

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
6 August 2009 (06.08.2009)

PCT

(10) International Publication Number
WO 2009/097032 A1

(51) International Patent Classification:

H04W 80/04 (2009.01) *H04W 4/16* (2009.01)
H04W 8/26 (2009.01)

(21) International Application Number:

PCT/US2008/082573

(22) International Filing Date:

6 November 2008 (06.11.2008)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

233/DEL/2008 28 January 2008 (28.01.2008) IN

(71) Applicant (for all designated States except US): **MOTOROLA, INC.** [US/US]; 1303 East Algonquin Road, Schaumburg, Illinois 60196 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **TRIPATHI, Shashi Bhusan**, [IN/IN]; 30, 5th B Cross, 16th Main, BTM Layout II Stage, Bangalore, Karnataka 560076 (IN). **AVASARALA, Ranjit**, [IN/IN]; bldg# 34, Apt#202,

Mantri Residency Banerghatta Road, Bangalore 560076 (IN). **DILIPUMAR SAKLIKAR, Samir**, [IN/IN]; B-15 Madhav Nagar R A Kidwai Rd, Wadala Mumbai, Maharashtra 400031 (IN). **SAHA, Subir**, [IN/IN]; B 602, Salarpuria Silverwood, Bangalore, Karnataka 560093 (IN).

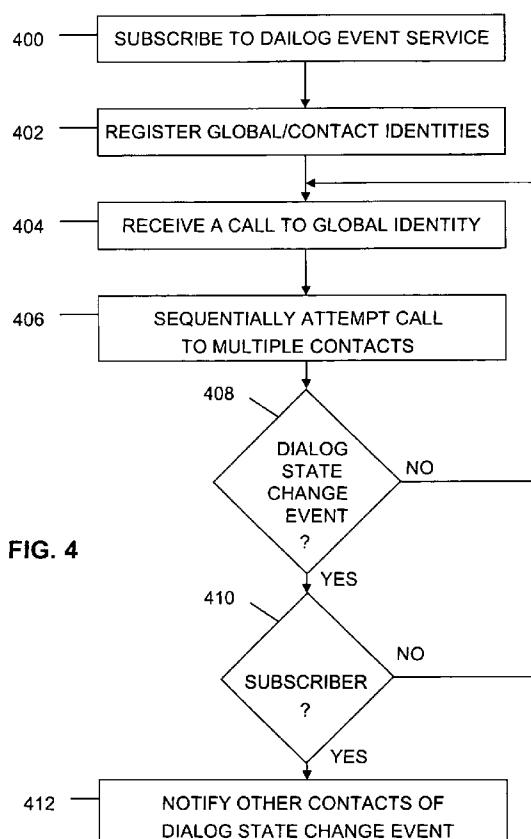
(74) Agent: **MANCINI, Brian M.**; 1303 East Algonquin Road, Schaumburg, Illinois 60196 (US).

(81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, 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,

[Continued on next page]

(54) Title: MANAGING CALL DELIVERY IN AN INTERNET PROTOCOL COMMUNICATION SYSTEM



(57) Abstract: A system and method for managing sequential call delivery in an internet protocol communication system includes a step 402 of registering a global identity and multiple contact identities for a user. A next step 404 includes receiving a call to the global identity of the user. A next step 406 includes sequentially attempting the call to the multiple contact identities of the user. A next step 408 includes determining that one of the contacts handles the call. A next step 412 includes notifying at least the sequentially-previous contacts that the call has been handled.



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, MT, NL, NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report*

MANAGING CALL DELIVERY IN AN INTERNET PROTOCOL COMMUNICATION SYSTEM

5 FIELD OF THE INVENTION

The present invention relates in general to communication systems and more specifically to techniques for managing call delivery in an internet protocol communication system.

10 BACKGROUND OF THE INVENTION

Recent technology innovations have expanded the capabilities of the Internet to include communication services. One such service for multimedia communication provides a call control protocol for use in an Internet Protocol (IP) Multimedia Core Network Subsystem (IMS). IMS conforms to Session Initiation Protocol (SIP) and
15 Session Description Protocol (SDP), as are known in the art, and where all IMS entities are allocated an IP address. In IMS, a mobile terminal or other communication device, referred to as User Equipment (UE) herein, provides the SIP User Agent (UA) role. A Call Session Control Function (CSCF) provides the SIP proxy server role. Comparable UE and proxy server roles are provided in non-IMS
20 communication systems, as are known in the art.

Typically in IMS, each UE or Subscriber is allocated a private user identity by a home communication network operator, such as a Home Subscriber System (HSS). The private user identity is available to the SIP application within the UE. The UE is also allocated at least one public user identity (PUI) provisioned on the HSS. The
25 PUI is of the form of a Session Initiation Protocol Uniform Resource Identifier (SIP

URI). A PUI may be shared among multiple UEs having different private user identities and IP addresses. A UE is required to submit a PUI, private user identity, and a domain name in a registration request in the IMS. Correspondingly, in non-IMS networks, the user has one global SIP URI and can have multiple contact URIs

5 registered for that global URI. At the time of registration, a registration message contains all the contacts identities. In either case, a call processing server can support registration on behalf of a known UE. In particular, the call processing server may subscribe (using SIP URI) on behalf of the PUI registered at a Serving CSCF (S-CSCF).

10 Where a user has multiple UEs, a call placed to the user's PUI may be delivered to all of the user's registered UEs (i.e. "forking"). In other words, when a caller calls a user's public user identity, the call processing server "forks" the call to all the registered UE contacts. This service is convenient to end-users because they increase their reachability beyond a single device. Such forking can be done either in
15 parallel or sequentially as per the call processing server's policy or user settings.

In parallel forking, the call processing server forwards the call from the caller to all of the user's UE contacts in parallel. In this case, either of the UE contacts can accept, reject, or ignore the call.

In sequential forking, the call processing server forwards the call from caller
20 to a first of the user's UEs. If the first UE does not answer after a preset number of rings, then the call is attempted at the remaining UE contacts one after the other. If one of the UE contacts accepts or rejects the call, the call is not attempted at the remaining UE contacts.

However, a problem arises in the above scenarios. In particular, only the UE contact that actually handled the call is aware of the actual outcome of the call.

Additionally, the other UE contacts get a different result due to the handling of the call at a particular UE. For example, if a second UE contact handles the call, the other

5 UE contacts may not get any information that the call has been handled. In particular, any UE contact that received the Call attempt before the handling (i.e. answering or rejecting) second UE, would show a “missed call” to that User, when the Call was actually handled at the handling UE. Additionally, the UE contacts after the Call Attempt do not have any idea that such a Call attempt was made and handled at the
10 second UE. Hence, when a subscriber looks at the various call-logs or on-screen notification on their other UEs, the subscriber gets an inconsistent experience.

One solution to the problem has been to supply a parallel CANCEL message to the remaining UE contacts that contains a SIP reason header explaining that this call is being cancelled because the call was already handled by another UE contact.

15 This CANCEL message is identical to when caller would have hung or if call was unanswered. However, the reason header in the CANCEL message does not provide adequate information to the other UEs as to why the call was cancelled. Moreover, this solution can only be used to solve the parallel forking scenario, and not the sequential forking scenario.

20 Another solution to the problem is to notify the original called party that a call has been diverted from him/her. This only applies to those cases where the original called party is skipped due to a diversion policy, which is different from the

ambiguous “Missed Call” problem solved in the present invention. Moreover, this solution still does not inform a user that the call has already been handled.

Another solution to the problem is to have a SIP UE SUBSCRIBE for information about certain call events for other users. Subsequently, the device
5 receives notifications through NOTIFY messages. However, these NOTIFY messages address other users and do not address the notification that the call has been handled by another UE of the same user. Moreover, these NOTIFY messages do not carry related information regarding CANCEL messages.

