputting the UI 500 of FIG. 5.

[0047] FIG. 6 is a flow diagram depicting a procedure 600 in an exemplary implementation in which the configured UI 308 of FIG. 3 is employed by a user to initiate a control function of the client 104. At block 602, the local device 134 forms and communicates a request 604 to initiate a control function of the client 104. For example, the configured UI 308, when executed on the local device 134, may provide one or more techniques for initiating the control function. A wireless phone, for instance, may execute the configured UI 308 that includes a voice menu to enable a user to navigate through a listing of available control functions, and initiate one or more of the control functions. A request is then formed for initiation of the desired control function and communicated to the client 104, such as over the local connection 136.

[0048] At block 606, the client 104 communicates the request 604 to the distribution server 118. In an implementation, the client 104 executes the UI module 132 of FIG. 2 to package the request 604 for forwarding over the network 106 to the distribution server 118. In another implementation, the request 604 is formatted for routing to the distribution server 118 by the local device 134 without packaging by the client 104.

[0049] At block 608, the distribution server 118 processes the request using the client state data 142(j). The remoting module 144, for instance, may be executed on the distribution server 118 to process the request 604 according to client state data 142(j) that is locally available to the distribution server 118, such as stored in a hard drive, RAID array, and so forth. The request 604 is processed according to client state data 142(j) that is specific to the client 104.

[0050] The request 604 may include a unique client identifier such that the remoting module 144 may locate client state data 142(j) that corresponds to the particular client 104. Once the client state data 142(j) is located, the remoting module 144, when executed, determines whether performance of the control function is permissible on the client 104. For example, the request 604 may be configured to initiate a control function on the client 104 to record content, e.g. a television program, and the distribution server 118, through examination of the client state data 142(j), may determine whether the client 104 is permitted to record the content. In another example, the request 604 is configured to tune the client 104 to a desired television channel, and the distribution server 118 may then determine whether the client 104 has a subscription to that particular television channel.

[0051] At block 610, the distribution server 118 forms and communicates a command 612 to perform the control function to the client 104. Continuing with the previous example, the remoting module 144, when executed, may determine that the client 104 is permitted to perform the control function and therefore form the command 612 for communication to the client 104. If the client 104 is not permitted to perform the control function, the remoting module 144 may form a response for communication to the local

device 134 to indicate that the control function is not permitted. In another implementation, an extra interface may be designed on the client 104 to “listen” to the head end for updates. For example, the UI module 132, when executed on the client 104, may periodically log onto the network 106 to receive and/or request commands provided by the head end 112 to perform one or more control functions.

[0052] At block 614, the client 104 performs the control function. For example, the client 104 may receive the command 612 from the distribution sever 118. The command may be encrypted utilizing a variety of encryption techniques to protect the command from being impersonated by an attacker. The navigation application 128 and/or the UI module 132, when executed, may decrypt the command 612 to determine which control function is to be performed. The navigation application 128 may then perform the control function indicated. The navigation application 128, for instance, may record a particular television program at a particular time, may change which broadcast channel is output by the client 104 to the display device 122 of FIG. 2, tune to a favorite channel, and so on. Although the implementations discussed in relation to FIGS. 3 and 6 describe the configuration and execution of a UI on a local device 134, the remote device 138 may also provide similar functionality, as is described in greater detail in the following implementation.

[0053] FIG. 7 is a flow diagram depicting a procedure 700 in an exemplary implementation in which a UI is configured for and executed by the remote device 138 of FIG. 2 to initiate one or more control functions of the client 104. At block 702, a remote device initiates a remote connection with the distribution server. For instance, the remote device 138 of FIG. 2 may initiate a remote connection over the network 106 with the

distribution server 188 of FIG. 2. The network 106 may assume a variety of configurations, such as a wide area network (WAN), e.g. the Internet, and so forth. A WAN may be considered as a geographically dispersed network, which includes a broader telecommunication structure than a local area network (LAN). A WAN, for instance, may be privately owned, leased or rented, but in most instances includes at least a portion of a public (e.g., shared user) network.

[0054] At block 704, the remote device forms a query for a UI from the distribution server that is suitable for initiating control functions of a client. The remote device, for instance, may form the query for communication over the network which identifies a desired client. In an implementation, the query includes the network address of the client. In another implementation, the distribution server includes a locally stored listing of client locations. In a further implementation, the distribution server queries the network 106 of FIG. 2 to locate the client. For instance, the remoting module 144, when executed on the distribution server 118, may act as a browser for locating desired network resources, which in this instance includes the client 104 of FIG. 2.

[0055] At block 706, the distribution server obtains the configurable UI from the client. For example, the distribution server may form an additional query that is communicated to the client over a network. The distribution server may then receive a response to the query that includes the configurable UI.

[0056] At block 708, the distribution server detects the hardware and/or software capabilities of the remote device over the remote connection. Like the client as previously described, the distribution server may detect processing speeds, available software, and so on, of the remote device 138 which indicate the capabilities of the

remote device to execute the UI. At block 710, the distribution server configures the configurable UI based on the detect capabilities to form a UI that is configured for the remote device. The distribution server, for instance, may arrange functions of the configurable UI, select functions based on available input devices, and so on. In another implementation, the UI may be configured based on a likely distance at which the user is positioned from the remote device when viewing the UI. For example, as previously described, device may be configured to be viewed at different distances, such as a television in a living room as opposed to a PDA that is held in the hand of the user. By configuring the configurable UI based on the likely distance of the user from the device, the configured UI may be optimized for viewing by the user.

[0057] At block 712, the distribution server stores client state data that includes an indication of the remote device and the client affiliation. The distribution server, for instance, may store data that includes an identifier of the remote device that received the configured UI and a corresponding indication of client affiliation for that particular UI. Therefore, when the distribution server receives a request from the remote device, the distribution server may readily identify a destination for the request, i.e. a particular client. Although configuration of the configurable UI is described in this implementation as being performed by the distribution server, the configurable UI may be configured in a variety of ways. For example, the configurable UI may be configured by the client as described in relation to FIG. 3, configured automatically by the configurable UI itself, and so on.

