



- (51) International Patent Classification:
H04M 3/48 (2006.01)
- (21) International Application Number:
PCT/IB2012/001662
- (22) International Filing Date:
23 July 2012 (23.07.2012)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
201110218297.5 1 August 2011 (01.08.2011) CN
- (71) Applicant (for all designated States except US): **ALCATEL LUCENT** [FR/FR]; 3, avenue Octave Gréard, F-75007 Paris (FR).
- (72) Inventors; and
- (75) Inventors/Applicants (for US only): **XIAO, Yu** [CN/CN]; 388# Ningqiao RD, Shanghai, Shanghai 201206 (CN). **LIU, Fangxin** [CN/CN]; 388# Ningqiao RD, Shanghai, Shanghai 201206 (CN).
- (74) Agents: **SCIAUX, Edmond** et al.; Alcatel-lucent International, 32 avenue Kléber, F-92700 Colombes (FR).

- (81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, 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, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, 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: METHOD AND APPARATUS OF CALL TO BUSY SUBSCRIBER ESTABLISHMENT

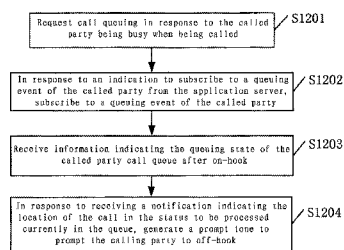


Figure 12a

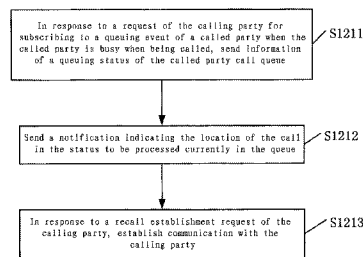


Figure 12b

(57) Abstract: The present invention relates to a method and apparatus for call to busy subscriber establishment. The method comprises: in response to the called party being busy when called, requesting for call queuing at a calling party subscriber terminal; in response to an indication to subscribe to a called party queuing event from an application server, subscribing to a queuing event of the called party at the calling party subscriber terminal; after the calling party on-hooks, receiving information indicating a queuing status of a queue of calls of the called party at the calling party subscriber terminal, wherein, the queuing status at least comprises a record-count indicating a location of the call in the queue; and in response to the receipt of a notification indicating the location of the call in a status to be processed currently in the queue, generating a prompt tone at the calling party subscriber terminal to prompt the calling party to off-hook. The introduction of a call queuing mechanism, and the use of continuous updated call queue information, facilitate the calling party in selecting a service and also provide user friendliness.



Method and Apparatus of Call to Busy Subscriber Establishment

Technical Field

The present invention relates to the field of communication, and more specifically, to method and apparatus of a call to busy subscriber establishment.

Description of the Related Art

With the rapid development of network technology, in recent years, mobile communication trends to evolve into an all-IP network architecture and IMS (IP Multimedia Subsystem)-based network integration becomes a hot spot of the application. IMS is a new core network domain defined by 3GPP (3rd Generation Partnership Project) in the UMTS R5 and later versions thereof, supporting IP-based networks that process traditional audio phone and multimedia services. 3GPP selects SIP (Session Initiation Protocol) as a core session signaling protocol that is developed by the Internet Engineering Task Force (IETF) for the establishment, change and removal of a session and is independent of the transmission of the media. Its specific content can refer to <http://www.3gpp.org/ftp/Specs/Latest-drafts/24229-201.zip> (entitled "IP Multimedia Call Control Protocol based on SIP and SDP").

Figure 1 illustrates a SIP-based application network structure. Shown in Figure 1, the network may comprise various types of SIP terminals, a SIP server and an application server, and even a media server. SIP signaling interaction may be used between various network entities. The SIP terminals are connected to the SIP server and the application server is connected to the SIP server. The application server an API physical entity carrying the generation and development of control services, and the IMS network may provide non-traditional SIP-based telecommunications services via the application server and implement a large number of value-added services. The SIP server may be a proxy server, a register server, a redirection server, a location server, etc. The SIP server can be responsible for the registration authentication and session control of the UE, perform a basic session routing function, and according to subscribed IMS trigger rules, conduct value-added service triggering and service control interaction to the application server when conditions are met. The media server provides a specific media resource service, such as multi-party conferencing, recording, announcements, voice recognition and the like. The media server accepts the control by the application server to complete the allocation, processing and play of media resources.

For those skilled in the art, the functionality of the SIP server is a basic functionality. When describing IMS-network-based specific services, for the sake of clarity, the introduction of the SIP server will be omitted in the following description. However, those skilled in the art should

appreciate that the services or processing described in the present invention are performed on the SIP server architecture.

In the IMS-network-based call process, if the called party is busy without the immediate establishment of a call, usually there are several processing solutions as follows:

I. With the use of small subscriber switch PBX

Figure 2 illustrates a communications network architecture using a small subscriber switch PBX. As shown in Figure 2, a subscriber terminal 10 is connected through a network 12 with a voice gateway 14 of the called party B, a called party voice gateway 142 is connected to a PBX 14 and the PBX 14 is connected to a service interface 16.

Figure 3 illustrates a call establishment flow applied on the network architecture in Figure 2. Shown in Figure 3, in step S301, the subscriber terminal 10 sends an INVITE request INVITE (From A to B) to a SIP-based application server 120 when the subscriber A off-hooks and dials the number of a called subscriber B. In step S302, the application server 120 forwards the INVITE request INVITE (From A, to B) to the called party voice gateway 142. In step S303, the called party voice gateway 142 sends to a ringing message the PBX 14. In step S304, the PBX returns a connection message to the subscriber terminal 142. Then, in step S305, the called party voice gateway 142 sends a 200 response message to the application server 120. If all the service interfaces are busy, then in step S306, the PBX 14 sends through the media stream to the calling party subscriber terminal 10 an announcement "All services are busy now. Please wait". If the PBX 14 monitors a service interface is idle, then in step S307, a ringing message is sent, and the service interface 16 returns a connection message. Alternatively, in step S308, the PBX 14 may also send through the media stream to the subscriber terminal 10 an announcement "Service XXX will serve you". Finally, in step S309, voice communication is established between the calling party subscriber terminal and the called party service interface.

Obviously for this solution, if the called party is busy, the PBX will keep the subscriber A online until some service interface is idle before access. In other words, the calling subscriber A needs to maintain the call and is always in a call waiting status, whereas this status will be annoying and inefficient, and the calling party has to pay for the waiting time. In addition, the purchase and maintenance cost of the PBX is high and not acceptable for subscribers.

II. CCBS (Completion of Call to Busy Subscriber) Service applied on the application server

Figure 4 illustrates a network architecture with CCBS service applied on the application server. Shown in Figure 4, the subscriber terminal 30 is connected via an application server 320 on a network 32 to a called party voice gateway 34 that connects a service interface 36, and a CCBS service module resides on the application server.

Figure 5 illustrates a call establishment flow applied on the network architecture in Figure 4. Shown in Figure 5, in step S501, the subscriber terminal 30 sends an INVITE request INVITE (From A to B) to the SIP-based application server 320 when the subscriber A off-hooks and dials the number of the called subscriber B. In step S502, the application server 320 forwards the INVITE (From A, to B) to the called party voice gateway 342. If it is detected that all of the service interfaces are busy, then in step S503, the voice gateway 342 returns a 486 subscriber-busy response.

In this solution, when the CCBS service is provided, the application servers needs to check if the subscriber A is a signatory subscriber, and if so, then in step S504, the application server 320 may control the provision of the media stream (such as voice announcement) from the Media Resource Function (MRF) 330 to the subscriber terminal 30. The MRF is used to control and handle a variety of media resource services. It may be divided into MRFC (Media Resource Function Control section) and MRFP (Media Resource Function Processing section), respectively, to complete the control and bearer functions of the media stream, and invoked by the application server, the MRFC controls the MRFP through H.248 and RTSP to complete corresponding encoding/decoding, conversion, mixing and play of the media stream.

For example, the application server 320 sends an INVITE request INVITE (SDP A) to the Media Resource Function (MRF) 330 to request for media type and all encoding. Thereafter, the MRF 330 sends a response 200 OK (SDP MRF) to the application server, which carries MRF-supported media type and encoding. The application server 320 sends a 183 response 183 (SDP) to the calling party subscriber terminal 30 to notify the MRF-supported media type and all encoding. Next, through the PRACK message and 200 OK response, the media type and encoding are negotiated and determined between the subscriber terminal 30 and the application server 320. Finally, the application server 320 sends an acknowledgement signal ACK to the MRF 330.

Thus, in step S505, the MRF 330 may send a media stream to the subscriber terminal 30 for announcement. For example, "The number you dial is busy. If you want to be called back, please dial X". In addition, when the subscriber wants to call back and dials the corresponding number, the subscriber terminal 30 may send the media stream "X" to the MRF 330. After the MRF 330 receives the media stream, it may also send a media stream to the subscriber terminal 30 for announcement, such as "The service you request has been registered. Please on-hook".

Thereafter in step S506, the application server sends a 500 message and a BYE message to the subscriber terminal 30 and the MRF 330, respectively, to tear down the link. Then in step S507, the subscriber terminal 30 generates a congestion tone and the subscriber A on-hooks.

When the application server 320 monitors the service interface of the subscriber B is idle, then in step S508, the application server may control the calling party subscriber terminal 30 and the voice

gateway 342 as a calling party to establish voice communication between the subscriber terminal 30 and the service interface 346 of the called party.

For example, the application server, like the calling party, calls back the subscriber terminal, which may send an INVITE request to the subscriber terminal 30. The subscriber terminal 30, like the called party, generates a ringing tone and returns the 180 response to the application server 320 to prompt it to generate the ringback tone. After the subscriber off-hooks to answer, a 200 response is sent to the application server 320.

After the subscriber terminal 30 off-hooks, the application server 320, like the calling party, may send an INVITE request to the voice gateway 342. Similarly, the service interface 346 generates a ringing tone and returns a 180 response to the application server 320 to prompt it to generate a ringback tone. After the service interface 346 off-hooks to answer, the voice gateway 342 sends a 200 response to the application server 320. Thereafter, the application server 320 sends an acknowledgement signal ACK to the subscriber terminal 30 and the voice gateway 342, respectively.

Finally, in step S509, voice communication is established between the subscriber terminal 30 and the service interface 346.

For this solution, although the calling party does not have to always online wait for an idle service interface, it still cannot know the time to wait, and it has no chance to cancel the previous request for CCBS during waiting and only can cancel the request for CCBS service by not off-hooking when the application server calls back, whereby even if the subscriber may not want to call back, the communication network still needs to do the above operations to call the subscriber back, increasing burdens of additional processing and additional signaling interactions to the communication network. In addition, if a callback is missed accidentally, then the calling party may still need to repeat the lengthy and annoying processes to get the call establishment. On the other hand, for the application server, the application server's resources are very valuable, but the use of the existing CCBS solution needs to request the application server to monitor the status of each number having the CCBS service, greatly wasting the application server's resources and leading to inefficiency of the application server. Meanwhile, the existing CCBS solution needs to attach additional functionality to the application server so as to control, as the calling party, the establishment of the communication between the calling party subscriber terminal and the called party's service interface during callback, which makes the structure and function of the application server complex and not conducive to the universality of intermediate network elements or the trend of a simple core network.

III. Without the PBX or the CCBS service applied on the application server

If the PBX is not used and the application server does not provide the CCBS service, or the subscriber A has not subscribed CCBS, the subscriber A needs to keep trying to call the subscriber B. In this way, when the number is busy, the calling party has to dial the number again and again to try to establish a call. For subscribers, such subscriber experience is extremely annoying, while every subscriber call failure will increase the risk of the company to lose the customer.

Therefore, there is an urgent need in the existing technology for a solution of establishing a call, which can improve experience degrees of subscribers, show friendliness to subscribers, and simplify the function or structure of the application server.

Summary of the Invention

In order to solve at least one of the above-mentioned problems, there is provided a method for establishing communication with a called party, a method for establishing communication with a calling party, a method for controlling the establishment of a call between a calling party and a called party, as well as a subscriber terminal and an application server.