Another solution to the problem creates a Missed Call Event Package that a
10 user can subscribe to where a callee receives notifications about all missed calls. However, this solution does not disclose any events to tell callee why the call was cancelled.

Another solution to the problem describes software on a phone that can correlate a missed-call-list with outgoing call-lists to determine which calls need to be
15 returned. However, this solution only works on a single device, and does not address the multiple phone scenario problem described for the present invention. Moreover, this solution does not disclose any events/information to a callee for why the call was cancelled.

What is needed is a synchronizing mechanism so that the actions taken at a
20 particular UE contact address are consistently updated across all the other registered UE contacts in a sequential forking scenario. For example, if one UE contact handles the call, the action taken at that UE contact should be updated at the other UE contacts. So if a call is answered or rejected or if a caller is blocked at the one UE

contact, the other sequentially-addressed UE contacts need to see the same information instead of “missed call” or any other information they may see.

BRIEF DESCRIPTION OF THE DRAWINGS

5 The accompanying figures, where like reference numerals refer to identical or functionally similar elements throughout the separate views and which together with the detailed description below are incorporated in and form part of the specification, serve to further illustrate various embodiments and to explain various principles and advantages in accordance with the present invention.

10 FIG. 1 is a flow diagram illustrating a prior art technique for managing call delivery;

 FIG. 2 is a block diagram illustrating a system for managing call delivery, in accordance with the present invention;

 FIG. 3 is a flow diagram illustrating a system for managing call delivery, in
15 accordance with the present invention; and

 FIG. 4 is a flow chart illustrating a method for managing call delivery, in accordance with the present invention.

 Skilled artisans will appreciate that common but well-understood elements that are useful or necessary in a commercially feasible embodiment are typically not
20 depicted or described in order to facilitate a less obstructed view of these various embodiments of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

The present invention defines a call delivery management protocol that provides a synchronizing mechanism so that the actions taken at a particular UE contact address are updated across all the registered UE contacts in a sequential
5 forking scenario. So if a call is answered or rejected or if a caller is blocked at the one UE contact, the other sequentially-addressed UE contacts need to see the same information instead of “missed call” or any other information they may see.

The present invention is described herein in terms of an Internet Protocol Multimedia Core Network Subsystem (IMS). However, it should be recognized that
10 the present invention is also applicable to non-IMS architectures.

The present invention provides an enhancement existing protocols in order to allow users to subscribe to state change and event that caused the change at any of their contacts. In one embodiment, this is accomplished by having the users’ registered contacts subscribe to a SIP Dialog Event package using their own URI as
15 the request URI. In this way, upon receiving SUBSCRIBE request with the event-package as “dialog” and the request URI being the same as the sender URI, the call processing server: a) understands that the subscription is for the user and should be used for notifying all the registered contacts of the request URI, a separate state machine needs to be maintained for every contact UE, and any change in the state
20 machine of any of the UE contacts should be notified to the remaining UE contacts, or at least the sequentially-previous UE contacts. In another embodiment, the UE contacts may choose to selectively send multiple SUBSCRIBE messages to other registered UE contacts, subscribing for their respective dialog-state even package.

This has the advantage that it allows more control, such as the ability to completely bypass (not ask for any dialog-state information from) any UE contacts, which usually end up “missing” the call. For example, a set-top based Client would surely miss most of the calls during the day time, when there is no one home. The other advantage is

5 that instead of enabling the default subscription for all the UE contacts, now only those UE contacts that are able to handle the multiple notification information (or can present it to the User in a GUI-based intuitive manner) may choose to send the Subscription request towards the other UE contacts.

The present invention also proposes an enhancement to existing calling

10 protocols by adding additional events and states to the dialog event package to complete the proposed solution of synchronizing mechanism between contacts. The additional events added are: blocked and diverted

Advantageously, the present invention reuses an existing dialog event subscription package, and provides further useful and detailed information to some or

15 all of a user’s UE contacts about an action that took place for a call. The information is conveyed using standard SIP event package mechanism as defined in RFC 3265. In addition, since this is subscription based, users can subscribe to it as and when required. Subscribers who have multiple contacts registered will find the present invention useful as all their contacts can be synchronized with any action that takes

20 place at any of their registered contacts.

As used herein, the terms such as user equipment (UE) or communication device generally refer to end user devices such as fixed or/and WiFi SIP phones, cellular or mobile radiotelephones, WiMax SIP UA, two-way radios, messaging

devices, personal digital assistants, personal assignment pads, personal computers equipped for wireless operation, a cellular handset or device, or the like, or equivalents thereof provided such units are arranged and constructed for operation in accordance with the various concepts and principles of the present invention and

5 operating under appropriate specifications, standards, and protocols as discussed and described herein. In the example herein, IMS is an overlay technology, which doesn't address under layers, as it can be from Ethernet, WiFi, WiMax, Cellular, to Cable set top box or DSL/DSLAM. It should be noted that the present invention is applicable to any type of access that provides an IP network layer.

10 The principles and concepts discussed and described may be particularly applicable to communication units, devices, and systems providing or facilitating packet based multimedia communications services or voice, data, or messaging services over communication networks, such as conventional two way systems and devices, various cellular phone systems including analog and digital cellular, CDMA
15 (code division multiple access) and variants thereof, GSM (Global System for Mobile communications), GPRS (General Packet Radio System), 2.5 G and 3G systems such as UMTS (Universal Mobile Telecommunication Service) systems, Integrated Digital Enhanced Networks, and variants or evolutions thereof. The principles and concepts described herein may further be applied in devices or systems with shorter range
20 communications capabilities, such as IEEE 802.xx, Bluetooth, WiFi or WiMAX and the like that preferably utilize CDMA, frequency hopping, orthogonal frequency division multiplexing, or TDMA access technologies and one or more of various networking protocols, such as TCP/IP (Transmission Control Protocol/Internet

Protocol), IPX/SPX (Inter-Packet Exchange/Sequential Packet Exchange), Net BIOS (Network Basic Input Output System) or other protocol structures.

Further in accordance with various exemplary and alternative exemplary embodiments, the packet based RAN can include a Code Division Multiple Access (CDMA) RAN, a Global System Mobile (GSM) RAN, Universal Mobile
5 Telecommunication System (UMTS) RAN, a Data Only (DO) RAN, a High Rate Packet Data Access (HRPDA) RAS, a Wireless Local Area Network (WLAN) RAN, or an Evolution Data Voice (EVDV) RAN. The exemplary RAN should support communications under the IP Multimedia Core Network Subsystem (IMS)
10 specifications, for example as outlined in the Third Generation Partnership Project (3GPP) Technical Specification (TS) 24.228 for communications using Session Initiation Protocol (SIP), Session Description Protocol (SDP) and variants thereof. It should be noted that in accordance with the above noted standards, such as 3GPP release 8 specification TS 24.228, multimedia streams can be transmitted over
15 RTP/UDP (Real Time Transfer Protocol/Universal Datagram Protocol).

It will be appreciated that other 3GPP specifications and standards may also be relevant herein. Further in accordance with various exemplary embodiments, the present invention can be implemented as a higher layer, such as application layer software application, in which case lower protocol layers, such as the data link layers,
20 can be interchangeable without departing from the intended scope of the invention, provided they support packet switched communication.

The instant disclosure is provided to further explain in an enabling fashion the best modes of making and using various embodiments in accordance with the present

invention. The disclosure is further offered to enhance an understanding and appreciation for the inventive principles and advantages thereof, rather than to limit in any manner the invention. The invention is defined solely by the appended claims including any amendments made during the pendency of this application and all
5 equivalents of those claims as issued.

It is further understood that the use of relational terms, if any, such as first and second, top and bottom, and the like are used solely to distinguish one from another entity or action without necessarily requiring or implying any actual such relationship or order between such entities or actions. The invention may further include a process
10 with steps, procedures, or the like. Where a plurality of processes or steps are indicated, they may be performed in any order, unless expressly and necessarily limited to a particular order. Steps or processes that are not so limited may be performed in any order. In certain cases, these steps or processes may be repeated a number of time or may loop infinitely as required or until a particular event occurs or
15 the like.

