

(43) International Publication Date
3 June 2010 (03.06.2010)(10) International Publication Number
WO 2010/062048 A2(51) International Patent Classification:
G06F 15/16 (2006.01) *G06F 3/048* (2006.01)(21) International Application Number:
PCT/KR2009/006386(22) International Filing Date:
2 November 2009 (02.11.2009)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
10-2008-0108628 3 November 2008 (03.11.2008) KR
10-2009-0036057 24 April 2009 (24.04.2009) KR
10-2009-0096467 9 October 2009 (09.10.2009) KR(71) Applicant (for all designated States except US): **SAM-SUNG ELECTRONICS CO., LTD.** [KR/KR]; 416, Maetan-dong, Yeongtong-gu, Suwon-si, Gyeonggi-do 442-742 (KR).(72) Inventors: **RYU, Young-Sun**; #412-504, Kkachimaetul Lotte APT., Gumi-dong, Bundang-gu, Seongnam-si, Gyeonggi-do 463-738 (KR). **LEE, Kook-Heui**; #433-103, Cheongmyeongmaeul 4-danji Samsung Raemian APT., Yeongtong 1-dong, Yeongtong-gu, Suwon-si, Gyeonggi-do 443-738 (KR). **SONG, Jae-Yeon**; #B-805, Seongbo APT., Yeoksam-dong, Gangnam-gu, Seoul 135-080 (KR).(74) Agent: **LEE, Keon-Joo**; Mihwa Bldg. 110-2, Myongryun-dong 4-ga, Chongro-gu, Seoul 110-524 (KR).

(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, 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.

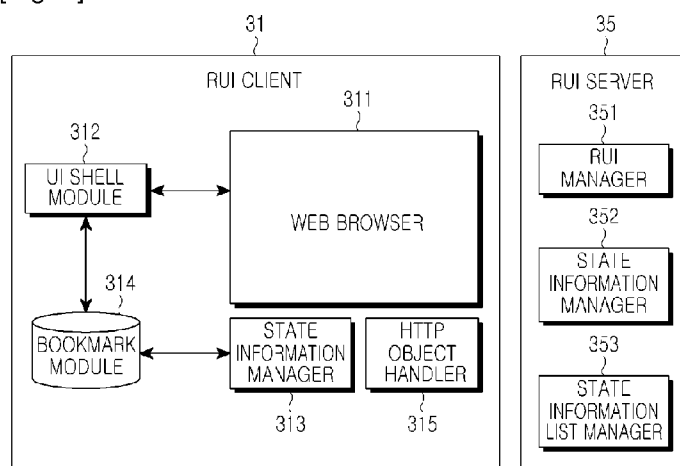
(84) Designated States (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT,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

Published:

— without international search report and to be republished upon receipt of that report (Rule 48.2(g))

(54) Title: METHOD AND APPARATUS FOR MANAGING STATE INFORMATION OF REMOTE USER INTERFACE

[Fig. 3]



(57) Abstract: A method for managing state information of a Remote User Interface (RUI) received from a server providing the RUI in a client. The method includes extracting the state information in response to a store request for the state information from a user; providing the state information to the server; receiving from the server location information indicating a route to a location where the state information is stored; and storing the location information in a bookmark module.

Description

Title of Invention: METHOD AND APPARATUS FOR MANAGING STATE INFORMATION OF REMOTE USER INTERFACE

Technical Field

- [1] The present invention relates generally to an apparatus and method for using Remote User Interfaces (RUIs), and more specifically, to an apparatus and method for managing state information of RUIs.

Background Art

- [2] Intensive research to improve home network technology is being conducted by many industrial standards organizations, such as Digital Living Network Alliance (DLNA), Home Audio-Video interoperability (HAVi), and Universal Plug and Play (UPnP). In such home networks, an RUI technology may be used to enable a device to control functions of other devices. In RUI technology based on a client-server architecture, an RUI client fetches a User Interface (UI) from an RUI server and enables a user to control the RUI server through the UR in the RUI client.
- [3] Consumer Electronics Association (CEA)-2014, a standard regarding RUI, defines a protocol and a framework for RUI in a UPnP network and the Internet. According to CEA-2014, a UPnP device (or an RUI server) provides a user interface to a Consumer Electronics-HyperText Markup Language (CE-HTML) type web page, and a user may remotely control an application of the UPnP device through the web page. CE-HTML is a special version of HTML, which is based on an eXtensible HyperText Markup Language (XHTML).
- [4] FIG. 1 illustrates a communication method between an RUI server and an RUI client based on the CEA-2014 standard. In order to use RUIs in a home network, discovery of an RUI server 20 should be performed first. CEA-2014 discovers the RUI server 20 through a UPnP discovery process because CEA-2014 is based on a UPnP network. An RUI client 10 and a UI Control Point (CP) 30 are shown in a separated manner in FIG. 1. However, when the UI CP 30 is embedded in the RUI client 10, the RUI client 10 will directly discover the RUI server 20. If the RUI client 10 has no CP 30, and a separate CP 30 intends to send a UI of the RUI server 20 to the RUI client 10, the CP 30 may discover the RUI client 10 through UPnP search.
- [5] After completing the discovery process, the RUI client 10 acquires an RUI page referring to a Uniform Resource Locator (URL) where the RUI page exists, which is included in a UPnP Device Description. More specifically, an XHTML browser of the RUI client 10 sends a request for an RUI page to a web server of the RUI server 20.

Since the RUI page is configured for control of an application of the RUI server 20, the RUI client 10 controls the application of the RUI server 20 using the RUI page. The conventional communication method between the RUI server 20 and the RUI client 10 is described in detail in documents of the CEA-2104 standard, so a description thereof is omitted herein.

- [6] A process of storing and restoring state information of RUIs in the conventional UPnP network is described with reference to FIG. 2. In step 201, the RUI server 20 provides RUIs to the RUI client 10. Upon receiving a store request for state information of the presently used RUI from a user, the RUI client 10 sends a store request for the state information to the RUI server 20, in step 202. In step 203, the RUI server 20 stores the state information of the RUI that is presently provided to the RUI client 10. In step 204, the RUI server 20 generates and stores a list of the stored RUI state information.

Disclosure of Invention

Technical Problem

- [7] As shown in FIG. 2, all state information of RUIs is managed by the RUI server 20. Therefore, in order to use the state information stored in the RUI server 20, the RUI client 10 must access a list of RUI state information, and a discovery process between the RUI client 10 and the RUI server 20 is required for the access. Therefore, the process of storing and restoring state information of RUIs in the conventional UPnP could be applied to Remote Desktop Protocol (RDP), eXtended Remote Technology (XRT), etc., which are remote protocol models in which all state information of RUIs is stored in the RUI server, but may not be applied to a remote protocol model having a stateless feature, such as Internet-based HyperText Transfer Protocol (HTTP).