[0058] At block 714, the remote device receives and executes the configured UI, such as by executing the UI 230 on the processor 226 of FIG. 2. At block 716, a user selects a

control function by interacting with the UI. For example, a user may utilize an input device of the remote device, such as a keyboard, mouse, microphone to receive a voice command, and so forth, to select one of a plurality of control functions for initiation. At block 718, the UI forms and communicates a request for the control function to the distribution server. Therefore, in this implementation, a request initiated by a remote device 138 is communicated directly to the distribution server 118 over the network 106 without first being communicated through the client 104.

[0059] At decision block 720 a determination is made as to whether the client is authorized to perform the requested control function. As previously described, the determination may be made by the distribution server by processing the request utilizing client state data of the client that is to perform the control function. If the client is not authorized to perform the control function, the procedure 700 ends at block 722 by sending a failure message to the remote device that indicates that the client is not authorized to perform the requested control function.

[0060] If the client is authorized to perform the control function according to the client state data, then at block 724 the distribution server translates the request to form a command for communication to the client to perform the control function. For example, the request for initiating the control function that was formed by the remote device may not be compatible with the client. Therefore, the distribution server may translate the request into a command that causes the client to perform the corresponding control function. Therefore, at block 726, the client may receive the command and perform the control function as indicated in the command.

[0061] Thus, an environment for configuration and utilization of UI is provided. In an implementation, the head end is configured to be the authoritative source for client state data in the environment as previously described. In addition, the head end may have the capacity to recognize capabilities of devices, such as a Web browser accessible over the Internet, a set-top box communicatively coupled to a standard definition television, a set-top box communicatively coupled to a high-definition television, a wireless phone, a PDA, and so on, both by client affiliation and by the hardware and/or software capabilities.

[0062] Although the invention has been described in language specific to structural features and/or methodological acts, it is to be understood that the invention defined in the appended claims is not necessarily limited to the specific features or acts described. Rather, the specific features and acts are disclosed as exemplary forms of implementing the claimed invention.

CLAIMS

What is claimed is:

1. A method comprising:
receiving a query from a device, wherein:
the query is for a user interface (UI) from a client;
the UI, when executed by the device, is configured to form a request for execution of a control function by the client;
a processing of the request determines whether the execution by the client of the control function is permitted; and
the processing is performed at a head end by using client state data;
detecting at least one of hardware and software capabilities of the device to execute the UI; and
configuring the UI based on the at least one of hardware and software capabilities of the device.
2. The method as described in claim 1, further comprising:
communicating the request over a local area network from the device to the client;
and
forwarding the request by the client to the head end over a wide area network.
3. The method as described in claim 1, further comprising receiving the request by the head end from the device over a wide area network.

4. The method as described in claim 3, wherein the request is communicated from the device to the head end without communicating the request to the client.

5. The method as described in claim 1, wherein:
the device is communicatively coupled to the head end over a wide area network such that the request is received by the head end from the device; and
if the head end determines that client execution of the control function is permitted, the head end forms and communicates a command to the client to cause the client to perform the control function.

6. The method as described in claim 1, wherein the plurality of control functions, when executed on the client, control content interaction on the client.

7. The method as described in claim 1, wherein the configuring is selected from the group consisting of the following:

selecting one or more of a plurality of functions for inclusion in the UI;
arrangement of said functions for navigation by a user;
visibility of said functions for display;
resolution of said functions for display;
typeface of said functions for display; and
color scheme of said functions for display.

8. The method as described in claim 1, wherein the UI is configured based on

a likely distance at which a user is positioned from the device when viewing the user interface.

9. The method as described in claim 1, wherein the client state data is selected from the group consisting of:

ratings limits of content for output by the client;
favorite channels of the client;
one or more levels of content service for the client; and
general operations data for content output that is specific to the client.

10. The method as described in claim 1, wherein the hardware and software capabilities of the device are selected from the group consisting of:

one or more applications that are available for execution on the device;
device drivers;
available input devices;
available output devices;
processing speed;
amount of available memory;
resolution of a display device included on the device; and
available color palette for display of colors on a display device of the device.

11. The method as described in claim 1, wherein the detecting and the configuring are performed by at least one of the client and the head end.

12. The method as described in claim 1, further comprising communicating the UI to the device, wherein the UI, when executed on the device, performs the detecting and the configuring.

13. The method as described in claim 1, wherein the client executes a user interface module implemented as an HTTP server that performs the detecting, the configuring, and further provides the UI in an HTML format to the device.

14. One or more computer readable media comprising computer executable instructions that, when executed on a computer, direct the computer to perform the method of claim 1.

15. A method comprising:

- receiving a request on a client to perform a control function for content interaction, wherein the request is:
 - received from a local device communicatively coupled to the client over a local area network; and
 - formed by a user interface (UI) that is executed on the local device;
- forwarding the request by the client to a head end over a wide area network;
- processing the request on the head end to determine whether performance of the control function by the client is permitted, wherein the head end:
 - includes client state data for a plurality of said clients; and

the request is processed using the client state data that corresponds to the forwarding client; and

if performance of the control function is permitted, forming a command for communication over the wide area network to cause the forwarding client to perform the control function.

16. The method as described in claim 15, further comprising:
receiving a query on the forwarding client from the local device for the UI;
detecting at least one of hardware and software capabilities of the local device to execute the UI; and
configuring the UI based on the detecting.

17. The method as described in claim 16, further comprising communicating the UI to the local device, wherein the UI, when executed on the local device, performs the detecting and the configuring.

18. The method as described in claim 15, wherein the client state data for each said client is selected from the group consisting of:

ratings limits of content for output;
favorite channels;
one or more levels of content service; and
general operations data for content output.

19. One or more computer readable media comprising computer executable instructions that, when executed, perform the method of claim 15.

20. A method comprising:

receiving a request on a head end, wherein the request is:

- received from a remote device communicatively coupled to the head end over a wide area network;
- to execute a control function on a particular client for content interaction;
- and
- formed by a user interface (UI) that is executed on the remote device;

processing the request on the head end to determine whether execution of the control function by the particular client is permitted, wherein the head end:

- includes client state data for a plurality of said clients; and
- the request is processed using the client state data that corresponds to the particular client; and

if execution of the control function is permitted, forming a command for communication to the particular client to perform the control function.