Much of the inventive functionality and many of the inventive principles are best implemented with or in software programs or instructions and integrated circuits (ICs) such as application specific ICs. It is expected that one of ordinary skill, notwithstanding possibly significant effort and many design choices motivated by, for
20 example, available time, current technology, and economic considerations, when guided by the concepts and principles disclosed herein will be readily capable of generating such software instructions and programs and ICs with minimal experimentation. Therefore, in the interest of brevity and minimization of any risk of

obscuring the principles and concepts according to the present invention, further discussion of such software and ICs, if any, will be limited to the essentials with respect to the present invention.

FIG. 1 is a diagram illustrating a prior art scenario for call delivery management. An originating caller 102 desires to engage in a communication session with a user (callee) associated with a PUI, so that a video and/or audio media stream can be exchanged therebetween, for example in accordance with IMS (Internet protocol Multimedia Subsystem) and SIP procedures. It will be appreciated that the session would be established through, for example, a call processing server 100 which can facilitate communication of the audio and/or video streams to a plurality of target communication units, 104, 106, 108 of the user. The call processing server has already provided the basic validation and registration of the user identity, as is known in the art. The call processing server is also aware of the multiple registered UE contacts 104, 106, 108 under the user's global PUI.

To initiate the communication the caller 102 inserts a P-Asserted-Identity representing the PUI (from the initial IMS registration) of the callee in a SIP INVITE call request message 110. In particular, the SIP INVITE message contains the URI of the callee in the topmost Route Header. The call processing server receives this message 110 on behalf of the callee. Knowing that the callee has three private UE identities, for example, the call processing server 100 attempts this call request 112 in parallel to all three UE contacts 104, 106, 108 of the callee. If none of the UE contacts answers the call within a specified time or number of rings, all of the UEs will then show that this call was missed.

If however, one of the UE contacts is used to answer the call request, the second UE contact 106 in this example, a SIP 200 OK message 118 is returned to the call processing server 100 by the answering UE 106, which then forwards this acknowledgement 120 back to the originating caller 102. Communications (not shown) can then proceed between the caller 102 and UE contact 106, as is known in the art.

However, the first UE contact 104 and the third UE contact 108 will still indicate that the call was missed when in fact it was actually completed through the second UE contact 106. Accordingly, the caller 102 sends a CANCEL message 122 to the call processing server 100 indicating in a reason header that the second UE contact 106 has responded to the call and that the call processing server 100 should send a message to the other private UEs of the callee to cancel their “missed call” indications. The call processing server 100, knowing that the second UE contact has answered the call from the reason header, then sends CANCEL messages 124, 126 in parallel to the first UE contact 104 and the third UE contact 108 indicating the that call was not in fact missed, but that the second UE contact answered the call. These UE contacts 104, 108 can then delete their missed call indications.

Referring to FIGs. 2 and 3, a call delivery management technique for a sequentially attempted call is shown, in accordance with the present invention. The following scenario is described herein where a user “Bob” has registered a global URI as SIP:bob@domain.com. The user has also registered following private UE contact URIs; SIP:bob@home.domain.com (assign to a first UE contact 104),

SIP:bob@office.domain.com (assigned to a second UE contact 106), and
SIP:bob@mobile.domain.com (assigned to a third UE contact 108).

In accordance with the present invention, the user subscribes 301 each of its private UE contacts to a Dialog Event service package using the user's own URI as a request URI. The Dialog Event service operates to provide a dialog state of the user's
5 UE contacts to the other of the user's UE contacts. The Dialog Event service package can be controlled by the call processing server 100 (as shown), or by any other network entity. Upon receiving the SUBSCRIBE request 301 with the event as dialog and the request URI the same as the sender URI, the call processing server 100 does
10 following: a) understands that the subscription is for one user under a global PUI and should be used for notifying all the registered contacts 104, 106, 108 of the request URI, b) a separate state machine needs to be maintained for every contact UE, and c) any change in the state machine of any of the contacts should be notified to the remaining contacts.

15 An originating caller 102 desires to engage in a communication session with the user, Bob (callee) at SIP:bob@domain.com, so that a video and/or audio media stream can be exchanged therebetween, for example in accordance with IMS (Internet protocol Multimedia Subsystem) and SIP procedures. It will be appreciated that the session would be established through, for example, a call processing server 100 which
20 can facilitate communication of the audio and/or video streams to a plurality of target communication units, 104, 106, 108 of the user. The call processing server has already provided the basic validation and registration of the user identity, as is known

in the art. The call processing server is also aware of the multiple private identities of the UE contacts 104, 106, 108 under the user's global PUI.

To initiate the communication the caller 102 inserts a P-Asserted-Identity representing the PUI (from the initial IMS registration) of the callee in a SIP INVITE call request message 110. In particular, the SIP INVITE message contains the URI (SIP:bob@domain.com) of the callee in the topmost Route Header. The call processing server receives this message 110 on behalf of the callee. Knowing that the callee has three private UE identities, for example, the call processing server 100 forwards ("forks") this call request 112 sequentially to all three registered UE contacts 104, 106, 108 of the callee.

In particular, the call processing server 100 attempts the SIP INVITE message 112 to a first UE contact 104. If the first UE contact 104 does not answer the call within a specified time or number of rings, the call processing server 100 then attempts the SIP INVITE message 114 to the second UE contact 106 and the first UE contact 104 would indicate a missed call). If the second UE contact 106 does not answer the call within a specified time or number of rings, the call processing server 100 then attempts the SIP INVITE message 116 to the third UE contact 108 (and the second UE contact 106 would indicate a missed call). If the third UE contact 108 does not answer the call within a specified time or number of rings, then all of the UEs will then show that this call was missed.

If however, one of the UE contacts is used to answer the call request, the second UE contact 106 in this example, a SIP 200 OK message 118 is returned to the call processing server 100 by the answering UE 106, which then forwards this

acknowledgement 120 back to the originating caller 102. Communications (not shown) can then proceed between the caller 102 and UE contact 106, as is known in the art.

However, the first UE contact 104 (and possibly the third UE contact 108 if
5 the call was attempted that far) will still indicate that the call was missed when in fact it was actually completed through the second UE contact 106. In present practice, the third UE contact has no idea about this Call, since this is a case of sequential forking. Hence, when the User sees the Call Records on the respective devices, she gets an inconsistent experience for the same calling User. Additionally, the User would need
10 to check all of her devices, to know whether any Calls were missed, rather than being able to check a single Device and have the updated information about all their calls as can be done in the present invention. Accordingly, if the call processing server 100 has ascertained that the user has subscribed to a Dialog Event service that operates to provide a dialog state to a user, the call processing server 100 sends a NOTIFY
15 message 300 to at least the sequentially-previous user equipment of the first UE contact 104 (and optionally 310 to the third UE contact 108 if an INVITE message 116 was sent), in accordance with the Dialog Event package service, indicating in a header a particular dialog state change event of the second UE contact 106 indicating that the second UE contact 106 has in fact responded, and how responded, to the call
20 and that the other UE contacts 104 (and optionally 108) of the callee should delete their “missed call” indications, which is acknowledged by the other UE contacts 104 (and possibly 108) in 200 OK acknowledgements 320 (and possibly 330). Preferably, the other UE contact(s) could also indicate other information on how the second UE contact 106 handled the call. For example, the dialog state change event from the

second UE contact 106 could indicate whether it blocked the call, accepted the call, or rejected the call. It should be noted that the call processing server 100 needs no prompting from the caller for sending the NOTIFY messages, inasmuch as the call processing server 100 knows that there was a dialog state change event from the response message 118 and that the second UE contact 106 has answered the call.