Solution to Problem

- [8] The present invention addresses at least the above-mentioned problems and/or disadvantages and to provide at least the advantages described below. Accordingly, an aspect of the present invention provides a method and apparatus in which an RUI client can manage state information of RUIs.
- [9] Another aspect of the present invention provides a method and apparatus for migrating state information of RUIs of a first RUI client to a second RUI client different from the first RUI client.
- [10] According to one aspect of the present invention, there is provided a method for managing state information of a Remote User Interface (RUI) received from a server providing the RUI in a client. The method includes extracting the state information in response to a request to store the state information from a user; providing the state information to the server; receiving, from the server, location information indicating a

route to a location where the state information is stored; and storing the location information in a bookmark module that stores a user interface list preferred by the user.

[11] According to another aspect of the present invention, there is provided a method for managing state information of a Remote User Interface (RUI) received from a server providing the RUI in a first client having a control point. The method includes receiving a request for a migration of the state information through the control point; receiving a user selection of a second client to which the state information is to be migrated, through the control point; checking and storing, by the control point, an identifier of a presently running RUI; providing the state information to the server; sending, by the second client, a request for the state information to the server upon receiving a request for a migration of the state information from the control point; and receiving, by the second client, the RUI and the state information from the server, and providing the RUI and the state information.

[12] According to another aspect of the present invention, there is provided a method for managing state information of a Remote User Interface (RUI) received from a server providing the RUI in a first client and a second client with a control point. The method includes receiving, by the control point, a request for a migration of the state information from a user; receiving, by the control point, a user selection of a first client to which the state information is to be migrated; checking and storing, by the control point, an identifier of a presently running RUI; providing, by the first client, the state information to the server; receiving, by the second client, the RUI and the state information from the server upon receiving a request for a migration of the state information from the control point; and providing, by the second client, the RUI and the state information.

[13] According to another aspect of the present invention, there is provided a client terminal for managing state information of a Remote User Interface (RUI). The client terminal includes a web browser module for rendering an RUI provided from a server; a state information manager for sending a request to store the state information to the server, and restoring and providing state information provided from the server and location information indicating a route to a location where the state information is stored; a User Interface (UI) shell module for providing the RUI to a user and handling an input from the user; and a bookmark module for storing the state information provided from the state information manager along with a user interface list preferred by the user, and providing the state information to the UI shell module.

[14] According to another aspect of the present invention, there is provided a client terminal for managing state information of a Remote User Interface (RUI). The client terminal includes a control point for discovering in a home network a second client to which the state information can be migrated and in which an RUI is running, providing

the discovery results, receiving a user selection of a second client to which the state information can be migrated, checking an identifier of an RUI, state information of which is to be migrated, sending a request to store state information of the second client to an RUI server, and generating a request to receive state information from the selected second client; a state information manager for receiving state information of the selected second client from the server in response to the request from the control point; and a web browser module for rendering an RUI provided from the server.

[15] According to another aspect of the present invention, there is provided a method for managing state information of a Remote User Interface (RUI) in a server providing the RUI to a client. The method includes providing an RUI whose state information can be stored, to the client; receiving a request to store the state information of the RUI from the client; storing the state information and updating location information indicating a route to a location where the state information is stored; and providing at least one of the location information to the client.

[16] According to another aspect of the present invention, there is provided a server for providing a Remote User Interface (RUI). The server includes an RUI manager for storing and managing an RUI to be provided to a client; a state information manager for storing and managing state information of the RUI provided from the client; and a state information list manager for managing a list of location information indicating a route to a location where the state information is stored.

Advantageous Effects of Invention

[17] As is apparent from the foregoing description, exemplary embodiments of the present invention have the following effects.

[18] First, as RUI state information is stored in the bookmark module prepared in the RUI client, the RUI state information can be stored and restored in an internet-based iBox model.

[19] Second, as RUI state information is stored in RUI clients, the RUI state information can be migrated between different RUI clients.

[20] Third, if multiple RUI clients exist in the home network, a user can freely migrate an RUI among different devices simply by selecting a migration function of RUI state information without a separate process.

[21] Fourth, migration is possible not only by the Push scheme in which a device in operation migrates RUI state information to another device, but also by the Pull scheme in which a device desiring to receive RUI state information migrates a UI by detecting migrable UIR state information.

[22] Fifth, various additional effects can be provided to users by providing state information of an RUI that is presently running in the RUI client.

Brief Description of Drawings

- [23] The above and other aspects, features and advantages of certain exemplary embodiments of the present invention will be more apparent from the following description taken in conjunction with the accompanying drawings, in which:
- [24] FIG. 1 is a diagram illustrating a communication method between an RUI server and an RUI client based on the CEA-2014 standard;
- [25] FIG. 2 is a diagram illustrating a process of storing and restoring state information of RUIs in the conventional UPnP network;
- [26] FIG. 3 is a block diagram illustrating architecture of an RUI system according to an embodiment of the present invention;
- [27] FIG. 4 is a diagram illustrating a bookmark list stored in a bookmark module of an RUI client according to an embodiment of the present invention;
- [28] FIG. 5 is a flow diagram illustrating an operation process of an RUI state information management method according to an embodiment of the present invention;
- [29] FIG. 6 is a block diagram illustrating architecture of an RUI system according to an embodiment of the present invention;
- [30] FIG. 7 is a diagram illustrating an operation in a Push mode of an RUI system according to an embodiment of the present invention;
- [31] FIG. 8 is a diagram illustrating an operation in a Pull mode of an RUI system according to an embodiment of the present invention;
- [32] FIG. 9 is a flow diagram illustrating an operation process of a state information management method according to an embodiment of the present invention; and
- [33] FIG. 10 is a flow diagram illustrating an operation process of a state information management method according to an embodiment of the present invention.
- [34] Throughout the drawings, the same drawing reference numerals will be understood to refer to the same elements, features and structures.

Best Mode for Carrying out the Invention

- [35] The terms and words used in the following description and claims are not limited to bibliographical meanings, but are merely used to enable a clear and consistent understanding of the invention. Accordingly, it should be apparent to those skilled in the art that the following description of embodiments of the present invention are provided for illustrative purposes, and not for the purpose of limiting the invention, as defined by the appended claims and their equivalents.
- [36] In a Remote User Interface (RUI) service to which the present invention is applied, an RUI server provides RUIs, and an RUI client can control remote devices through the provided RUIs. The present invention provides a method and apparatus in which an RUI client manages state information of provided RUIs in the RUI service.

