



US 20170272489A1

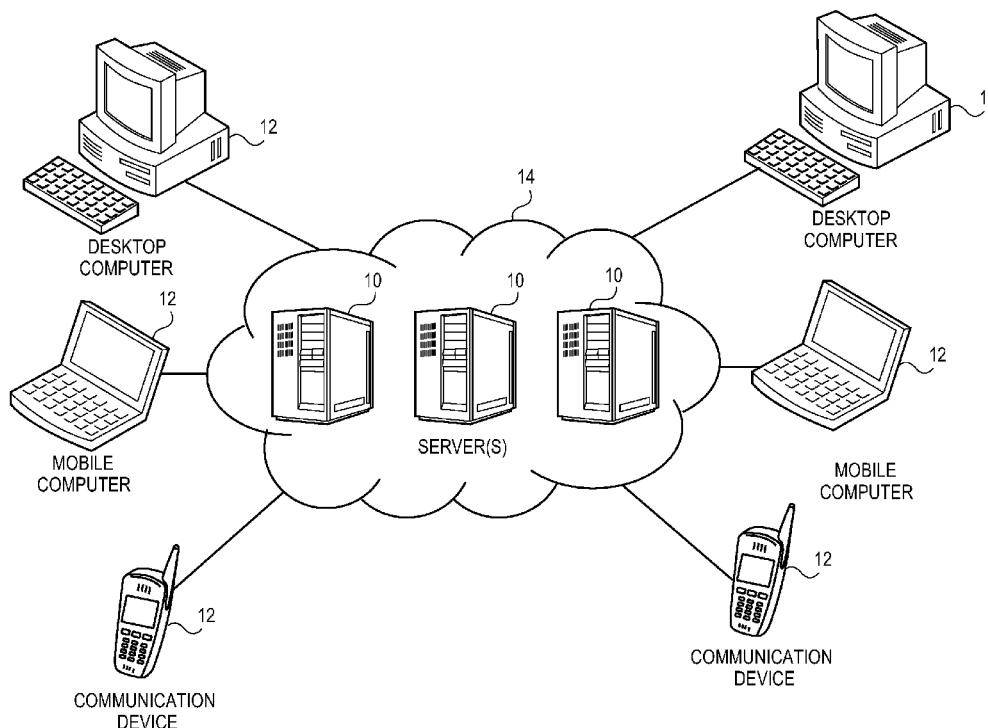
(19) **United States**(12) **Patent Application Publication**
KATIS et al.(10) **Pub. No.: US 2017/0272489 A1**(43) **Pub. Date: Sep. 21, 2017**(54) **METHOD FOR DOWNLOADING AND USING
A COMMUNICATION APPLICATION
THROUGH A WEB BROWSER**(71) Applicant: **Voxer IP LLC**, San Francisco, CA (US)(72) Inventors: **Thomas E. KATIS**, Jackson, WY (US);
James T. PANTTAJA, San Francisco,
CA (US); **Mary G. PANTTAJA**, San
Francisco, CA (US); **Matthew J.
RANNEY**, Oakland, CA (US)(21) Appl. No.: **15/615,406**(22) Filed: **Jun. 6, 2017****Related U.S. Application Data**

(60) Division of application No. 12/883,116, filed on Sep. 15, 2010, which is a continuation-in-part of application No. 12/028,400, filed on Feb. 8, 2008, now Pat. No. 8,180,029, said application No. 12/883,116 is a continuation-in-part of application No. 12/561,089, filed on Sep. 16, 2009, now Pat. No. 8,533,611, said application No. 12/883,116 is a continuation-in-part of application No. 12/419,861, filed on Apr. 7, 2009, now abandoned, which is a continuation-in-part of application No. 12/552,980, filed on Sep. 2, 2009, now Pat. No. 8,645,477, which is a continuation-in-part of application No. 12/857,486, filed on Aug. 16, 2010, now Pat. No. 9,178,916.

(60) Provisional application No. 60/937,552, filed on Jun. 28, 2007, provisional application No. 60/999,619, filed on Oct. 19, 2007, provisional application No. 61/232,627, filed on Aug. 10, 2009, provisional application No. 61/148,885, filed on Jan. 30, 2009.

Publication Classification(51) **Int. Cl.**
H04L 29/06 (2006.01)
H04L 12/18 (2006.01)
H04L 12/58 (2006.01)
(52) **U.S. Cl.**
CPC **H04L 65/604** (2013.01); **H04L 12/1831**
(2013.01); **H04L 51/04** (2013.01); **H04L**
65/1083 (2013.01); **H04L 65/4015** (2013.01);
H04L 51/32 (2013.01)(57) **ABSTRACT**

A method of enabling communication over a network by maintaining a server on a network and receiving a request at the server from a user of a communication device. In response to the request, a communication application is downloading over the network to the communication device. The communication application enabling the user to participate in a conversation on the communication device in either (i) a real-time mode or (ii) a time-shifted mode and (iii) to seamlessly transition the conversation between the two modes (i) and (ii).