Since the present invention provides sequential forking of the INVITE message to the UE contacts, it may be that one of the contacts (e.g. second contact 106) answers the call before the call processing server can attempt the INVITE message to further UE contacts (e.g. third UE contact 108). If this is so, the third UE contact 108, may never know that there was a call and that it was missed at the first UE and answered at the second UE, whereas the first UE contact 104 would know about the missed call. Therefore, there would be no need to delete a missed call indication from the third UE contact 108, and the call processing server would know not to send a NOTIFY message 310 to the third UE contact 108. However, even in this case it may be beneficial to send the NOTIFY message 310 to the third UE contact 108 in order for a user at the third contact 108 to realize that a call was completed to another contact 106.

As used herein, the communication system is an Internet Protocol (IP) Multimedia Core Network Subsystem (IMS) network, and the global identity is a public user identity of the user and the multiple contact identities are private user identities of the user. Where the communication system is a non-IMS network, the global identity is a global Session Initiation Protocol Uniform Resource Identifier

(SIP URI) of the user and the multiple contact identities are multiple contact URIs of the user.

In operation, the present invention modifies exist call control protocols that define support for forking of incoming SIP requests to multiple registered contacts in both sequential and parallel fashion depending on user configuration. The main problem in both the types of forking is that only the registered contact that actually handled the call is aware of the actual outcome of the call. The rest of the contacts either get a missed call notification or nothing at all. This is not a desirable situation when the users want all their registered devices to be synchronized with whatever call treatment happened at either of their registered devices – be it call acceptance, diversion of call, blocking of caller, etc.

These behaviors could be undesirable when there is a need for all the registered contacts to be synchronized with each other.

When the registered contact addresses want to be synchronized with the actual outcome of the forked request, then each of the registered contact address shall exhibit following behavior:

1. Each of the contact addresses SHALL subscribe to the registration event package of the UE to receive the contact addresses registered for the UE. Upon receiving the list, each contact address would filter its own URI and generate a subscription to the remaining contact address' dialog event package.
2. The registered contacts MAY also specify a filter criteria for receiving the notifications about any change in the dialog state at any of the registered contacts.

When multiple registered contact addresses of UE have subscribed to each other's dialog event package, the following behaviors SHALL be exhibited by the S-CSCF when an incoming SIP request is forked:

1. For every forked request, the S-CSCF generates a notification to all the
5 contact addresses other than the contact address that actually handled the call.
2. The notification information is sent using the dialog event package semantics, as are known in the art.

It should be noted that the enhanced SIP Dialog event package, as is known in the art, contains additional events that are required for notifying the actual outcome of the
10 forked request.

When multiple contacts of a UE subscribe to each other's dialog event package for receiving notification about each other's dialog state and events, the following events of the dialog event package, as are known in the art, could be used:

1. Cancelled – The user at one of the multiple contacts has cancelled the
15 incoming call.
2. Rejected – The user at one of the multiple contacts has rejected the incoming call.
3. Local-bye – The user at one of the multiple contacts has terminated the call by sending a BYE.
- 20 4. Remote-bye – The incoming call at one of the multiple contacts got terminated by remote user by receiving a BYE.

5. Caller-blocked – The user at one of the multiple contacts has blocked the caller from further calling, and
 6. Diverted – The user at one of the multiple contacts has diverted the incoming call to another URI.
- 5 When multiple contacts of a UE subscribe to each other's dialog event package for receiving notifications about each other's dialog state and events, the following events can be used for specifying the filter criteria to receive notifications pertaining to forking scenario:
1. local-bye,
 - 10 2. remote-bye,
 3. block-caller,
 4. divert-call, and
 5. rejected.

The present invention proposes an extension to the SIP Dialog event package

15 by defining additional events to the dialog event package schema for addressing the issues that arise when a SIP request is forked to multiple contact addresses.

As in introduction, usually in SIP based networks, a single SIP AOR could have multiple contact addresses registered for it. In such cases when there is an incoming session for the AOR, the SIP call processing server could fork the call

20 request to all the registered contact addresses either in a sequential or simultaneous pattern. Now in this scenario, only the contact address that handled the call is aware of the actual outcome of the call. The remaining contact addresses either have a

missed call notification or would not have been contacted at all. So in order to keep all the contact address synchronized with the actual outcome at either of the contacts, it is required that all the contacts subscribe to each other's dialog event package.

The present invention outlines the need for extending the dialog event
5 package. When the SIP server forks the incoming call request to all the registered contacts of a SIP URI wither in sequential or simultaneous fashion, the following outcomes are possible at each registered contact:

Missed : implies the contact missed the call - i.e. it neither answered or cancelled or answered the call

10 Cancelled : implies the contact cancelled the request without accepting it

Terminated : implies the contact sent a local bye or received bye from remote to terminate the call request

Diverted : implies the contact diverted the call to another contact or URI

15 Blocked : implies the contact blocked the caller from further calling.

The existing dialog event package as is known in the art, defines events corresponding to cancelled and terminated, but does not define events for other possible outcomes. The present invention proposes the above additional events for the
20 dialog event package for addressing the possible outcomes that could arise when a SIP request is forked to all the registered contact addresses of a SIP AOR.

The present invention can be better understood with reference to an Enhanced SIP Dialog State Machine. Referring to Table 1, the section 3.7.1 of RFC 4235 shows

the dialog state model for both UAC and UAS. The model depicted here extends it to show the proposed events.