According to one aspect of the present invention, there is provided a method for establishing communication with a called party, comprising: in response to the called party being busy when called, requesting for call queuing at a calling party subscriber terminal; in response to an indication to subscribe to a called party queuing event from an application server, subscribing to a queuing event of the called party at the calling party subscriber terminal; after the calling party on-hooks, receiving information indicating a queuing status of a queue of calls of the called party at the calling party subscriber terminal, wherein, the queuing status at least comprises a record-count indicating a location of the call in the queue; and in response to the receipt of a notification indicating the location of the call in a status to be processed currently in the queue, generating a prompt tone at the calling party subscriber terminal to prompt the calling party to off-hook. Preferably, the prompt tone comprises a ringing tone.

Optionally, the method may further comprise: in response to the off-hook, sending a call establishment request at the calling party subscriber terminal to request again for establishing communication with the called party.

Optionally, the method may further comprise: during a call waiting period, canceling queuing waiting at the calling party subscriber terminal by canceling the subscription.

Optionally, the step of said canceling queuing waiting may further comprise: initiating cancelling the subscription by trying calling so as to cancel queuing waiting.

Optionally, the method may further comprise: if the called party is busy when being called, receiving information notifying that the called party is busy at the calling party subscriber terminal,

wherein the information contains information about a queuing status of the called party call queue.

Optionally, the method may be applied to a Session Initiation Protocol SIP environment.

According to another aspect of the present invention, there is provided a method for establishing communication with a calling party, comprising: in response to a request from the calling party for subscribing to a queuing event of a called party when the called party is busy when being called, sending at a called party subscriber terminal information of a queuing status of the called party call queue, wherein the queuing status at least comprises a record-count indicating a location of the call in the queue; sending a notification indicating a location of the call in a status to be processed currently in the queue; and establishing communication with the calling party in response to a recall establishment request of the calling party. Preferably, the recall request of the calling party is initiated in response to the notification indicating the location of the call in a status to be processed currently in the queue.

Optionally, the method may further comprise: if the called party is busy when being called, sending information notifying that the called party is busy at a subscriber terminal of the called party, wherein the information contains information about a queuing status of the called party call queue.

Optionally, the method may further comprise: during a call waiting period, in response to the receipt of a request of canceling the subscription from the calling party, notifying the calling party to cancel the subscription so as to notify to cancel queuing waiting.

According to another aspect of the present invention, there is provided a method for controlling the establishment of a call between a calling party and a called party, comprising: in response to a request for call queuing of the calling party when the called party is busy when called, instructing the calling party to subscribe to a queuing event of the called party at an application server; forwarding to the called party a request for subscribing to the queuing event of the calling party; forwarding to the calling party information of status of the queuing event of the called party, wherein the queuing status at least comprises a record-count indicating a location of the call in the queue; and forwarding to the calling party a notification of a location of the call in a status to be processed currently in the queue so that the calling party generates a prompt tone to prompt off-hook.

Optionally, the method further comprise: forwarding to the called party a call establishment request from the calling party in response to the off-hook to request again for establishing communication.

Optionally, the method further comprise: during a call waiting period, in response to the receipt of a request from the calling party for canceling queuing waiting, instructing at the application server the calling party to cancel the subscription so as to cancel queuing waiting.

Optionally, the method further comprise: forwarding to the called party a request from the

calling party of canceling the subscription so as to cancel the subscription to the queuing event of the called party.

Optionally, the method further comprise: forwarding to the calling party a notification of canceling the subscription from the called party so as to notify canceling queuing waiting.

Optionally, the method further comprise: receiving information notifying that the called party is busy from the called party when the called party is busy when being called, wherein the information contains information about the queuing status of the called party call queue.

Optionally, the method further comprises: invoking Media Resource Function to provide to the calling party information of the queuing status of the called party call queue.

According to another aspect of the present invention, there is provided a computer program product, comprising computer executable instructions, which cause, when executed on a computer device, the device to be configured to execute any one of the methods.

Optionally, the queuing status at least comprises one of the following: statuses such as the called party is out of service, the queue being empty, queuing or the queue being full; call waiting number in the queue; the phone number of the call waiting party; the serial number or location of the calling party in the called party queue, indicating the sequence of the call to be processed (i.e., the record-count); and expected time to preserve the call.

In an embodiment of the present invention, a mechanism to subscribe to a queue-event of a called party subscriber terminal is introduced to notify the calling party in real time of queuing status of its call, improving subscriber friendliness, and further a mechanism to cancel subscriber subscription is provided, enabling subscribers to take the initiative to cancel waiting when desired, so as to reduce the burden on the communication network. In addition, initiating by the calling party subscriber terminal, may simplify the function and structure of the application server, facilitate the setting of the service and adapt to the trend of intelligent subscriber agent and simple core network.

Brief Description of the Drawings

Exemplary embodiments will be understood more clearly according to the following specific description in conjunction with the drawings. Figure 1-12 show non-limiting exemplary embodiments, in which:

Figure 1 illustrates a SIP-based application network architecture;

Figure 2 illustrates a communications network architecture using a small subscriber switch PBX.

Figure 3 illustrates a call establishment flow applied on the network architecture in Figure 2.

Figure 4 illustrates a network architecture with CCBS service applied on the application server.

Figure 5 illustrates a call establishment flow applied on the network architecture in Figure 4.

Figure 6 illustrates a flow chart of a call to busy subscriber establishment method according to an embodiment of the present invention;

Figure 7 illustrates a flow chart of processing of canceling the subscription during waiting for busy subscriber to cancel waiting according to an embodiment of the present invention;

Figure 8 illustrates a flow chart of processing of call to busy subscriber establishment based on a SIP environment according to the embodiment of the present invention;

Figure 9 illustrates a flow chart of processing of canceling the subscription during waiting for busy subscriber based on a SIP environment according to the embodiment of the present invention;

Figure 10 illustrates a block diagram of a subscriber terminal according to an embodiment of the present invention;

Figure 11 illustrates a block diagram of an application server according to an embodiment of the invention;

Figure 12 illustrates a flow chart of processing performed by a subscriber terminal according to an embodiment of the present invention.

Detailed Description of the Preferred Embodiment

Embodiments of the present invention will be described more completely now with reference to drawings in which some but not all of embodiments of the present invention are shown. Of course, the present invention can be implemented through many different forms and should not be understood as limitations on the embodiments described here; on the contrary, these embodiments are provided so that the present invention will satisfy applicable legal requirements. Where, similar signs refer to similar elements.

It should also be noted that, in some alternative embodiments, the proposed functionality/behavior may occur in a different order than shown in the drawings. The two graphs shown continuously, for example, may in fact be performed concurrently or may sometimes be performed in a reverse order, depending on the involved functionality/behavior.

Unless defined on the contrary, all terms (comprising technical and scientific terms) used here have the same meaning as understood by those skilled in the art to which exemplary embodiments belong. It should also be understood that the terms (for example, as defined in common dictionaries) should be construed as the same meanings as in an environment of related fields, and should not be construed in an idealized or overly formal way, unless defined explicitly here.

The description of parts of the present invention and corresponding details thereof is provided according to symbol representations of operations of data bits in computer memory and software or

algorithm. These description and representations are means via which those skilled in the art effectively communicate their work to other technicians. As a term used here, and as commonly used, algorithm can be considered as a self-consistent sequence of steps leading to a desired result. Steps are those requiring physical operations of physical quantities. Generally, although unnecessarily, these quantities adopt the forms capable of storing, transferring, combining, comparing and manipulating optical, electrical or magnetic signals. It is often proved and convenient that, universally in principle, these signals are referred to as bits, values, elements, symbols, words, items, numbers, etc..

In the following description, the exemplary embodiments will be described with reference to behaviors of operations that may be implemented as program modules or functional processing (comprising routines, programs, objects, components, data results and the like, which perform specific tasks or implement specific abstract data types and may be implemented using existing hardware at existing network elements or control nodes) and symbol representations (for example, in the form of a flow chart). Such existing hardware may comprise one or more central processing units (CPU), digital signal processor (DSP), ASIC, field programmable gate array (FPGA) computer and the like.

However, it should be remembered that all of these and similar terms are associated with appropriate physical quantities and are only convenient labels applied to these quantities. Unless otherwise explained, or clear from the discussion, terms such as “processing” or “computing” or “determine” or “display” and the like refer to behaviors and processing of a computer system or similar electronic computing device, which operates data expressed as physical, electronic quantities in a register and memory in the computer system, and transforms it into other data similarly expressed as physical quantities in a register and memory in the computer system or other such information storage devices, transmitting or displaying devices.

It should also be noted that software-implemented aspects of the present invention are typically encoded on some kind of program storage medium, or on some type of transmission medium. The program storage medium may be magnetic (e.g., floppy disk or hard drive) or optical (e.g., compact disk read only memory or “CD-ROM”), and may be read-only or random access. Similarly, the transmission medium may be twisted pair, coaxial cable, fiber, or some other suitable transmission medium known in the field. The present invention is not limited by these aspects of any given solution.

In an embodiment of the present invention, a call queue status is indicated by introducing a mechanism to subscribe to a queue-event of a called party subscriber terminal in a communication network, and the call queue status may comprise the following: statuses such as the called party is

out of service, the queue being empty, queuing or the queue being full; call waiting number in the queue; the phone number of the call waiting party; the serial number or location of the calling party in the called party queue, indicating the sequence of the call to be processed (i.e., the record-count); and expected time to preserve the call, etc.

To clearly show the queue-event in the network introduced in the embodiment of the present invention, the principle of operation thereof is shown by example of Session Initiation Protocol SIP signaling:

call-queue-network = queue-status-line CRLF

[waiting-number CRLF]

[*record-line CRLF]

[*due-time]

queue-status-line = "Call-Queue" HCOLON queue-status

queue-status = "empty" / "queuing" / "full" / "out-of-service"

waiting-number = "Waiting-Number" HCOLON waitnum

waitnum = 1*DIGIT

record-line = record-info CRLF

record-count

record-info = "Record-Info" HCOLON SIP-URI / SIPS-URI / absoluteURI

record-count = "Record-Count" HCOLON rcdcount

rcdcount = 1*DIGIT

due-time = "Due" HCOLON minutes "min"

minutes = 1*DIGIT

For example, there are 12 calls waiting to be processed in the queue of the called party, A@alcatel-lucent.com is the 12th one. The following instance is given by example of SIP signaling:

Call-Queue: queuing

Waiting-Number: 12

Record-Info: A@alcatel-lucent.com

Record-Count: 12

For example, there are 15 calls waiting to be processed in the queue of the called party. It is now

A@alcatel-lucent.com's turn to get the service, and some flag or numeric values may be set to represent the status, for example, a value of the record-count may be set to 1 to express that the call in the location in the queue is to be processed. The right may be preserved for 15 minutes. The following instance is given by example of SIP signaling:

Call-Queue: queuing

Waiting-Number: 15

Record-Info: A@alcatel-lucent.com

Record-Count: 1

Due: 15min

Those skilled in the art may understand that other types of signaling may be used to define a queue-event of a network in the embodiment of the present invention, and the embodiment of the invention is not limited to SIP signaling.

Figure 6 illustrates a flow chart of processing of call to busy subscriber establishment according to an embodiment of the present invention. Shown in Figure 6, in step S601, the calling party subscriber terminal A off-hooks and dials subscriber terminal B of the called party.

Then, in step S602, the subscriber terminal of the called party is busy, and then the calling party is notified that the called party is busy and of its queuing status at the time. Optionally, the application server may also provide to the calling party A an announcement about the registration in the waiting queue. For example, where the calling party may be instructed to queue and wait.

In the embodiment of the present invention, the announcement may be implemented by invoking MRF via the application server to perform voice play, and may also be implemented by controlling the messaging server via the application server to send short messages. Those skilled in the art may appreciate that messaging methods in other existing technologies may be used here and the embodiment of the present invention is not limited to the listed ways.

In the embodiment of the present invention, a queue-event may be introduced in the message of the announcement. For example, when there are 11 calls waiting to be processed in the queue, the message of the notification may carry a queue-event as follows:

Call-Queue: queuing

Waiting-Number: 11

Then, in the embodiment of the present invention, the calling party subscriber terminal may be

announced there are 11 calls (that is, 11 persons) waiting for service. Thus, the calling party may decide whether to wait based on the queue status of the called party, thereby improving subscriber friendliness. Of course, the calling party subscriber terminal may also be announced of its sequence to be processed in the queue. Those skilled in the art may appreciate that other types of signaling may also perform the operation.

