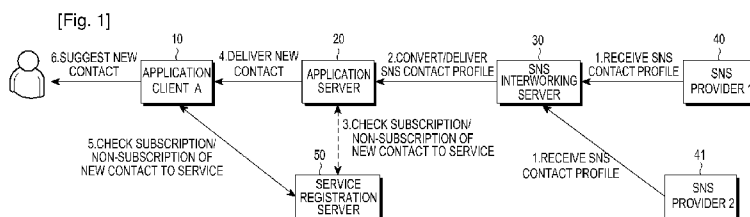




- (51) **International Patent Classification:** Not classified
 - (21) **International Application Number:** PCT/KR2011/001436
 - (22) **International Filing Date:** 2 March 2011 (02.03.2011)
 - (25) **Filing Language:** English
 - (26) **Publication Language:** English
 - (30) **Priority Data:** 10-2010-0018734 2 March 2010 (02.03.2010) 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; and**
 - (75) **Inventors/Applicants (for US only):** **LEE, Kyung-Tak** [CH/KR]; #110-303, Jugong Yeongtongvillage, Yeongdeok-dong, Giheung-gu, Yongin-si, Gyeonggi-do 446-967 (KR). **OH, Gyu-Bong** [KR/KR]; 970-3, Yeongtong 2-dong, Yeongtong-gu, Suwon-si, Gyeonggi-do 443-472 (KR). **KIM, Wuk** [KR/KR]; #131-2201, Samsung Raemian APT., Bisan 1-dong, Dongan-gu, Anyang-si, Gyeonggi-do 431-760 (KR). **LEE, Seung-Yong** [KR/KR]; #101-301, I-Readville, 185-11, Songpa-dong, Songpa-gu, Seoul 138-170 (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, TH, 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, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, 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, RS, 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:** APPARATUS AND METHOD FOR PROVIDING CONTACTS THROUGH INTERWORKING BETWEEN MESSAGING SERVICE AND SOCIAL NETWORK SERVICE



(57) **Abstract:** A method and apparatus is provided for providing a contact through interworking with a Social Network Service (SNS). A client using the other service through SNS interworking receives a contact provided in the SNS, determines whether the other party's client having subscribed to the same service as that of the client is present in the received contact, and informs a user of the determination results. Based on the determination results, the client may determine subscription/non-subscription of the other party's client to the service, and may send an invite request for the service if the other party's client has not subscribed to the same service as that of the client. By doing so, the number of targets, subscription/non-subscription of which to a specific service can be determined, may be extended.

WO 2011/108844 A2

Description

Title of Invention: APPARATUS AND METHOD FOR PROVIDING CONTACTS THROUGH INTERWORKING BETWEEN MESSAGING SERVICE AND SOCIAL NETWORK SERVICE

Technical Field

- [1] The present invention relates generally to a messaging service apparatus and method, and more particularly, to an apparatus and method for providing contacts through interworking between a messaging service and a Social Network Service (SNS).

Background Art

- [2] A mobile communication terminal is a means capable of providing and exchanging the necessary information anywhere, anytime. Conventionally, in most cases, the mobile terminal was used simply for calls over a phone. However, with the increasing prevalence of mobile communication terminals, the features of conversation and information exchange through texts and multiple media have become important. In line with the trend, various messaging services such as Short Message Service (SMS), Multimedia Messaging Service (MMS) and MicroSoft™ Network (MSN), and various SNS services such as Facebook™ and Twitter™ have been provided.
- [3] Although such various services have been provided based on different technologies, users may be provided with duplicate features and user experiences even though they use different services. In addition, because each service has its own features distinguishable from other services, a user should subscribe to every service in order to experience all the features, causing many burdens.
- [4] In order to minimize the user burdens, new services integrating such various user experiences have been introduced and standardized. Typically, these new services may include Open Mobile Alliance (OMA) Converged IP Messaging (CPM) and Global System for Mobile communication Association (GSMA) Rich Communication Suite (RCS). Because these messaging services use different address books, there is a need for management of the address books. To meet the need, OMA has proposed OMA Converged Address Book (CAB). The OMA CAB enables the services to manage a variety of information with an improved network address book using the network storage, and to share inter-group information.

Disclosure of Invention

Technical Problem

- [5] In case of the existing messaging service among the newly introduced services, even

though a user has subscribed to a specific service, if the other party has not subscribed to the specific service or other compatible services, conversation and information exchange through the specific service may not be possible. Taking this into consideration, an RCS service provides an address book management function, and may provide a function of notifying subscription/non-subscription to the RCS service. However, since even the RCS service does not share address books with other services, subscription/non-subscription to the RCS service may be notified only to the address book used in the RCS service. Therefore, if it interworks with the SNS, the messaging service may be provided with an extended address book, extending the number of targets (or contacts), subscription/non-subscription of which to the service can be (or is desired to be) determined. Therefore, there is a need for a method capable of extending the number of targets, subscription/non-subscription of which to a specific service can be determined.

Solution to Problem

- [6] Exemplary embodiments of the present invention provide an apparatus and method for providing contacts through interworking between a messaging service and an SNS, capable of extending the number of targets (or contacts), subscription/non-subscription of which to a specific service can be determined.
- [7] In accordance with one aspect of the present invention, there is provided a method for receiving a contact in a client terminal through interworking between a messaging service and a Social Network Service (SNS). The method includes sending a server a request for a contact profile list provided in the SNS; receiving from the server a contact profile list, which is one of contact profile lists corresponding to the request and filtered by the server; and determining whether each contact in the received contact profile list has subscribed to the same service as that of the client terminal.
- [8] In accordance with another aspect of the present invention, there is provided a client terminal for receiving a contact through interworking between a messaging service and a Social Network Service (SNS). The client terminal includes a controller for sending a server a request for a contact profile list provided in the SNS, receiving from the server a contact profile list, which is one of contact profile lists corresponding to the request and filtered by the server, and determining whether each contact in the received contact profile list has subscribed to the same service as that of the client terminal; an Input/Output (I/O) interface for exchanging a request and a response with the server; a memory for temporarily storing the received contact profile list; an address book manager for storing an address book synchronized with the server; and a user interface for informing a user whether a specific contact has subscribed to the same service as that of the user.

[9] In accordance with further another aspect of the present invention, there is provided a method for providing a contact to a client terminal in a server through interworking between a messaging service and a Social Network Service (SNS). The method includes receiving a client's requirement for a contact profile list provided in the SNS; examining the received client's requirement; sending a contact list request using the SNS; receiving a contact list in response to the contact list request; filtering a contact corresponding to the examined requirement by comparing the received contact list with the existing contact; and providing the filtered contact.

[10] In accordance with yet another aspect of the present invention, there is provided a server for providing a contact to a client terminal through interworking between a messaging service and a Social Network Service (SNS). The server includes a controller for receiving a client's requirement for a contact profile list provided in the SNS, examining the received client's requirement, sending a contact list request using the SNS, receiving a contact list in response to the contact list request, filtering a contact corresponding to the examined requirement by comparing the received contact list with the existing contact, and providing the filtered contact; and an Input/Output (I/O) interface for transmitting the request and receiving a response to the request.

Advantageous Effects of Invention

[11] According to exemplary embodiments of the present invention, an extended address book can be provided, contributing to extending the number of targets (or contacts), subscription/non-subscription of which to a specific service can be determined. Accordingly, a user can be automatically provided with new contacts which are not in his or her address book, and thus can easily make contact not only with the friends and family members in the address book, but also with the contacts, who are related to his or her profile, but whom he or she is not aware of. In addition, when having newly subscribed to a specific service, the user can enjoy the service right away, enhancing user conveniences. Besides, for example, if the service provider sends a join invite message, the user may not be interested in it. However, if a senior of a specific school finds an address of his or her school junior through the system proposed by the present invention and sends a join invite message to him or her, then the junior is highly likely to join the service. Taking this into consideration, the service provider may easily induce many users having not subscribed to its service to subscribe to the service, contributing to a sharp increase in the number of its service subscribers.

Brief Description of Drawings

[12] 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:

- [13] FIG. 1 is a diagram illustrating a system configuration for providing new contacts according to an embodiment of the present invention;
- [14] FIG. 2 is a flow diagram illustrating a process of providing contacts through SNS interworking according to an embodiment of the present invention;
- [15] FIG. 3 is a block diagram illustrating an internal structure of the application server in FIG. 2;
- [16] FIG. 4 is a flowchart illustrating an operation of the application server in FIG. 2;
- [17] FIG. 5 is a block diagram illustrating a structure of the application client in FIG. 2;
- [18] FIG. 6 is a flowchart illustrating an operation of the application client in FIG. 2;
- [19] FIG. 7 is a diagram illustrating architecture for a CAB service, used in the present invention;
- [20] FIG. 8 is a flow diagram illustrating an operation in a Presence enabler used in the present invention;
- [21] FIGs. 9a and 9b are flow diagrams illustrating a process of providing contacts according to a first embodiment of the present invention;
- [22] FIGs. 10a and 10b are flow diagrams illustrating a process of providing contacts according to a second embodiment of the present invention;
- [23] FIGs. 11a and 11B are flow diagrams illustrating a process of providing contacts according to a third embodiment of the present invention;
- [24] FIGs. 12a and 12b are flow diagrams illustrating a process of providing contacts according to a fourth embodiment of the present invention; and
- [25] FIGs. 13a and 13b are flow diagrams illustrating a process of providing contacts according to a fifth embodiment of the present invention.
- [26] 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

- [27] Exemplary embodiments of the present invention will now be described in detail with reference to the accompanying drawings. In the following description, specific details such as detailed configuration and components are merely provided to assist the overall understanding of exemplary embodiments of the present invention. Therefore, it should be apparent to those skilled in the art that various changes and modifications of the embodiments described herein can be made without departing from the scope and spirit of the invention. In addition, descriptions of well-known functions and constructions are omitted for clarity and conciseness.
- [28] For a better understanding of the present invention, the terms of entities defined in Converged Address Book (CAB) of Open Mobile Alliance (OMA) and Rich Communication Suite (RCS) of Global System for Mobile communication Association

(GSMA) will be used herein. However, the use of such standards and terms defined therein is not intended to limit the scope of the present invention, and the present invention may be applied to any systems having the similar technical backgrounds.

[29] The present invention provides a method for providing contacts through interworking with a Social Network Service (SNS). To this end, the present invention includes a process in which a client using the other service through interworking with an SNS is provided with a contact provided in the SNS, and a process in which the client determines whether the other party's client having subscribed to the same service as that of the client itself is present in the provided contact, and notifies its user of the determination results. Based on the determination results, a client using the other service may determine whether the other party's client has subscribed to the desired service, and may send an invite request for the service if the other party's client has not subscribed to the same service as that of the client itself. By doing so, it is possible to extend the number of targets (or contacts), subscription/non-subscription of which to a specific service can be (or is desired to be) determined.

[30] A system configuration for providing contacts, in which the above-described function is implemented, will be described with reference to FIG. 1.

[31] Referring to FIG. 1, an application client A 10 is an application messaging service program running in a user terminal, also known as a User Equipment (UE) or a Mobile Station (MS). The application client A 10 delivers user requirements to an application server 20 in the network, and informs its user of the event and/or message received from the application server 20 or a service registration server 50. As an example of the application messaging service, an RCS service will be described in embodiments of the present invention. The application client A 10 may serve as a CAB or Presence client.

[32] The application server 20 handles the user requirements received from the application client A 10. The application server 20 delivers the event and/or message received from an SNS interworking server 30 or the other network to the application client A 10.