21. The method as described in claim 20, further comprising:

- receiving a query for the UI;
- detecting at least one of hardware and software capabilities of the remote device to execute the UI; and
- configuring the UI based on the detecting.

22. The method as described in claim 21, further comprising communicating the UI to the remote device, wherein the UI, when executed on the remote device, performs the detecting and the configuring.

23. The method as described in claim 20, wherein the client state data for each said client is selected from the group consisting of:

- ratings limits of content for output;
- favorite channels;
- one or more levels of content service; and
- general operations data for content output.

24. One or more computer readable media comprising computer executable instructions that, when executed, perform the method of claim 20.

25. A client comprising:

- a processor; and

- memory configured to maintain:

- a navigation application that is executable on the processor to perform a plurality of control functions for controlling content interaction;

- a configurable user interface (UI) for forming a request to initiate one or more said control functions, wherein the request is for processing by a head end utilizing client state data; and

a UI module that is executable on the processor to:

detect capabilities of a device to execute the configurable UI; and
 configure the configurable UI based on the detection.

26. The client as described in claim 25, further comprising:

a local network interface for providing a local network connection with the device; and

a remote network interface for providing a remote network connection with the distribution server over a wide area network,

wherein the UI module is executable on the processor to:

receive a request from the local network interface that is provided by the configured UI when executed on the device, wherein the request is for initiating one or more said control functions;

forward the request to the remote network interface for communication to the distribution server; and

receive a response to the request at the remote network interface to initiate the one or more said control functions.

27. The client as described in claim 25, further comprising a remote network interface configured for communication with the distribution server over a wide area network, wherein:

the UI module is executable on the processor to receive a command at the network interface from the head end to perform one or more said control

functions; and

the command was provided in response to the request.

28. The client as described in claim 25, wherein the UI module is executable on the processor to configure the configurable UI in a manner selected from the group consisting of the following:

selecting one or more of a plurality of functions;
arrangement of said functions for navigation by a user;
visibility of said functions for display;
resolution of said functions for display;
typeface of said functions for display; and
color scheme of said functions for display.

29. The client as described in claim 25, wherein the client state data for the client is selected from the group consisting of:

ratings limits of content;
favorite channels;
one or more levels of content service; and
general operations data for content output.

30. The client as described in claim 25, wherein the hardware and software capabilities of the device are selected from the group consisting of:

one or more applications that are available for execution on the device;

device drivers;
available input devices;
available output devices;
processing speed;
amount of available memory;
resolution of a display device included on the device; and
available color palette for display of colors on a display device of the device.

31. The client as described in claim 25, wherein the UI module is implemented as an HTTP server that further provides the configured UI in an HTML format.

32. A system comprising:

a network;

a client communicatively coupled to the network, wherein the client:

is configured to perform a plurality of control functions for content interaction; and

includes a user interface (UI) module for providing a plurality of UIs such that each said UI is configured based on at least one of hardware and software capabilities of a device that executes the UI;

a local device communicatively coupled to the client over a local connection and including a first said UI for forming a first request to initiate a first said control function;

a remote device communicatively coupled to the network and including a second

said UI for forming a second request to initiate a second said control function; and

a head end communicatively coupled to the network and including:

client state data of the client; and

a remoting module for processing the first and second said requests utilizing the client state data to determine whether performance of the first and second said control functions is permitted on the client; and if so, forming a command for communication to the client over the network to cause the client to perform the first and second said control functions.

33. The system as described in claim 32, wherein the network is a wide area network and the local connection is provided by a local area network.

34. The system as described in claim 32, wherein:

the local device is communicatively coupled to the client over the local connection provided by a local area network; and

the first request is received by the client over the local area network and forwarded to the head end over the network, wherein the network is configured as a wide area network.

35. The system as described in claim 32, wherein:

the network is configured as a wide area network; and

the second request is received by the head end over the network.

36. The system as described in claim 32, wherein the UI module configures each said UI in a manner selected from the group consisting of the following:

- selecting one or more of a plurality of functions;
- arrangement of said functions for navigation by a user;
- visibility of said functions for display;
- resolution of said functions for display;
- typeface of said functions for display; and
- color scheme of said functions for display.

37. The system as described in claim 32, wherein the client state data of the client is selected from the group consisting of:

- ratings limits of content;
- favorite channels;
- one or more levels of content service; and
- general operations data for content output.

38. The system as described in claim 32, wherein the UI module is implemented as an HTTP server which, when executed, provides each said UI in an HTML format.

39. The system as described in claim 32, wherein the UI module configures each said UI based on a likely distance at which a user is positioned from the device when viewing the UI.

40. The system as described in claim 32, wherein the hardware and software capabilities are selected from the group consisting of:

one or more applications that are available for execution;

device drivers;

available input devices;

available output devices;

processing speed;

amount of available memory;

resolution of a display device included on the device; and

available color palette for display of colors on a display device of the device.