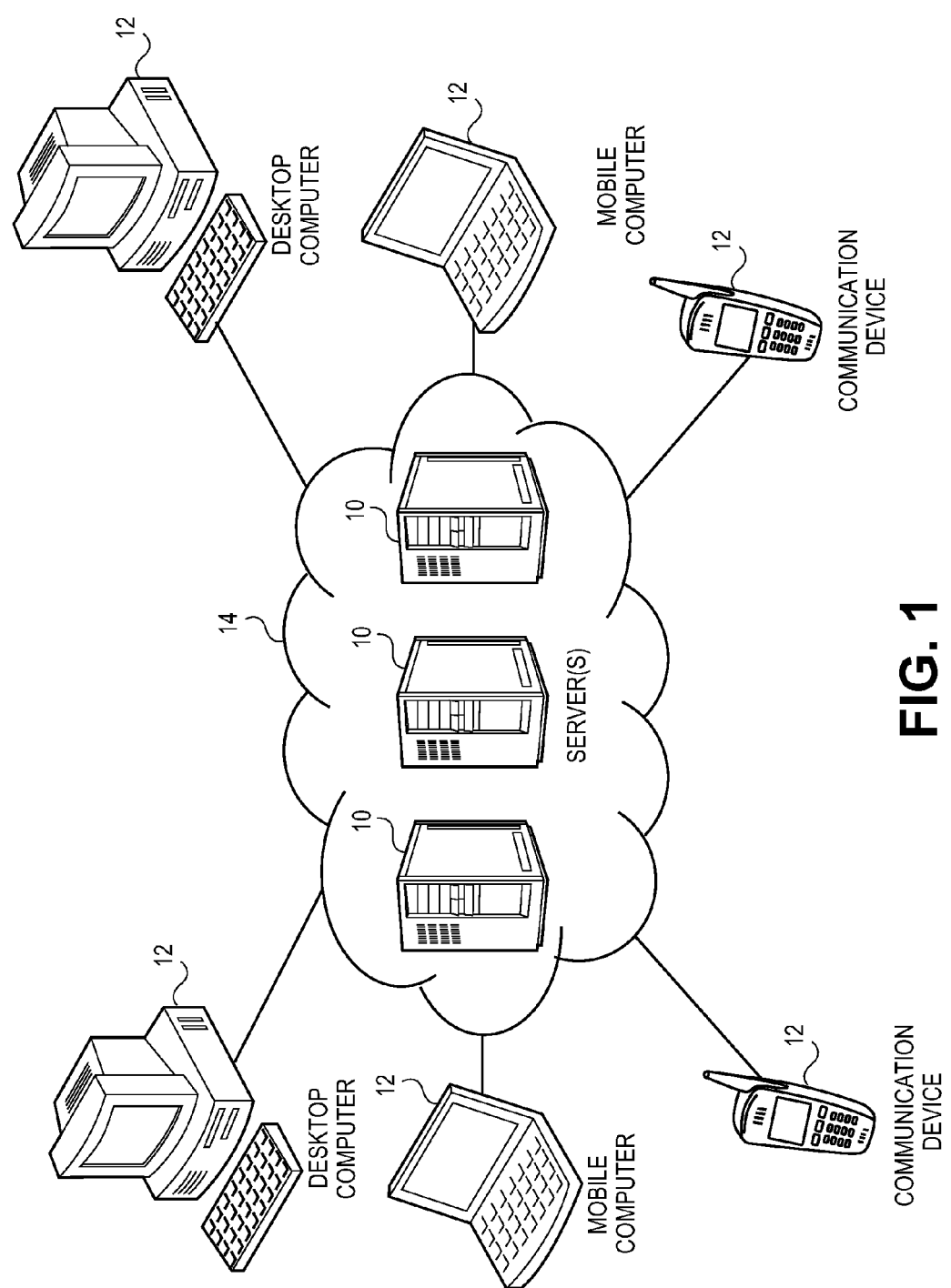


FIG. 1

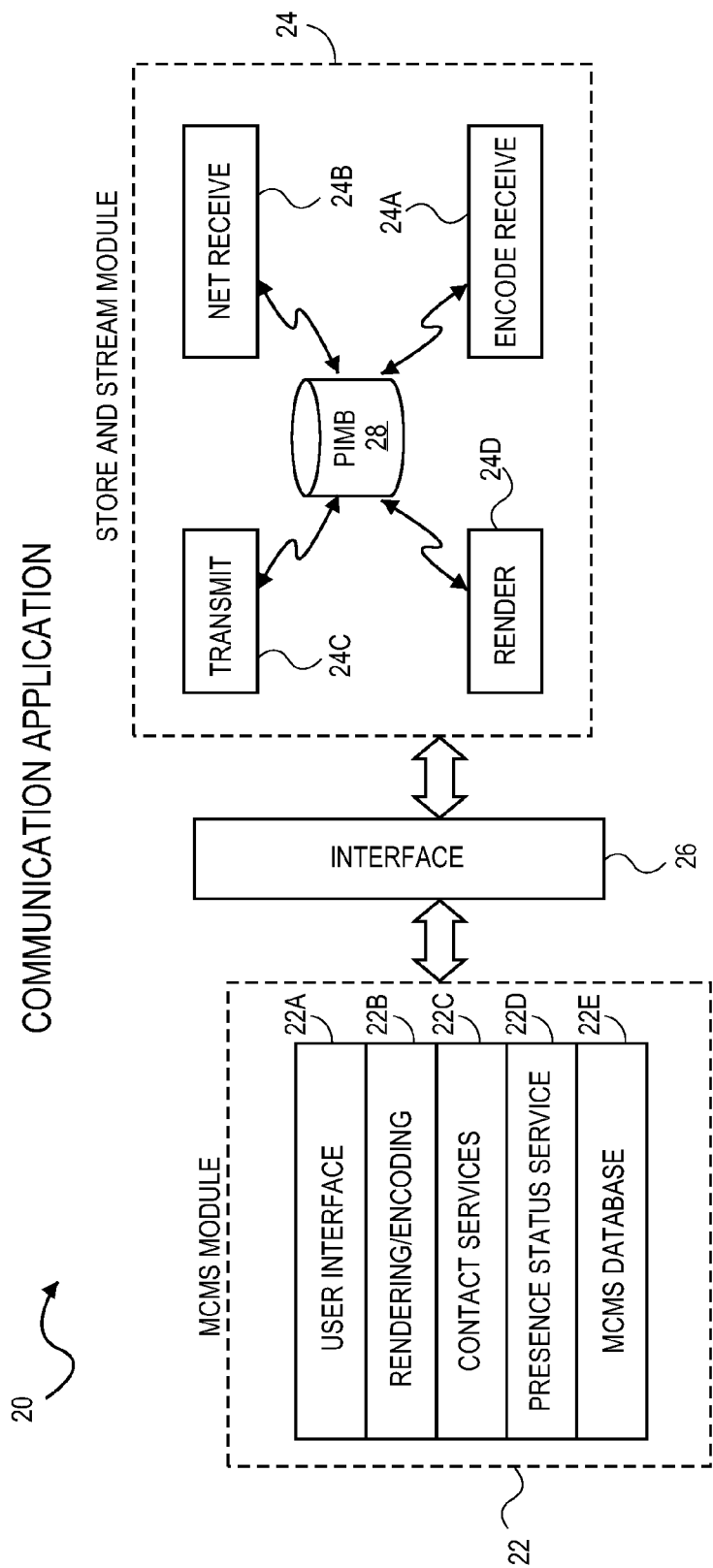