- [37] An RUI state information management method according to an embodiment of the present invention provides a method and apparatus in which an RUI client receives RUI state information provided from an RUI server, and stores and manages an RUI state information list in a bookmark module prepared in the RUI client.
- [38] The state information may include at least one of a value that is input to a web form, audio state information, video state information, cookie information, a user Identifier (ID), and a password.
- [39] Referring to FIG. 3, an RUI system according to an embodiment of the present invention includes an RUI client 31 and an RUI server 35.
- [40] The RUI client 31 includes a web browser module 311 for rendering RUIs provided from the RUI server 35, a UI shell module 312 for providing RUIs to a user and handling an input from the user, a state information manager 313 for managing state information, a bookmark module 314 for storing a state information list along with a list of RUIs preferred by the user, and an HTTP object handler 315.
- [41] RUIs provided from the RUI server 35 may be composed in an Extensible Hypertext Markup Language (XHTML) format, and the web browser module 311 renders the provided RUIs in the XHTML format and displays the provided RUIs on a display (not shown) of the RUI client 31.
- [42] The UI shell module 312 is a module for providing a list of RUIs to the user and handling an input from the user. For example, the UI shell module 312 displays a list of RUIs available in the system for the user and causes the user to select any one RUI included in the available RUI list. Alternatively, the UI shell module 312 reads an RUI state information list stored in the bookmark module 314, displays the RUI state information list for the user, and causes the user to select any state information included in the RUI state information list.
- [43] The state information manager 313 is a module for managing storage of state information requested by the user. Upon receipt of a store request for state information from the user, the state information manager 313 generates current state information for the RUI that is presently provided through the web browser module 311, and sends a store request for the generated state information to the RUI server 35. The state information manager 313 receives location information, such as a Uniform Resource Locator (URL), which indicates a route to a location where the state information is stored, from the RUI server 35, and stores the received location information in the bookmark module 314.
- [44] If any one in the RUI state information list stored in the bookmark module 314 is selected through the UI shell module 312, then the state information manager 313 provides a URL of the selected state information to the RUI server 35, thereby requesting restoration of the state information.

- [45] The bookmark module 314, connected to the UI shell module 312 and the state information manager 313, acquires URLs of RUI state information from the RUI server 35 through interaction with the state information manager 313, and stores the acquired URLs as a state information list. The bookmark module 314 provides the state information list to the UI shell module 312 through interaction with the UI shell module 312, and provides state information selected through the UI shell module 312 to the state information manager 313, so that the selected state information may be restored.
- [46] For example, the bookmark module 314 may store a bookmark such as the list illustrated in FIG. 4. The bookmark list stored in the bookmark module 314 may include a list 41 of RUIs preferred by the user and a state information list 43. The state information list 43 may be stored in a script 45 in an XML format. For example, the RUI list 41 may include RUIs preferred by the user, such as RUIs corresponding to the American Broadcasting Corporation (ABC), Cable News Network (CNN®)-Sports, and Naver.com, while the state information list 43 may include state information, such as state information of Movie 1 and station information of Movie 2.
- [47] Meanwhile, the RUI server 35 includes an RUI manager 351, a state information manager 352 for storing and managing the state information store-requested by the RUI client 31, and a state information list manager 353 for managing a list of the state information using a route to a location where the state information is stored.
- [48] The RUI manager 351 is a module for storing and managing RUIs that the RUI server 35 provides to the RUI client 31. For example, the RUI manager 351 can be a web server consisting of a software module for providing RUI pages. Each of the RUI pages being provided to the RUI client 31 can be identified by a Universally Unique Identifier (UUID). The RUI page is composed of XHTML content, which includes a plug-in object for managing state information of RUIs. Accordingly, the RUI client 31 includes the HTTP object handler 315, which may analyze and handle the plug-in object. For example, Table 1 below describes an object with which the HTTP object handler 315 handles storage and management of the state information.
- [49] Table 1

[Table 1]

[Table]

Property	
setURL()	It appoints a location where state information of RUI is to be stored.
isRestoring(): Boolean	It notifies whether restoring is in progress or not.
saveStates(): Number	An API that stores state information of RUI in the RUI server through HTTP POST.
getUIStateURL(): String	An API that gets from the RUI server a location where state information of RUI is stored.
addBookmark(): Boolean	An API that stores in the bookmark module a location where state information of RUI received from the RUI server is stored.

- [50] The state information manager 352 is a module for storing and managing, in a database, the state information store-requested by the RUI client 31. Upon receiving a store request for the state information from the RUI client 31, the state information manager 352 stores the state information in the database. Upon receiving a restore request for the state information from the RUI client 31, the state information manager 352 extracts the restore-requested state information from the database and provides the extracted state information.
- [51] The state information list manager 353 manages a route, such as aURL, of a location where the state information is stored. For example, if state information is stored upon request of the RUI client 31, the state information list manager 353 adds a URL indicating where the state information is stored to the state information list, and provides the added URL to the RUI client 31. Upon receipt of a restore request for state information corresponding to the URL from the RUI client 31, the state information list manager 353 provides a restore request for state information to the state information manager 313 along with the URL.
- [52] FIG. 5 illustrates an operation process of an RUI state information management method according to an embodiment of the present invention. In the following description with reference to FIG. 5, it is assumed that the RUI client 31 provides a list of at least one RUI page provided from the RUI server 35 through the UI shell module 312. Further, a description of the process of receiving an RUI list from the RUI server 35 in the RUI client 31 is omitted.
- [53] In step 501, a user inputs a request for an RUI page using the UI shell module 312 prepared in the RUI client 31. In response to the request, the RUI client 31 sends a

request for the RUI page to the UI server 35 in step 502.

[54] Upon receiving the RUI page request from the RUI client 31, the RUI server 35 transmits an RUI page composed by the web server 351 to the RUI client 31, which provides the RUI page to the user through the web browser module 311, in step 503.

[55] In step 504, as the user who uses the RUI page provided through the RUI client 31, inputs a store request for state information of the currently used RUI page, the UI shell module 312 provides the input from the user to the state information manager 313. In response to the request, the state information manager 313 checks state information of the current RUI page and makes a store request for the state information by providing the checked state information to the RUI server 35 in step 505.

[56] Upon receipt of the store request for the state information from the RUI client 31, the RUI server 35 stores in step 506 the state information in a database through the state information manager 352 and provides location information, such as a URL, which indicates a route to a location where the state information is stored, to the state information list manager 353. Then, the state information list manager 353 adds the URL of the state information to the state information list and transmits the added URL to the RUI client 31 in step 507.

[57] As the RUI server 35 transmits the added URL, the RUI client 31 receives the URL. In step 508, the state information manager 313 in the RUI client provides the received URL to the bookmark module 314, and the bookmark module 314 adds the state information to a bookmark list using the URL.

[58] When the user desires to use the state information stored in the RUI client 31, the user checks the bookmark list stored in the bookmark module 314 through the UI shell module 312.