[33] The SNS interworking server 30 serves as a gateway connecting the application server 20 to its SNS provider 40, and also serves as an important intermediary offering appropriate format conversion between the application server 20 and the SNS provider 40. In actual implementation of the system according to the present invention, the SNS interworking server 30 may be disposed in the same place as that of the application server 20, or may be provided as an internal component of the application server 20.

[34] SNS providers 40 and 41 are agencies offering SNSs such as Facebook™ and Twitter™.

[35] The service registration server 50 is a server that handles and manages registration of all services.

- [36] Reference will be made to FIG. 2 to describe a process of providing contacts in the system with the above-described configuration. In FIG. 2, it is assumed that a user A's terminal, i.e., the application client A 10 has subscribed to both a service M and an SNS service. Accordingly, there are login and password of the user A accessible to the SNS service, and the SNS interworking server 30 has logged in the SNS providers 40 and 41 on behalf of the user A, and is accessible thereto. It is also assumed that if the SNS interworking server 30 cannot directly log in the SNS providers 40 and 41, it performs the login by requesting the user A to log in the SNS service providers 40 and 41 through the application client A 10.
- [37] Referring to FIG. 2, the application client A 10 sends a request for a new contact list to the application server 20 in step 201. The request for a new contact list may be made when the user A runs the application client A 10 or accesses the service M. This request is made to request a new contact profile list meeting a relevant condition to determine the presence/absence of a contact having the same characteristics as the user A, for example, a contact corresponding to the friend who has the same other contact and/or the same graduate school as the user A. The new contact profile includes a name, an e-mail address, a phone number, etc.
- [38] Then, the application server 20 forwards the request to the SNS interworking server 30 in step 203. In response thereto, the SNS interworking server 30 sends a request for at least one necessary requirement to the SNS providers 40 and 41 to meet the request from the application server 20 in steps 205a and 205b. Thereafter, upon receiving responses to the requests from the SNS providers 40 and 41 in steps 207a and 207b, the SNS interworking server 30 determines in step 209 whether a profile list has been received.
- [39] To be specific, the SNS interworking server 30 determines whether the received response(s) is the new contact profile list requested by the application server 20, and if not so, the SNS interworking server 30 returns to steps 205a and 205b and sends the next necessary request to the SNS providers 40 and 41. On the other hand, upon receiving the requested new contact profile list in response to the request, the SNS interworking server 30 proceeds to step 211. For example, if the SNS interworking server 30 receives only a contact ID list and fails to receive the detailed profile such as a name, an e-mail address and a phone number in step 209, the SNS interworking server 30 re-sends the request to the SNS providers 40 and 41, determining that the received response is not a new contact profile list.
- [40] Upon receiving the new contact profile list in this way, the SNS interworking server 30 forwards the received new contact profile list to the application server 20 in step 211. Then the application server 20 filters or extracts only the contacts valid for the user A from each of the received contact profiles in step 213. An address book of the

user A is stored even in the network, and the application server 20 may access the address book. Accordingly, the application server 20 may perform contact filtering, and a detailed description thereof will be made below with reference to FIG. 4.

[41] Thereafter, the application server 20 sends the filtered contact list to the application client A 10 in step 215. In step 217, the application client A 10 determines whether each received contact has subscribed to the service M, and informs its user of the determination results. A detailed description of the process of determining whether the contact has subscribed to the service, and informing the user of the determination results will be made below with reference to FIG. 6.

[42] If the application client A 10 sends a request for a contact registration service list to the service registration server 50 in step 217a, in order to determine to which service M which contact has subscribed, and the service registration server 50 sends a service list registered to correspond to the contact to the application client A 10 in step 217b. Accordingly, the user A may be provided with an already connected SNS friend(s), or an SNS contact whose characteristics are consistent with the user, and may also be provided with the fact that the contact has subscribed to the service M, and with the method capable of connecting with the contact through the service. In addition, if the provided contact has not subscribed to the service M or its subscription/non-subscription is unknown, even though information only about the contact is provided, the user A may send an invite message for subscription to the service M to the other party's client corresponding to the contact, if needed.

[43] Prior to a description of an operation of an application server performing contact filtering, an internal structure of the application server will be described below with reference to FIG. 3.

[44] FIG. 3 is a block diagram illustrating an internal structure of the application server 20.

[45] Referring to FIG. 3, a controller 301 controls the overall operation of the application server 20, and informs when and what it should perform, in cooperation with other internal components. An Input/Output (I/O) interface 303 is used to exchange information with other system entities, and a memory 305 is a storage in which processing data is temporarily stored when the application server 20 performs its operations. An address book manager 307 stores the existing contact information of a user(s), and also stores an address book synchronized with an address book manager 507 used in the application client 10. A mapping table manager 309 translates the existing addresses/numbers such as e-mail numbers and phone numbers into specific addresses used for a service M, e.g., Session Initiation Protocol (SIP) URI. The address book manager 307 and the mapping table manager 309 may be implemented out of the application server 20 according to proposed embodiments of the present invention.

- [46] The application server 20 constructed as described above performs contact filtering as in FIG. 4. The contact filtering is performed under control of the controller 301 in the application server 20.
- [47] Referring to FIG. 4, the controller 301 stores a contact profile list received via the I/O interface 303 in the memory 305 in step 401. In step 403, the controller 301 extracts contact-identification data, e.g., a name, an e-mail address, and a phone number, from a profile stored in the memory 305. In step 405, the controller 301 compares the extracted data with its associated data in the address book manager 307 and determines whether the extracted data corresponds to a contact already existing in an address book. If the extracted data corresponds to a contact existing in the address book, the controller 301 determines in step 407 whether the contact has already subscribed to a service M. To be specific, the controller 301 determines whether an indicator indicating subscription of the contact to the service M is present in the address book manager 307. If the contact has already subscribed to the service M, meaning that even the user A knows this fact, the controller 301 deletes the profile for the contact from the memory 305 in step 409, because this contact may no longer undergoes filtering.
- [48] If the extracted data does not correspond to a contact existing in the address book in step 405, or if the contact has not already subscribed to the service M in step 407, the controller 301 determines in step 411 whether there is an address available in the service M. To be specific, the controller 301 determines whether an address available in the service M (hereinafter referred to as a 'service-M address') is present in the address book manager 307 in association with the contact. Therefore, if the extracted data does not correspond to a contact existing in the address book manager 307 or there is no service-M address, e.g., SIP URI, in the address book manager 307, the controller 301 accesses the mapping table manager 309 and determines the presence/absence of a service-M address matched to the contact's e-mail address or phone number extracted in step 403. For example, in case of an RCS service, although a SIP URI address is available, because the SIP URI address is used by several services, even though the SIP URI address is present, it is not necessary that the contact has subscribed to the RCS service.
- [49] Therefore, the presence/absence of a service-M address may be determined in the following method, and the method is optionally used depending on the situation. In this method, although no matched contact's service-M address is present in the mapping table manager 309, if the service-M address is likely known by a server in the other network because the other party's client corresponding to the contact is a subscriber to the other network, then the controller 301 may deliver a profile of the contact to the other network's server through the I/O interface 303, and request a matched service-M address. If a response is received from the other network's server through the I/O

interface 303 and the received response includes a matched service-M address, the controller 301 adds the service-M address in the mapping table manager 309 together with the contact's e-mail address and phone number.

- [50] If it is determined by the above method that the contact's service-M address is present, the controller 301 adds the service-M address in the contact profile in step 413, because the other party's client corresponding to the contact will likely subscribe to the service M. On the other hand, if the contact's service-M address is not present or cannot be known, the controller 301 determines in step 415 whether there are more contacts to be examined. If so, the controller 301 returns to step 403 and repeats the above process, and if not so, the controller 301 ends the contact filtering.
- [51] Prior to a description of an operation of an application client A performing a process of determining whether the contact has subscribed to the service and informing the user of the determination results, an internal structure of the application client A will be described with reference to FIG. 5.
- [52] Referring to FIG. 5, a controller 501, a component controlling the overall operation of the application client A 10, informs when and what it should perform, in co-operation with other internal components. In I/O interface 503 is used to exchange information with other system entities. A memory 505 is a storage in which processing data is temporarily stored when the application client A 10 performs its operations. An address book manager 507, a storage containing user's existing contact information, is synchronized with the address book manager 307 used in the application server 20. A user interface 509 is used to show all information about the user, or is used by the user to input information.
- [53] The application client A 10 constructed as described above performs a process of determining whether a contact has subscribed to a specific service and informing its user of the determination results, as in FIG. 6. This operation is performed under control of the controller 501 of the application client A 10.
- [54] Referring to FIG. 6, the controller 501 stores a filtered-contact profile list received through the I/O interface 503 in the memory 505 in step 601. Thereafter, the controller 501 determines in step 603 whether a service-M address is present in the contact profile stored in the memory 505. If the service-M address is present, the controller 501 adds the contact profile in a 'subscription list' and then stores it in the memory 505 in step 605, because the contact will likely subscribe to the service M. On the other hand, if the service-M address is not present, the contact will unlikely subscribe to the service M, or cannot be known. Therefore, in step 607, the controller 501 adds the contact profile in a 'non-subscription list', and stores it in the memory 505.
- [55] The controller 501 determines in step 609 whether a more profile to be examined is present in the received list. If a more profile to be examined is present, the controller

501 returns to step 603 and repeats the above process. However, if no more profile to be examined is present, the controller 501 sends contacts in the 'subscription list' stored in the memory 505 to the service registration server 50 through the I/O interface 503, and requests a service list in which contacts in the 'subscription list' are registered, in step 611. The controller 501 receives a service list, which is registered to correspond to each contact in response to the request, from the service registration server 50 through the I/O interface 503 in step 613. That is, the controller 501 receives a list of services to which the other party's clients corresponding to the contacts have registered.

- [56] The controller 501 determines in step 615 whether the service M is present in a contact service list. If the service M is not present, the controller 501 moves the contact stored in the memory 505 from the 'subscription list' to the 'non-subscription list' in step 617, because the contact has not subscribed to the service M. The controller 501 determines in step 619 whether a more profile to be examined is present in the received contact profile list. If a more profile to be examined is present, the controller 501 returns to step 615 and repeats the above process. On the other hand, if no more profile is present, the controller 501 notifies the user A of the contacts in the 'subscription list' and 'non-subscription list' stored in the memory 505 using the user interface 509 in step 621.
- [57] In case of the contacts in the 'subscription list', the application client A 10 notifies the user A of the fact that a relevant contact uses the same service M, and provides a method of adding a contact in the address book manager 507 and a method capable of connecting with the contact through the service M. In case of the contacts in the 'non-subscription list', the application client A 10 notifies the user A of the fact that a relevant contact is present, and provides a method of adding a contact in the address book manager 507 and a method capable of sending a subscription invite message to the service M, to the contact.
- [58] Assuming that the service M, to which a user of the application client A 10 has subscribed, is an RCS service, the RCS service provides the service by means of a combination of a plurality of enablers and profiling. In embodiments of the present invention, the number of targets, subscription/non-subscription of which to a specific service can be determined, can be extended by interworking the RCS service with an SNS service. The RCS service may be provided based on OMA CAB and OMA Presence, which are examples of enablers providing an address book integration function.
- [59] Accordingly, the application client A 10 in FIG. 2 may operate as a CAB client to receive a new contact, and may also operate as a Presence client to receive service subscription/non-subscription of a client corresponding to the received new contact.

Therefore, the system providing contacts through interworking with an SNS service according to the present invention may be implemented based on OMA CAB and OMA Presence enablers.

[60] Prior to a description of the present invention, reference will be made to FIG. 7 to describe architecture of the general CAB system used in the present invention.