FIG. 2

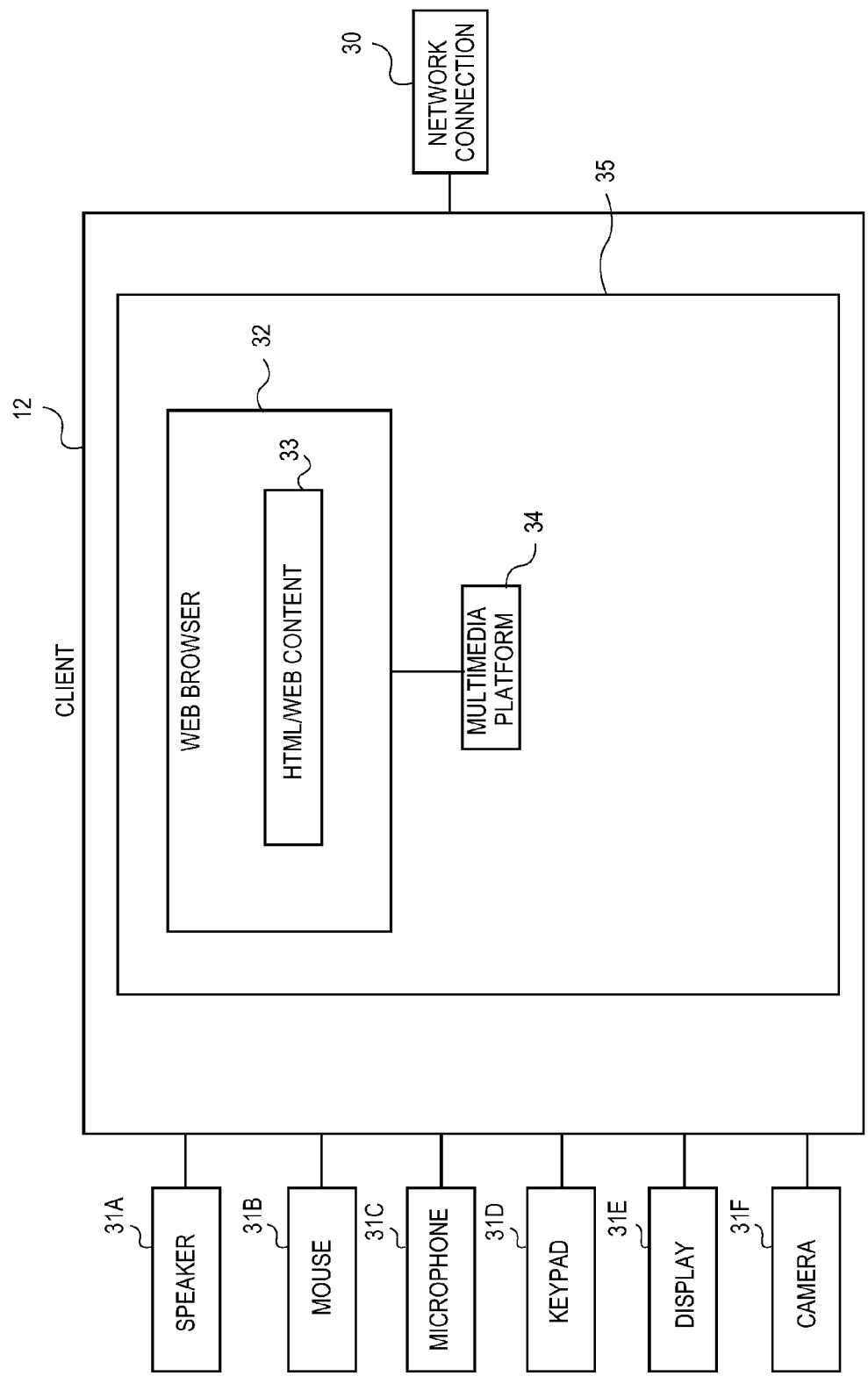


FIG. 3A

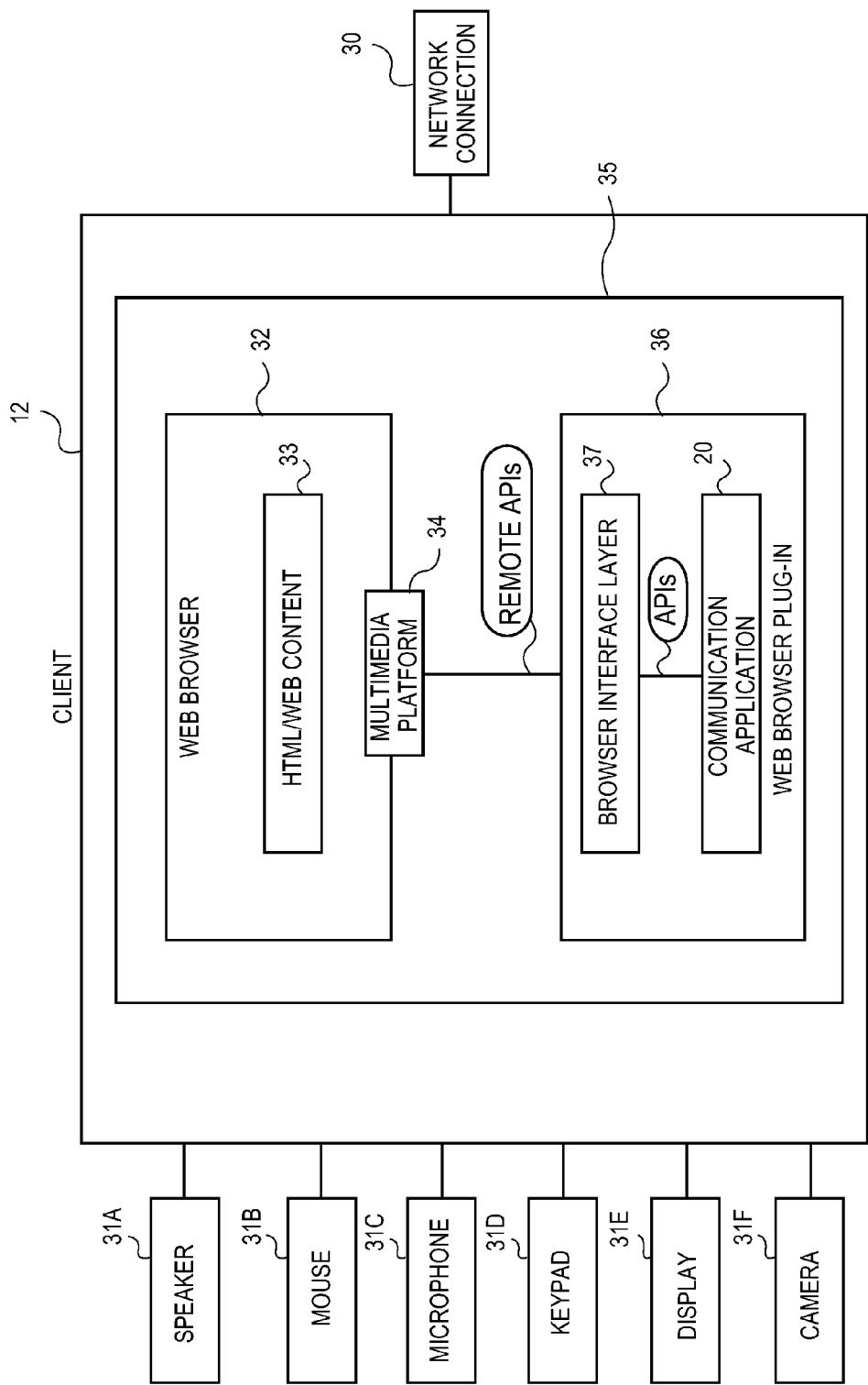