[59] If information for selecting state information included in the bookmark list is received from the user through the UI shell module 312 in step 511, the bookmark module 314 provides a URL of the received state information to the state information manager 313, and the state information manager 313 provides the URL to the RUI server 35 and sends a request for state information corresponding to the URL to the RUI server 35 in step 512.

[60] As the RUI server 35 receives the URL and the request for state information corresponding to the URL, the state information manager 352, which is located within the RUI server 35, detects state information stored in the URL in step 513. In step 514, the RUI manager 351 provides the detected state information to the RUI client 31 along with the RUI page. Then the RUI client 31 browses an RUI in which the state information is reflected, in step 515.

[61] Meanwhile, in the RUI state information management method according to an embodiment of the present invention, as the RUI client manages the RUI state information

list from the RUI server, it is also possible for a first RUI client to migrate the RUI state information to a second RUI client. Therefore, an RUI state information management method according to a second embodiment of the present invention provides a method and apparatus for migrating RUI state information from the first RUI client to the second RUI client.

- [62] Referring to FIG. 6, an RUI system according to another embodiment of the present invention includes a first RUI client 61, a second RUI client 63, and an RUI server 65.
- [63] The first RUI client 61, like the RUI client 31 illustrated in FIG. 3, includes a web browser module 611, a UI shell module 612, and an HTTP object handler 615. Functions and operations of the web browser module 611, the UI shell module 612, and the HTTP object handler 615 are equal to the respective functions and operations of the web browser module 311, the UI shell module 312, and the HTTP object handler 315 provided in the RUI client 31.
- [64] The first RUI client 61 includes a device description module 616 for storing a list of the currently running applications, and a state information manager 617 for managing storage and migration of state information. The state information manager 617 is a module for managing storage of state information requested by the user. Upon receipt of a migration request for state information from the user, the state information manager 617 generates the current state information for an RUI or an application that is presently provided through the web browser module 611, and sends a store request for the generated state information to the RUI server 65. The state information manager 617 receives the state information and the RUI or application from the RUI server 65, and provides the received state information and RUI or application.
- [65] The second RUI client 63 includes a Control Point (CP) 631. The CP 631 discovers the first RUI client 61 and the RUI server 65, and provides a list of the searched first RUI client 61 and RUI server 65. In particular, the CP 631 provides a list of RUIs to be migrated, to the user, and handles migration of state information of the RUI included in the provided list.
- [66] The RUI server 65 includes an RUI manager 651, a state information manager 652 for storing and managing the state information store-requested by the first RUI client 61, and a state information list manager 653 for managing a list of the state information using a route to a location where the state information is stored.
- [67] The RUI server 65 is equal in structure to the RUI server 35. However, the state information list manager 653 can manage the state information list using a Universally Unique Identifier (UUID) of an RUI page. That is, upon receipt of the state information migration request from the CP 631, the state information list manager 653 in the RUI server 65 stores the UUID along with a URL in the state information list.
- [68] The RUI server 65, unlike the RUI server 35, receives a migration request for the

state information from the second RUI client 63 (or the first RUI client 61). At this point, the RUI server 65 receives a URL indicating where state information to be migrated is stored, and based on the received URL, the state information list manager 653 makes an instruction to transmit the state information stored in the URL to the second RUI client 63 (or the first RUI client 61).

- [69] Although the state information list manager 653 receives the URL from the second RUI client 63 and based on the received URL, requests state information corresponding to the URL in the embodiment of the present invention described with reference to FIG. 6, the present invention is not limited to this embodiment. For example, as an alternative to this example, when receiving a migration request for the state information, the state information list manager 653 may receive a UUID from the second RUI client 63, extract a URL matched to the UUID, and generate an instruction to transmit state information stored in the extracted URL to the second RUI client 63 (or the first RUI client 61).
- [70] While the CP 631 is provided in the second RUI client 63 as an example, the present invention is not limited to this example. The CP 631 may be provided as a separate device, or provided in the first RUI client 61.
- [71] The RUI system described with reference to FIG. 6 may be embodied in a Push mode or a Pull mode according to the place where the CP 631 is provided. An operation in the Push mode and the Pull mode of the RUI system according to the embodiment of the present invention described with reference to FIG. 6 will now be described in detail.
- [72] FIG. 7 illustrates an operation in the Push mode of the RUI system according to an embodiment of the present invention described with reference to the RUI system illustrated in FIG. 6. Referring to FIG. 7, the RUI system includes the first RUI client 61, the second RUI client 63, and the RUI server 65, and the CP 631 is provided in the first RUI client 61.
- [73] While the first RUI client 61 receives an RUI page identified by a UUID and provides the RUI page to the user, the first RUI client 61 may allow the user to input a migration request for state information of the currently used RUI page or application through the CP 631 provided in the first RUI client 61.
- [74] The CP 631 discovers a client to which state information can be migrated, provides the discovered client to the user, and allows the user to select the second RUI client 63, a target to which the state information is to be migrated.
- [75] In response to the migration request, the CP 631 stores a UUID of the currently used RUI page or application. Further, the first RUI client 61 provides the state information to the RUI server 65 and makes a store request to migrate the state information. In response to the store request, the RUI server 65 stores the state information in a

database, generates a route, such as a URL, to a location where the state information is stored, and adds the URL to a state information list. A UUID of the state information is stored together in the state information list.

- [76] Meanwhile, the CP 631 in the first RUI client 61 accesses the RUI server 65, and checks a URL matched to the UUID using the state information list. The CP 631 provides the URL to the second RUI client 63, which is selected by the user and to which the state information is to be migrated, and sends a restore request for the state information corresponding to the URL to the second RUI client 63.
- [77] The second RUI client 63 sends a request for state information corresponding to the URL to the RUI server 65, receives the state information and an application or an RUI page corresponding thereto from the RUI server 65, and displays the .
- [78] Upon receiving a request to delete the state information stored in the URL from the CP 631, the second RUI client 63 sends a delete request for the state information stored in the URL to the RUI server 65.
- [79] According to an embodiment of the present invention as described above with reference to FIG. 7, the CP 631 accesses the RUI server 65, checks a URL matched to the UUID, and migrates the state information using the checked URL. As an alternative to this embodiment, however, the CP 631 may provide the UUID to the second RUI client 63 when making a migration request for the state information, without performing an operation of accessing the RUI server 65 and checking the URL matched to the UUID, and the second RUI client 63 may make a request for state information matched to the UUID.
- [80] FIG. 8 illustrates an operation in the Pull mode of the RUI system according to an embodiment of the present invention described with reference to the RUI system illustrated in FIG. 6. Referring to FIG. 8, the RUI system includes the first RUI client 61, the second RUI client 63, and the RUI server 65, and the CP 631 is provided in the second RUI client 61.
- [81] The user inputs a migration request for state information of the currently used RUI page or application through the CP 631 provided in the second RUI client 63.
- [82] The CP 631 checks and provides a client to which the state information can be migrated, and further checks and provides the application or RUI page that is presently running in the first RUI client 63. The CP 631 allows the user to select the first RUI client 61 to which the state information is to be migrated.
- [83] In reply to the user's selection, the CP 631 checks and stores a UUID of the currently running application or RUI page, and sends a migration request for the state information of the currently running application or RUI page to the first RUI client 61.
- [84] In response to the migration request from the CP 631, the first RUI client 61 provides the state information of the currently running application or RUI page to the RUI

server 65, and makes a store request to migrate the state information. In response to the store request, the RUI server 65 stores the state information in a database, generates a route, such as a URL, to a location where the state information is stored, and adds the URL to a state information list. A UUID of the presently running application or RUI page is stored together in the state information list.