Smart & Biggar
Ottawa, Canada
Patent Agents

1/6

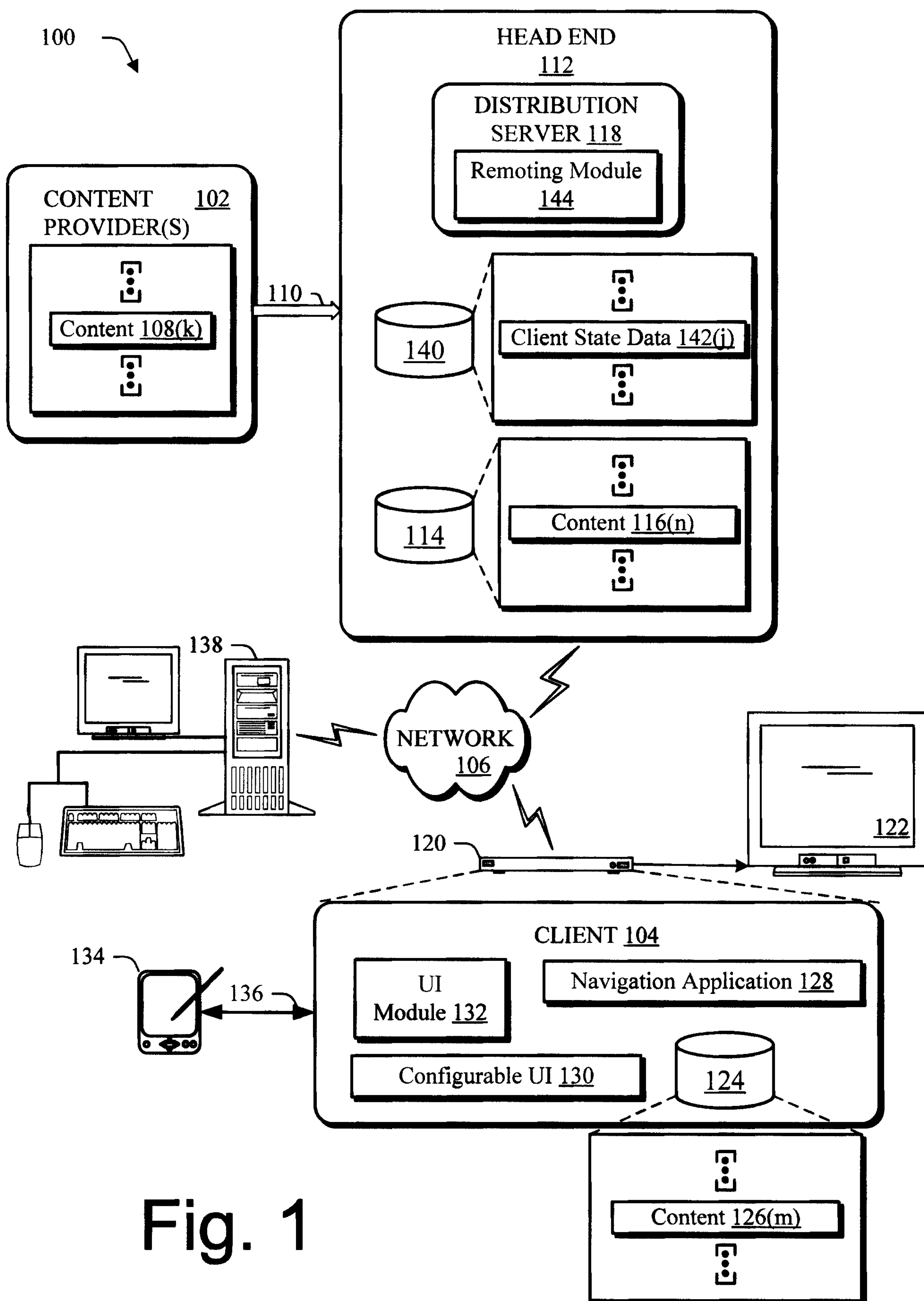


Fig. 1

2/6

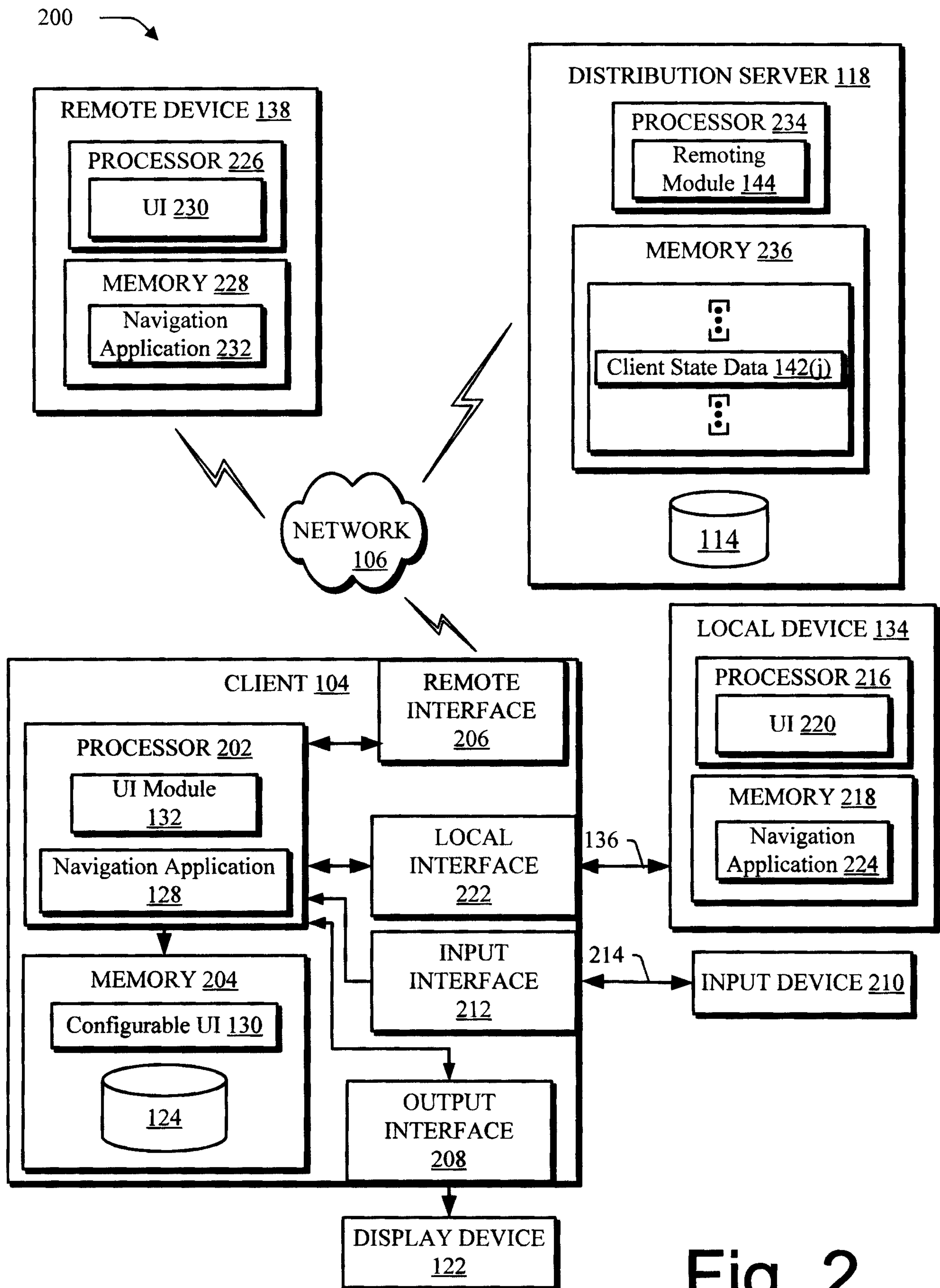


Fig. 2

3/6