[61] Referring to FIG. 7, a CAB Client 71 is an application program for managing and controlling an address book in a user terminal. A CAB Server 72 is a major network entity for handling user's various requirements received from the CAB Client 71. A CAB Address Book (AB) XML Document Management Server (XDMS) 73 is a server for storing and managing address books of users. A CAB Personal Contact Card (PCC) XDMS 74 is a server for storing and managing profiles of users. A CAB User Preferences (UP) XDMS 75 is a server for storing and managing preferences and specific requirements of users. A Non-CAB Address Book System 76 includes an address book system used in a CAB service, and all other address book systems. For example, vCard and the like correspond thereto. In the present invention, it can be assumed that the SNS providers 40 and 41 described in FIG. 2 are also included in the Non-CAB Address Book System 76.

[62] The interfaces and protocols/technologies used in the present invention are as follows:

[63] A CAB-1 interface is used for synchronization between an address book in the CAB Client 71 and an address book stored in the CAB AB XDMS 73 by way of the CAB Server 72. The CAB-1 interface is also used when the CAB Server 72 notifies the CAB Client 71 of a difference between the two address books. SyncML[3] is used as its protocol/technology.

[64] A SIC-1 interface is used to inform the CAB Client 71 of changes, every time data stored in the CAB PCC XDMS 74 and the CAB UP XDMS 75 are changed. For reference, the CAB-1 interface is used to inform the CAB Client 71 of a change in data stored in the CAB AB XDMS 73. SIP-Specific Event Notification [4] is used as its protocol/technology.

[65] A SIC-2 interface is used to inform the CAB Server 72 of changes, every time data stored in the CAB AB XDMS 73, the CAB PCC XDMS 74, and the CAB UP XDMS 75 are changed. SIP-Specific Event Notification [4] is used as its protocol/technology.

[66] An XDM-3i interface is used when the CAB Client 71 manages data stored in the CAB PCC XDMS 74 and the CAB UP XDMS 75. XML Configuration Access Protocol/Hyper Text Transfer Protocol (XCAP/HTTP) [5] is used as its protocol/technology.

[67] An XDM-4i interface is used when the CAB Server 72 manages data stored in the CAB AB XDMS 73, the CAB PCC XDMS 74, and the CAB UP XDMS 75. XCAP/

HTTP [5] is used as its protocol/technology.

[68] An operation flow in a general Presence enabler used in the present invention is illustrated in FIG. 8. An OMA Presence enabler performs the operation of FIG. 8 when a user wants to determine the service used by the other user. The operation flow is carried out using the SIP-Specific Event Notification [4] protocol.

[69] Referring to FIG. 8, a Presence client 81, a fetcher, sends a SIP SUBSCRIBE to receive Presence information that the other user has registered in a Presence server 84, in step 801. To receive the Presence information only once, an "Expires" header is set to '0' in the SIP SUBSCRIBE. On the other hand, in order to periodically receive changes in the Presence information, the "Expires" header may be set to a higher value. The SIP SUBSCRIBE is forwarded to the Presence server 84 through a Fetcher SIP/IP Core 82 corresponding to a SIP-based sub-transmission network and a Presentity SIP/IP Core 83 in the side of the Presence server 84 in steps 803 and 805.

[70] In step 807, the Presence server 84 performs authorization check to determine whether the Presence client 81 is approved to receive the requested information. Upon approval of the request, the Presence server 84 sends a 200 OK to the Presence client 81 via the SIP/IP Cores 82 and 83 in steps 809 to 813. Thereafter, in steps 815 to 819, the Presence server 84 delivers the requested information to the Presence client 81 via the SIP/IP Cores 82 and 83 using a SIP NOTIFY. In response to the SIP NOTIFY, the Presence client 81 sends an Acknowledgement (ACK) of the Presence information to the Presence server 84 through the SIP/IP Cores 82 and 83 using a 200 OK in steps 821 to 825.

[71] The operation flow of FIG. 2 according to an embodiment of the present invention may be reconstructed based on the foregoing description of FIGS. 7 and 8, and will be described in detail below. According to embodiments of the present invention, a configuration of the system, which is reconstructed based on FIGS. 7 and 8, and interworks with an RCS service and an SNS service, may be implemented as in FIGS. 9a to 13b. Here, FIGs. 9b, 10b, 11B, 12b, and 13b are diagrams succeeding FIGs. 9a, 10a, 11a, 12a, and 13a, respectively.

[72] FIGs. 9a and 9b illustrate a process of providing contacts according to a first embodiment of the present invention, in which the application server 20 and the SNS interworking server 30 in FIG. 2 are integrated. For convenience of description, entities for routing and other purposes, for example, SIP/IP Cores, IMS, and Aggregation Proxy, are omitted herein, but it can be assumed that the process undergoes these entities, if needed. In the following drawings, since the RCS service of FIG. 2 is reconstructed based on the CAB service, the RCS-related terms may be replaced with the terms used in CAB, and for convenience of description, the RCS terms and the CAB terms will be used in a mixed way. Likewise, the SNS terms in FIG. 2 may also be

used together with their associated non-RCS terms.

[73] Referring to FIG. 9a, when a user A accesses an RCS service, the RCS (CAB) client A 71 makes a request to store detailed information about a request in a "CAB Feature Handler" document managed by the CAB UP XDMS 75, to receive a new contact suggestion list. To this end, a "Get Contact suggestions" request is sent to the CAB UP XDMS 75 among the CAB XDMSs in the form of XCAP/HTTP PUT, in step 901. A detailed description of the "CAB Feature Handler" document will be made below.

[74] In step 903, the CAB UP XDMS 75 stores requirements in the "CAB Feature Handler" document in its management, and sends a 200 OK, or a store complete response, to the RCS (CAB) client A 71. In step 905, the CAB UP XDMS 75 notifies the RCS (CAB) server 72 of the requirements and associated information of the RCS (CAB) client A 71, which are stored in the "CAB Feature Handler" document. For this notification, a SIP NOTIFY message is used. In step 907, upon receiving the notification, the RCS (CAB) server 72 sends a 200 OK, or an ACK message, to the CAB UP XDMS 75. In step 909, the RCS (CAB) server 72 checks the requirements, converts them in a format appropriate for the Non-RCS (CAB) address book system 76, and sends a request for a Non-RCS (CAB) contact list to the Non-RCS (CAB) address book system 76. The Non-RCS (CAB) address book system 76 corresponds to an SNS provider. Although a Non-RCS (CAB) address book system corresponding to one SNS provider is illustrated in FIGs. 9a and 9b, if there are a plurality of SNS providers, the RCS (CAB) server 72 may receive contact lists from the plurality of SNS providers and combines them.

[75] In step 911, the Non-RCS (CAB) address book system 76 sends the requested Non-RCS (CAB) contact list to the RCS (CAB) server 72. To exactly receive the necessary list, steps 909 and 911 may be repeated. In step 913, the RCS (CAB) server 72 converts the received Non-CAB contact list in an RCS (CAB) format. In step 915, the RCS (CAB) server 72 sends a request for the existing address book of the user A, stored in the network, to the CAB AB XDMS 73. For this request, an XCAP/HTTP GET may be used. In response, the CAB AB XDMS 73 sends a 200 OK including the requested address book of the user A, to the RCS (CAB) server 72 in step 917. The RCS (CAB) server 72 performs functions of the application server 20 and the SNS interworking server 30 in FIG. 2.

[76] In step 919, the RCS (CAB) server 72 filters contacts by comparing the existing contacts of the user A with the new contact profiles received from the Non-RCS (CAB) address book system 76. This contact filtering operation is performed as described in FIG. 4. The filtered contacts are stored in the CAB UP XDMS 75 in step 921. To request the storage, an XCAP/HTTP PUT is used. In step 923, the CAB UP XDMS 75 stores the received contact profiles and sends a 200 OK to the RCS (CAB)

server 72 as an ACK.

- [77] In step 925, the CAB UP XDMS 75 notifies the RCS (CAB) client A 71 of the fact that new contact profiles are stored therein, and then sends the profiles to the RCS (CAB) client A 71 using a SIP NOTIFY message. The received profile is a response to the request that the RCS (CAB) client A 71 sent in step 901. In step 927, the RCS (CAB) client A 71 sends an ACK message for the received SIP NOTIFY to the CAB UP XDMS 75.
- [78] In this manner, the RCS (CAB) client A 71 may extend the number of targets, subscription/non-subscription of which to a specific service can be determined, through interworking with the Non-RCS provider, i.e., an SNS provider. To this end, in step 929, the RCS (CAB) client A 71 extracts a contact having a SIP address from the received profile. To be specific, according to the description of FIG. 6, the RCS (CAB) client A 71 determines the presence/absence of an address available for the RCS service, i.e., a SIP URI, among the received contact profiles, and extracts the contacts. In step 931, based on the SIP contact addresses, the RCS (CAB) client A 71 sends the RCS (Presence) server 84 a request for “service capability information” of each SIP contact, which indicates a list of services available on the RCS (Presence) client A 71. For this request, a SIP SUBSCRIBE message is used. The RCS (CAB) client A 71 and the RCS (Presence) client A 71 are application programs, which correspond to the application client A 10 in FIG. 2 and are installed in the same user terminal. However, in the present invention, when requesting a contact, the client A 71 is called the RCS (CAB) client A 71, since it operates as an RCS (CAB) client. When requesting to determine subscription/non-subscription to a service, the client A 71 is called the RCS (Presence) client A 71, since it operates as an RCS (Presence) client.
- [79] If the RCS (Presence) server 84 accepts the request after performing authorization check to determine whether the RCS (Presence) client A 71 is allowed to receive the requested information, the RCS (Presence) server 84 sends a 200 OK or an accept message to the RCS (Presence) client A 71 in step 933. In step 935, the RCS (Presence) server 84 delivers the requested “service capability information” of contacts to the RCS (Presence) client A 71 using a SIP NOTIFY message. Upon receiving this message, the RCS (Presence) client A 71 sends a 200 OK or an ACK of the “service capability information” to the RCS (Presence) server 84 in step 937.
- [80] Thereafter, in step 939, the RCS (CAB) client A 71 determines whether the “service capability information” of each contact is for an RCS user and informs its user of the determination results by means of the RCS (Presence) client A 71 according to steps 615 to 621 in FIG. 6.
- [81] As described above, in the first embodiment of the present invention, the RCS (CAB) server 72 performs both the filtering function and the interworking function, and the

filtered contacts are stored independently in the “CAB Feature Handler” document managed in the CAB UP XDMS 75. On the other hand, however, the filtered contacts may be stored not in the “CAB Feature Handler” document, but in an address book of the RCS (CAB) client A 71. A configuration according to a second embodiment of the present invention is as illustrated in FIGs. 10a and 10b.

- [82] Prior to a description of the second embodiment of the present invention, details in the “CAB Feature Handler” document will be described.
- [83] In the present invention, a <contact_suggestions> element is defined. The <contact_suggestions> element is an element that the RCS (CAB) client A 71 includes in the “CAB Feature Handler” document when it desires to receive a new contact suggestion list. Low-level elements of this element includes <non-CAB source>, <credentials>, <criteria>, <scheduled-interval>, <max-suggestions>, <id>, <code>, <phrase>, <contact_suggestions_response>, etc.
- [84] <non-CAB source> is an element for recording an arbitrary agency from which the RCS (CAB) client A 71 desires to receive a new contact suggestion. For example, a domain name of the agency is recorded in this element. In the absence of this element, the agency is determined depending on the service provider policy.
- [85] <credentials> means the information needed for authentication during access to the agency. Low-level elements of <credentials> include <username> in which login information is recorded, and <password> in which a password is recorded.
- [86] <criteria> may be replaced with <preferences> or <keywords>, and is used to set a criteria based on which a new contact suggestion is to be selected. Low-level elements of this element may include <friend-of-friend>, <same-school>, <same-work>, <same-hobby>, etc.
- [87] First, <friend-of-friend>, also called <mutual-friend>, is set to ‘1’ when the RCS (CAB) client A 71 desires to receive contacts of the contacts registered in a user address book in the agency, for example, when the RCS (CAB) client A 71 desires to receive contacts of friends of the user’s friend.
- [88] In <same-school> is recorded a name of a specific school when the RCS (CAB) client A 71 desires to receive a suggestion of contacts who went to the same school.
- [89] In <same-work> is recorded a name of a specific company or a type of a specific work when the RCS (CAB) client A 71 desires to receive a suggestion of contacts having the same work.
- [90] In <same-hobby> is recorded a specific hobby when the RCS (CAB) client A 71 desires to receive a suggestion of contacts having the same hobby.
- [91] The above low-level elements are mere examples, and desired keywords may be directly input in the <criteria> (or <preferences>, and <keywords>) element. Even the <criteria>, if not recorded, may be determined according to the user profile and service

provider policy as the RCS (CAB) server 72 accesses the user profile or the CAB PCC XDMS 74.