Table 1
Enhanced SIP Dialog State Machine

5

10

15

20

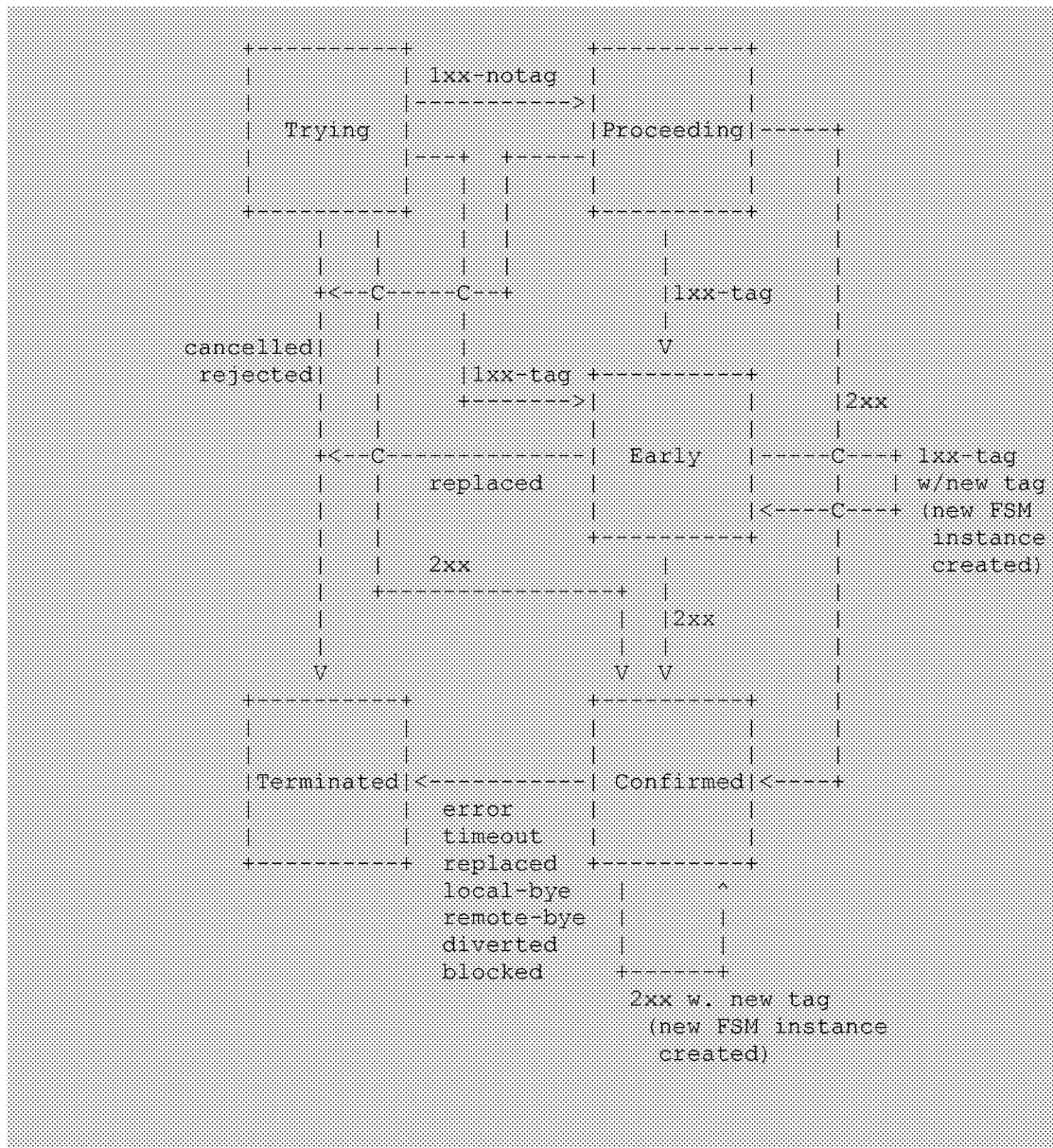
25

30

35

40

45



It should be noted that there is no state change when the event is missed, and the contact sends a BYE with Reason header to block the caller from further calling.

The present invention also proposed an enhancement to XML Schema. In particular, section 4.4 of RFC 4235 defines the XML schema for the dialog event package. The present invention extends the schema for the above proposed events. Table 2 shows the schema for the application/dialog-info+xml type:

5

Table 2

XML Schema

10

15

20

25

30

35

40

45

50

```

    <?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  targetNamespace="urn:ietf:params:xml:ns:dialog-info"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="urn:ietf:params:xml:ns:dialog-info"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified">
  <!-- This import brings in the XML language
                                attribute xml:lang-->
  <xs:import namespace="http://www.w3.org/XML/1998/namespace"
    schemaLocation="http://www.w3.org/2001/03/xml.xsd"/>

  <xs:element name="dialog-info">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="tns:dialog" minOccurs="0"
          maxOccurs="unbounded"/>
        <xs:any namespace="##other" processContents="lax"
          minOccurs="0" maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:attribute name="version" type="xs:nonNegativeInteger"
        use="required"/>
      <xs:attribute name="state" use="required">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="full"/>
            <xs:enumeration value="partial"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:attribute>
      <xs:attribute name="entity" type="xs:anyURI"
        use="required"/>
    </xs:complexType>
  </xs:element>

  <xs:element name="dialog">
    <xs:complexType>
      <xs:sequence>
        <xs:element ref="tns:state" minOccurs="1" maxOccurs="1"/>
        <xs:element name="duration" type="xs:nonNegativeInteger"
          minOccurs="0" maxOccurs="1"/>
        <xs:element name="replaces" minOccurs="0" maxOccurs="1">
          <xs:complexType>
            <xs:attribute name="call-id" type="xs:string"
              use="required"/>
            <xs:attribute name="local-tag" type="xs:string"

```

```

        use="required"/>
        <xs:attribute name="remote-tag" type="xs:string"
            use="required"/>
    </xs:complexType>
5   </xs:element>
    <xs:element name="referred-by" type="tns:nameaddr"
        minOccurs="0" maxOccurs="1"/>
    <xs:element name="route-set" minOccurs="0" maxOccurs="1">
10   <xs:complexType>
        <xs:sequence>
            <xs:element name="hop" type="xs:string"
                minOccurs="1" maxOccurs="unbounded"/>
        </xs:sequence>
    </xs:complexType>
15 </xs:element>
    <xs:element name="local" type="tns:participant"
        minOccurs="0" maxOccurs="1"/>
    <xs:element name="remote" type="tns:participant"
        minOccurs="0" maxOccurs="1"/>
20 <xs:any namespace="##other" processContents="lax"
        minOccurs="0" maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="id" type="xs:string" use="required"/>
    <xs:attribute name="call-id" type="xs:string"
25   use="optional"/>
    <xs:attribute name="local-tag" type="xs:string"
        use="optional"/>
    <xs:attribute name="remote-tag" type="xs:string"
        use="optional"/>
30 <xs:attribute name="direction" use="optional">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:enumeration value="initiator"/>
            <xs:enumeration value="recipient"/>
35 </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    </xs:complexType>
    </xs:element>
40
    <xs:complexType name="participant">
        <xs:sequence>
            <xs:element name="identity" type="tns:nameaddr"
                minOccurs="0" maxOccurs="1"/>
45 <xs:element name="target" minOccurs="0" maxOccurs="1">
            <xs:complexType>
                <xs:sequence>
                    <xs:element name="param" minOccurs="0"
                        maxOccurs="unbounded">
50 <xs:complexType>
                        <xs:attribute name="pname" type="xs:string"
                            use="required"/>
                        <xs:attribute name="pval" type="xs:string"
                            use="required"/>
55 </xs:complexType>
                    </xs:element>
                </xs:sequence>
                <xs:attribute name="uri" type="xs:string"
60   use="required"/>
            </xs:complexType>
        </xs:element>
    <xs:element name="session-description" type="tns:sessd"
        minOccurs="0" maxOccurs="1"/>
    <xs:element name="cseq" type="xs:nonNegativeInteger"

```

```

minOccurs="0" maxOccurs="1"/>
<xs:any namespace="##other" processContents="lax"
minOccurs="0" maxOccurs="unbounded"/>
5  </xs:sequence>
</xs:complexType>
<xs:complexType name="nameaddr">
  <xs:simpleContent>
    <xs:extension base="xs:anyURI">
10    <xs:attribute name="display-name" type="xs:string"
      use="optional"/>
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="sessd">
15  <xs:simpleContent>
    <xs:extension base="xs:string">
      <xs:attribute name="type" type="xs:string"
        use="required"/>
    </xs:extension>
  </xs:simpleContent>
20  <xs:element name="state">
    <xs:complexType>
      <xs:simpleContent>
        <xs:extension base="xs:string">
25        <xs:attribute name="event" use="optional">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="cancelled"/>
30              <xs:enumeration value="rejected"/>
              <xs:enumeration value="replaced"/>
              <xs:enumeration value="local-bye"/>
              <xs:enumeration value="remote-bye"/>
              <xs:enumeration value="error"/>
              <xs:enumeration value="timeout"/>
35              <xs:enumeration value="missed"/>
              <xs:enumeration value="blocked"/>
              <xs:enumeration value="diverted"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
40        <xs:attribute name="code" use="optional">
          <xs:simpleType>
            <xs:restriction base="xs:positiveInteger">
              <xs:minInclusive value="100"/>
45              <xs:maxInclusive value="699"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:attribute>
        </xs:extension>
50      </xs:simpleContent>
    </xs:complexType>
  </xs:element>
</xs:schema>
55

```

As an example, the present invention considers a SIP Forking scenario where in Alice calls Bob and Bob's SIP call processing server forks the call request from Alice to Bob's contacts Bcob1, Bob2 and Bob3. And the contact at Bob2 accepts the

call request from Alice while the contact at Bob1 and Bob3 get a missed call indication. So the notification from Bob2 to Bob1 and Bob3 would proceed as in Table 3.