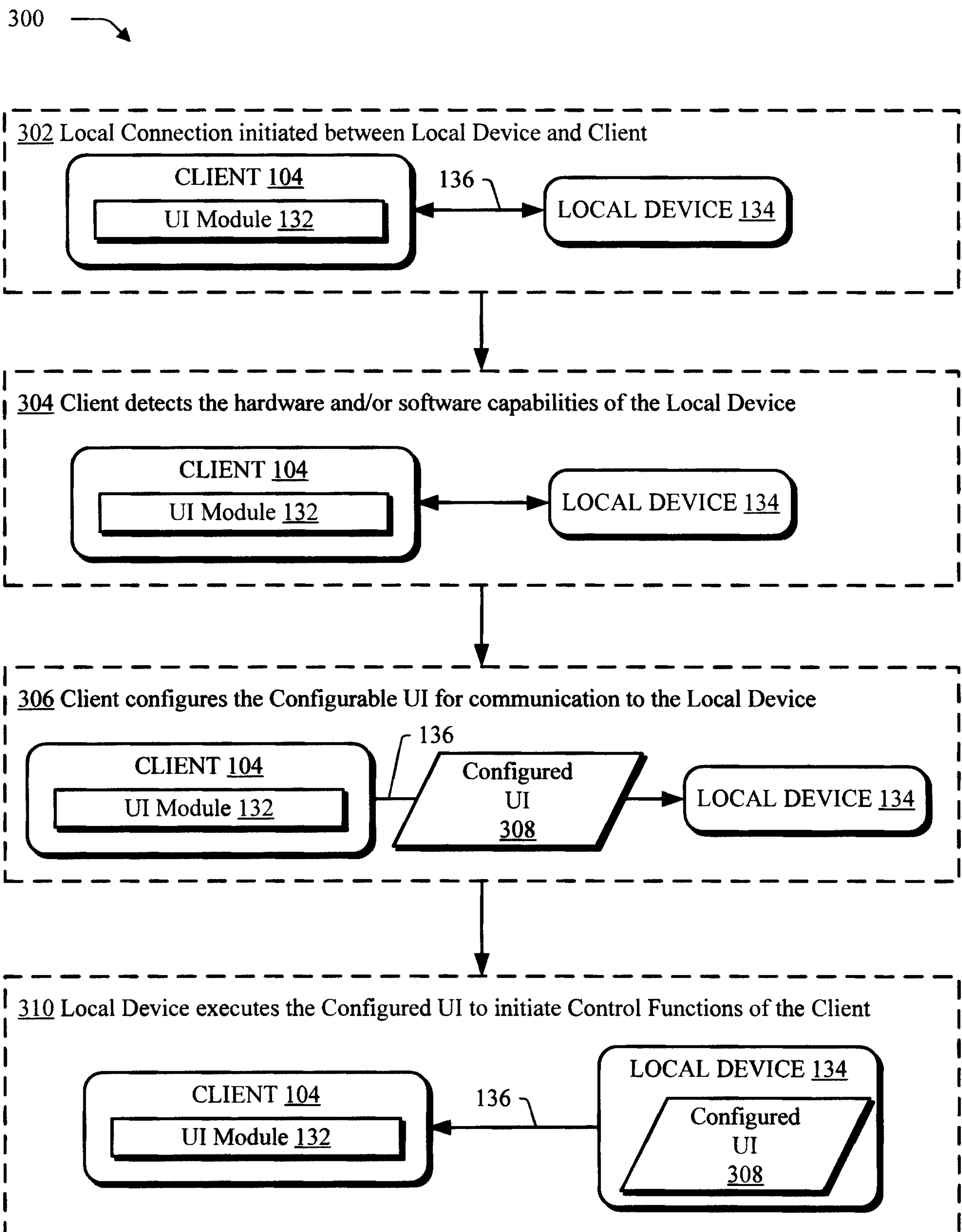


Fig. 3

4/6

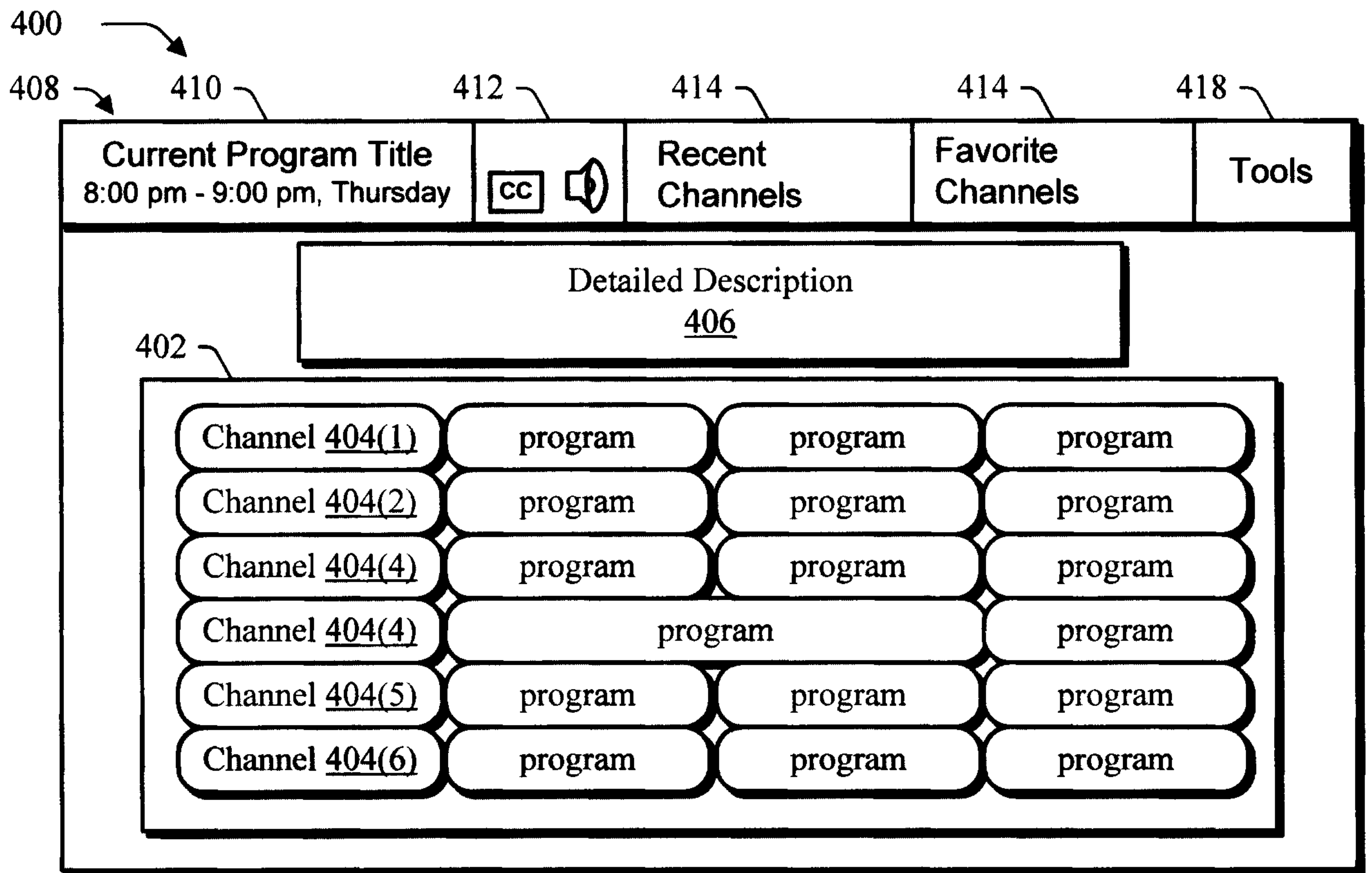


Fig. 4

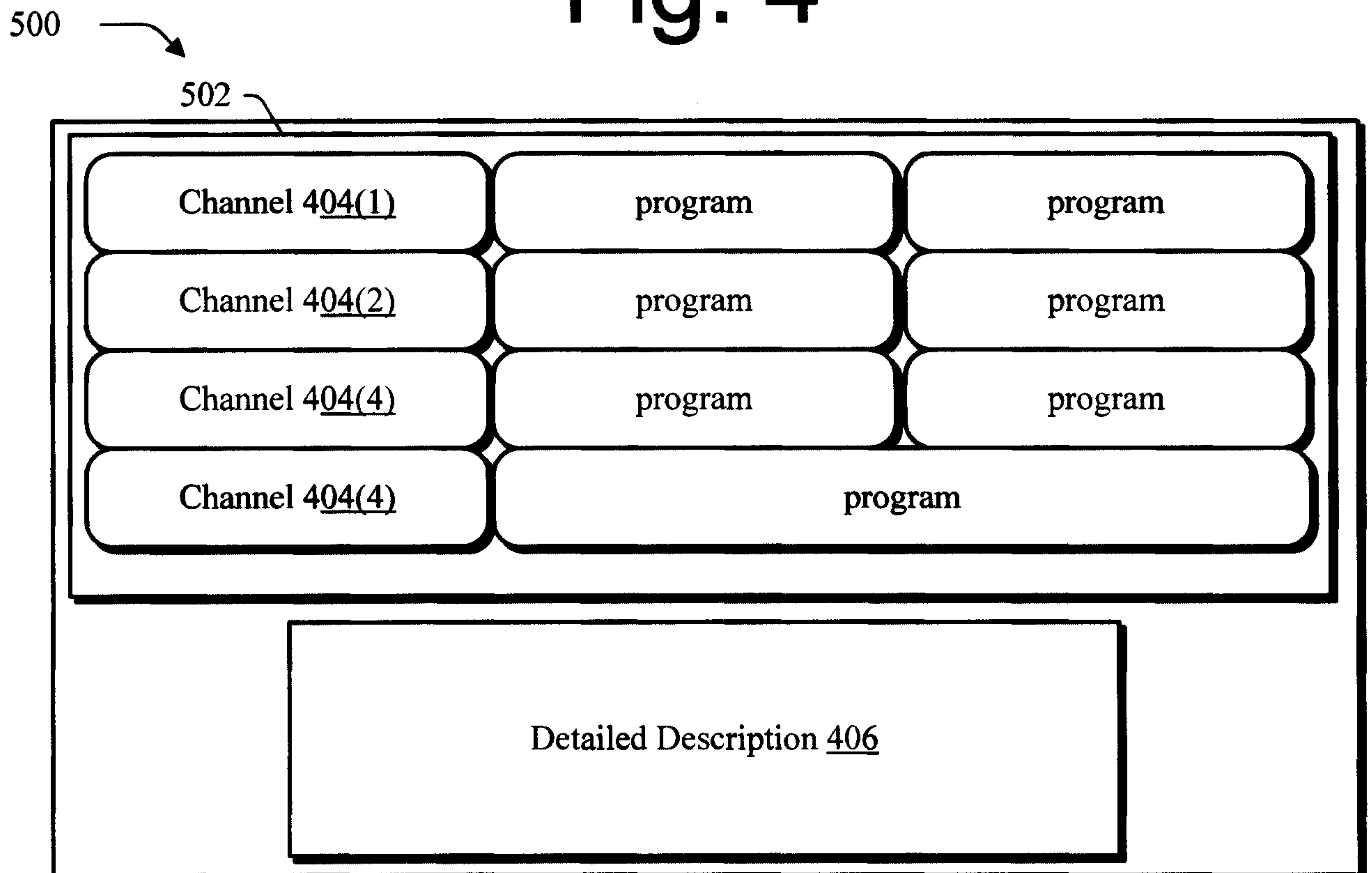


Fig. 5

5/6

600

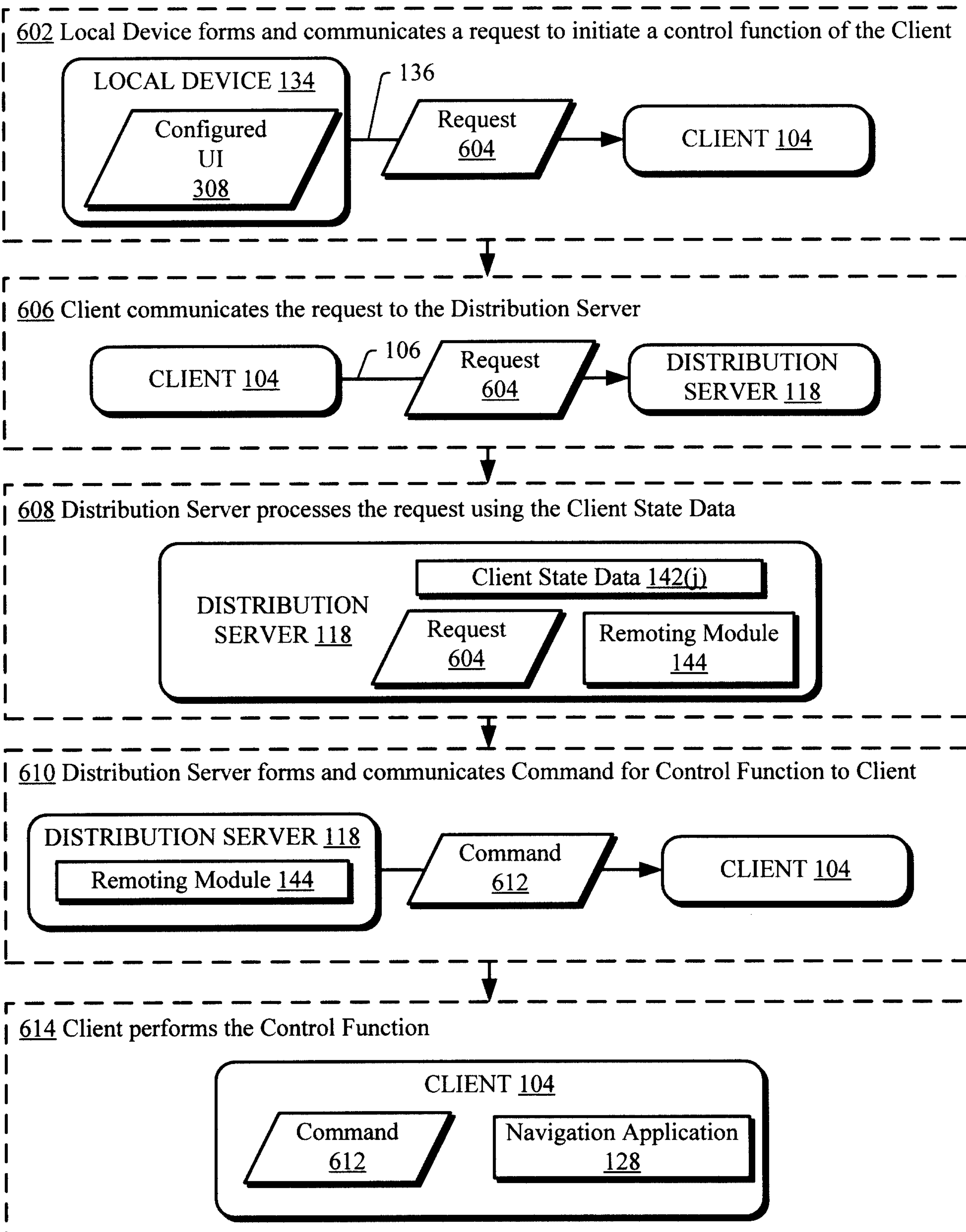