- [92] In <scheduled-interval> is recorded a suggested time interval when the RCS (CAB) client A 71 desires to receive a contact suggestion not only once, but by periods.
- [93] In <max-suggestions> is set the maximum number of contacts for which the RCS (CAB) client A 71 desires to receive a suggestion.
- [94] <id> is an ID for identifying each request.
- [95] The above-described <contact_suggestions> element is a low-level element of a cab_feature element. A low-level element of the cab_feature element also includes a <response> element. The <response> element is an element that the RCS (CAB) server 72 fills with data when it responds to a request made by the RCS (CAB) client A 71 as in step 921. Low-level elements of the <response> element include the following elements.
- [96] In <code> is recorded, as a status code for the response, a status code indicating whether the response is successful or failed.
- [97] <phrase> is a high-level phrase for the response.
- [98] In <contact_suggestions_response> is recorded a detailed response to the above-described <contact_suggestions> request. This element carries new contact suggestion information, and a detailed information structure for each contact follows the structure defined by OMA CAB.
- [99] Although it is assumed in the first embodiment of the present invention that the <contact_suggestions_response> element is put in the same document as that of the new contact suggestion request information, this element may be recorded in other documents.
- [100] In the second embodiment of the present invention, filtered contacts are stored in the address book of the RCS (CAB) client A 71. For a description thereof, reference will be made to FIGs. 10a and 10b.
- [101] Steps 1001 to 1019 in FIG. 10a are equal in operation to steps 901 to 919 in FIG. 9a, except that in the second embodiment of the present invention, filtered contacts are stored in the CAB AB XDMS 73. Accordingly, in step 1021, the RCS (CAB) server 72 requests the CAB AB XDMS 73 to store the filtered contacts. For this request, a XCAP/HTTP PUT is used. The filtered contacts may be directly stored in the user address book, or may be stored in another separate document.
- [102] In step 1023, the CAB AB XDMS 73 stores the received contact profiles and sends a 200 OK or an ACK to the RCS (CAB) server 72. In step 1025, the RCS (CAB) server 72 notifies the RCS (CAB) client A 71 of the change in the address book. For this notification, OMA Data Synchronization (OMA DS) is used. In step 1027 of FIG. 10b, the RCS (CAB) client A 71 performs address book synchronization with the RCS (CAB)

server 72, and they receive in common the new contact suggestions through the synchronization. Steps 1029 to 1039 in FIG. 10b are equal in operation to steps 929 to 939 in FIG. 9b. However, when directly stored in the address book, the added contact suggestions may be deleted from the address book depending on the use/nonuse of the RCS service by the contacts, and on the user's disposition or choice.

[103] In a third embodiment of the present invention, the contact filtering step and the step of storing the filtered contact list in the CAB UP XDMS 75 in the second embodiment of the present invention are performed not by the RCS (CAB) server 72, but by the CAB AB XDMS 73. Reference will be made to FIGs. 11a and 11B to describe the third embodiment of the present invention.

[104] Steps 1101 to 1113 in FIG. 11a are equal in operation to steps 901 to 913 in FIG. 9a, except that the CAB AB XDMS 73 performs an operation of filtering contacts and storing them in the CAB UP XDMS 75. Accordingly, in step 1115, the RCS (CAB) server 72 requests the CAB AB XDMS 73 to store the changed contacts, using an XCAP/HTTP PUT. In step 1117, the CAB AB XDMS 73 stores the received contact profiles and sends a 200 OK to the RCS (CAB) server 72 as an ACK. In step 1119, the CAB AB XDMS 73 filters contacts by comparing the existing contacts of the user A with (changed) new contact profiles received from the RCS (CAB) server 72 according to the operation described in FIG. 4.

[105] In step 1121, the CAB AB XDMS 73 stores the filtered contacts in the CAB UP XDMS 75. Because the operation of storing the filtered contacts is an operation carried out between internal CAB XDMS entities, a separate protocol or an ACK message is not used. Other steps 1125 to 1139 in FIGs. 11a and 11B are equal in operation to steps 925 to 939 in FIGs. 9a and 9b.

[106] In a fourth embodiment of the present invention, the contact filtering step and the process of extracting SIP contacts and requesting "service capability information" in the first embodiment of the present invention are performed together by the RCS (CAB) server 72 in advance. Reference will be made to FIGs. 12a and 12b to describe the fourth embodiment of the present invention. Steps 1201 to 1219 in FIG. 12a are equal in operation to steps 901 to 919 in FIG. 9a. Although steps 1221 to 1229 in FIGs. 12a and 12b are equal in operation to steps 929 to 937 in FIG. 9b, this operation is performed not by the RCS (Presence) client 81, but by the RCS (CAB) server 72. Therefore, it is assumed that a Presence function serving as the Presence client 81 is included in the RCS (CAB) server 72.

[107] In addition, steps 1231 to 1237 in FIG. 12b are equal in operation to steps 921 to 927 in FIG. 9a. However, in a contact suggestion profile stored in step 1231 is included an indicator indicating whether each contact has subscribed to the RCS (CAB) service, according to the operation results of steps 1231 to 1237. In step 1239, the RCS (CAB)

client A 71 provides the received contact information to its user without the need for analysis.

[108] In the above-described fourth embodiment of the present invention, the entire analysis and filtering process has been performed by the RCS (CAB) server 72 in advance, by way of example. On the contrary, however, in a fifth embodiment of the present invention, the entire process may be performed by the RCS (CAB/Presence) client 71 later on, instead of being performed by the RCS (CAB) server 72 in advance. Reference will be made to FIGs. 13a and 13b to describe the fifth embodiment of the present invention.

[109] Steps 1301 to 1313 in FIG. 13a are equal in operation to steps 901 to 913 in FIG. 9a. In addition, steps 1315 to 1321 are also equal in operation to steps 921 to 927 in FIG. 9a. However, while the contacts stored in step 921 are converted in the RCS (CAB) format, those stored in step 1315 are stored in the CAB UP XDMS 75 without undergoing additional analysis and filtering. Further, although step 1323 in FIG. 13b is equal in operation to step 919 in FIG. 9a, its filtering operation is performed by comparing by the RCS (CAB) client A 71 the existing contacts of the user A with contacts in its address book manager 507, instead of being performed by the RCS (CAB) server 72. Subsequent steps 1325 to 1335 in FIG. 13b are equal in operation to steps 929 to 939 in FIG. 9b.

[110] As is apparent from the foregoing description, according to exemplary embodiments of the present invention, an extended address book can be provided, contributing to extending the number of targets (or contacts), subscription/non-subscription of which to a specific service can be determined. Accordingly, a user can be automatically provided with new contacts which are not in his or her address book, and thus can easily make contact not only with the friends and family members in the address book, but also with the contacts, who are related to his or her profile, but whom he or she is not aware of. In addition, when having newly subscribed to a specific service, the user can enjoy the service right away, enhancing user conveniences. Besides, for example, if the service provider sends a join invite message, the user may not be interested in it. However, if a senior of a specific school finds an address of his or her school junior through the system proposed by the present invention and sends a join invite message to him or her, then the junior is highly likely to join the service. Taking this into consideration, the service provider may easily induce many users having not subscribed to its service to subscribe to the service, contributing to a sharp increase in the number of its service subscribers.

[111] While the invention has been shown and described with reference to certain exemplary 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 invention as defined by the appended claims and their equivalents.

Claims

- [Claim 1] A method for receiving a contact in a client terminal through interworking between a messaging service and a Social Network Service (SNS), the method comprising:
sending a server a request for a contact profile list provided in the SNS;
receiving from the server a contact profile list, which is one of contact profile lists corresponding to the request and filtered by the server; and
determining whether each contact in the received contact profile list has subscribed to the same service as that of the client terminal.
- [Claim 2] The method of claim 1, wherein the contact profile list comprises at least one of a name, an e-mail address, and a phone number.
- [Claim 3] The method of claim 1, wherein the determining comprises determining whether an address of the same service as that of the client terminal is present in the received contact profile list.
- [Claim 4] The method of claim 3, further comprising:
if an address of the same service as that of the client terminal is present in the received contact profile list, adding at least one contact having the address of the same service as that of the client terminal, in a subscription list.
- [Claim 5] The method of claim 4, further comprising:
sending a service registration server a request for a service list in which the other party is registered, which corresponds to the at least one contact added in the subscription list; and
receiving the service list from the service registration server.
- [Claim 6] The method of claim 5, further comprising:
determining whether the same service as that of the client terminal is present in the service list; and
if the same service as that of the client terminal is present in the service list, notifying that the other party corresponding to the at least one contact uses the same service as that of the client terminal.
- [Claim 7] The method of claim 1, wherein the contact profile list is filtered by comparing an address book received from a Converged Address Book (CAB) XML Document Management Server (XDMS) with contact profile list received from the SNS.
- [Claim 8] A client terminal for receiving a contact through interworking between a messaging service and a Social Network Service (SNS), the client terminal comprising:

a controller for sending a server a request for a contact profile list provided in the SNS, receiving from the server a contact profile list, which is one of contact profile lists corresponding to the request and filtered by the server, and determining whether each contact in the received contact profile list has subscribed to the same service as that of the client terminal;

an Input/Output (I/O) interface for exchanging a request and a response with the server;

a memory for temporarily storing the received contact profile list;

an address book manager for storing an address book synchronized with the server; and

a user interface for informing a user whether a specific contact has subscribed to the same service as that of the user.

[Claim 9] The client terminal of claim 8, wherein the contact profile list comprises at least one of a name, an e-mail address, and a phone number.

[Claim 10] The client terminal of claim 8, wherein the controller determines whether an address of the same service as that of the client terminal is present in the received contact profile list.

[Claim 11] The client terminal of claim 10, wherein if an address of the same service as that of the client terminal is present in the received contact profile list, the controller adds at least one contact having the address of the same service as that of the client terminal, in a subscription list.

[Claim 12] The client terminal of claim 11, wherein the controller sends a service registration server a request for a service list in which the other party is registered, which corresponds to the at least one contact added in the subscription list, and receives the service list from the service registration server.