In the embodiment of the present invention, the calling party subscriber terminal may be announced of “If you want to enter the queue (or wait), please press the key X”.

In the embodiment of the present invention, for an intelligent terminal, the issued call may also be recorded at the calling party subscriber terminal and an announcement is provided by the calling party subscriber terminal, and such processing principle is similar to the above and will be omitted.

Then, in step S603, the calling party subscriber terminal A may request for call queuing. In the embodiment of the present invention, call queuing may be initiated by pressing a pre-set specific key associated with the function (for example, X in the above text may represent any key) or by sending a specific message.

In step S604, the application server receives a call queuing request from the calling party subscriber terminal, to instruct the calling party subscriber terminal A to subscribe from the subscriber terminal B of the called party to its call queue-event status. In the embodiment of the present invention, a call queue-event may be carried in the message. How to carry call queue-event to subscribe to the B's call queue-event is shown below by example of a REFER message in SIP signaling. However, those skilled in the art may appreciate that other types of signaling may also perform the operation:

REFER A@alcatel-lucent.com SIP 2.0

...

Refer-To:B@abc.com;method=SUBSCRIBER;

Event=call-queue-network

Then in step S605, the calling party subscriber terminal A may subscribes to the queue-event status of the subscriber terminal B of the called party. How to carry the call queue-event is shown below still by example of SIP signaling. However, those skilled in the art may appreciate that other types of signaling may also perform the operation:

SUBSCRIBE B@abc.com SIP 2.0

Event: call-queue-network

In the embodiment of the present invention, a subscription request of the calling party subscriber terminal A may be forwarded to the called party subscriber terminal B via the application server.

Optionally, in the embodiment of the present invention, after the called party subscriber terminal receives a subscription message from the calling party subscriber terminal, the calling party subscriber terminal may be announced to on-hooks. Optionally, the announcement may be implemented through a media stream or short message. For example, the application server may control the removal of the link so as to generate a congestion tone at the calling party subscriber terminal to prompt the calling party subscriber terminal to on-hook. Thus, the subscriber's waiting on the line is avoided, to provide more satisfaction.

Then, in step S606, the subscription message of the calling party subscriber terminal A is received, and the called party subscriber terminal may send a notification message on the queue status to the calling party subscriber terminal via the application server. For example, on the receipt of the subscription message of the calling party subscriber terminal A, the called party subscriber terminal may immediately send to the calling party the total number of waiting and the record-count of the calling party A in the queue, and when the record-count of the calling party A in the queue is changed due to the previous call is terminated, the called party may notify the calling party of the queue status (for example, the number of call waiting and/or the record-count).

In the embodiment of the present invention, although the message carrying the call queue status is shown as follows still by example of SIP signaling, those skilled in the art may appreciate that other types of messages may also perform the operation:

NOTIFY A@alcatel-lucent.com SIP 2.0

Content-Type: application/call-queue-network

Content-Length: xx

Call-Queue: queuing

Waiting-Number: 12

Record-Info: A@alcatel-lucent.com

Record-Count: 12

In the embodiment of the present invention, the called party subscriber terminal may notify the calling party subscriber terminal upon the change of the local call queue status to update the record-count of the A. The calling party subscriber terminal may also be notified of the update of the

call queue status periodically or at fixed change intervals. Those skilled in the art may understand that the way to notify may be set according to needs, and the embodiment of the present invention is not limited to the instances above.

For example, by example of the SIP signaling, the called party subscriber terminal continues to send updated messages about the record-count of A:

NOTIFY A@alcatel-lucent.com SIP 2.0

Content-Type: application/call-queue-network

Content-Length: xx

Call-Queue: queuing

Waiting-Number: 15

Record-Info: A@alcatel-lucent.com

Record-Count: 11

In the embodiment of the present invention, optionally, after the calling party subscriber terminal on-hooks, a notification message on the queue status may be sent thereto.

Then, in step S607, the called party subscriber terminal B sends via the application server to the calling party subscriber terminal B a notification indicating the location of the call in states of the call queue to be processed, that is, a notification that it is the A's turn to get the service. In the embodiment of the present invention, the position number or the record-count may be set to e.g. "1" to indicate that the call is to be processed currently.

In the embodiment of the present invention, different with the prior art, the time to maintain the call in such status or for appropriate location may also be set, to prevent subscribers from failing to receive the notification due to some reasons and then canceling call queuing waiting.

For example, this status may be maintained for 15 minutes, where if the subscriber does not off-hook after ringing, the subscriber also may off-hook within 15 minutes to make a call, and then may establish communication with the called party without waiting.

An instance carrying a call queuing event that it is the A's turn to get the service is shown below by example of SIP signaling, wherein the expected time is 15 minutes:

NOTIFY A@alcatel-lucent.com SIP 2.0

Content-Type: application/call-queue-network

Content-Length: xx

Call-Queue: queuing

Waiting-Number: 15

Record-Info: A@alcatel-lucent.com

Record-Count: 1

Due: 15min

In the embodiment of the present invention, a message notifying the call queue-event status of the called party subscriber terminal B may be forwarded to the calling party subscriber terminal A via the application server.

Then in step S608, a call establishment request is issued at the calling party subscriber terminal A to request again for establishing communication with the called party. Optionally, in the embodiment of the present invention, when a notification indicating the A is to get the service is received, a prompt tone may be generated at the calling party subscriber terminal to remind the calling party subscriber of off-hooking to initiate a recall. For example, the prompt tone may comprise a ringing tone. Of course, sound and light prompts may also be triggered to remind the subscriber of off-hooking.

In the embodiment of the present invention, a call request may be generated again to initiate the establishment of a call by off-hooking after the prompt tone is heard.

Next, in step S609, just like a normal call establishment process, communication is established between the calling party and the called party.

In an embodiment of the present invention, when the calling party subscriber terminal establishes communication with the called party subscriber terminal, the called party subscriber terminal may terminate the subscription of the calling party subscriber terminal to the B's call queue-event. As shown below, by example of SIP message, the subscription process may be terminated by setting the subscription status (for example, Subscription-Status) in a NOTIFY message to terminated:

NOTIFY A@alcatel-lucent.com SIP 2.0

Subscription-Status: terminated; reason=noresource

Content-Type: application/call-queue-network

Content-Length: xx

Call-Queue: queuing

Waiting-Number: 14

Record-Info: A@alcatel-lucent.com

Record-Count: 0

Here, the content in “reason” may be set differently according to specific environments, for example, may also be set to “timeout”, namely, reason = timeout.

Those skilled in the art may understand that the SIP type of the message is only an example and the embodiments of the present invention is not limited to the above types of messages to implement the operation or processing.

In the embodiment of the present invention, with continuous updated call queue information and a mechanism to cancel the registration in the waiting queue (which will be described in more detail below), subscriber friendliness is improved. Since call queue processing and logic is mainly located at the subscriber terminal, and the re-call request is initiated by the calling party subscriber terminal without the need for the application server as the calling party to initiate calls with the calling party and the called party respectively to establish communication between the calling party and the called party, thus the functionality of the application server is simplified, and the resources of the application server are saved, adapting to the trend of intelligent subscriber agent and a simple core network.

In the embodiment of the present invention, queuing waiting may also be cancelled during the call to busy subscriber processing. Figure 7 illustrates a flow chart of processing of canceling the subscription during waiting for busy subscriber to cancel waiting according to an embodiment of the present invention. Shown in Figure 7, in the embodiment of the present invention, there is also provided a mechanism to cancel queuing during waiting for busy subscriber.

In step S701, after the calling party subscriber terminal A requests to subscribe to a call queuing event, of the called party subscriber terminal and on-hooks (at the time the subscriber terminal A has already been in a call queue), and wants to cancel queuing waiting during the waiting period, the calling party re-tries to call the B. In the embodiment of the present invention, the waiting period refers to a period of time from when the called party is busy to the access to the called party's call service.

In step S702, when the service is busy, the called party subscriber terminal sends a response that the subscriber is busy. In the embodiment of the present invention, similar to the steps shown in Figure 6, the call queuing event may be carried in the response to announce the calling party subscriber terminal A of the call queuing-event status of the called party subscriber terminal B.

In the embodiment of the present invention, the calling party subscriber terminal may also be checked whether to have been registered in the queue at the called party subscriber terminal B with respect to the calling party. If it has been already registered in the queue, the call queuing event status may be carried in the response, such as the number of waiting people or the record-count

indicating the location of the calling party's call in the queue and the sequence to be processed. Thus, by checking the status of the calling party in the queue, overflow of the call queue due to the continuous calls of the calling party or multiple registrations may be prevented.

In the embodiment of the present invention, it may be also at the calling party subscriber terminal A to check whether the subscriber B has been called as well as whether call waiting has been requested. If so, the subscriber A may be prompted to exit the call in order to avoid repetitive processing of the communication network. In addition, the called party's call queue may also be prevented from overflow due to the continuous calls of the calling party or multiple registrations.

Thereafter in step S703, the calling party is announced that the called party is busy and of the queuing status thereof at the time. Optionally, it may also instruct the calling party to select a services option. For example, if the calling party wants to cancel waiting, the calling party may press a pre-set specific key associated with the function (for example, y in the following text may represent any key) or send a specific message to initiate the processing of canceling waiting. If the calling party wants to wait, the calling party may on-hook. Through an announcement such as "You are already in the queue. N persons are waiting before you. If you want to cancel the waiting, please dial y. If you want to keep waiting, please on-hook", the calling party subscriber terminal A is provided with a chance to keep waiting and be aware of the record-count, and is also provided with an option to cancel the waiting.

In the embodiment of the present invention, similarly, the announcement may also be implemented by invoking MRF via the application server to perform voice play, and may also be implemented by controlling the messaging server via the application server to send short messages. Those skilled in the art may appreciate that messaging methods in other existing technologies may be used here and the embodiment of the present invention is not limited to the listed ways.

Then, in step S704, the calling party subscriber terminal A may choose to cancel the waiting, for example, the waiting may be cancelled by pressing the key y to request for canceling the waiting.

In step S705, the application server receives a request for canceling the waiting from the calling party subscriber terminal, instructs the calling party subscriber terminal A to cancel the subscription of the called party subscriber terminal B's queuing event. As shown below, by example of SIP signaling, it is shown that a REFER message may be sent to the calling party subscriber terminal to cancel the subscription:

REFER A@alcatel-lucent.com SIP 2.0

...

Refer-To:B@abc.com;method=SUBSCRIBER;

Event=call-queue-network;Expires: 0

Here, the content of “Expires” may be set to 0 to indicate cancel-subscription-event.

Thereafter, in step S706, the calling party subscriber terminal A requests to cancel the subscription of the queuing event status of the called party subscriber terminal B.

In the embodiment of the present invention, how to cancel the subscription of the call queue event may be shown by example of sending SIP signaling such as SUBSCRIBE message. However, those skilled in the art may appreciate that other types of signaling may also perform the operation:

SUBSCRIBE B@abc.com SIP 2.0

Event: call-queue-network

Expires: 0

In the embodiment of the present invention, the message of canceling the subscription may be forwarded via the application server.

Thereafter, in step S707, the called party subscriber terminal B notifies the calling party subscriber terminal A to terminate the subscription of call queuing event to the B, thus canceling the queuing waiting. The above operation is shown by example of the NOTIFY message in SIP signaling:

NOTIFY A@alcatel-lucent.com SIP 2.0

Subscription-Status: terminated; reason=timeout

Content-Type: application/call-queue-network

Content-Length: xx

Call-Queue: queuing

Waiting-Number: 14

Record-Info: A@alcatel-lucent.com

Record-Count: 0

In the embodiment of the present invention, optionally, the calling party subscriber terminal may be announced to on-hook after the called party subscriber terminal receives the unsubscribe message from the calling party subscriber terminal. Optionally, the announcement may be implemented through a media stream or short message. For example, the application server may control the removal of the link, so as to generate a congestion tone at the calling party subscriber

terminal to prompt the calling party subscriber terminal to on-hook.

In the embodiment of the present invention, optionally, a notification of call event status from the called party subscriber terminal may be received after the calling party on-hooks.

In order to more clearly show advantages of the embodiment of the present invention as compared with the existing technology, the processing of call to busy subscriber establishment according to the embodiment of the present invention will be described specifically by example of operations in a SIP environment.