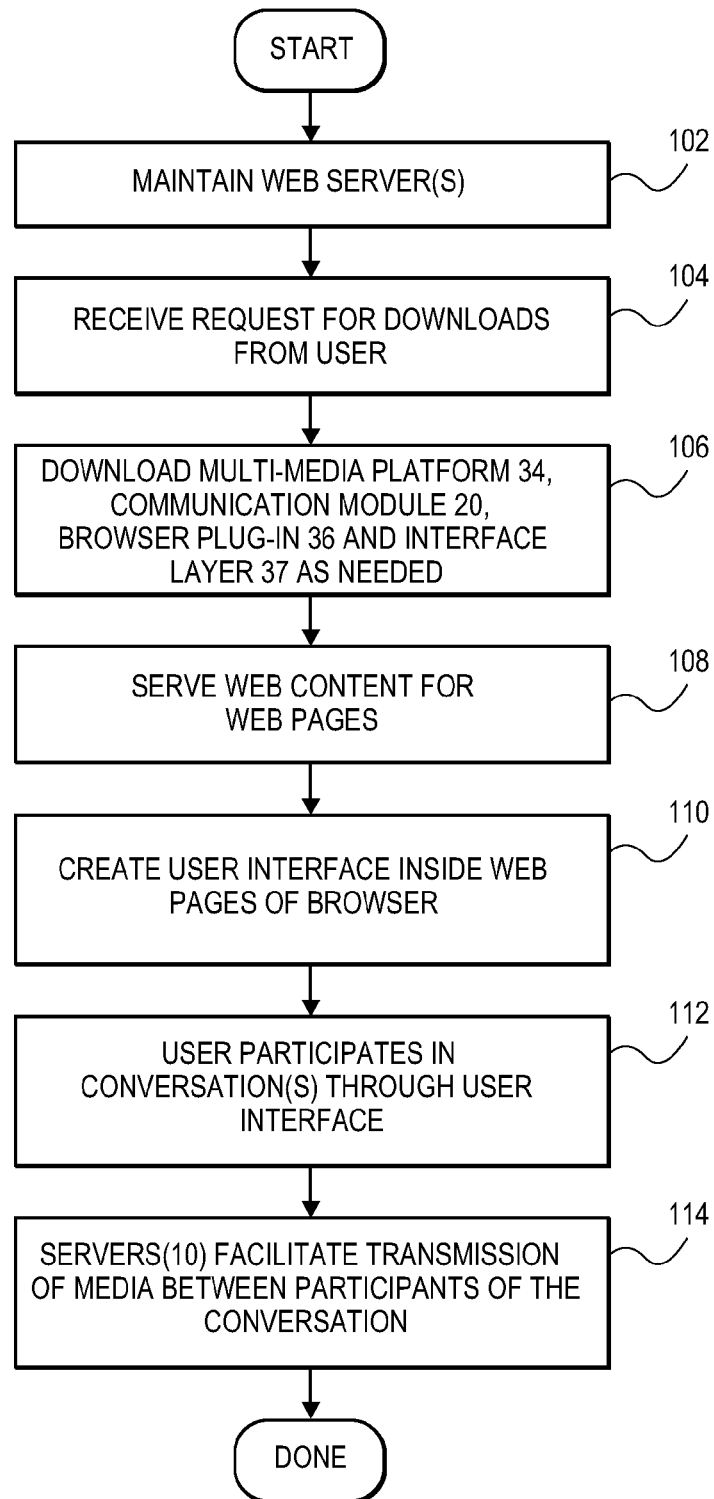
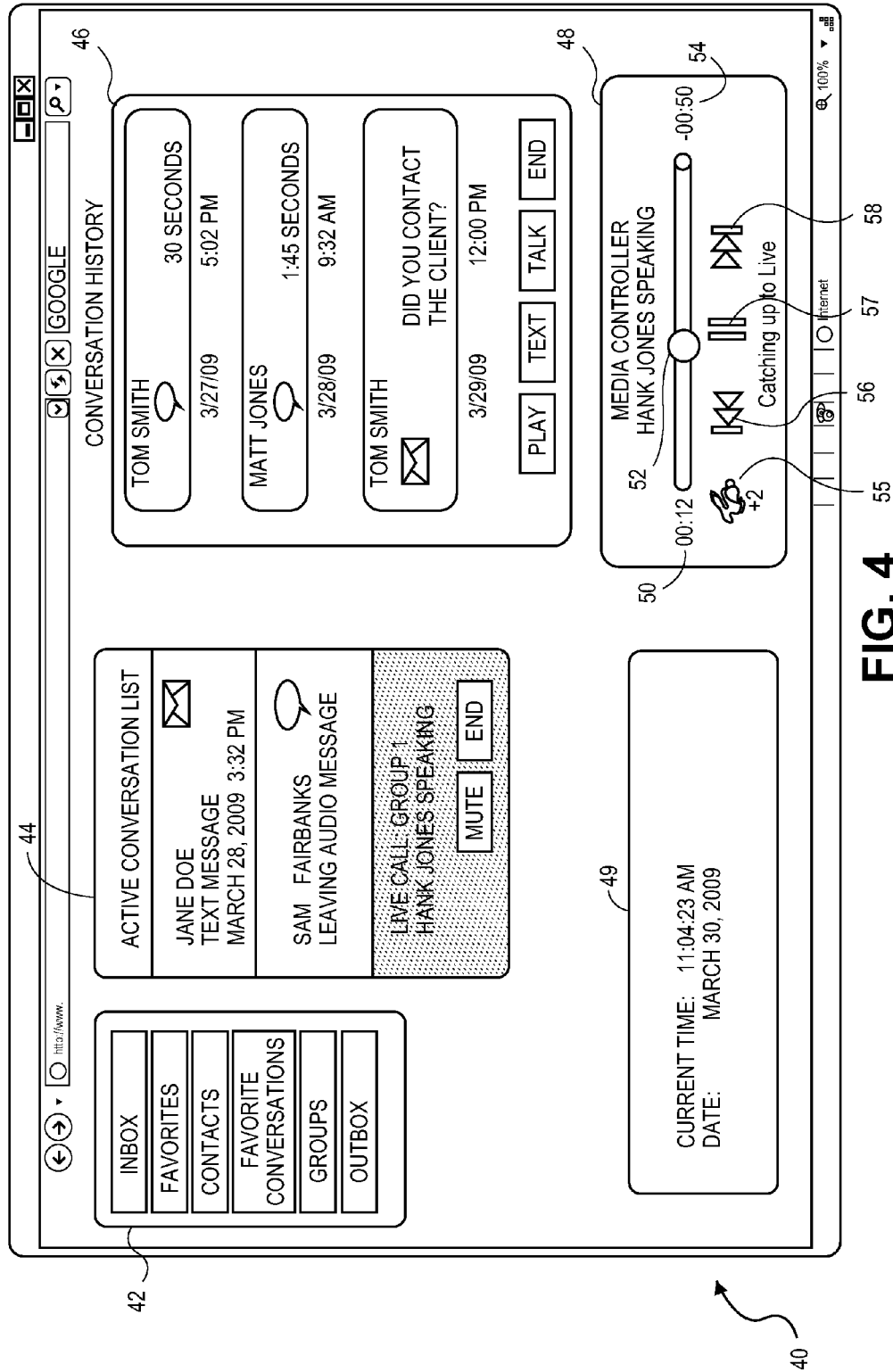