[85] Meanwhile, the CP 631 in the second RUI client 63 accesses the RUI server 65 and checks a URL matched to the UUID to restore the state information. The CP 631 sends a restore request for state information stored in the URL to the RUI server 65. In response to the restore request, the RUI server 65 provides the state information stored in the URL and an application or RUI page corresponding to the state information to the second RUI client 63, and the second RUI client 63 reflects the state information in the application or RUI page and displays the state information.

[86] Next, the CP 631 in the second RUI client 63 sends a request to delete the state information stored in the URL to the RUI server 65, and the RUI server 65 deletes the state information in response to the delete request.

[87] While, as in the Push mode, the CP 631 accesses the RUI server 65 and checks the URL matched to the UUID even in the Pull mode, the CP 631 may send in an alternative manner a migration request for state information using the UUID, without performing the operation of accessing the RUI server 65 and checking the URL matched to the UUID.

[88] FIG. 9 illustrates an operation process of a state information management method according to an embodiment of the present invention described with reference to the RUI system illustrated in FIG. 6.

[89] In step 801, the user selects an RUI provided by the CP 631, and makes a request for an RUI page corresponding thereto.

[90] In step 802, the CP 631 is connected to the first RUI client 61. In step 803, the first RUI client 61 sends a request for an RUI page identified by a UUID to the RUI server 65.

[91] As the first RUI client 61 makes the request for an RUI page, the RUI server 65 receives the RUI page request, and transmits the RUI page composed by the web server 651 to the first RUI client 61 in step 804. In response, the first RUI client 61 receives the RUI page and provides the RUI page to the user through the web browser module 611.

[92] In step 805, the user who uses the RUI page provided through the first RUI client 61, inputs a migration request for state information of the presently used RUI page or application through the CP 631.

[93] In response to the migration request for state information, the CP 631 accesses the device description module 616 in the first RUI client 61 and checks the presently

running application or RUI page in step 806 to thereby determine whether there is any application or RUI page that can be migrated. In steps 807 to 810, the CP 631 checks the UUID of the presently running application or RUI page, and stores state information of the application or RUI page selected by the user in the RUI server 65 using a state information storage mechanism. The RUI server 65 generates a state information list including location information, such as a URL, which indicates a route to a location where the state information is stored. The RUI server 65 stores the UUID together in the state information list.

[94] In step 811, the CP 631 determines the completed storage of the state information. In step 812, the CP 631 accesses the RUI server 65 and checks a URL matched to the UUID in the state information list.

[95] In step 813, the CP 631 provides the URL to the second client 63 to which the state information is to be migrated, and makes a restore request for the state information corresponding to the URL.

[96] In step 814, the second RUI client 63 sends a request for state information stored in the URL to the RUI server 65. In step 815 to 817, the second RUI client 63 receives the state information stored in the URL and an application or RUI page corresponding thereto from the RUI server 65, and displays the state information along with the application or the corresponding RUI page.

[97] In step 818, the second RUI client 63 provides the handling results of the state information to the CP 631. If the migration of the state information is successful, the CP 631 sends a delete request for state information stored in the URL to the second RUI client 63 in step 819, and the second RUI client 63 sends a delete request for the state information stored in the URL to the RUI server 65 in step 820. In step 821, the second RUI client 63 browses the application or RUI page provided from the RUI server 65.

[98] FIG. 10 illustrates an operation process of a state information management method according to an embodiment of the present invention. The state information management method according to the embodiment of the present invention described with reference to FIG. 10 is similar to the state information management method according to the embodiment described with reference to FIG. 9. Compared with the state information management method according to the embodiment described with reference to FIG. 9, in which the CP 631 migrates the state information using the URL in the state information list stored in the RUI server 65, the state information management method according to the embodiment described with reference to FIG. 10 enables the CP 631 to migrate the state information using a UUID.

[99] Accordingly, steps 901 to 911 of the state information management method according to FIG. 10 are the same as steps 801 to 811 of the state information management method according to FIG. 9.

- [100] In step 912 of the third embodiment, the CP 631 provides the UUID of the presently running application or RUI page to the second RUI client 63 to which the state information is to be migrated, without the process of acquiring the URL from the RUI server 65, and generates a restore request for state information corresponding to the UUID.
- [101] In step 913, the second RUI client 63 sends a request for state information corresponding to the UUID to the RUI server 65. In steps 914 to 916, the second RUI client 63 receives the state information corresponding to the UUID and an application or RUI page corresponding to the state information from the RUI server 65, and displays them. The RUI server 65 receives the UUID from the second RUI client 63, checks the state information list, and then checks a URL matched to the UUID. The RUI server 65 checks state information stored in the URL and provides the state information to the second RUI client 63.
- [102] In step 917, the second RUI client 63 provides the handling results of the state information to the CP 631. If the migration of the state information is successful, the CP 631 sends a request to delete state information corresponding to the UUID to the second RUI client 63 in step 918, and the second RUI client 63 sends a delete request for the state information corresponding to the URL to the RUI server 65 in step 919. In step 920, the second RUI client 63 browses the application or RUI page provided from the RUI server 65.
- [103] The RUI state information management method according to embodiments of the present invention can also be embodied as computer-readable codes on a computer-readable medium. A computer-readable medium is any data storage device that can store data, which can thereafter be read by a computer system. Examples of computer-readable mediums include, but are not limited to, computer-readable recording mediums such as Read-Only Memory (ROM), Random-Access Memory (RAM), CD-ROMs, magnetic tapes, floppy disks, and optical data storage devices. Computer-readable mediums also include carrier waves (such as data transmission through the Internet via wired or wireless transmission paths). The computer-readable medium can also be distributed over network-coupled computer systems so that the computer-readable code is stored and executed in a distributed fashion. Also, function programs, codes, and code segments for accomplishing the present invention can be easily construed as within the scope of the invention by programmers skilled in the art to which the present invention pertains.
- [104] While the present invention has been shown and described with reference to certain embodiments thereof, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the spirit and scope of the present invention as defined by the appended claims and their equivalents.