Figure 8 illustrates a flow chart of processing of call to busy subscriber establishment based on a SIP environment according to the embodiment of the present invention. Shown in Figure 8, in step S801, when the calling party subscriber off-hooks and dials the subscriber B, a call establishment request INVITE message is sent at the SIP subscriber terminal A.

Then in step S802, the called party subscriber terminal is busy, and as follows, it is notified via a 486 message that the calling party is busy and the queuing status thereof at the time:

486 SIP 2.0

...

Content-Type: application/call-queue-network

Content-Length: xx

Call-Queue: queuing

Waiting-Number: 11

In the message, it may be indicated that there are 11 people (or 11 calls) waiting for the service.

Optionally, the application server may also send to the calling party A an announcement to instruct the calling party to queue and wait. In the embodiment of the present invention, as shown in Figure 9, the announcement may also be implemented by invoking MRF via the application server to perform voice play, and may also be implemented by controlling the messaging server via the application server to send short messages. Those skilled in the art may appreciate that messaging methods in other existing technologies may be used here and the embodiment of the present invention is not limited to the listed ways.

Then, in step S803, the calling party subscriber terminal A may request for call queuing. In the embodiment of the present invention, call queuing may be initiated by pressing a pre-set specific key associated with the function or by sending a specific message.

In step S804, if the subscriber A wants call queuing, the application server instructs with the REFER message the calling party subscriber terminal A to subscribe from the called party subscriber

terminal B to its call queue-event status.

Then, in step S805, the calling party subscriber terminal A subscribes with the above SUBSCRIBE message to the called party subscriber terminal B's queuing event status.

In the embodiment of the present invention, the calling party subscriber terminal A's SUBSCRIBE message may be forwarded via the application server to the called party subscriber terminal B.

In step S806, optionally, after the announcement that the request has been registered and the calling party on-hooks is received, the application server may control the removal of the link so as to generate a congestion tone at the calling party subscriber terminal to prompt the calling party subscriber terminal to on-hook.

In step S807, the subscription message of the calling party subscriber terminal A is received, and the called party subscriber terminal may send continuously the NOTIFY message to the calling party subscriber terminal via the application server to announce the called party B's call queue status until the call is in the position of the status to be processed in the call queue, for example, the record-count is 1, and send the NOTIFY message to notify that it is the A's turn to get the service.

In the embodiment of the present invention, the NOTIFY message announcing the call queue status may be received from the called party subscriber terminal after on-hooking. The NOTIFY message notifying that it is the A's turn to get the service may also trigger the generation of a prompt tone or a ringing tone, event sound and light prompts, at the calling party subscriber terminal.

In the embodiment of the present invention, when starting to send the NOTIFY message, the subscriber terminal A may be notified of the total number of waiting and the record-count of the A in the queue. When the status of the queue event is changed due to the termination of the previous call, a changed queue event status may also be sent, such as the changed record-count.

In the embodiment of the present invention, the expected time of the call may also be set to an appropriate time, such as 15 minutes.

Then, in step S808, with off-hook, the calling party subscriber terminal A issues a call establishment request INVITE message to recall the subscriber terminal B.

Next, in step S809, just as a normal call establishment process, communication is established between the calling party and the called party.

In the embodiment of the present invention, when the calling party subscriber terminal establishes communication with the called party subscriber terminal, the called party subscriber terminal may notify via the above NOTIFY message that the calling party subscriber terminal's subscription of the B's call queue-event has been terminated.

In the embodiment of the present invention, the subscriber terminal may be a SIP subscriber

terminal. However, for traditional types of subscriber terminals such as POTS line subscribers, in the embodiment of the present invention, new data link message types may be introduced to distribute the call queue status on the POTS line.

In the embodiment of the present invention, for traditional types of subscriber terminals, it is required to access a voice gateway to participate in SIP sessions. To those skilled in the art, the traditional subscriber terminal and the voice gateway may be seen as subscriber terminals in the embodiment of the present invention.

In the embodiment of the present invention, on receipt of the NOTIFY message, the voice gateway may transmit a data message with a new message type and a new parameter to the subscriber A on the POTS line, and the parameter may comprise the call record-count.

As shown below, FSK data may be used to notify the subscriber A record-count:

Message type: Call Queue Message

Parameters:

Date and Time

Calling Line ID: A

Called Line ID: B

Waiting Number: 12

Record-count: 12

Due: 15min

For example, the voice gateway receives a NOTIFY (queue-event: 12 waiting; record-count: 12) message, and then sends a FSK (recall, B, record-count: 12) message to the traditional subscriber terminal, whereby the record-count will be displayed on the traditional subscriber terminal through calling number display.

When the voice gateway receives a NOTIFY (queue-event: an 15 waiting; record-count: 1; due: 15min) message, then it sends a FSK (recall B now) message to the traditional subscriber terminal and rings the subscriber terminal or makes it to send a prompt tone and the like to prompt the subscriber to off-hook.

In the embodiment of the present invention, there is also provided a mechanism to cancel queuing waiting during SIP-environment-based call to busy subscriber processing. Figure 9 illustrates a flow chart of processing of canceling the subscription during waiting for busy subscriber based on a SIP environment according to the embodiment of the present invention. Shown in Figure 9, in step S901, similarly, after the calling party subscriber terminal requests to subscribe to a call

queuing event, of the called party subscriber terminal and on-hooks, and wants to cancel queuing waiting during the waiting period, the calling party re-tries to call the B to send a call establishment request INVITE message.

In step S902, similarly, when the service is busy, the called party subscriber terminal sends a 486 response that the subscriber is busy. In the embodiment of the present invention, the call queuing event may be carried in the response to announce the calling party subscriber terminal A of the call queuing-event status of the called party subscriber terminal B.

In the embodiment of the present invention, similarly, a check mechanism may be also performed at the called party subscriber terminal B or at the calling party subscriber terminal A to check whether the calling party subscriber terminal has been registered in the queue at with respect to the calling party so that the call queue may be prevented from being overflowed or blocked due to multiple calls of the calling party.

Thereafter in step S903, the calling party is announced that the called party is busy and of the queuing status thereof at the time. Shown in Figure 9, the application server may invoke the MRF to announce the called party being busy, the queuing status and service options.

Those skilled in the art may appreciate that messaging methods in other existing technologies may be used here and the embodiment of the present invention is not limited to the listed ways.

Then, in step S904, the calling party subscriber terminal A requests to cancel the waiting. As shown in Figure 9, the processing of canceling the waiting may be initiated by pressing a predetermined key.

In step S905, the application server receives a request for canceling the waiting from the calling party subscriber terminal, and instructs via the REFER message the calling party subscriber terminal A to cancel the subscription of the called party subscriber terminal B's queuing event.

Thereafter, in step S906, the calling party subscriber terminal requests via the SUBSCRIBE message to cancel the subscription of the queuing event status of the called party subscriber terminal B.

In the embodiment of the present invention, the message of canceling the subscription may be forwarded via the application server.

Thereafter, in step S907, optionally, after an announcement that the queuing waiting has been cancelled with on-hook is received, the application server may control the removal of the link, so as to generate a congestion tone at the calling party subscriber terminal to prompt the calling party subscriber terminal to on-hook.

In step S907, optionally, the application server may be received after waiting in line and on-hooks the circular has been cancelled, the control to remove the link to generate congestion tone

to the calling party subscriber terminal to prompt the calling party subscriber terminal on-hooks.

In step S908, in response to receiving the request of canceling the subscription from the calling party, the called party subscriber terminal B notifies via the above NOTIFY (queue-event: XX waiting; Record-Count: 0; Subscription-Status: terminated; reason = timeout) message the calling party subscriber terminal A to terminate the subscription of the call queuing event of B, thereby canceling the queuing waiting.

In the embodiment of the present invention, the NOTIFY message may also be received after on-hook from the called party B. For the traditional phone, the FSK message may also be used to notify that the calling party A has cancelled the subscription, with the principle similar to the above which is not discussed here.

Obviously, the introduction of a call queuing mechanism in the embodiment of the present invention, the use of continuous updated call queue information, and the use of a mechanism to cancel the registration in the waiting queue, facilitate the calling party in selecting a service and provide good user friendliness. Meanwhile, since call queue processing and call establishment logic are mainly located at the subscriber terminal, the processing of the application server is also simplified, and the resources of the application server are saved, adapting to the trend of intelligent subscriber agent and a simple core network.

Figure 10 shows an exemplary subscriber terminal 100 configured to establish a call in the communication network according to the embodiments of the present invention. The apparatus 100 (specifically, a processor 105) may be configured to implement the operation and function generally described above, such as processing about the initiation and establishment of the call. In addition, the apparatus 100 (specifically, the processor 105) may be configured to perform some or all of the operations described in regard to Figures 6-9 and Figure 12 (detailed below).

In some exemplary embodiments, the apparatus 100 may be implemented as or included as components computing devices and/or communications devices with wired or wireless communication capabilities. Some instances of the apparatus 100 comprise computers, servers, mobile terminals (e.g. mobile phones, portable digital assistants (PDAs), pagers, mobile TV, gaming devices, mobile computers, laptop computers, cameras, video recorders, audio/video players, radio, and/or Global Positioning System (GPS) device) or any combination thereof and so on. In addition, the apparatus 100 may be configured to implement various aspects of the present invention described herein, for example, various exemplary methods of the present invention, wherein the methods may be implemented by a processor (such as the processor 105) which configures hardware and/or software, firmware or a combination thereof, computer readable media, and the like.

The apparatus 100 may comprise the processor 105, a memory device 110 and a communication

interface 115 or communicate therewith. In addition, in some embodiments, for example an embodiment in which the apparatus 100 is a mobile terminal, the apparatus 100 further includes a user interface 125. The processor 105 may be implemented as a variety of devices, comprising for example, microprocessors, coprocessors, controller, or a variety of other processing devices comprising integrated circuits (for example, ASIC (Application Specific Integrated Circuit), FPGA (field programmable gate array), or hardware accelerator). As to a processor used as FPGA, ASIC and the like, the processor in particular may be hardware configured to perform the operations of the processor 105 described here. In the exemplary embodiment, the processor 105 is configured to execute instructions stored in the memory device 110 or instructions accessible to the processor 105. The processor 105 may be configured to facilitate communication, for example, by controlling control hardware and/or software included in the communication interface 115 or via the communication interface 115.

The memory device 110 may be configured to store various information involved in the embodiment of the present invention. The memory device 110 may be computer-readable storage media and may include volatile and/or nonvolatile memory. For example, the memory device 110 may include a random access memory (RAM) (comprising dynamic and/or static RAM), on-chip or off-chip cache memory and/or other. In addition, the memory device 110 may include non-volatile memory which may be embedded and/or be removable, and may include, for example, read-only memory, flash memory, magnetic storage (such as hard disk, floppy disk drive, tape, etc.), CD-ROM drive and/or media, non-volatile random access memory (NVRAM), and/or others. The memory device 110 may include a cache area for temporarily storing data. As a result, some or all of the memory devices 110 may be included in the processor 105.

In addition, the memory device 110 may be configured to store information, data, applications, and computer readable program code instructions and the like, for making the processor 105 and the apparatus 100 perform functions according to the exemplary embodiments of the present invention. For example, the memory device 110 may be configured to buffer input data for the processor 105 to process. Additionally or alternatively, the memory device 110 may be configured to store instructions executed by the processor 105.

The user interface 125 may communicate with the processor 105, so as to receive user input and/or output to the subscriber at the user interface 125, such as audio, visual, mechanical, or other output indications. The user interface 125 may include for example, keyboard/keypad, mouse, joystick, display (for example, touch-screen display), microphone, speaker, or other input/output mechanism. In some exemplary embodiments, the display of the user interface 125 may be configured to exhibit the call queue status according to the embodiment of the invention.

The communication interface 115 may be any device or apparatus implemented by hardware, computer program product, or a combination of hardware and software, which is configured to receive data from and/or send data to the network in communication with the apparatus 100 and/or any other device or module. Thus, the communication interface 115 may comprise for example, antennas, transmitters, receivers, transceivers and/or support hardware, comprising the processor or computer program product for supporting communication with the network.