FIG. 3B

**FIG. 3C**



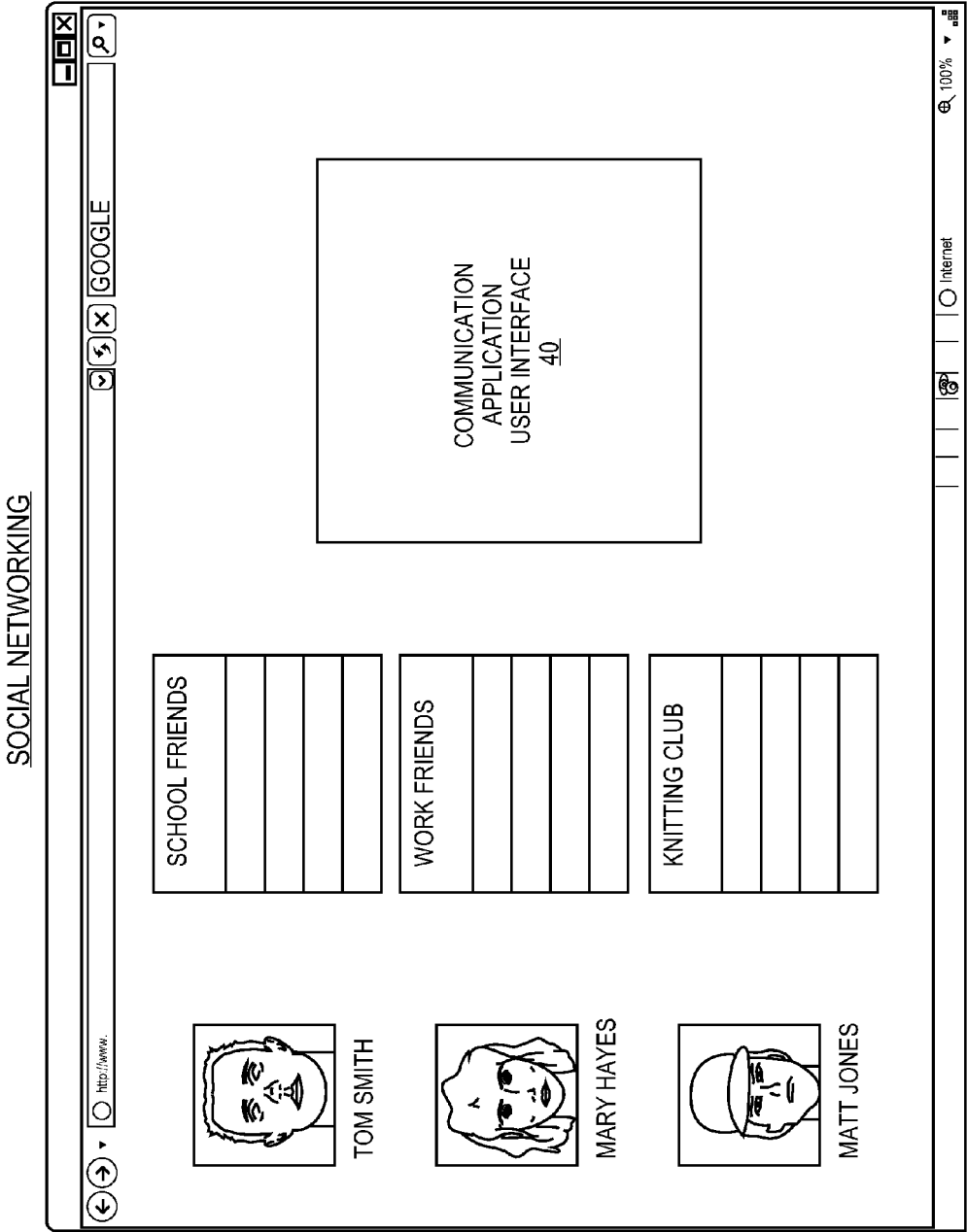


FIG. 5A

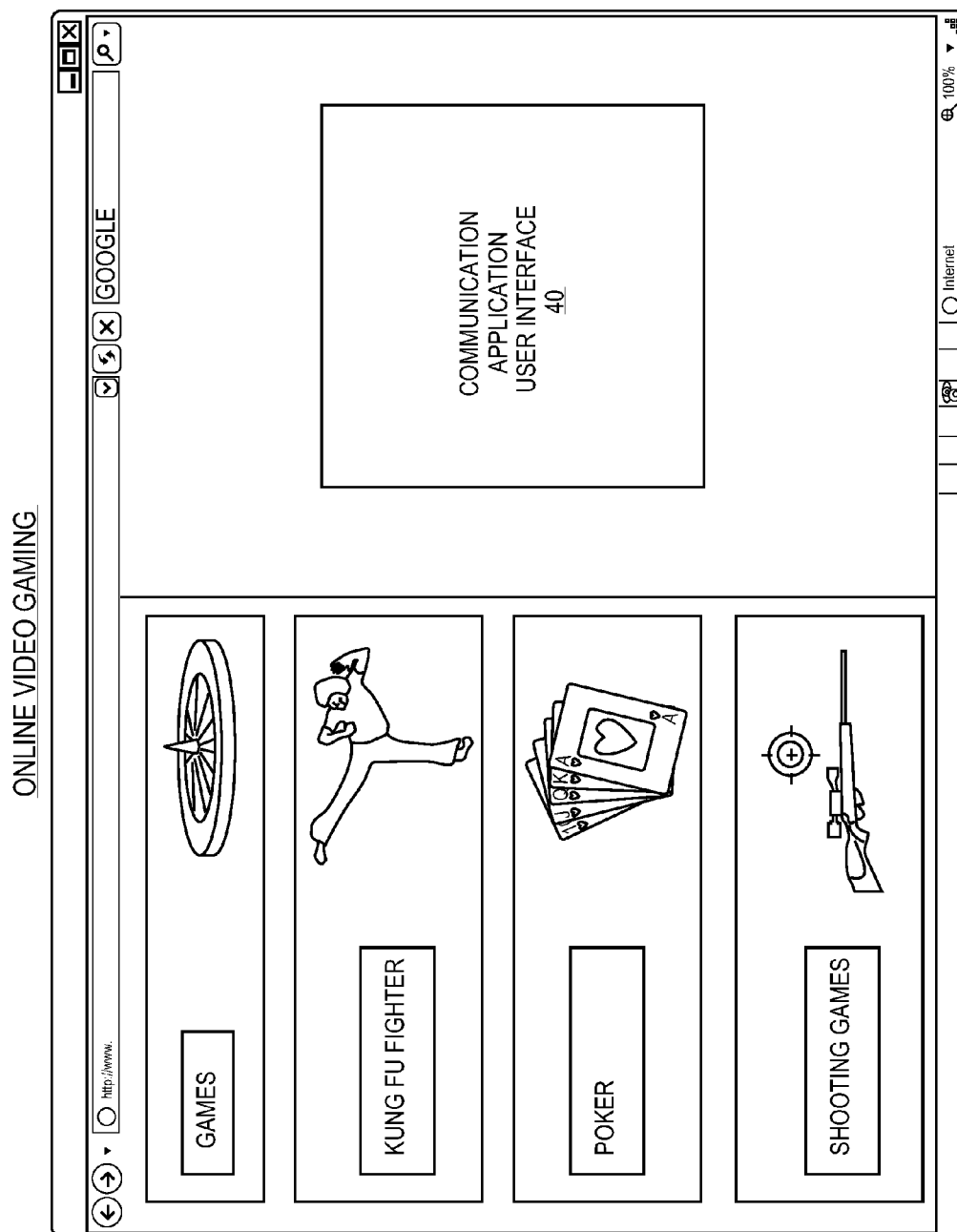