Fig. 6

6/6

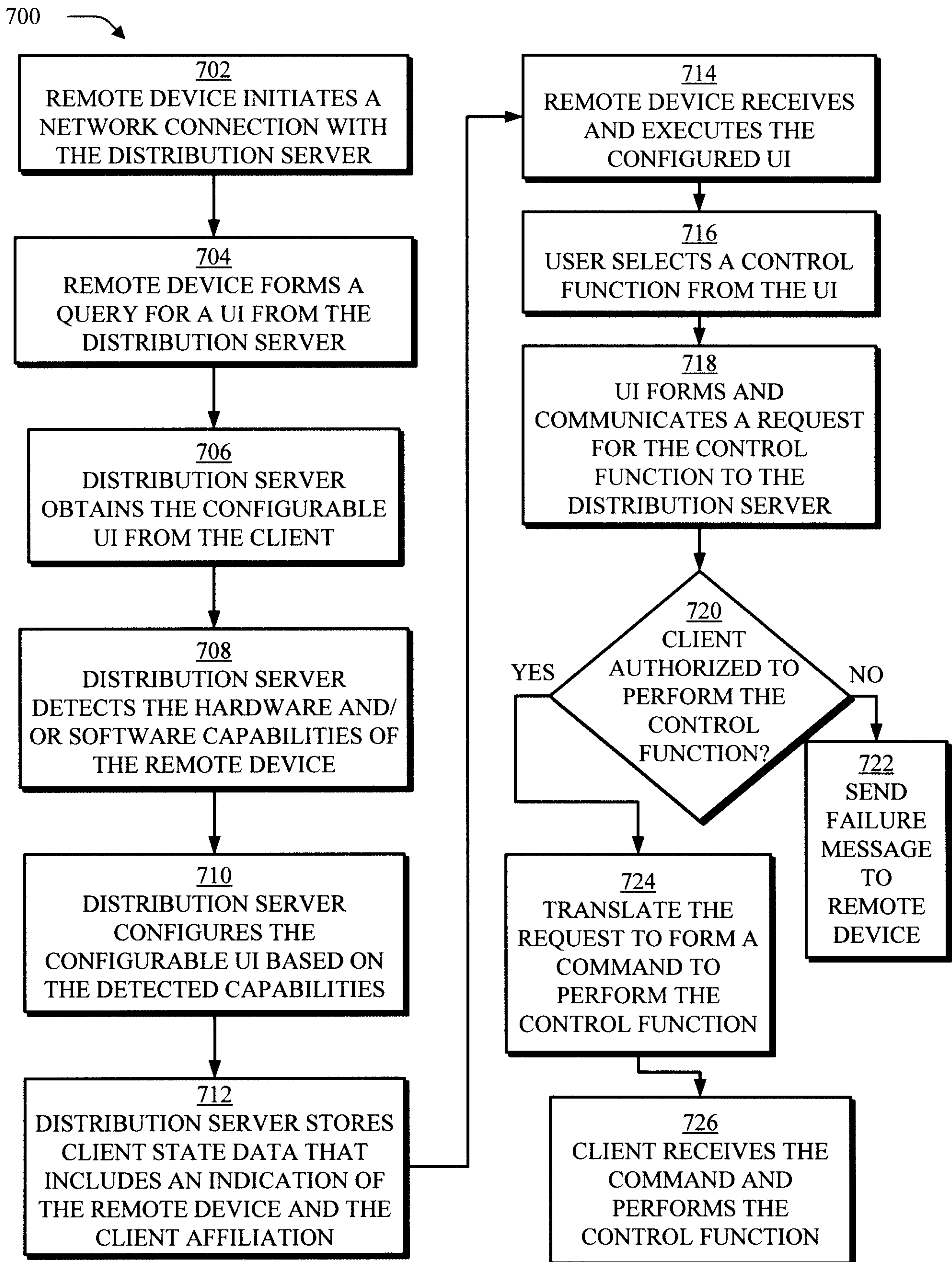
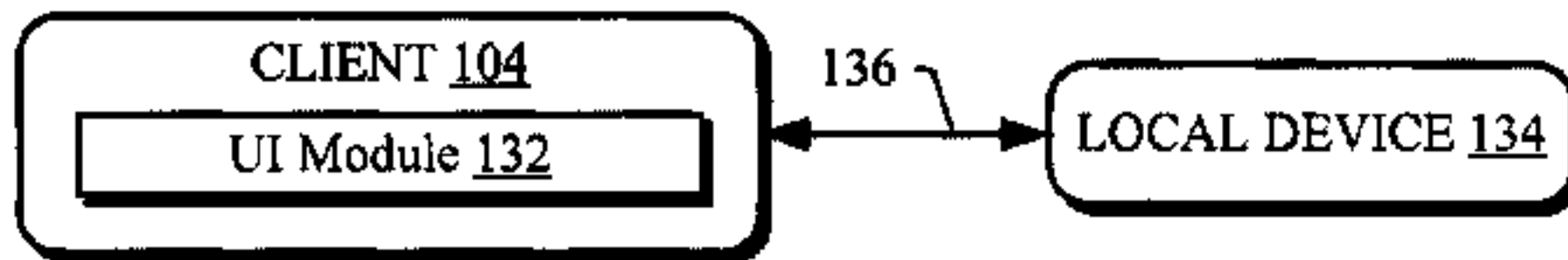
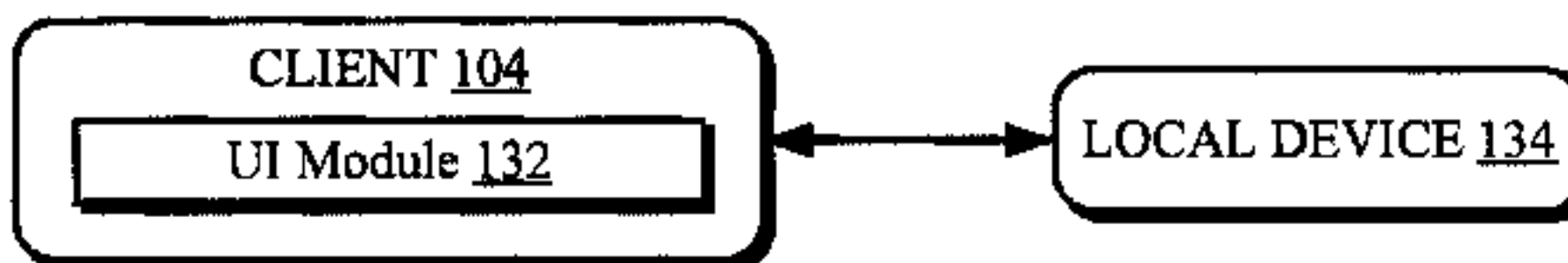


Fig. 7