The communication interface 115 may be configured to provide communications based on any wired or wireless communication standards. For example, the communication interface 115 may be configured to provide communications based on the following criteria: the second generation (2G) wireless communication protocol IS-136 (Time Division Multiple Access (TDMA)), GSM (Global System for Mobile Communications), IS-95 (Code Division Multiple Access (CDMA)), the third generation (3G) wireless communication protocols (such as the Universal Mobile Telecommunications System (UMTS), CDMA2000, wideband CDMA (WCDMA) and Time Division Synchronous CDMA (TD-SCDMA,)), the 3.9th generation (3.9G) wireless communication protocols (such as Evolved Universal Terrestrial Radio Access Network (E-UTRAN)), the fourth generation (4G) wireless communication protocols, International Mobile Telecommunications-Advanced (IMT-Advanced) protocol, the Long Term Evolution (LTE) protocol (comprising LTE-Advanced), etc. In addition, the communication interface 115 may be configured to provide communications according to the following technologies: radio frequency (RF), infrared (IrDA), or any one of a number of different wireless networking technologies, comprising the WLAN technology, such as IEEE 802.11 (e.g. 802.11a, 802.11b, 802.11g, 802.11n), Wireless LAN (WLAN) protocol, Microwave Access Worldwide Interoperability (WiMAX) technology (e.g. IEEE 802.16), and/or wireless personal area network (WPAN) technology, such as IEEE 802.15, Bluetooth (BT), ultra-wideband (UWB) and/or other.

In the embodiment of the present invention, the subscriber terminal may be a user terminal in a traditional communication network (e.g. PSTN) and may also be a SIP terminal. For a traditional subscriber terminal, it can be used together with the voice gateway to perform the method of establishing a call in the embodiment of the present invention.

In the embodiment of the present invention, the subscriber terminal 100 shown in Figure 10 may be configured to perform the method of establishing communication according to the embodiment of the present invention as shown in Figure 12a. Figure 12a illustrates a flow chart of a method of establishing communication with the called party according to the embodiment of the present invention. As shown in the figure, in step S1201, the subscriber terminal 100 may request call queuing at the calling party subscriber terminal in response to the called party being busy when

called. As mentioned above, call queuing may be requested via a SIP message.

Then, in step S1202, the subscriber terminal 100 may, in response to an indication to subscribe to a queuing event of the called party from the application server, subscribe to a queuing event of the called party. In the embodiment of the present invention, optionally, upon the completion of the subscription, an announcement prompting that the subscription is completed and the calling party should on-hook may be received at the subscriber terminal 100.

In step S1203, in response to a subscription request, the subscriber terminal 100 may receive information indicating the queuing state in the called party call queue. The queuing state at least comprises the record-count indicating the location of the call in the queue, and may also comprise the total number of waiting persons in the queue and/or phone numbers of waiting persons, etc. In the embodiment of the present invention, once the queue of the called party is changed, the subscriber terminal 100 may receive from the called party information notifying the queuing state of the called party call queue, and may also be configured in a certain mode to receive information notifying the queuing state of the called party call queue, for example, once two calls are reduced from the queue, the change of the status is notified.

In the embodiment of the present invention, optionally, after the subscriber terminal 100 on-hooks, information notifying the queuing state of the called party call queue from the called party may be received.

In step S1204, the subscriber terminal 100 may receive a notification indicating the location of the call in the status to be processed currently in the queue. In the embodiment of the present invention, if the notification is received after on-hook, a prompt tone is generated to prompt the calling party subscriber to off-hook. For example, the prompt tone may be a ringing tone or sound and light alarm mode.

In the embodiment of the present invention, the subscriber terminal 100 may, in response to off-hook, send a call establishment request to request for establishing communication with the called party again.

In the embodiment of the present invention, the subscriber terminal 100 may also cancel queuing waiting by canceling the subscription during the call waiting period. The process may be initiated by trying calling. In addition, the subscriber terminal 100 may also receive from the called party a message indicating the subscription has been cancelled to notify that the queuing waiting has been cancelled.

In the embodiment of the present invention, the subscriber terminal 100 may receive information notifying that the called party is busy when the called party is busy when being called, and this information may contain information on the queuing status of the called party call queue.

In the embodiment of the present invention, the subscriber terminal 100 shown in Figure 10 may be configured to perform the method of establishing communication according to the embodiment of the present invention as shown in Figure 12b. Figure 12b illustrates a flow chart of a method of establishing communication with the calling party according to the embodiment of the present invention. As shown in the figure, in step S1211, in response to a request from the calling party for subscribing to a queuing event of a called party when the called party is busy when being called, information of a queuing status of the called party call queue is sent. The queuing state at least comprises the record-count indicating the location of the call in the queue, and may also comprise the total number of waiting persons in the queue and/or phone numbers of waiting persons, etc. In the embodiment of the present invention, once the queue of the called party is changed, the subscriber terminal 100 may send information of the queuing status of the called party call queue, and may also be configured in a certain mode to send information of the queuing status of the called party call queue.

In step S1212, when the call in the called party call queue is to be processed, the subscriber terminal 100 may send a notification indicating the location of the call in the status to be processed currently in the queue.

Finally, in step S1213, in response to a recall establishment request of the calling party, the subscriber terminal 100 may, in response to the request, establish communication with the calling party.

In the embodiment of the present invention, preferably, if the calling party receives the notification indicating the location of the call in the status to be processed currently in the queue after on-hook, then the notification may trigger the generation of a prompt tone at the calling party subscriber terminal. In the embodiment of the present invention, the recall request of the calling party may be initiated in response to the notification indicating the location of the call in the status to be processed currently in the queue, for example, a call establishment request may be generated by off-hook.

In the embodiment of the present invention, the subscriber terminal 100 may send information notifying that the called party is busy if the called party is busy when being called, wherein the information contains information about the queuing status of the called party call queue.

In the embodiment of the present invention, the subscriber terminal 100 may also check whether the call from the calling party is in the call queue when the called party is busy when being called, and if so, does not register the call request in the queue again to avoid overflow of the call queue.

In the embodiment of the present invention, the subscriber terminal 100 may also notify, during a call waiting period, in response to the receipt of a request of canceling the subscription from the

calling party, the calling party to cancel the subscription to notify to cancel queuing waiting.

In the embodiment of the present invention, the queuing status at least comprises one of the following: statuses such as the called party is out of service, the queue being empty, queuing or the queue being full; call waiting number in the queue; the phone number of the call waiting party; the serial number or location of the calling party in the called party queue, indicating the sequence of the call to be processed (i.e., the record-count); and expected time to preserve the call.

In the embodiment of the present invention, the above processing may be performed based on SIP.

In the embodiment of the present invention, the operations on the subscriber terminal 100 may be performed by the processor 105 which may be configured specifically to perform the operations of Figure 12 and may also be configured to perform part of the operations in Figures 6-9.

Figure 11 illustrates a block diagram of an application server 200 configured to provide a call queue status in the communications network to establish a call according to an embodiment of the invention. The apparatus 200 (specifically, a processor 205) may be configured to implement the operation and function generally described above, such as processing about the initiation and establishment of the call. In addition, the apparatus 200 (specifically, the processor 205) may be configured to perform some or all of the operations described in regard to Figures 6-9 and Figure 12.

Similarly, the apparatus 200 may comprise the processor 205, a memory device 210 and a communication interface 215 or communicate therewith. In the embodiment of the present invention, the apparatus 200 may also include a user interface 225. In addition, in some embodiments, for example an embodiment in which the apparatus 200 is an application server. Similarly, the processor 205 may be implemented as a variety of devices, comprising for example, microprocessors, coprocessors, controller, or a variety of other processing devices including integrated circuits (for example, ASIC (Application Specific Integrated Circuit), FPGA (field programmable gate array), or hardware accelerator). As to a processor used as FPGA, ASIC and the like, the processor in particular may be hardware configured to perform the operations of the processor 105 described here. In the exemplary embodiment, the processor 205 is configured to execute instructions stored in the memory device 210 or instructions accessible to the processor 205. The processor 205 may be configured to facilitate communication, for example, by controlling control hardware and/or software included in the communication interface 215 or via the communication interface 215.

In the embodiment of the present invention, the application server 200 may be configured to perform a method for controlling the establishment of a call between a calling party and a called party, comprising: in response to a request for call queuing of the calling party when the called party is busy when being called, instructing the calling party to subscribe to a queuing event of the called

party; forwarding to the called party a request for subscribing to the queuing event of the calling party; forwarding to the calling party information of status of the queuing event of the called party, wherein the queuing status at least comprises a record-count indicating a location of the call in the queue; forwarding to the calling party an announcement of the location of the call in a status to be processed currently in the queue to instruct the calling party to send a call establishment request to request again for establishing communication.

In the embodiment of the present invention, the call establishment request may be initiated in response to the notification of the location of the call in the status to be processed currently in the queue, for example, a call establishment request may be generated via off-hook.

In the embodiment of the present invention, the application server 200 may also be configured to instruct, during a call waiting period, in response to the receipt of a request from the calling party for canceling queuing waiting, the calling party to cancel the subscription so as to cancel queuing waiting.

In the embodiment of the present invention, the application server 200 may also be configured to forward to the called party the request from the calling party of canceling the subscription so as to cancel the subscription to the queuing event of the called party.

In the embodiment of the present invention, the application server 200 may also be configured to forward to the calling party a notification of canceling the subscription from the called party so as to notify canceling queuing waiting.

In the embodiment of the present invention, the application server 200 may also be configured to receive information notifying that the called party is busy from the called party when the called party is busy when being called, wherein the information contains information about the queuing status of the called party call queue.

In the embodiment of the present invention, the application server 200 may also be configured to invoke Media Resource Function to provide to the calling party information of the queuing status of the called party call queue.

In the embodiment of the present invention, the above processing may be performed based on SIP.

In one exemplary embodiment, one or more processes described here may be implemented by a computer program product comprising program code instructions. As a result, the program code instructions implementing the processes described here may be stored by a memory device (such as memory devices 110, 210) of the apparatus (such as apparatus 100 and/or 200) or stored thereon, and be executed by the processor (such as processor 105, 205). It may be appreciated that any such program code instructions may be loaded onto a computer, processor or other programmable

apparatus (e.g., processor 105 and 205, memory device 110 and 210) to produce a machine, which comprises an apparatus for implementing functions specified in blocks, steps or operations of the flow charts. In some exemplary embodiments, these program code instructions are also stored in a computer readable storage medium, instruct the computer, processor or other programmable apparatus to run in a specific way, so that the instructions stored in the computer readable storage medium make products, which also provide an apparatus for implementing functions specified in blocks, steps or operations of the flow charts. The program code instructions may also be loaded onto the computer, processor or other programmable apparatus to perform a serial of operation steps on the computer, processor or other programmable apparatus or thereby, to generate computer-implemented processing, so that instructions executed on the computer, processor or other programmable apparatus provide steps for implementing functions specified in blocks, steps or operations of the flow charts.

Therefore, blocks, steps or operations of the flow charts support a combination of apparatuses for perform specific functions, a combination of steps for performing specific functions, and program code instructions for performing specific functions. It may also be appreciated that in some exemplary embodiments, one or more blocks, steps or operations of the flow charts, and the combination of the blocks, steps or operations of the flow charts, are implemented by combinations of a computer system or processor based on dedicated hardware, or dedicated hardware and program code instructions for performing specific functions or steps.

Those skilled in the art should understand that although embodiments of the present invention are shown in 3G using SIP as instances in the description, other appropriate systems and protocols may also be used.

Many modifications and other embodiments of the present invention recited here will enable those skilled in the art related to the present invention to benefit from teaching provided in the description and related drawings. Therefore, it can be appreciated that the present invention is not limited to the disclosed specific embodiment, and that the modifications and other embodiments are intended to be encompassed within the scope of the appended claims. In addition, although the above description and related drawings describe exemplary embodiments in an environment of some exemplary combination of parts and/or functions, it should be appreciated that different combinations of parts and/or functions may be provided through alternative embodiments, without departing from the scope of the appended claims. Thus, for example, just as recited in some of the appended claims, different combinations other than those of parts and/or functions explicitly described hereinabove may also be envisaged. Although specific terms are used here, they are used only to have a general and descriptive meaning, rather than for limiting purpose.