FIG. 5B

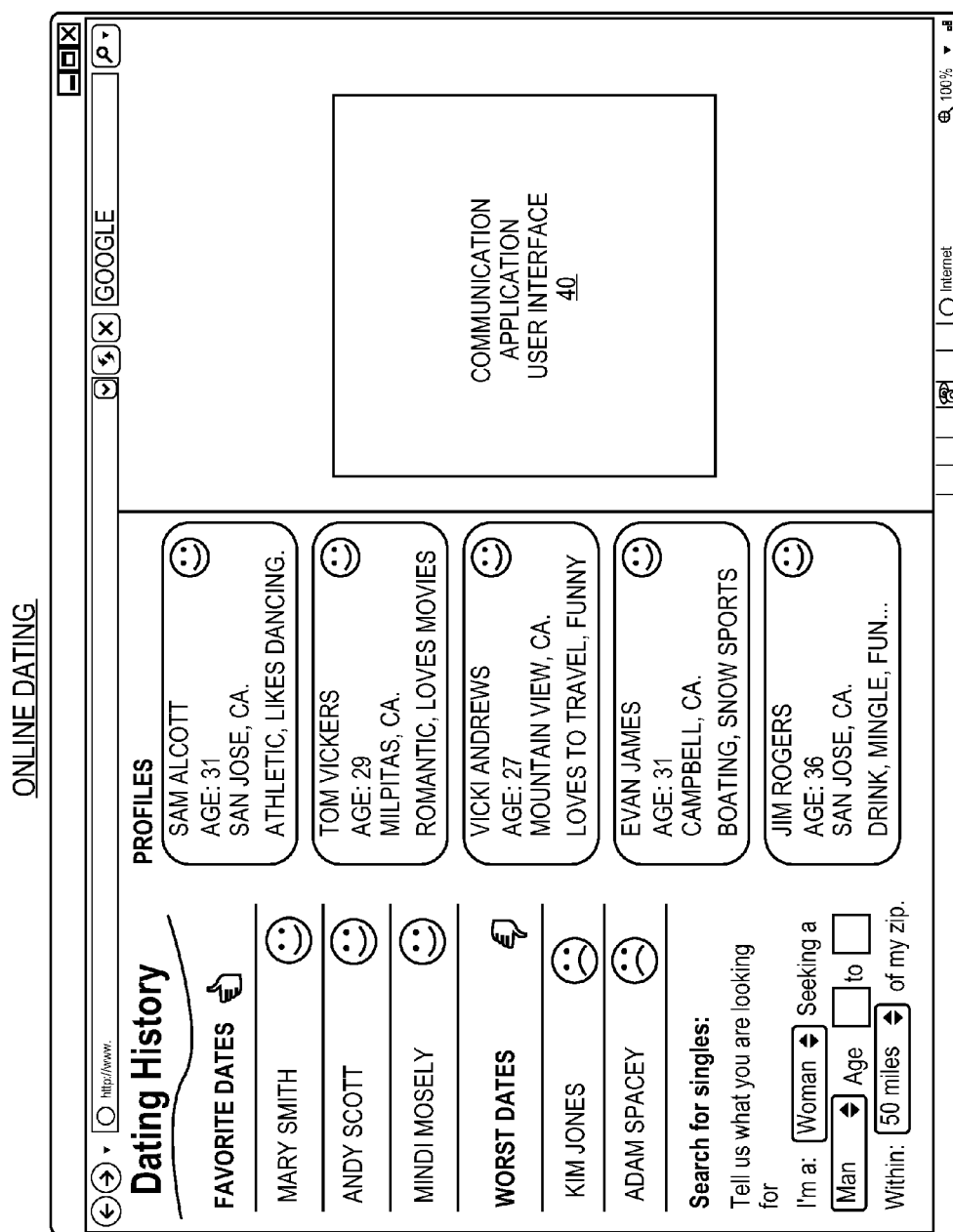
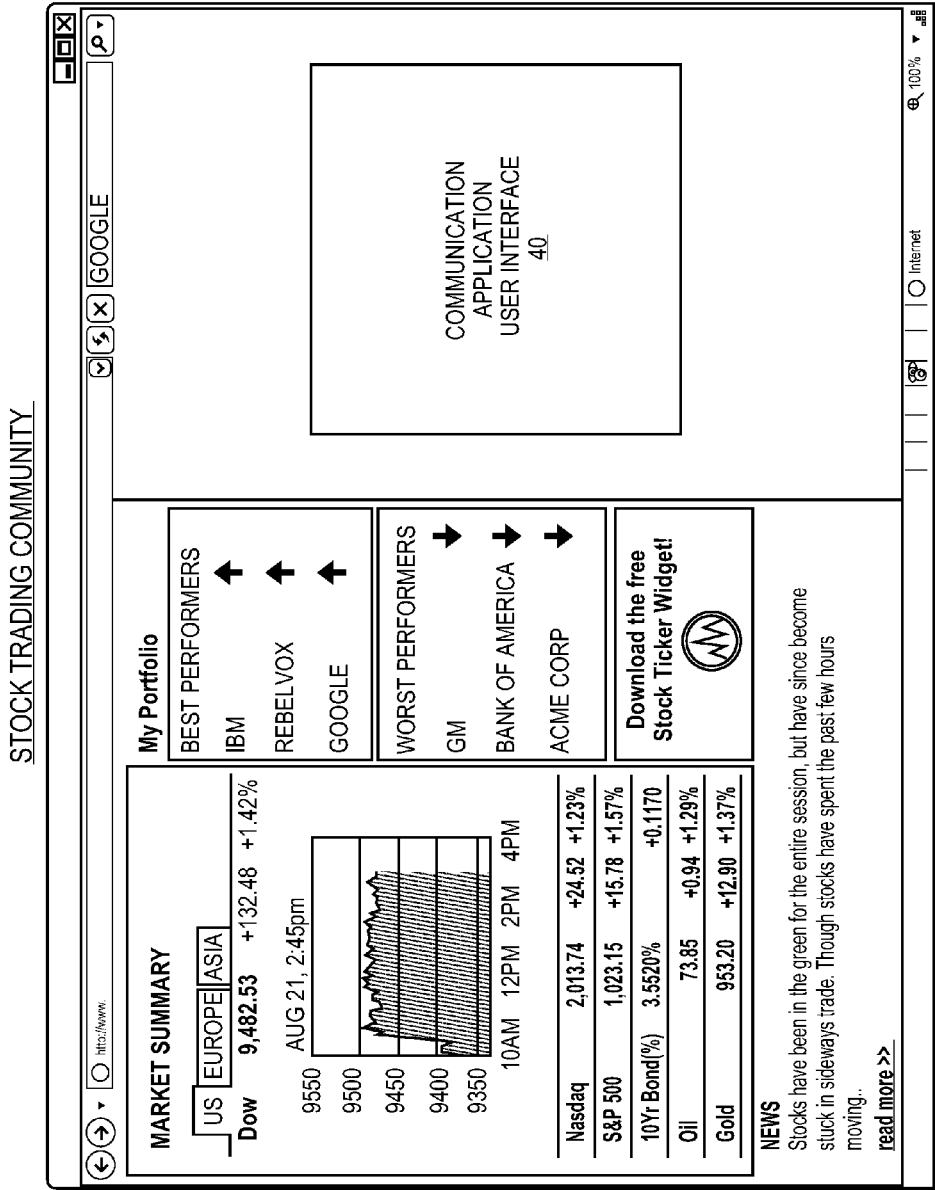