Table 3

```

5      <?xml version="1.0"?>
      <dialog-info xmlns="urn:ietf:params:xml:ns:dialog-info"
10          version="1" state="full"
          entity="sip:alice@example.com">
      <dialog id="1">
          <state event="terminated" ></state>
      </dialog>
      </dialog-info>

```

15

As far as security considerations; subscriptions to dialog state can reveal sensitive information. For this reason, known section 3.6 of RFC 4235 discusses authentication and authorization of subscriptions, and provides guidelines on sensible authorization policies. All implementations of this package MUST support the digest authentication mechanism. Since the data in notifications is sensitive as well, end-to-end SIP encryption mechanisms using S/MIME MAY be used to protect it. User agents that implement the dialog package SHOULD also implement SIP over TLS and the sips: scheme.

20

It will be appreciated by one of ordinary skill in the art that the interfaces between various modules described herein exist as software interfaces taking the form, for example, of function calls and the like, or may take the form of real time interrupt processing, or other software related processing. Alternatively, the functionality and interfaces may exist in hardware, or may be a combination of hardware and software. Bearer requirements involve an exemplary radio access

25

network providing bearers to transport the application flows as noted herein. The bearer requirements to support the services described herein are consistent with TS 24.228 section 4.2.7, for example.

It will be appreciated from the above discussion that many of the features of the present invention are susceptible to being implemented in a software program such as an application program or in a series of intercommunicating software programs, application, routines, modules, operating systems and the like. In addition, much of the functionality can be conducted as a method or procedure with a series of steps or the like.

FIG. 4 illustrates an exemplary method for call delivery management in a communication system, in accordance with the present invention. The method includes a first step 400 of subscribing to a Dialog Event service package that operates to provide a dialog state to a subscribing user.

A next step 402 includes registering a global identity and multiple contact identities for the user.

A next step 404 includes receiving a call to the global identity of the user.

A next step 406 includes sequentially attempting the call to the multiple contact identities of the user.

A next step 408 includes determining that one of the contacts has had a dialog state change event. The dialog state change event generally consists of either accepting the call or rejecting the call (e.g. cancel, terminate, divert, block).

A next step 410 includes ascertaining whether the user has subscribed to the Dialog Event service.

A next step 412 includes notifying at least the sequentially-previous contacts of the dialog state change event of the one contact. Preferably, this includes not only
5 that there was a state change event, but what particular state change event occurred. Optionally, all of the multiple contact identities of the user could be notified of the dialog state change event.

In the example herein, the communication system is an Internet Protocol (IP) Multimedia Core Network Subsystem (IMS) network, and wherein in the registering
10 step 402 the global identity is a public user identity of the user and the multiple contact identities are private user identities of the user. Alternatively, the communication system could be a non-IMS network, and wherein in the registering step the global identity is a global Session Initiation Protocol Uniform Resource Identifier (SIP URI) of the user and the multiple contact identities are multiple contact
15 URIs of the user.

Advantageously, the present invention allows subscribers who have multiple contacts registered to have all their contacts synchronized with any action that takes place at any of their registered contacts, unlike the prior art.

The sequences and methods shown and described herein can be carried out in
20 a different order than those described. The particular sequences, functions, and operations depicted in the drawings are merely illustrative of one or more embodiments of the invention, and other implementations will be apparent to those of ordinary skill in the art. The drawings are intended to illustrate various

implementations of the invention that can be understood and appropriately carried out by those of ordinary skill in the art. Any arrangement, which is calculated to achieve the same purpose, may be substituted for the specific embodiments shown.

The invention can be implemented in any suitable form including hardware,
5 software, firmware or any combination of these. The invention may optionally be implemented partly as computer software running on one or more data processors and/or digital signal processors. The elements and components of an embodiment of the invention may be physically, functionally and logically implemented in any suitable way. Indeed the functionality may be implemented in a single unit, in a
10 plurality of units or as part of other functional units. As such, the invention may be implemented in a single unit or may be physically and functionally distributed between different units and processors.

Although the present invention has been described in connection with some embodiments, it is not intended to be limited to the specific form set forth herein.
15 Rather, the scope of the present invention is limited only by the accompanying claims. Additionally, although a feature may appear to be described in connection with particular embodiments, one skilled in the art would recognize that various features of the described embodiments may be combined in accordance with the invention. In the claims, the term comprising does not exclude the presence of other elements or
20 steps.

Furthermore, although individually listed, a plurality of means, elements or method steps may be implemented by e.g. a single unit or processor. Additionally, although individual features may be included in different claims, these may possibly

be advantageously combined, and the inclusion in different claims does not imply that a combination of features is not feasible and/or advantageous. Also the inclusion of a feature in one category of claims does not imply a limitation to this category but rather indicates that the feature is equally applicable to other claim categories as
5 appropriate.

Furthermore, the order of features in the claims do not imply any specific order in which the features must be worked and in particular the order of individual steps in a method claim does not imply that the steps must be performed in this order. Rather, the steps may be performed in any suitable order. In addition, singular
10 references do not exclude a plurality. Thus references to "a", "an", "first", "second" etc do not preclude a plurality.

CLAIMS

What is claimed is:

- 1 1. A method for managing sequential call delivery in an internet protocol
- 2 communication system, the method comprising the steps of:
- 3 registering a global identity and multiple contact identities for a user;
- 4 receiving a call to the global identity of the user;
- 5 sequentially attempting the call to the multiple contact identities of the user;
- 6 determining that one of the contacts handles the call; and
- 7 notifying at least the sequentially-previous contacts that the call has been handled.

1 2. The method of claim 1, wherein the communication system is an Internet
2 Protocol (IP) Multimedia Core Network Subsystem (IMS) network, and wherein in
3 the registering step the global identity is a public user identity of the user and the
4 multiple contact identities are private user identities of the user.

5

1 3. The method of claim 1, wherein the communication system is a non-IMS
2 network, and wherein in the registering step the global identity is a global Session
3 Initiation Protocol Uniform Resource Identifier (SIP URI) of the user and the multiple
4 contact identities are multiple contact URIs of the user.

5

1 4. The method of claim 1, wherein in the determining step handling the call
2 consists of one of the group of:
3 blocking the call,
4 accepting the call, and
5 rejecting the call.

6

1 5. The method of claim 1, further comprising the step of ascertaining whether the
2 user has subscribed to a Dialog Event service that operates to provide a dialog state to
3 a user.

4

1 6. The method of claim 1, further comprising the steps of:
2 subscribing to a Dialog Event service that operates to provide a dialog state to the
3 user; and
4 ascertaining whether the user has subscribed to the Dialog Event service.

1 7. The method of claim 6, wherein the subscribing step includes a user
2 subscribing the Dialog Event service using the user's own URI as a request URI.

3

1 8. The method of claim 1, wherein the determining step consists of determining a
2 dialog state change event of the one contact, and wherein the notifying step consists
3 of notifying at least the sequentially-previous contacts of a dialog state change event
4 of the one contact.

5

1 9. The method of claim 8, wherein the notifying step includes notifying at least
2 the sequentially-previous contacts of the particular dialog state change of the one
3 contact.

1 10. A system for managing a sequential call delivery in an internet protocol
2 communication system, the system comprising:
3 a plurality of user equipment, each having a common global identity and
4 individual contact identities for a user; and
5 a call processing server that operates to register the global identity and multiple
6 contact identities of the user equipment, the call processing server also
7 operably to receive a call to the global identity of the user and sequentially
8 attempt the call to the multiple contact identities of the user, wherein if the call
9 processing server determines that one of the user equipment handles the call;
10 the call processing server notifies at least the sequentially-previous user
11 equipment that the call has been handled.