CLAIMS

1. A method for establishing communication with a called party, comprising:
 - in response to the called party being busy when called, requesting for call queuing at a calling party subscriber terminal;
 - in response to an indication to subscribe to a called party queuing event from an application server, subscribing to a queuing event of the called party at the calling party subscriber terminal;
 - after the calling party on-hooks, receiving information indicating a queuing status of a queue of calls of the called party at the calling party subscriber terminal, wherein, the queuing status at least comprises a record-count indicating a location of the call in the queue; and
 - in response to the receipt of a notification indicating the location of the call in a status to be processed currently in the queue, generating a prompt tone at the calling party subscriber terminal to prompt the calling party to off-hook.
2. The method according to claim 1, wherein the method further comprises: in response to the off-hook, sending a call establishment request at the calling party subscriber terminal to request again for establishing communication with the called party.
3. The method according to claim 1 or 2, wherein the method further comprises: during a call waiting period, canceling queuing waiting at the calling party subscriber terminal by canceling the subscription.
4. The method according to claim 3, wherein the step of said canceling queuing waiting further comprises: initiating cancelling the subscription by trying calling so as to cancel queuing waiting.
5. The method according to claim 1 or 2, wherein the method further comprises: if the called party is busy when called, receiving information notifying that the called party is busy at the calling party subscriber terminal, wherein the information contains information about the queuing status of the called party call queue.
6. A method for establishing communication with a calling party, comprising:
 - in response to a request from the calling party for subscribing to a queuing event of a called party when the called party is busy when called, sending at a called party subscriber terminal information of a queuing status of the called party call queue, wherein the queuing status at least comprises a record-count indicating a location of the call in the queue;
 - sending a notification indicating a location of the call in a status to be processed

currently in the queue; and

establishing communication with the calling party in response to a recall establishment request of the calling party.

7. The method according to claim 6, wherein the method further comprises: if the called party is busy when called, sending information notifying that the called party is busy at the called party subscriber terminal, wherein the information contains information about a queuing status of the called party call queue.

8. The method according to claim 6 or 7, wherein the method further comprises: during a call waiting period, in response to the receipt of a request of canceling the subscription from the calling party, notifying the calling party to cancel the subscription so as to notify cancelling queuing waiting.

9. A method for controlling the establishment of a call between a calling party and a called party, comprising:

in response to a request for call queuing of the calling party when the called party is busy when called, instructing the calling party to subscribe to a queuing event of the called party at an application server;

forwarding to the called party a request for subscribing to the queuing event of the calling party;

forwarding to the calling party information of status of the queuing event of the called party, wherein the queuing status at least comprises a record-count indicating a location of the call in the queue; and

forwarding to the calling party a notification of a location of the call in a status to be processed currently in the queue so that the calling party generates a prompt tone to prompt off-hook.

10. The method according to claim 9, wherein the method further comprises: forwarding to the called party a call establishment request from the calling party in response to the off-hook to request again for establishing communication.

11. The method according to claim 9 or 10, wherein the method further comprises at least one of the following steps:

during a call waiting period, in response to the receipt of a request from the calling party for canceling queuing waiting, instructing at the application server the calling party to cancel the subscription so as to cancel queuing waiting; and

forwarding to the called party a request from the calling party of canceling the

subscription so as to cancel the subscription to the queuing event of the called party.

12. The method according to claim 11, wherein the method further comprises: forwarding to the calling party a notification of canceling the subscription from the called party so as to notify canceling queuing waiting.

13. The method according to any one of claims 9 to 12, wherein the method further comprises: receiving information notifying that the called party is busy from the called party when the called party is busy when called, wherein the information contains information about the queuing status of the called party call queue.

14. The method according to claim 13, wherein the method further comprises: invoking Media Resource Function to provide to the calling party information of the queuing status of the called party call queue.

15. A computer program product, comprising computer executable instructions, which cause, when executed on a computer device, the device to be configured to execute the method according to any one of claims 1 to 14.