Claims

- [Claim 1] A method for managing state information of a Remote User Interface (RUI) received from a server providing the RUI in a client, comprising:
receiving, from a user, a request to store state information;
extracting the state information in response to the request;
providing the state information to the server;
receiving, from the server, location information indicating a route to a location where the state information is stored; and
storing the location information in a bookmark module.
- [Claim 2] The method of claim 1, further comprising:
receiving, from the user, a request for stored state information;
providing, in response to the request, a user interface list including the location information to the user;
receiving a user selection the location information, and transmitting the selected location information to the server;
receiving the stored state information corresponding to the location information, from the server; and
providing the received state information to the user.
- [Claim 3] The method of claim 1, wherein the state information includes at least one of a value that is input to a web form, audio state information, video state information, cookie information, a user Identifier (ID), and a password.
- [Claim 4] The method of claim 1, wherein the location information includes Uniform Resource Locator (URL) information of the route.
- [Claim 5] The method of claim 1, further comprising:
receiving a request for a migration of the state information from the user; and
providing the state information to be migrated to the user.
- [Claim 6] A method for managing state information of a Remote User Interface (RUI) received from a server providing the RUI in a first client having a control point, comprising:
receiving a request for a migration of the state information through the control point;
receiving a user selection of a second client to which the state information is to be migrated, through the control point;
checking and storing, by the control point, an identifier of a presently running RUI;

- providing the state information to the server;
sending, by the second client, a request for the state information to the server upon receiving the migration request for the state information from the control point; and
receiving, by the second client, the RUI and the state information from the server, and providing the RUI and the state information to the user.
- [Claim 7] The method of claim 6, further comprising sending a request to delete the state information to the server.
- [Claim 8] The method of claim 6, wherein the receiving the user selection of the second client comprises:
discovering, by the control point, at least one second client to which the state information is to be migrated;
providing the at least one second client to the user; and
receiving a user selection of any one of the discovered second clients.
- [Claim 9] The method of claim 6, further comprising:
accessing, by the control point, the server and checking location information indicating a route to a location where the state information is stored; and
transmitting, by the control point, the location information to the second client,
wherein the second client receives the RUI and the state information from the server using the location information.
- [Claim 10] The method of claim 6, wherein the control point generates a request for a migration of the state information by providing an identifier of the RUI to the second client.
- [Claim 11] A method for managing state information of a Remote User Interface (RUI) received from a server providing the RUI in a first client and a second client with a control point, comprising:
receiving, by the control point, a request for a migration of the state information from a user;
receiving, by the control point, a user selection of a first client to which the state information is to be migrated;
checking and storing, by the control point, an identifier of a presently running RUI;
providing, by the first client, the state information to the server;
receiving, by the second client, the RUI and the state information from the server upon receiving the migration request for the state information from the control point; and

- providing, by the second client, the RUI and the state information to the user.
- [Claim 12] The method of claim 11, further comprising sending, by the control point, a request to delete the state information to the server.
- [Claim 13] The method of claim 11, wherein the user selection of the first client comprises:
discovering at least one first client operating in a home network and an RUI running in the first client;
providing state information for the discovered at least one first client and RUI to the user; and
receiving a user selection of any one of the discovered first clients.
- [Claim 14] The method of claim 11, further comprising:
accessing, by the control point, the server and checking location information indicating a route to a location where the state information is stored; and
transmitting, by the control point, the location information to the second client,
wherein the second client receives the state information from the server using the location information.
- [Claim 15] The method of claim 11, wherein the control point generates a migration request for the state information by providing an identifier of the RUI to the second client,
wherein the second client receives the state information matched to the identifier of the RUI from the server.
- [Claim 16] A client terminal for managing state information of a Remote User Interface (RUI), comprising:
a web browser module for rendering an RUI provided from a server;
a state information manager for sending a request to store the state information to the server, and restoring and providing, to a user, state information provided from the server and location information indicating a route to a location where the state information is stored;
a User Interface (UI) shell module for providing the RUI to the user and handling an input from the user; and
a bookmark module for storing the state information provided from the state information manager along with a user interface list preferred by the user, and providing the state information to the UI shell module.
- [Claim 17] The client terminal of claim 16, further comprising a Hypertext Transfer Protocol (HTTP) object handler for handling data exchange

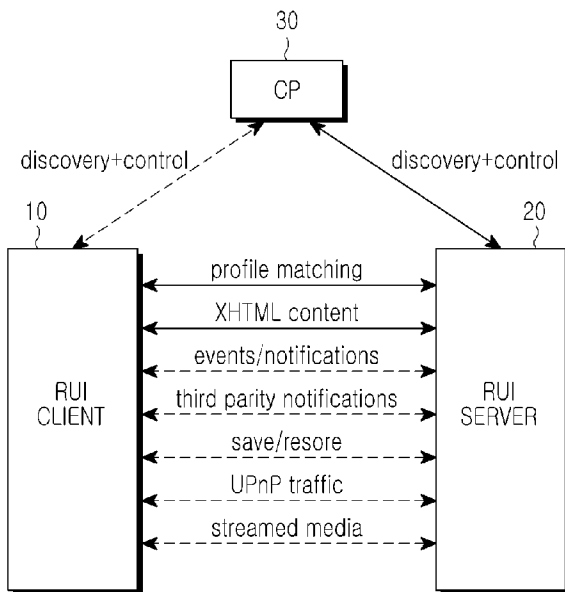
- with the server.
- [Claim 18] The client terminal of claim 16, wherein the state information includes at least one of a value that is input to a web form, audio state information, video state information, cookie information, a user Identifier (ID), and a password.
- [Claim 19] The client terminal of claim 16, wherein the location information includes Uniform Resource Locator (URL) information of the route.
- [Claim 20] The client terminal of claim 16, further comprising a control point for discovering, in a home network, a second client to which the state information can be migrated, providing results of the discovery to the user, receiving a user selection of a second client to which the state information is to be migrated, checking an identifier of an RUI, state information of which is to be migrated, and sending a migration request for the state information to the selected second client.
- [Claim 21] The client terminal of claim 20, wherein the control point sends a request for a migration of the state information using the location information.
- [Claim 22] The client terminal of claim 20, wherein the control point sends a request for a migration of the state information using an identifier of the RUI.
- [Claim 23] The client terminal of claim 20, wherein the control point sends a request to delete the state information to the server.
- [Claim 24] A client terminal for managing state information of a Remote User Interface (RUI), comprising:
a control point for discovering, in a home network, a second client to which the state information can be migrated and in which an RUI is running, providing results of the discovery to a user, causing the user to select a second client to which the state information can be migrated, checking an identifier of an RUI, state information of which is to be migrated, sending a request to store state information of the second client to an RUI server, and generating a request to receive state information from the selected second client;
a state information manager for receiving state information of the selected second client from the server in response to the request from the control point; and
a web browser module for rendering an RUI provided from the server.
- [Claim 25] The client terminal of claim 24, wherein the control point discovers state information of an RUI that is presently running in the second