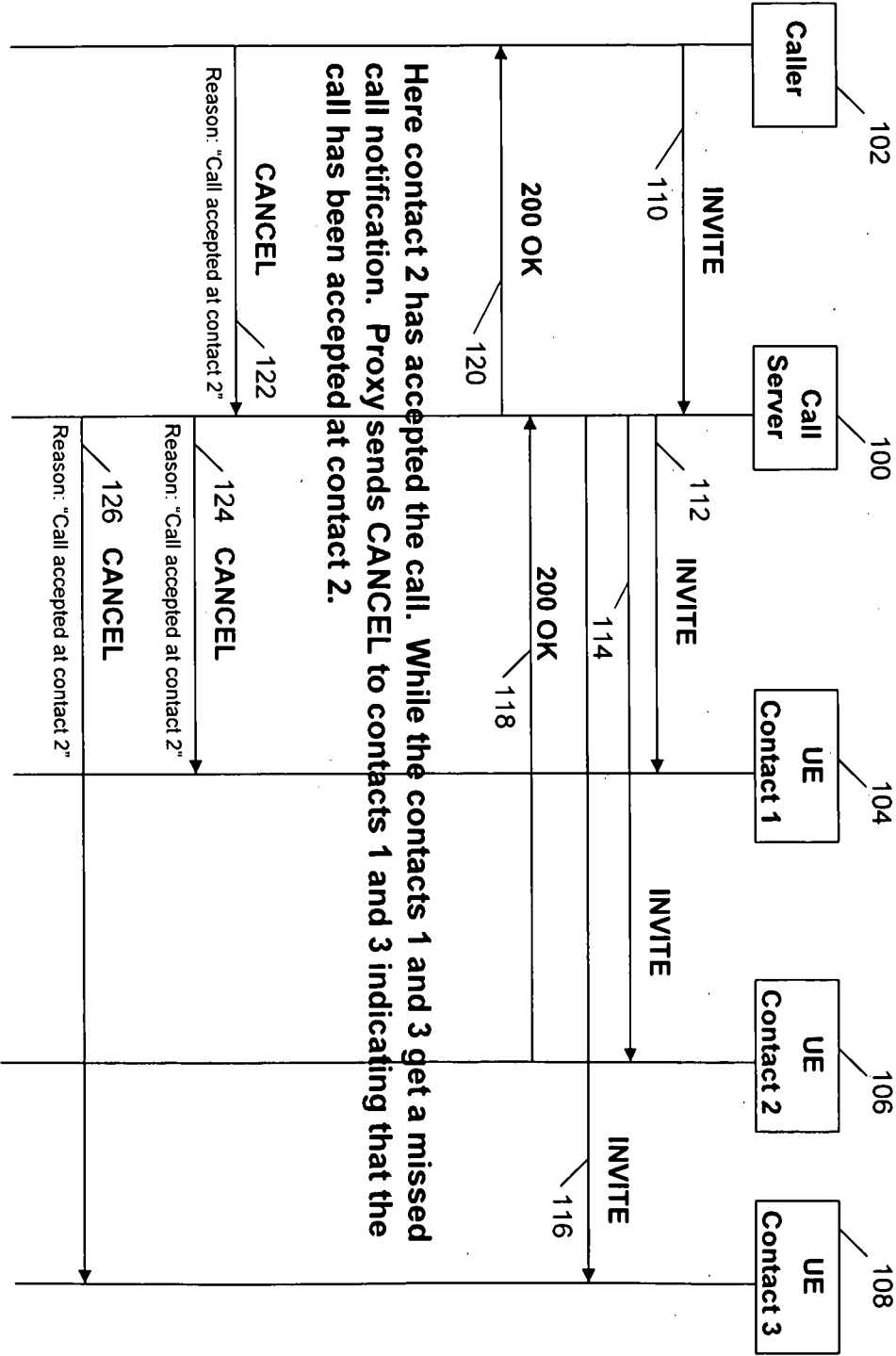


FIG. 1

- PRIOR ART -

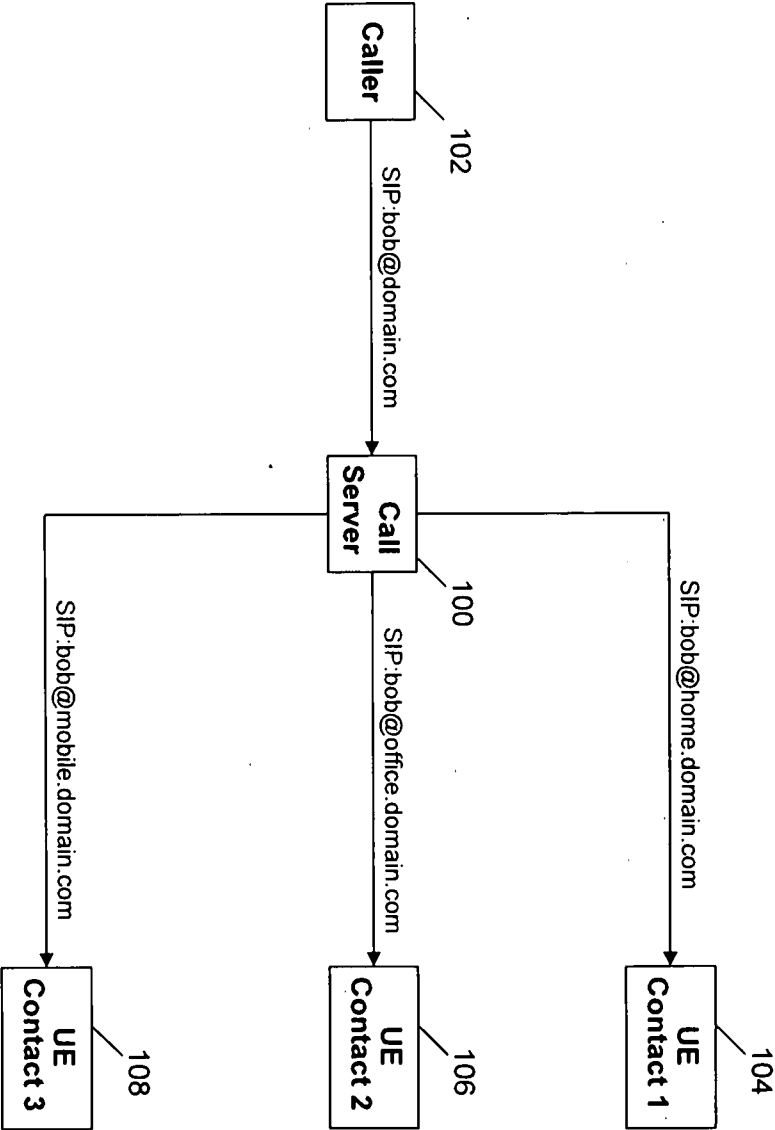


FIG. 2

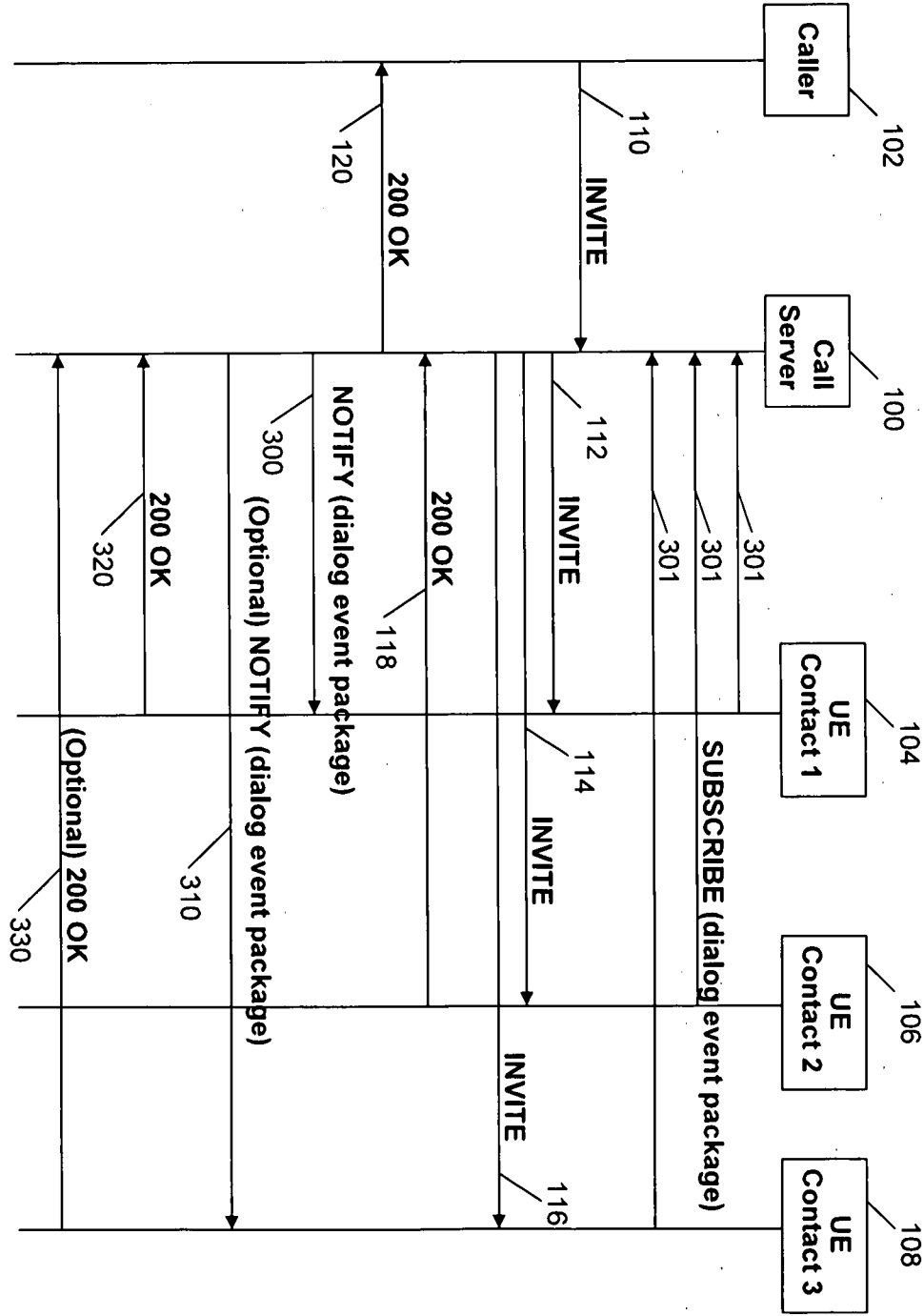
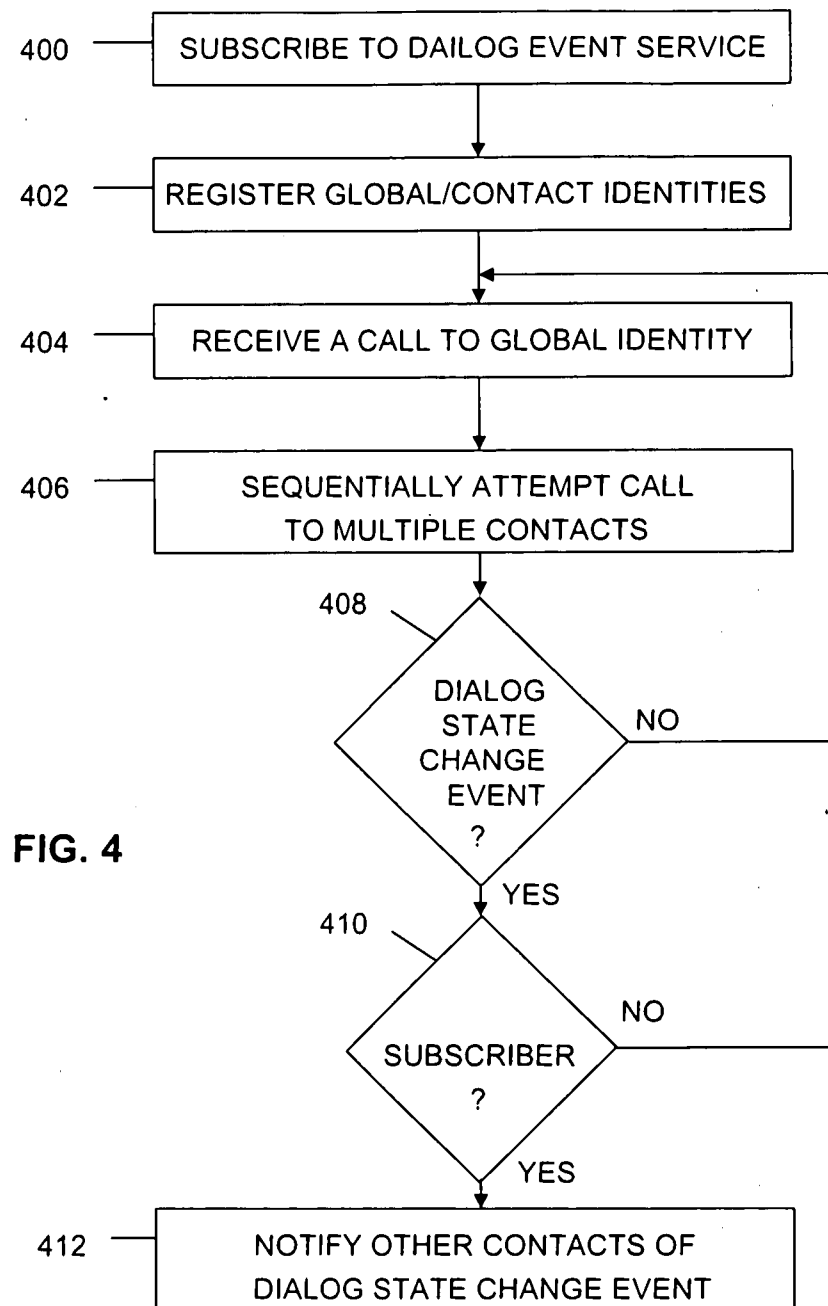


FIG. 3

4/4



A. CLASSIFICATION OF SUBJECT MATTER***H04W 80/04(2009.01)i, H04W 8/26(2009.01)i, H04W 4/16(2009.01)i***

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

IPC G10K, G10L

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Korean Utility models and applications for Utility Models since 1975

Japanese Utility models and applications for Utility Models since 1975

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKIPASS(KIPO internal) & keywords: "SIP", "call", "delivery, forwarding"

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	KR 10-2007-0037666 A (UANGEL) 06 April 2007 See the abstract, claims 1-5 and figures 2-4.	1-10
A	US 2004-0037406 A1 (CHRISTOPHE GOURRAND) 26 February 2004 See the abstract, claims 1-3 and figure 3.	1-10
A	KR 10-2004-0043344 A (LG ELECTRONICS INC.) 24 May 2004 See the abstract, claims 1-3 and figure 3.	1-10
A	KR 10-2005-0008033 A (SK TELECOM CO., LTD.) 21 January 2005 See the abstract, claims 1,4,6,7 and figures 2-4.	1-10



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

24 APRIL 2009 (24.04.2009)

Date of mailing of the international search report

28 APRIL 2009 (28.04.2009)

Name and mailing address of the ISA/KR

Korean Intellectual Property Office
Government Complex-Daejeon, 139 Seonsa-ro, Seo-
gu, Daejeon 302-701, Republic of Korea

Facsimile No. 82-42-472-7140

Authorized officer

JUNG, Sung Yun

Telephone No. 82-42-481-8483



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2008/082573Patent document
cited in search reportPublication
datePatent family
member(s)Publication
date

KR 10-2007-0037666 A

06.04.2007

KR 10-0734740 B1

03.07.2007

US 2004-0037406 A1

26.02.2004

AU 2003-250677 A1

11.03.2004

WO 2004-019554 A1

04.03.2004

KR 10-2004-0043344 A

24.05.2004

KR 10-0574656 B1

28.04.2006

KR 10-2005-0008033 A

21.01.2005

KR 10-0673165 B1

22.01.2007