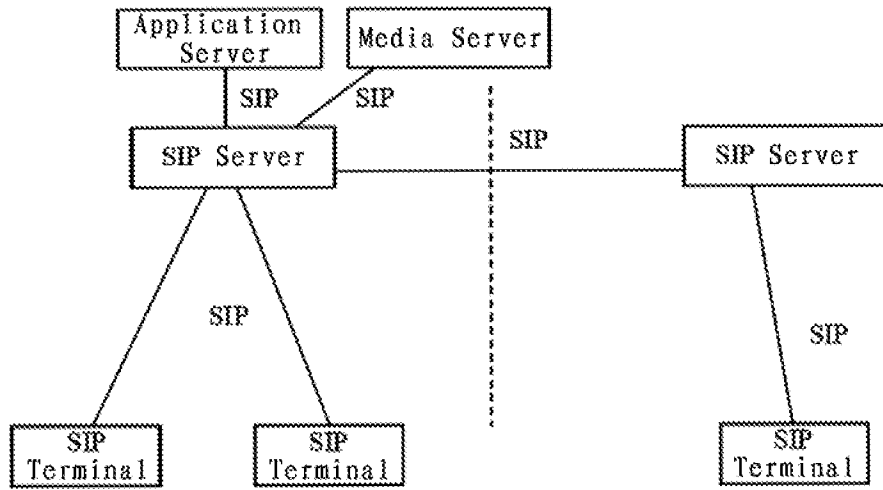


Figure 1

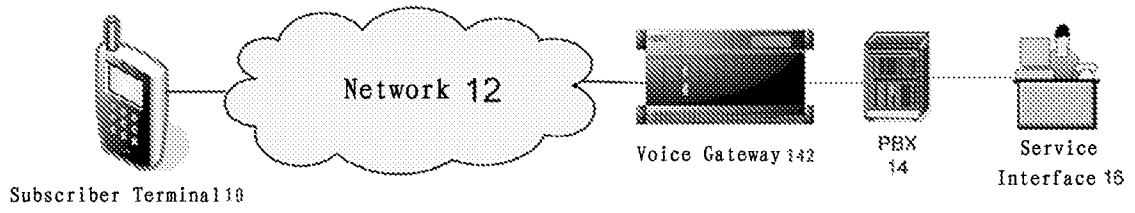


Figure 2

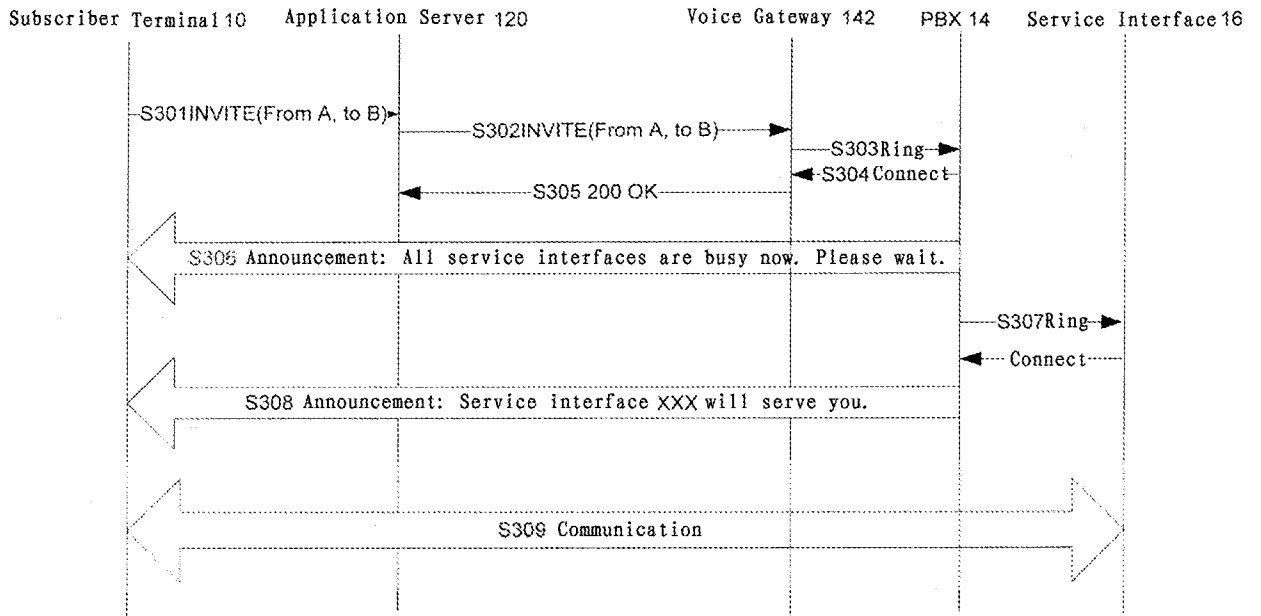


Figure 3

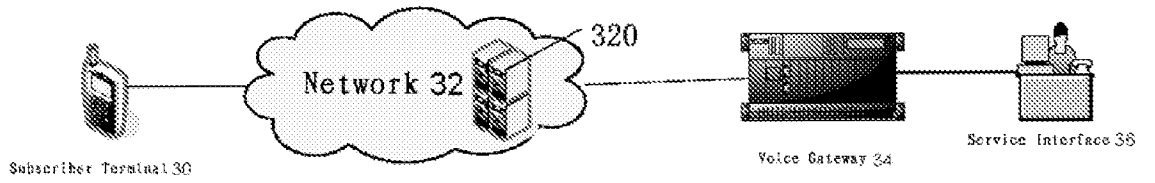


Figure 4

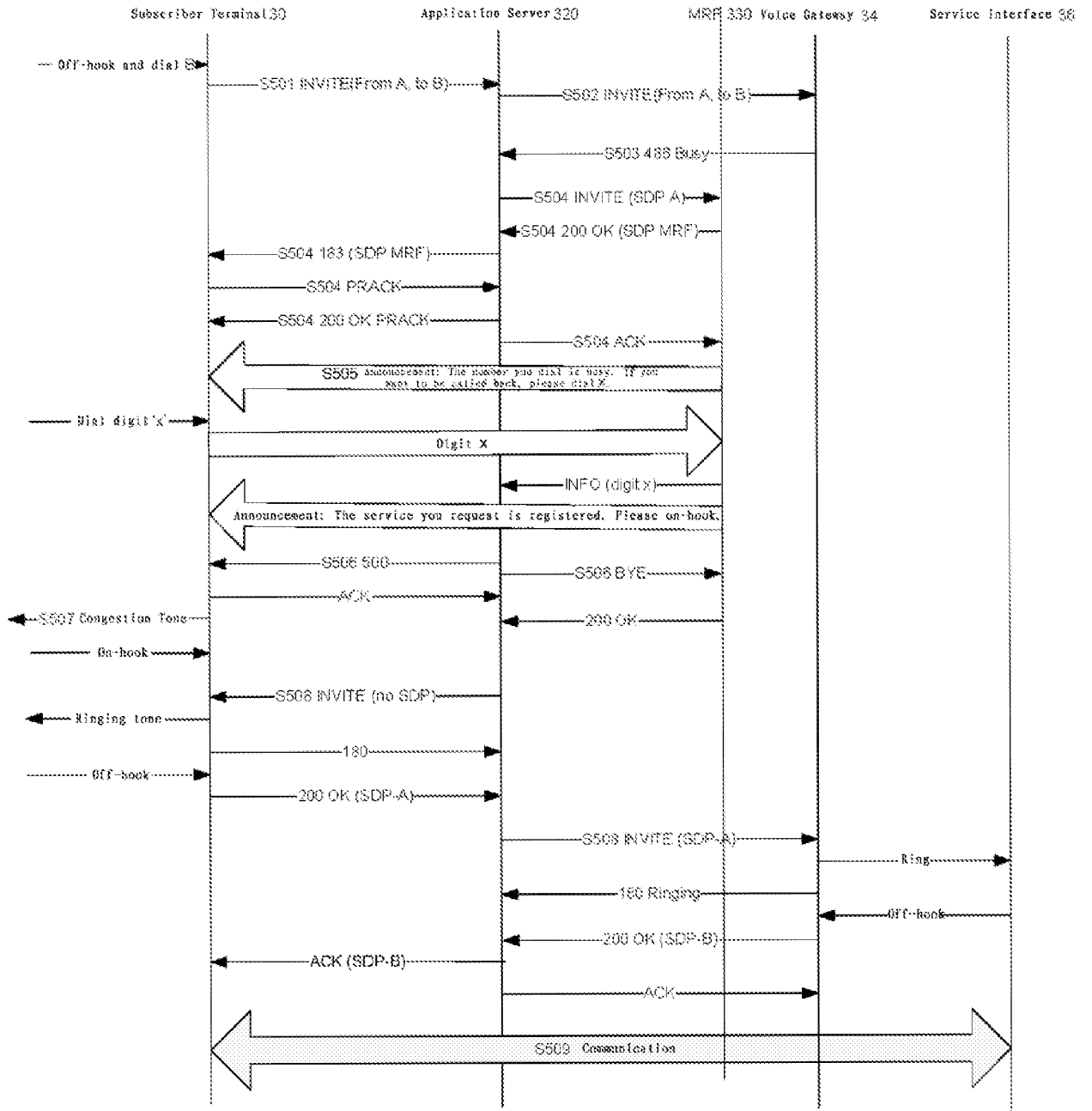


Figure 5

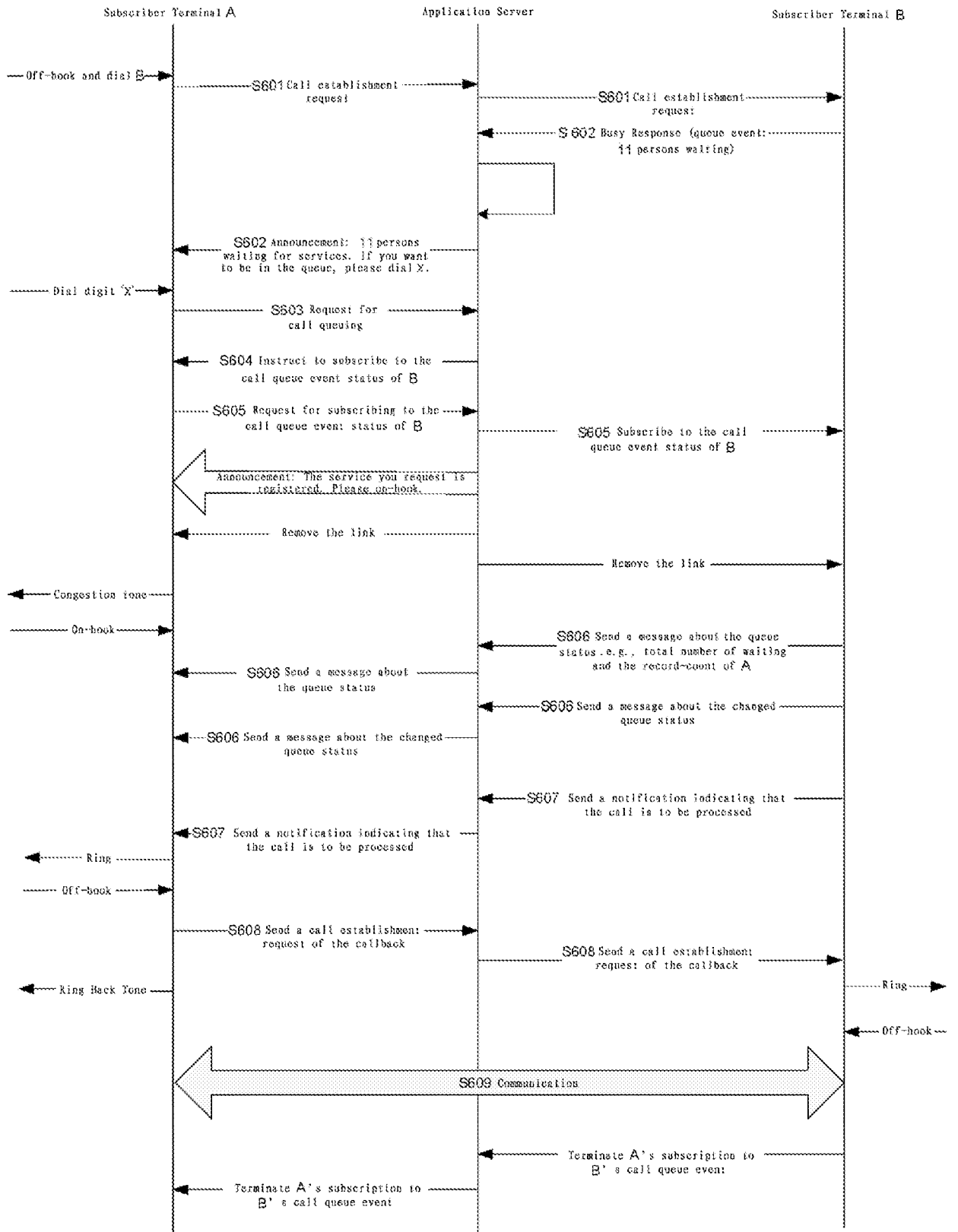


Figure 6

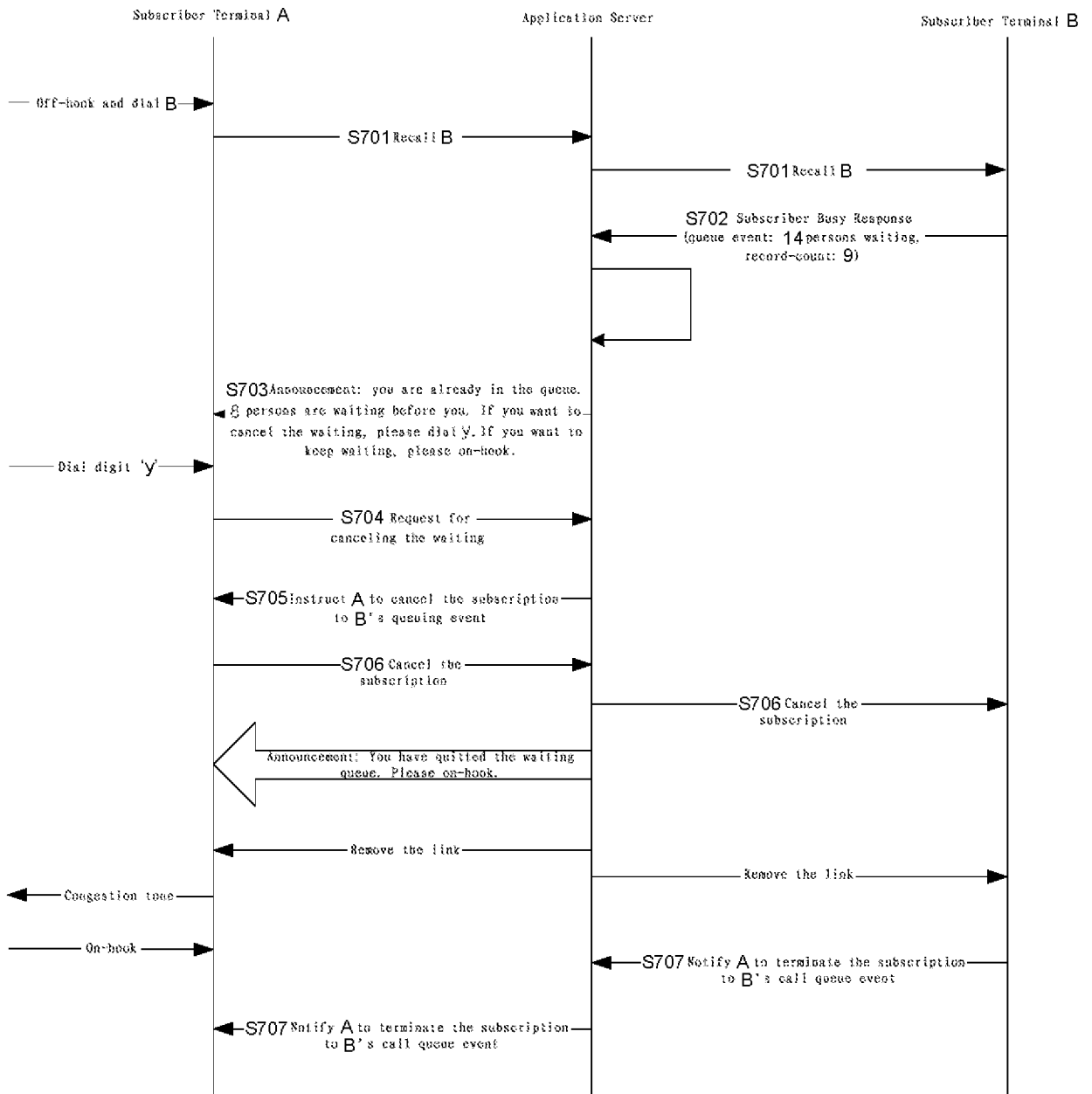