- client, and provides results of the discovery of the state information along with the state information of the RUI.
- [Claim 26] A method for managing state information of a Remote User Interface (RUI) in a server providing the RUI to a client, comprising:
providing an RUI whose state information can be stored, to the client;
receiving a request to store the state information of the RUI from the client;
storing the state information and updating location information indicating a route to a location where the state information is stored; and
providing at least one element of the location information to the client.
- [Claim 27] The method of claim 26, further comprising:
receiving a user selection of at least one element of the location information, and receiving a request for state information corresponding to the selected element of the location information; and
providing the requested state information to the client.
- [Claim 28] The method of claim 26, further comprising:
receiving the state information from the client, and storing the received state information;
receiving a request for a migration of the stored state information; and
providing the stored state information to a second client.
- [Claim 29] The method of claim 28, further comprising sending a request to delete the state information from the control point to the server.
- [Claim 30] The method of claim 28, further comprising receiving a request for a migration of the state information using the location information.
- [Claim 31] The method of claim 28, further comprising receiving a request for a migration of the state information using an identifier of the RUI.
- [Claim 32] A server for providing a Remote User Interface (RUI), comprising:
an RUI manager for storing and managing an RUI to be provided to a client;
a state information manager for storing and managing state information of the RUI provided from the client; and
a state information list manager for managing a list of location information indicating a route to a location where the state information is stored.
- [Claim 33] The server of claim 32, wherein the state information manager receives the state information from the client, stores the received state information, receives a request for a migration of the state information, and provides the state information to a second client.

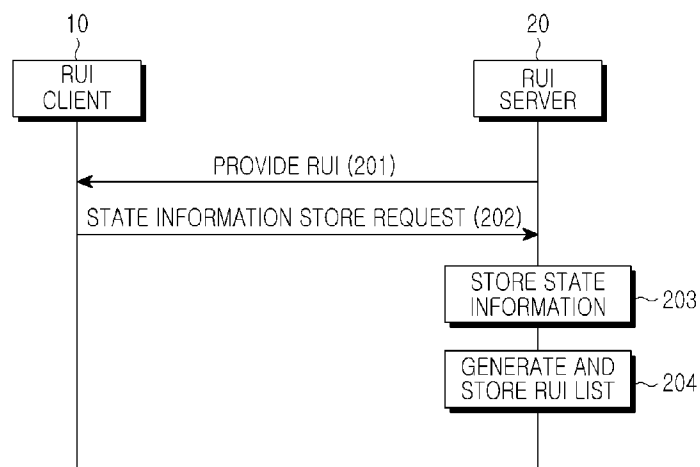
[Claim 34]

The server of claim 33, wherein the state information manager deletes the state information upon receiving a delete request for the state information from the client.

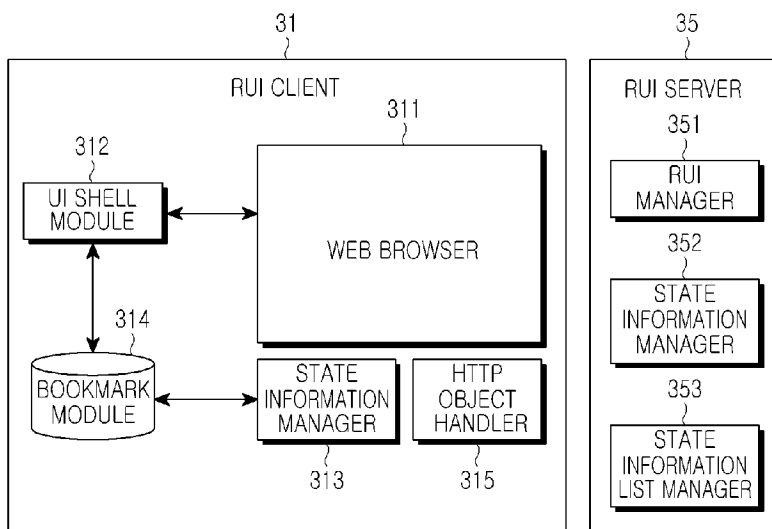
[Fig. 1]



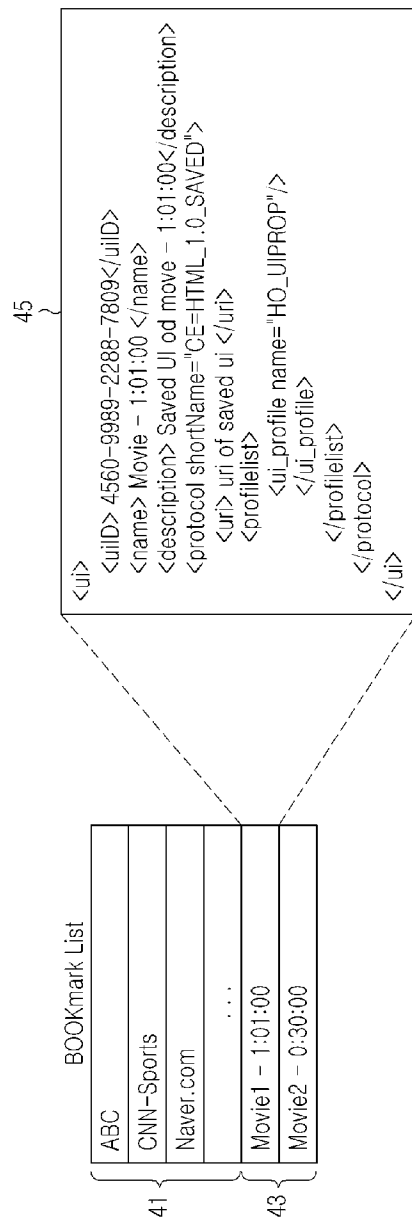
[Fig. 2]