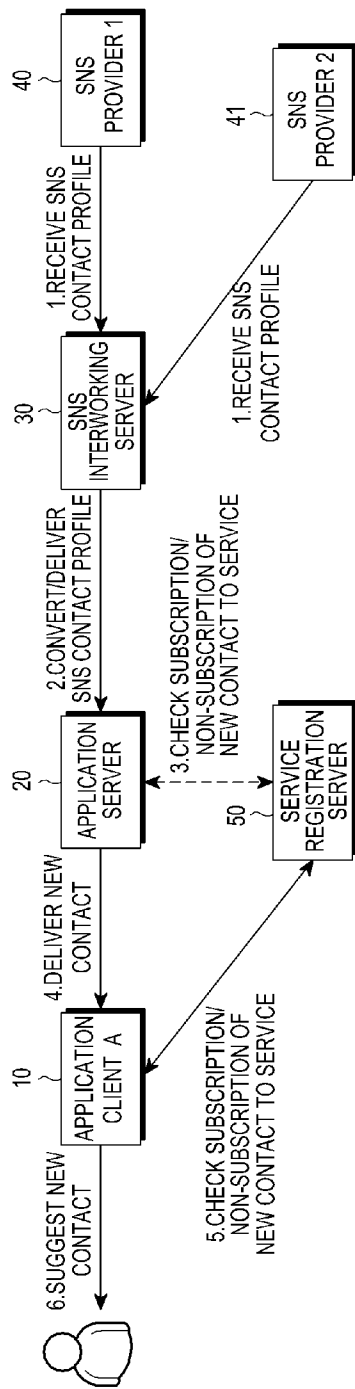
[Claim 13] The client terminal of claim 12, wherein the controller determines whether the same service as that of the client terminal is present in the service list, and if the same service as that of the client terminal is present in the service list, notifies on the user interface that the other party corresponding to the at least one contact uses the same service as that of the client terminal.

[Claim 14] The client terminal of claim 8, wherein the contact profile list is filtered by comparing an address book received from a Converged Address Book (CAB) XML Document Management Server (XDMS) with contact profile list received from the SNS.

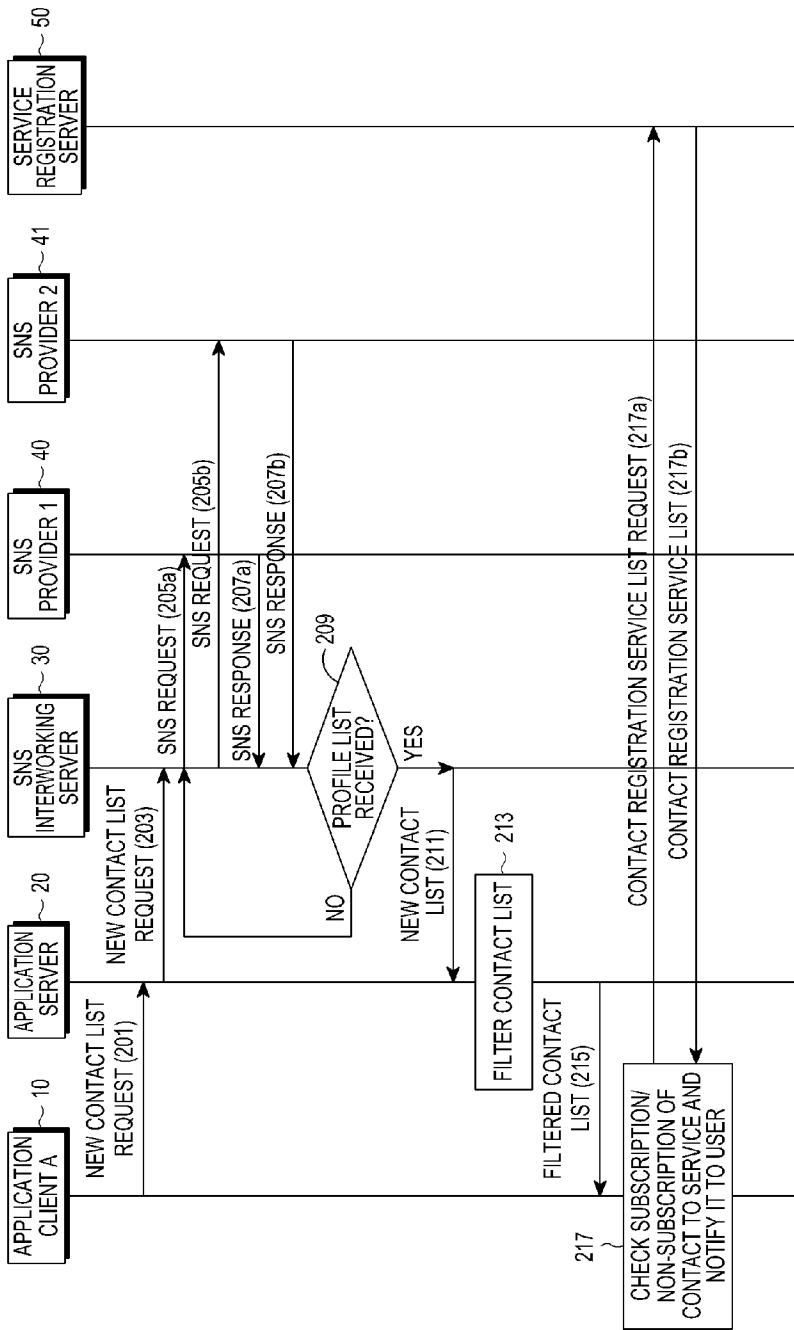
[Claim 15]

The client terminal of claim 8, wherein the a request for a contact profile list is converted in a format for the SNS and transmitted to the SNS through the server.

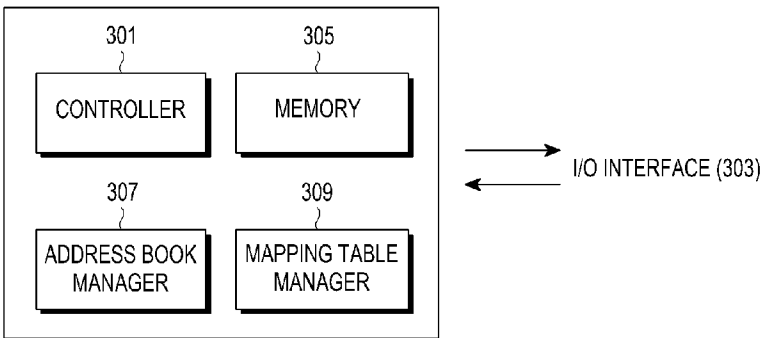
[Fig. 1]



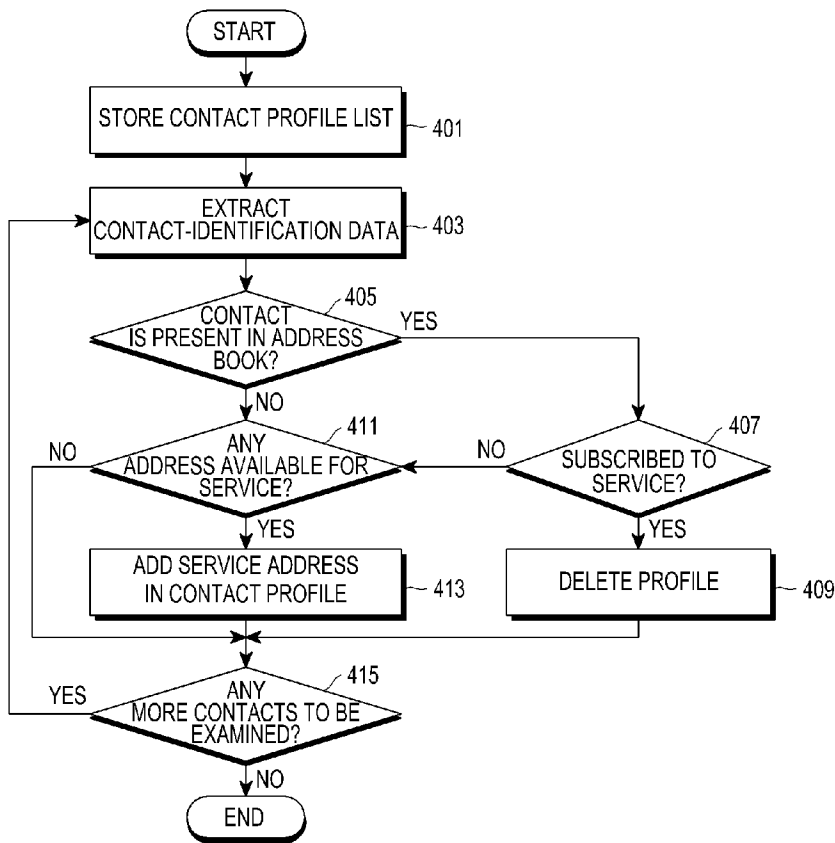
[Fig. 2]



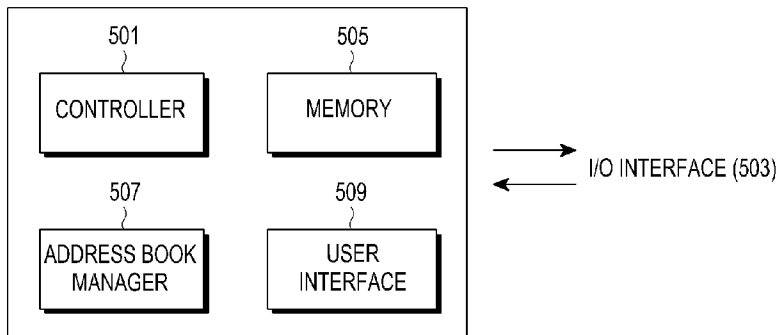
[Fig. 3]



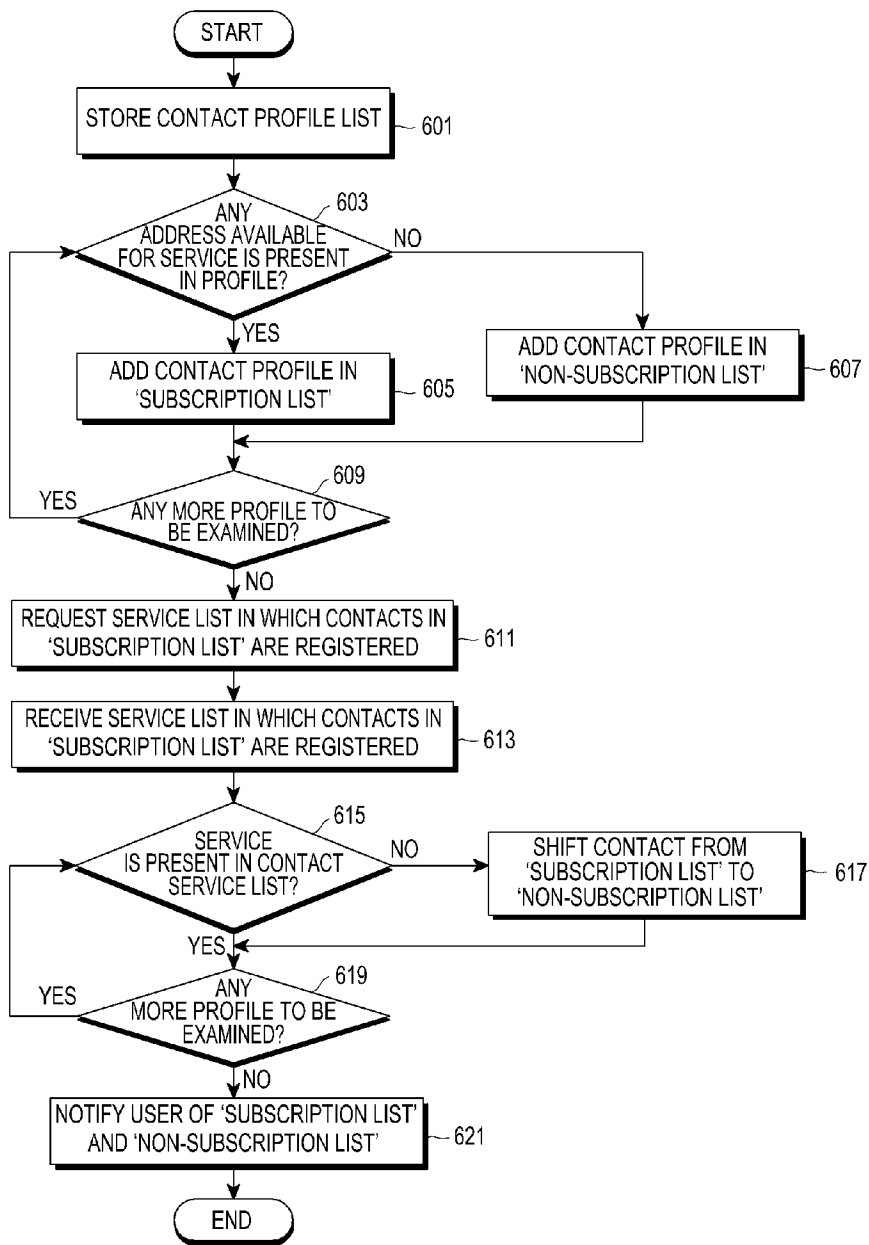
[Fig. 4]