Figure 7

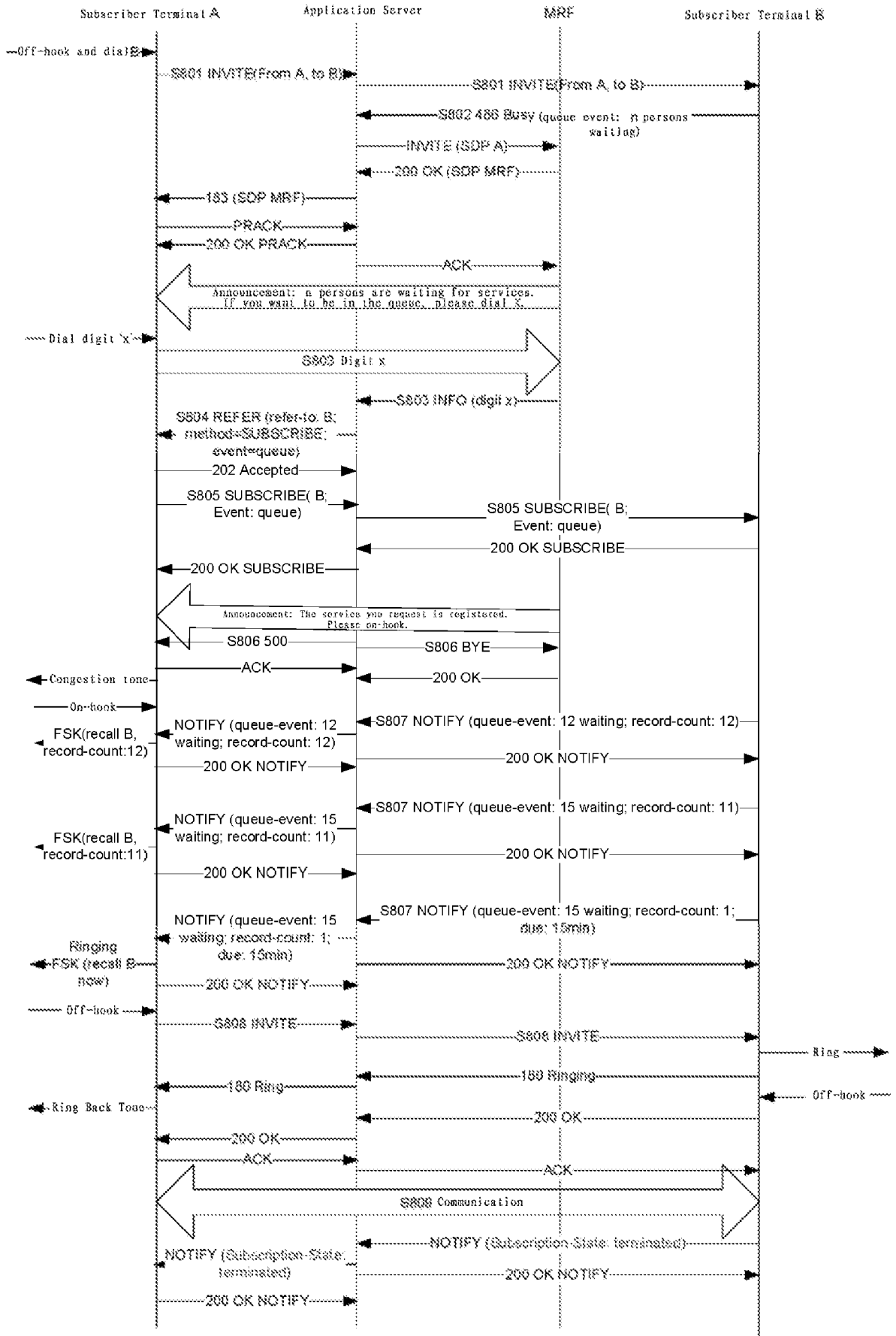


Figure 8

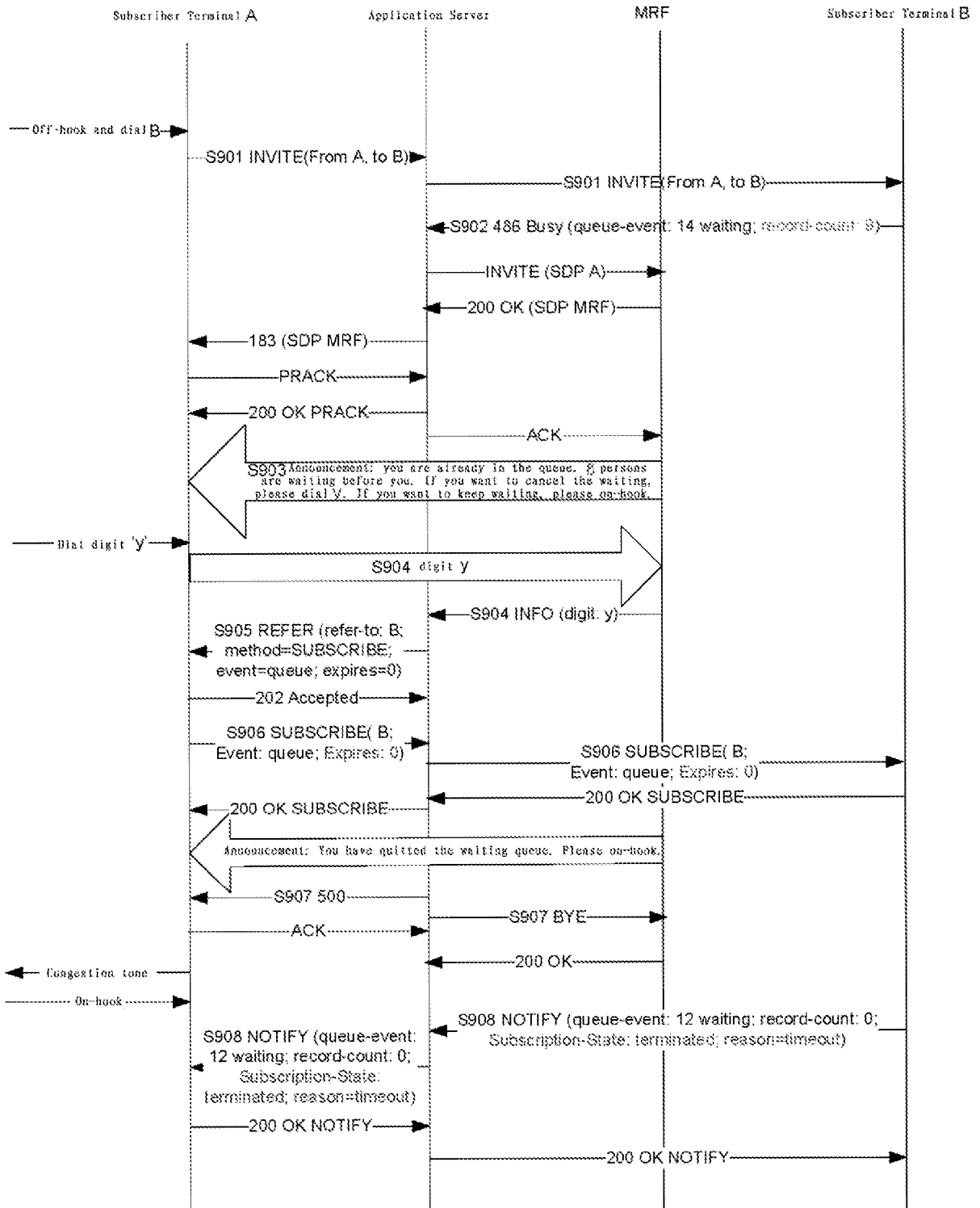


Figure 9

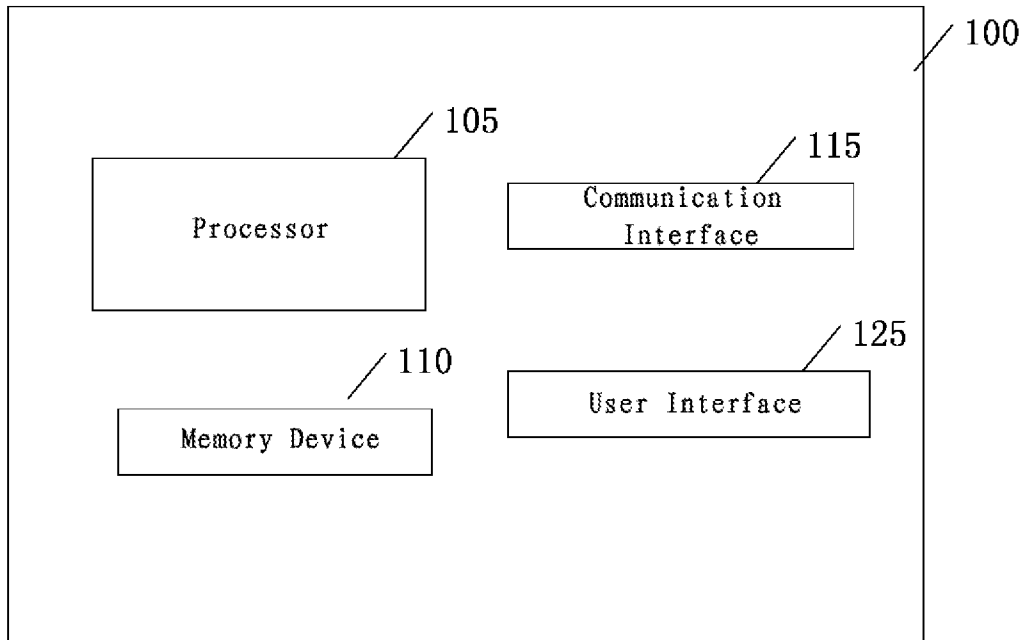


Figure 10

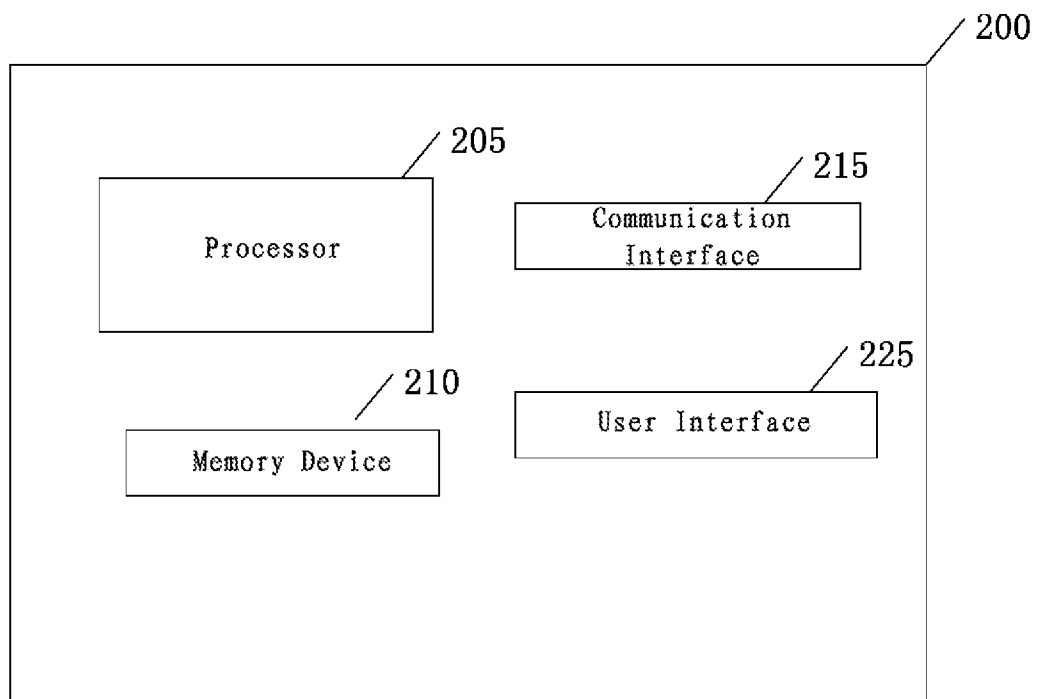


Figure 11

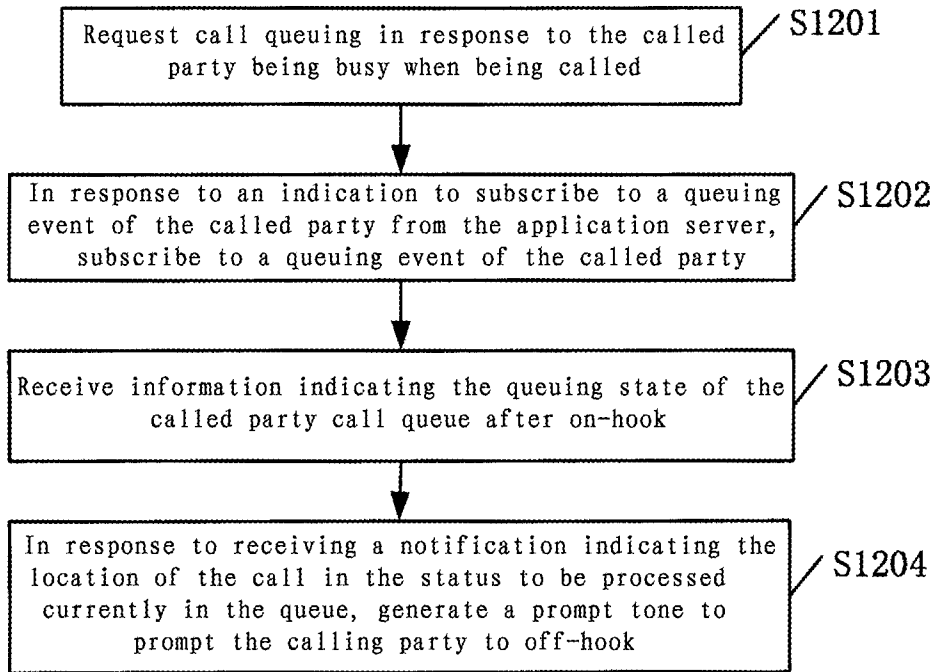


Figure 12a

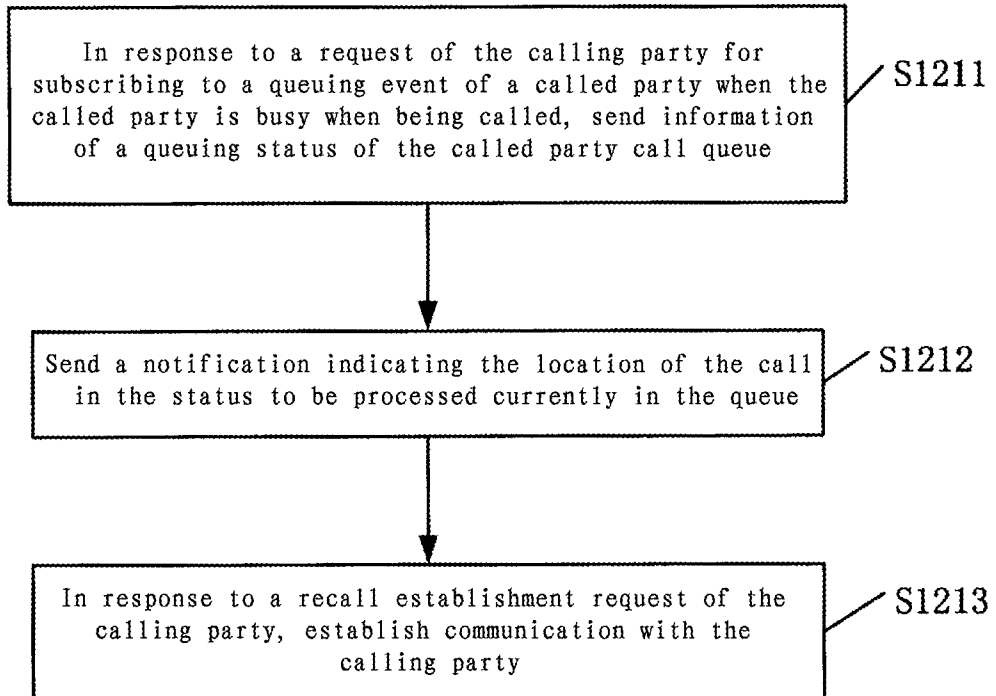


Figure 12b