FIG. 5C



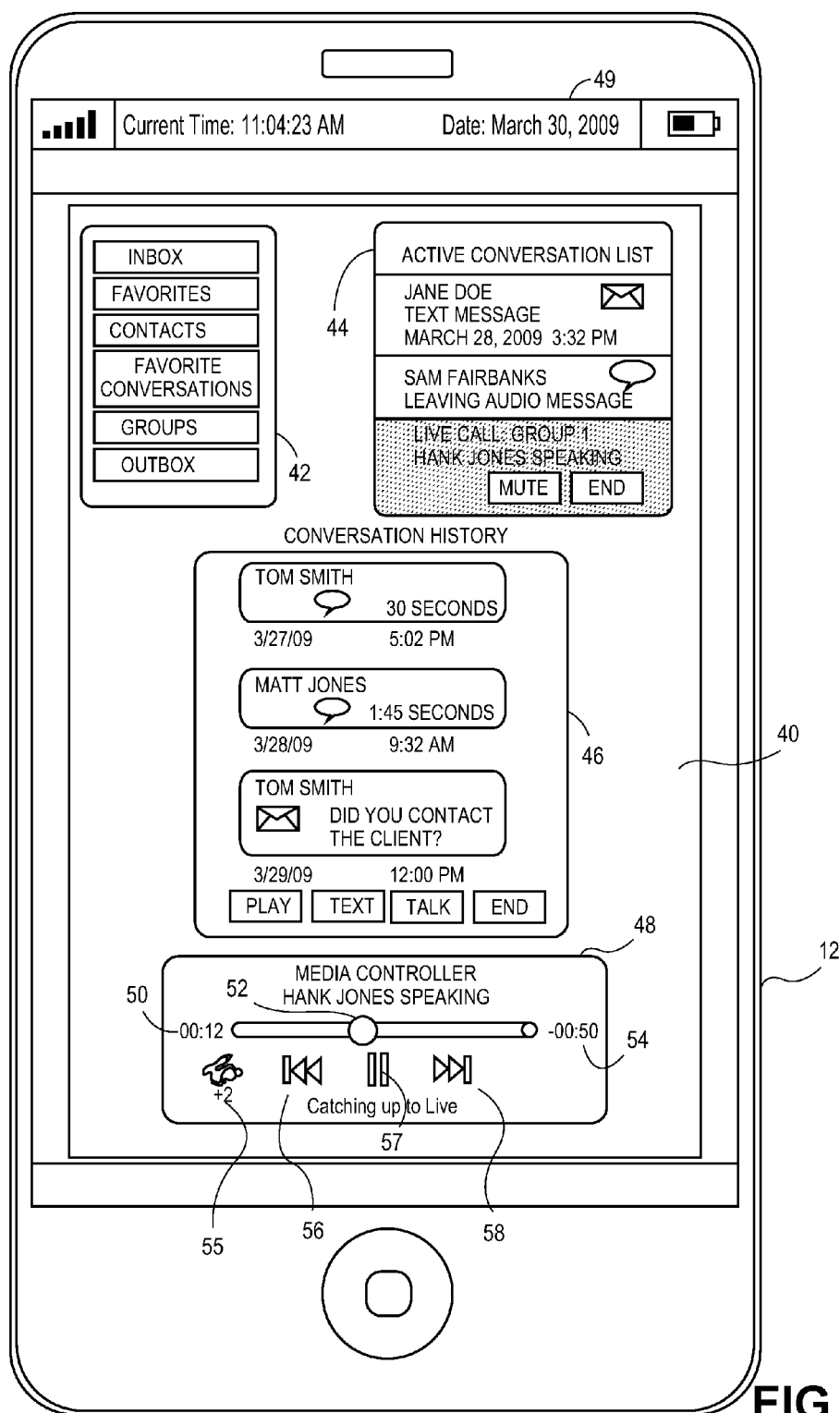


FIG. 6A

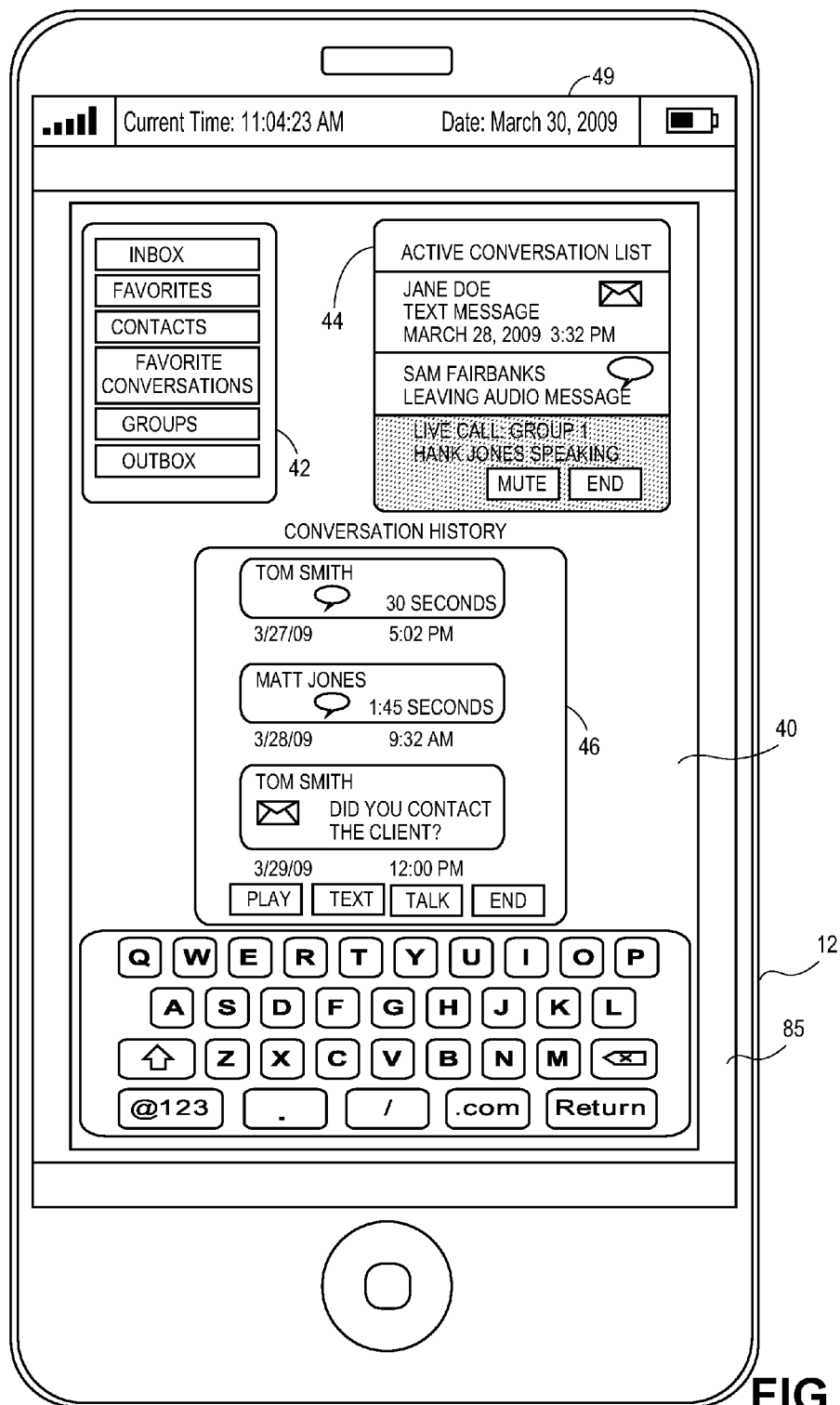


FIG. 6B

METHOD FOR DOWNLOADING AND USING A COMMUNICATION APPLICATION THROUGH A WEB BROWSER

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application is a Divisional of U.S. application Ser. No. 12/883,116, filed on Sep. 15, 2010. U.S. application Ser. No. 12/883,116 is a Continuation-in Part of U.S. application Ser. No. 12/028,400 (now U.S. Pat. No. 8,180,029), filed Feb. 8, 2008, which claims the benefit of priority to U.S. Provisional Application Nos. 60/937,552, filed Jun. 28, 2007, and 60/999,619, filed Oct. 19, 2007. U.S. application Ser. No. 12/883,116 is also a Continuation-in Part of U.S. application Ser. No. 12/561,089 (now U.S. Pat. No. 8,533,611), filed Sep. 16, 2009, which claims the benefit of priority to U.S. Provisional Patent Application No. 61/232,627, filed Aug. 10, 2009. U.S. application Ser. No. 12/883,116 is further a Continuation-in Part of U.S. application Ser. No. 12/419,861, filed Apr. 17, 2009, Ser. No. 12/552,980 (now U.S. Pat. No. 8,645,477), filed Sep. 2, 2009, and Ser. No. 12/857,486 (now U.S. Pat. No. 9,178,916), filed Aug. 16, 2010, each of which claim priority to U.S. Provisional Application No. 61/148,885, filed Jan. 30, 2009. The above-listed provisional and non-provisional applications are each incorporated herein by reference for all purposes.

BACKGROUND

[0002] Field of the Invention

[0003] This invention pertains to communications, and more particularly, to downloading and using a communication application through a web browser, the communication application enabling users to conduct voice conversations in either a synchronous real-time mode, asynchronously in a time-shifted mode, and with the ability to seamlessly transition between the two modes.

[0004] Description of Related Art

[0005] Electronic voice communication has historically relied on telephones and radios. Conventional telephone calls required one party to dial another party using a telephone number and waiting for a circuit connection to be made over the Public Switched Telephone Network or PSTN. A full-duplex conversation may take place only after the connection is made. More recently, telephony using Voice over Internet Protocol (VoIP) has become popular. With VoIP, voice communication occurs using IP over a packet-based network, such as the Internet.

[0006] Many full-duplex telephony systems have some sort of message recording facility for unanswered calls such as voicemail. If an incoming call goes unanswered, it is redirected to a voicemail system. When the caller finishes the message, the recipient is alerted and may listen to the message. Various options exist for message delivery beyond dialing into the voicemail system, such as email or “visual voicemail”, but these delivery schemes all require the entire message to be left by the caller before the recipient can listen to the message.

[0007] Many home telephones have answering machine systems that record missed calls. They differ from voicemail in that the caller’s voice is often played through a speaker on the answering machine while the message is being recorded. The called party can pick up the phone while the caller is leaving a message, which causes most answering machines

to stop recording the message. With other answering machines, however, the live conversation will be recorded unless the called party manually stops the recording. In either situation, there is no way for the called party to review the recorded message until after the recording has stopped. As a result, there is no way for the recipient to review any portion of the recorded message other than the current point while the message is ongoing and is being recorded. Only after the message has concluded can the recipient go back and review the recorded message.