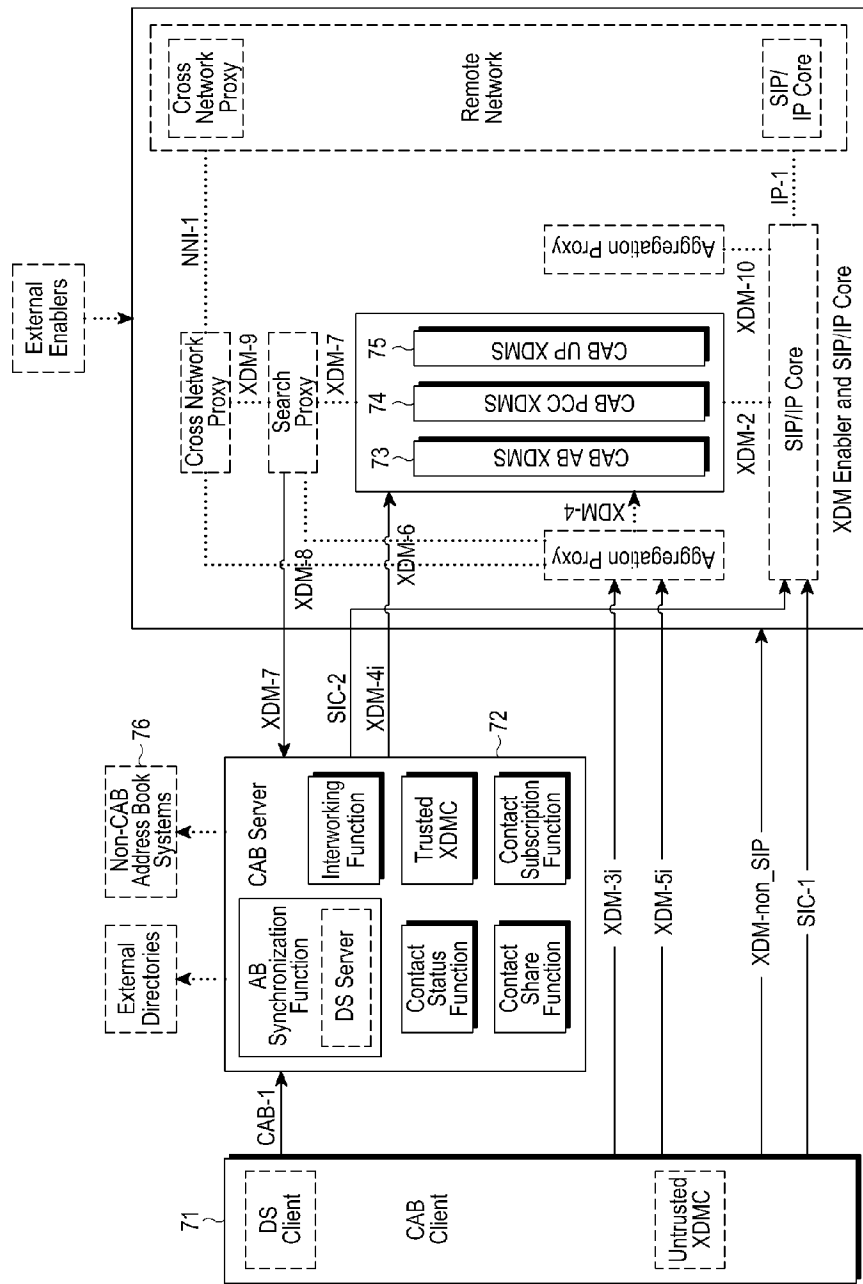
[Fig. 5]



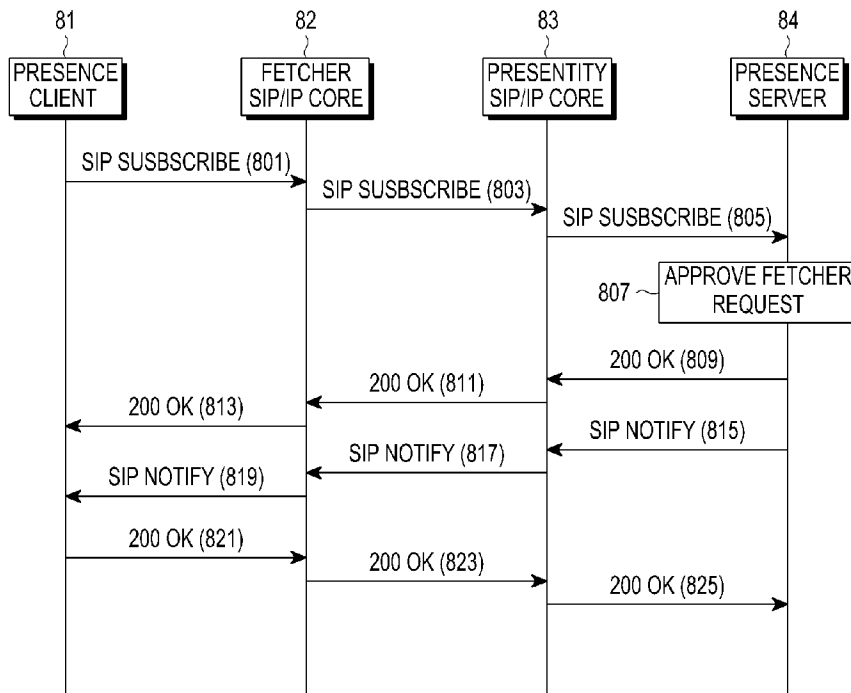
[Fig. 6]



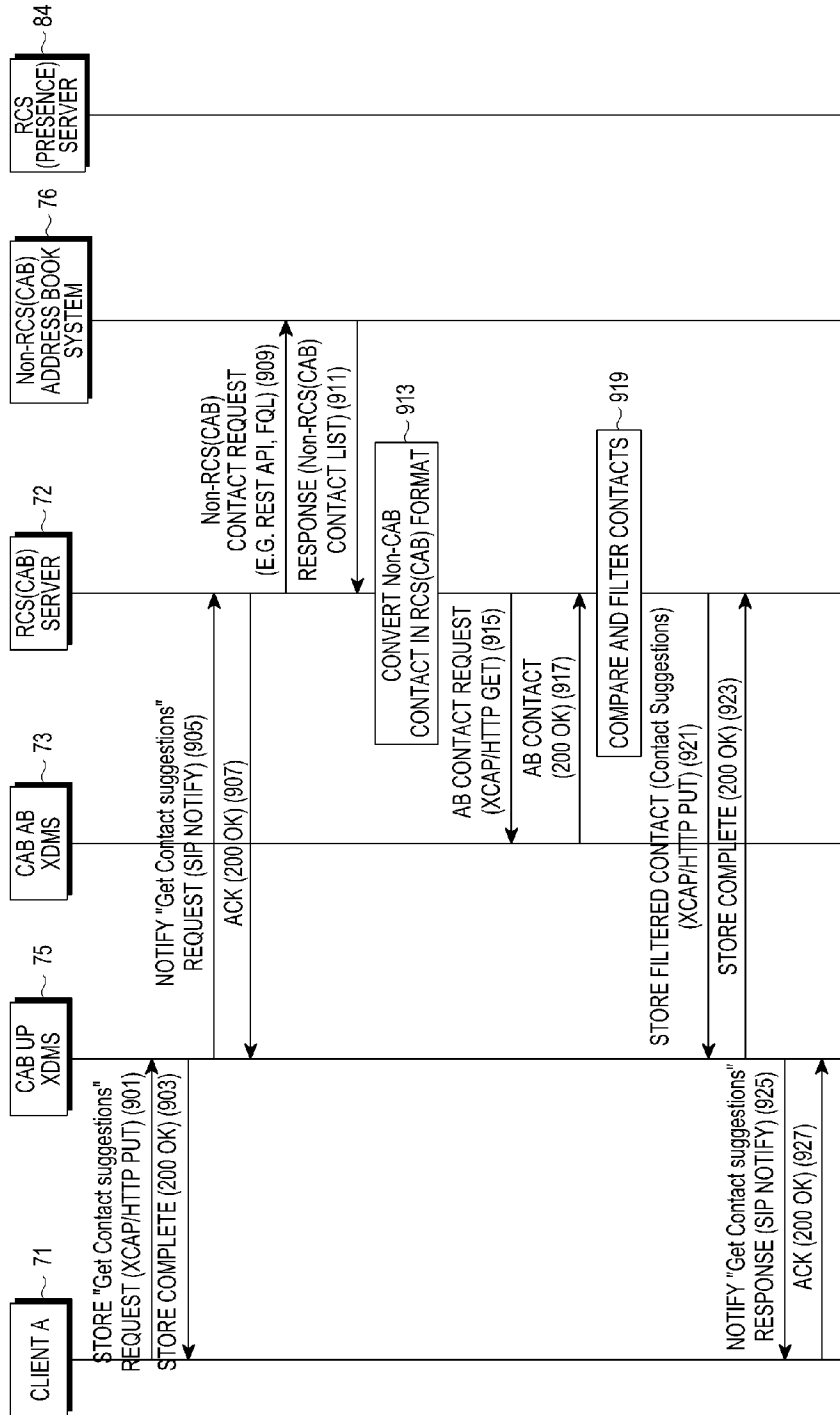
[Fig. 7]



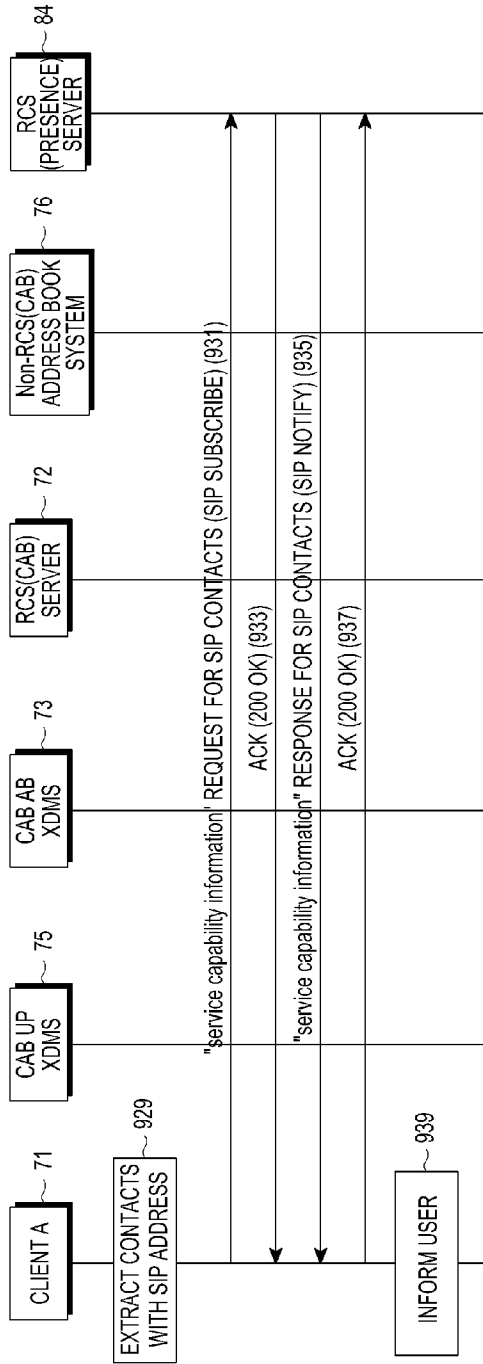
[Fig. 8]