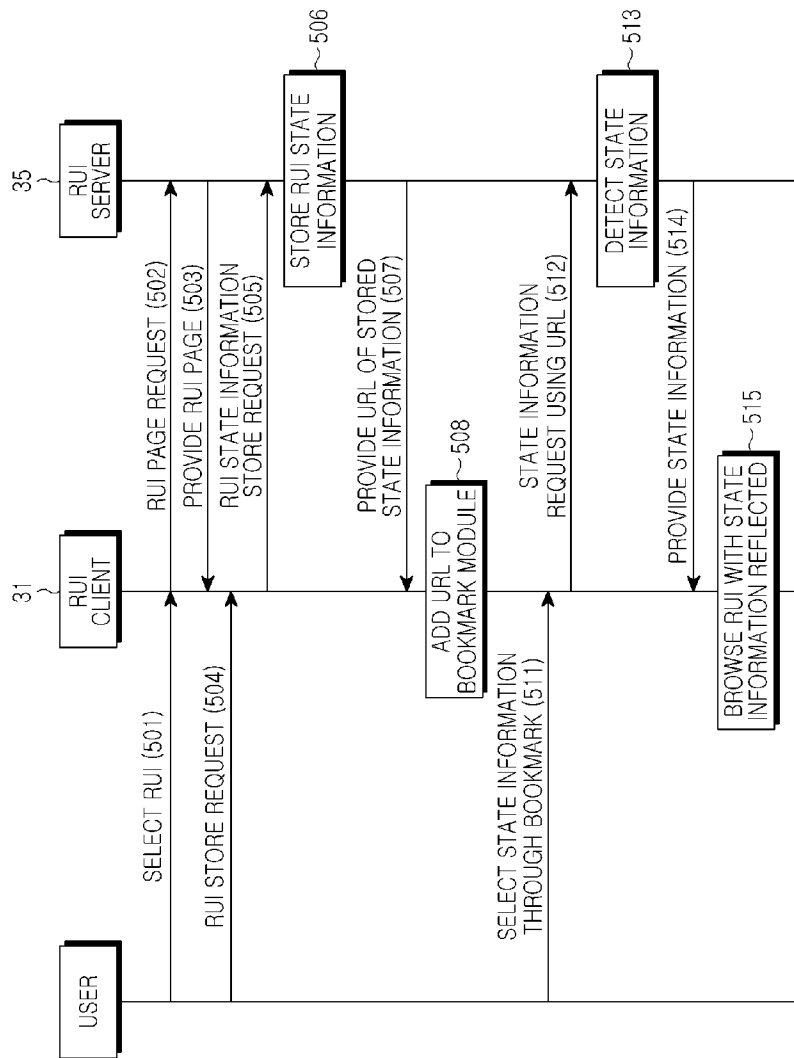
[Fig. 3]



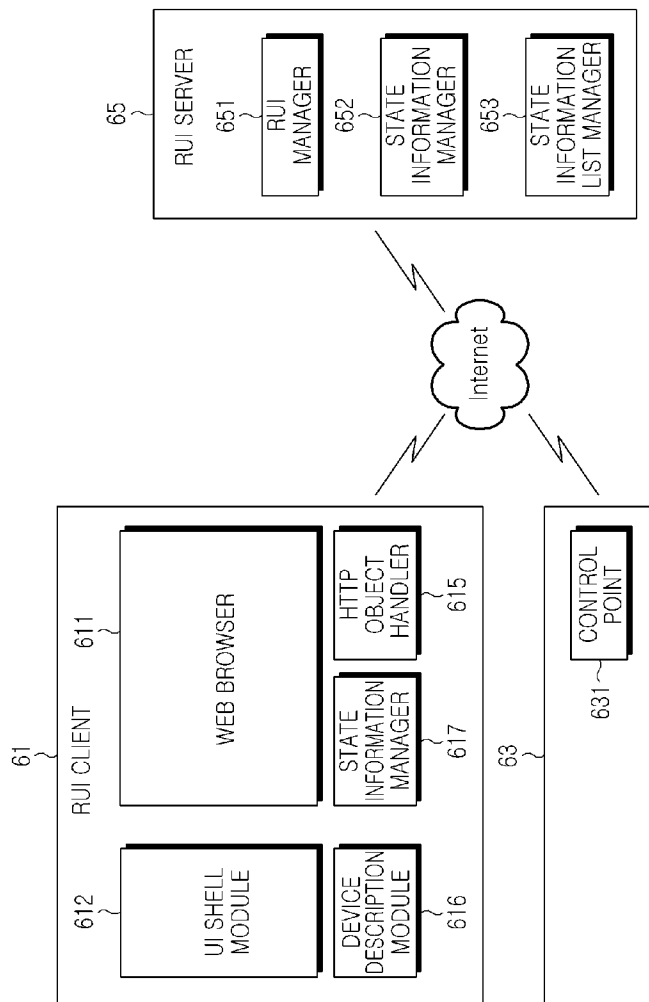
[Fig. 4]



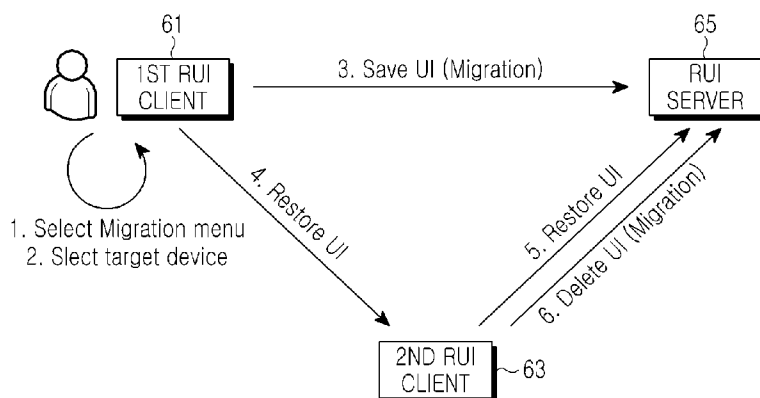
[Fig. 5]



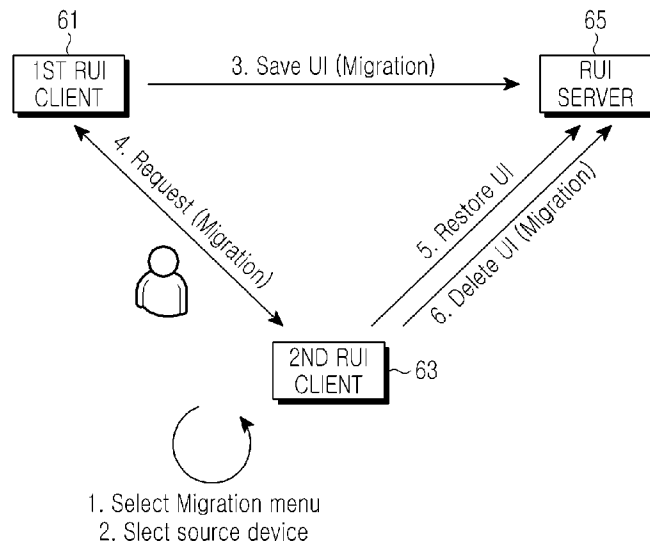
[Fig. 6]



[Fig. 7]



[Fig. 8]



[Fig. 10]