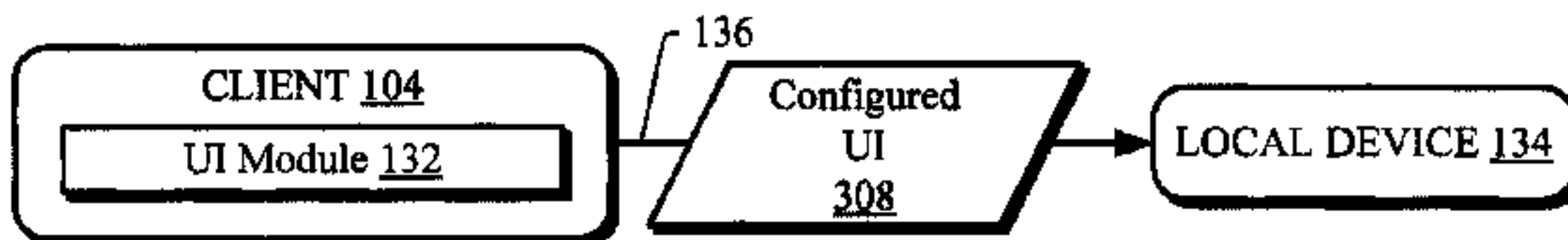
302 Local Connection initiated between Local Device and Client



304 Client detects the hardware and/or software capabilities of the Local Device



306 Client configures the Configurable UI for communication to the Local Device



310 Local Device executes the Configured UI to initiate Control Functions of the Client