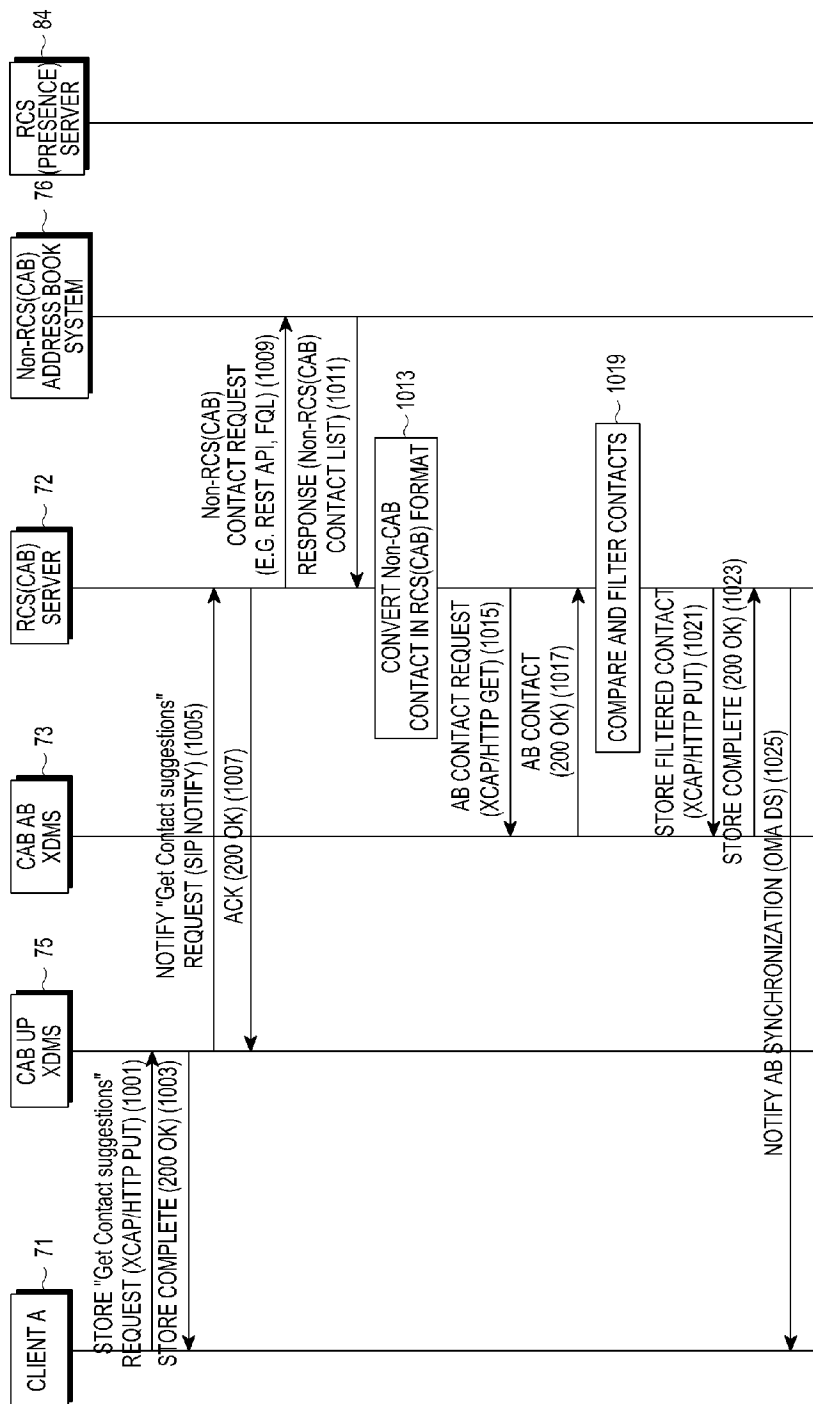
[Fig. 9a]



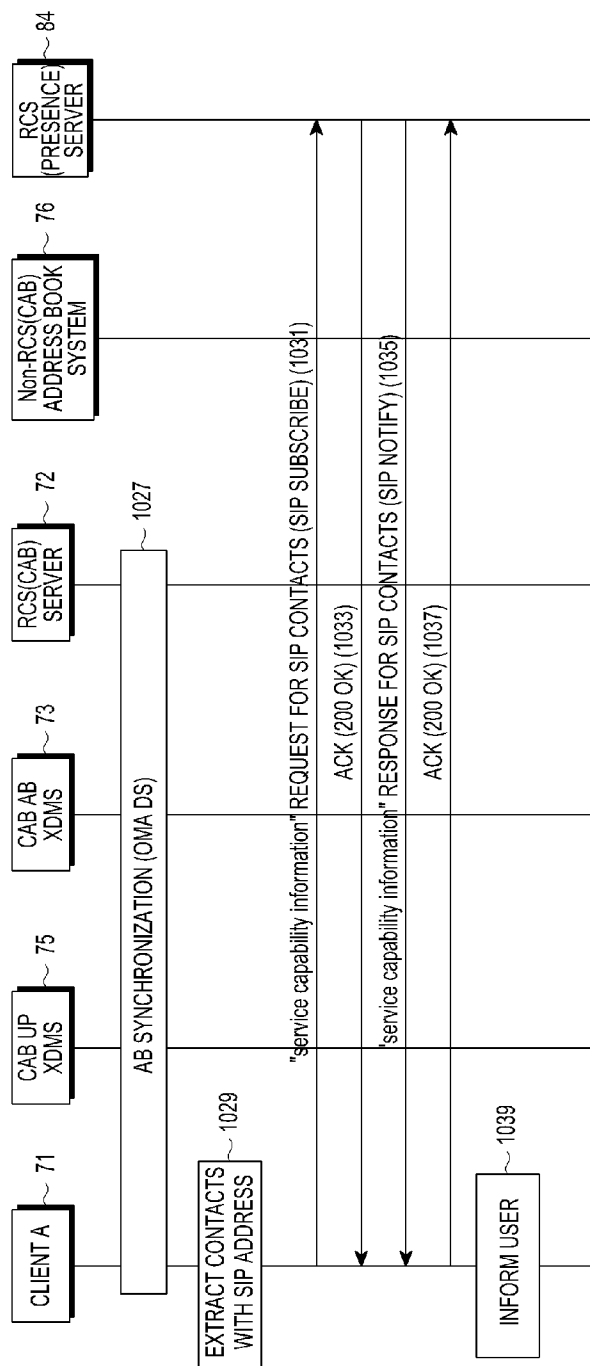
[Fig. 9b]



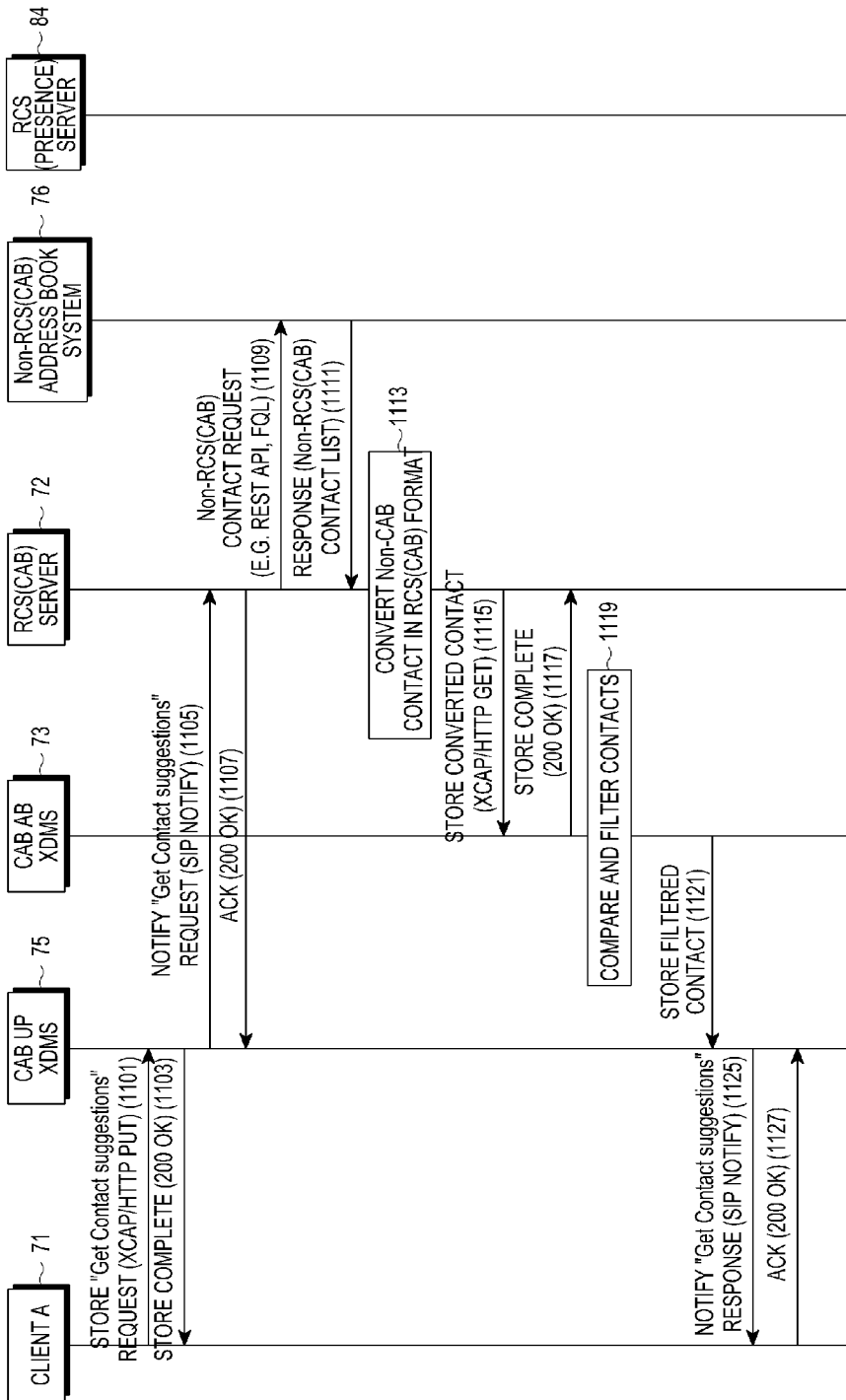
[Fig. 10a]



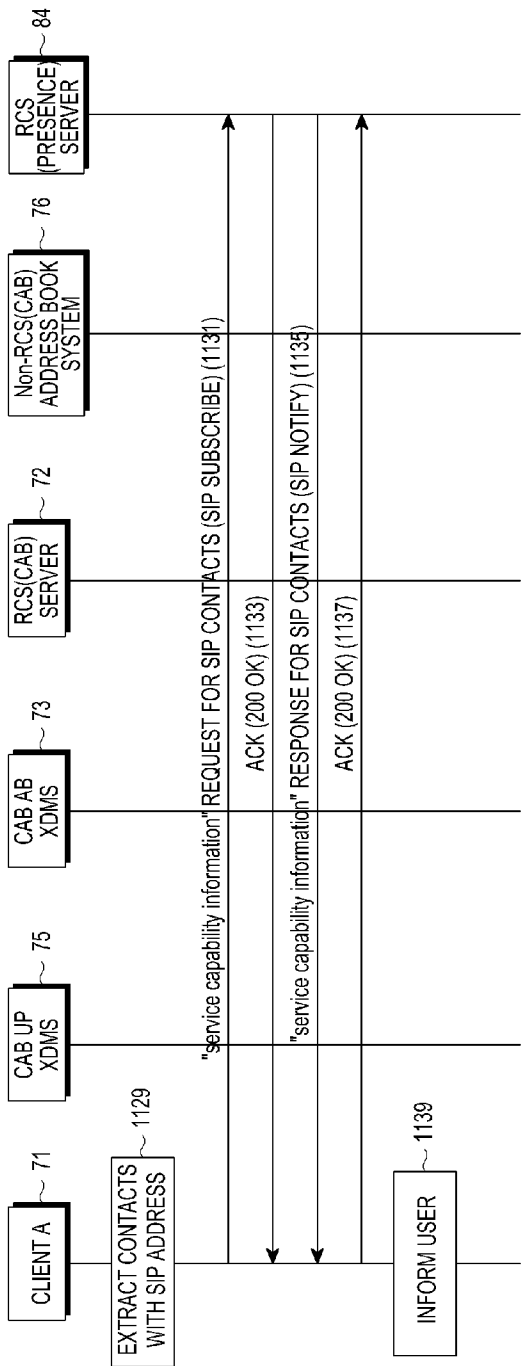
[Fig. 10b]



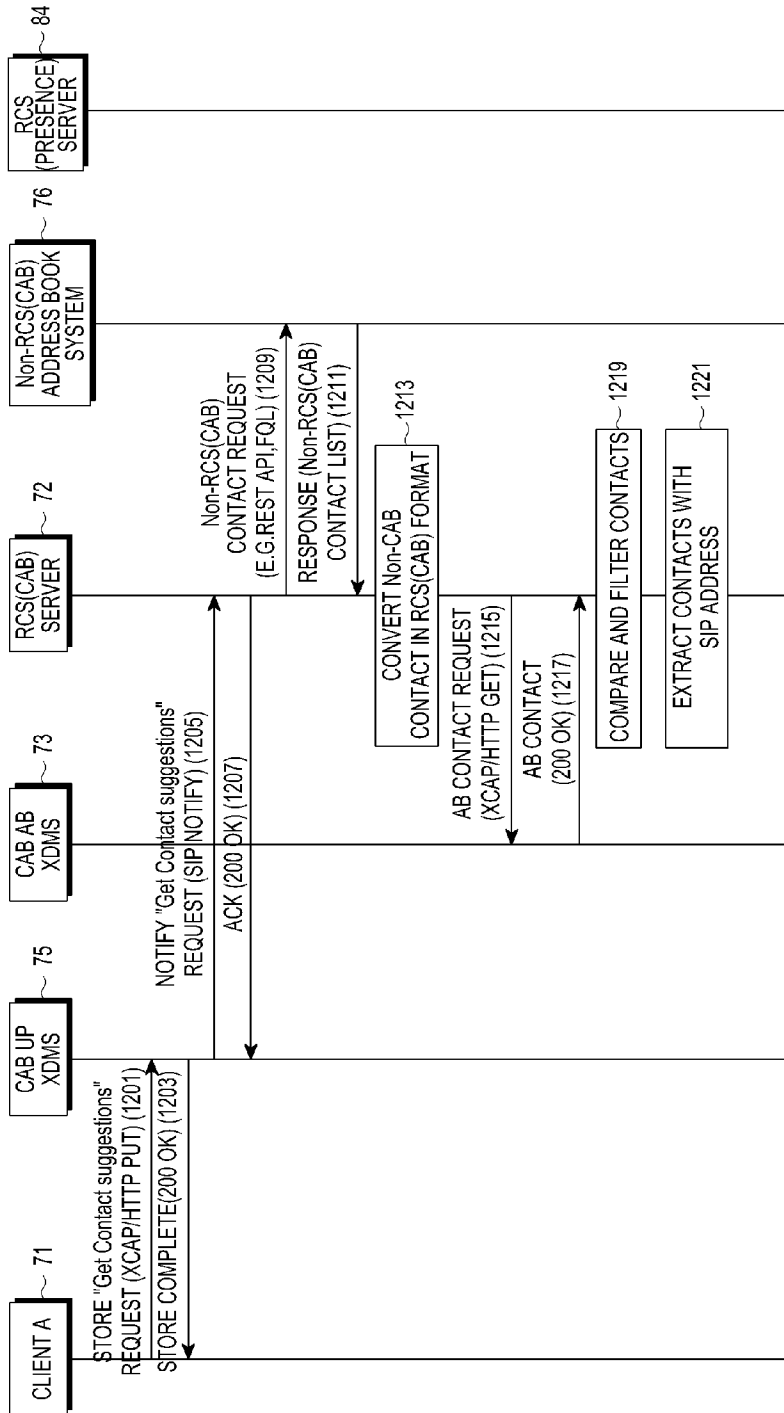
[Fig. 11a]



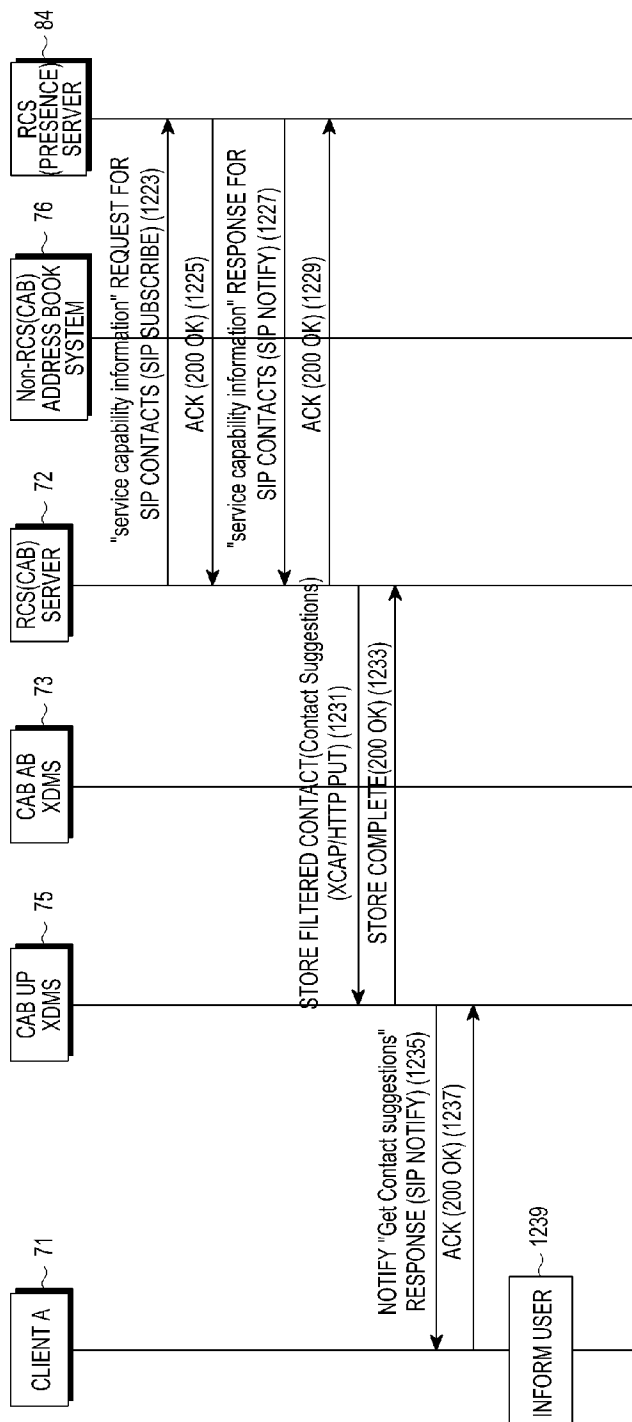
[Fig. 11b]



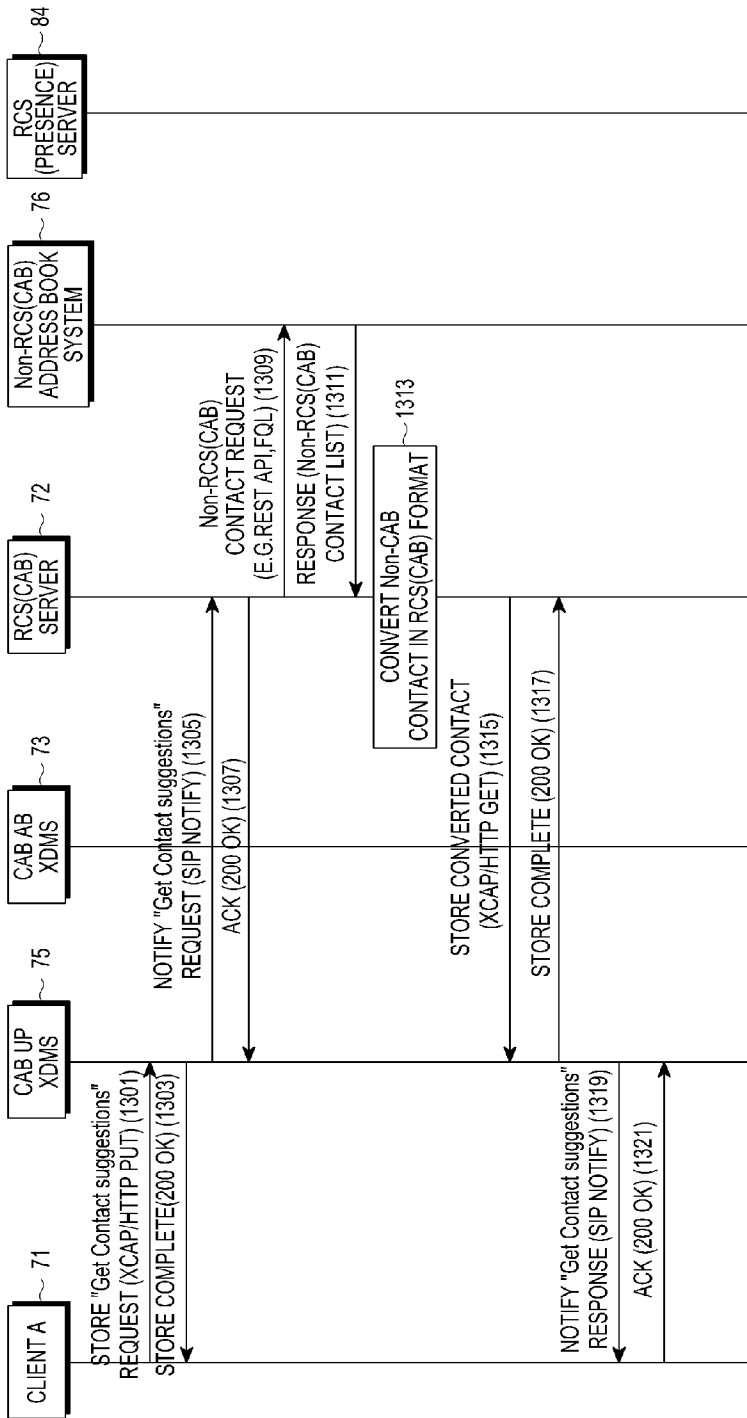
[Fig. 12a]



[Fig. 12b]



[Fig. 13a]



[Fig. 13b]