[0008] Some more recent call management systems provide a “virtual answering machine”, allowing callers to leave a message in a voicemail system, while giving called users the ability to hear the message as it is being left. The actual answering “machine” is typically a voicemail-style server, operated by the telephony service provider. Virtual answering machine systems differ from standard voice mail systems in that the called party may use either their phone or a computer to listen to messages as they are being left. Similar to an answering machine as described in the preceding paragraph, however, the called party can only listen at the current point of the message as it is being left. There is no way to review previous portions of the message before the message is left in its entirety.

[0009] Certain mobile phone handsets have been equipped with an “answering machine” feature inside the handset itself that behaves similarly to a landline answering machine as described above. With these answering machines, callers may leave a voice message, which is recorded directly on the phone of the recipient. While the answering machine functionality has been integrated into the phone, the limitations of these answering machines, as discussed above, are still present.

[0010] With most current PTT systems, incoming audio is played on the device as it is received. If the user does not hear the message, for whatever reason, the message is irretrievably lost. Either the sender must resend the message or the recipient must request the sender to retransmit the message. PTT messaging systems are known. With these systems, message that are not reviewed live are recorded. The recipient can access the message from storage at a later time. These systems, however, typically do not record messages that are reviewed live by the recipient. See for example U.S. Pat. No. 7,403,775, U.S. Publications 2005/0221819 and 2005/0202807, EP 1 694 044 and WO 2005/101697.

[0011] With the growing popularity of the world wide web, more people are communicating through the Internet. With most of these applications, the user is interfacing through a browser running on their computer or other communication device, such as a mobile or cellular phone or radio, communicating with others through the Internet and one or more communication servers.

[0012] With email for example, users may type and send text messages to one another through email clients, located either locally on their computer or mobile communication device (e.g., Microsoft Outlook) or remotely on a server (e.g., Yahoo or Google Web-based mail). In the remote case, the email client “runs” on the computer or mobile communication device through a web browser. Although it is possible to send time-based (i.e., media that changes over time, such as voice or video) as an attachment to an email, the time-based media can never be sent or reviewed in a “live” or real-time mode. Due to the store and forward

nature of email, the time-based media must first be created, encapsulated into a file, and then attached to the email before it can be sent. On the receiving side, the email and the attachment must be received in full before it can be reviewed. Real-time communication is therefore not possible with conventional email.

[0013] Skype is a software application intended to run on computers that allows people to conduct voice conversations and video-conferencing communication. Skype is a type of VoIP system, and it is possible with Skype to leave a voice mail message. Also with certain ancillary products, such as Hot Recorder, it is possible for a user to record a conversation conducted using Skype. However with either Skype voice mail or Hot Recorder, it is not possible for a user to review the previous media of the conversation while the conversation is ongoing or to seamlessly transition the conversation between a real-time and a time-shifted mode.

[0014] Social networking Web sites, such as Facebook, also allow members to communicate with one another, typically through text-based instant messaging, but video messaging is also supported. In addition, mobile phone applications for Facebook are available to Facebook users. Neither the instant messaging, nor the mobile phone applications, however, allow users to conduct voice and other time-based media conversations in both a real-time and a time-shifted mode and to seamlessly transition the conversation between the two modes.

SUMMARY OF THE INVENTION

[0015] The invention involves a method for downloading a communication application onto a communication device. Once downloaded, the communication application is configured to create a user interface appearing within one or more web pages generated by a web browser running on the communication device. The communication enables the user to engage in voice conversations in (i) a real-time mode or (ii) a time-shifted mode and provides the ability to seamless transition the conversation back and forth between the two modes (i) and (ii). In the real-time mode, the communication application is configured to transmit voice media as the user speaks and render voice media as it is transmitted and received from a sender. The communication application also provides for the persistent storage of transmitted and received voice media. With persistent storage, the voice media may be rendered at a later arbitrary time defined by the user in the time-shifted mode.

[0016] The communication application is preferably downloaded along with web content. Accordingly, when the user interface appears within the web browser, it is typically within the context of a web site, such as an on-line social networking, gaming, dating, financial or stock trading, or any other on-line community. The user of the communication device can then conduct conversations with other members of the web community through the user interface within the web site appearing within the browser.

[0017] In another embodiment, both the communication device and communication servers responsible for routing the voice media of the conversation between participants are “late-binding”. With late-binding, voice media is progressively transmitted as it is created and as soon as a recipient is identified, without having to first wait for a complete discovery path to the recipient to be discovered. Similarly, the communication servers can progressively transmit received voice media as it is available, before the voice

media is received in full, as soon as the next hop is discovered, and before the complete delivery route to the recipient is fully known. Late binding thus solves the problems with current communication systems, including the (i) waiting for a circuit connection to be established before “live” communication may take place, with either the recipient or a voice mail system associated with the recipient, as required with conventional telephony or (ii) waiting for an email to be composed in its entirety before the email may be sent.

[0018] In yet another embodiment, a number of addressing techniques may be used, including unique identifiers that identify a user within a web community, or globally unique identifiers, such as telephone numbers or email addresses. The unique identifier, regardless if global or not, may be used for both authentication and routing. Anyone of a number of real-time transmission protocols, such as SIP, RTP, VoIP, Skype, UDP, TCP or CTP, may be used for the actual transmission of the voice media.

[0019] In yet another embodiment, email addresses, the existing email infrastructure and DNS may be used for addressing and route discovery. In addition with this embodiment, existing email protocols may be modified so that voice media of conversations may be transmitted as it is created and rendered as it is received. This embodiment, sometimes referred to as “progressive emails”, differs significantly from conventional emails, which are store and forward only and are unable to support the transmission of “live” voice media in real-time.

BRIEF DESCRIPTION OF THE DRAWINGS

[0020] The invention may best be understood by reference to the following description taken in conjunction with the accompanying drawings, which illustrate specific embodiments of the invention.

[0021] FIG. 1 is diagram of a non-exclusive embodiment of a communication system embodying the principles of the present invention.

[0022] FIG. 2 is a diagram of a non-exclusive embodiment of a communication application embodying the principles of the present invention.

[0023] FIG. 3A is a block diagram of an exemplary communication device.

[0024] FIG. 3B is a block diagram illustrating the communication application of FIG. 2 running on a client communication device.

[0025] FIG. 3C is a diagram illustrating a non-exclusive embodiment of a sequence for implementing the principles of the present invention.

[0026] FIG. 4 is a diagram of an exemplary graphical user interface for managing and engaging in conversations on a client communication device according to the principles of the present invention.

[0027] FIGS. 5A through 5D are diagrams illustrating a non-exclusive examples of web browsers incorporating a user interface of the communication application within the context of various web pages according to the principles of the present invention.

[0028] FIGS. 6A and 6B are diagrams of an exemplary user interface displayed on a mobile client communication device within the context of web pages according to the principles of the present invention.

[0029] It should be noted that like reference numbers refer to like elements in the figures.

[0030] The above-listed figures are illustrative and are provided as merely examples of embodiments for implementing the various principles and features of the present invention. It should be understood that the features and principles of the present invention may be implemented in a variety of other embodiments and the specific embodiments as illustrated in the Figures should in no way be construed as limiting the scope of the invention.

DETAILED DESCRIPTION OF SPECIFIC EMBODIMENTS

[0031] The invention will now be described in detail with reference to various embodiments thereof as illustrated in the accompanying drawings. In the following description, specific details are set forth in order to provide a thorough understanding of the invention. It will be apparent, however, to one skilled in the art, that the invention may be practiced without using some of the implementation details set forth herein. It should also be understood that well known operations have not been described in detail in order to not unnecessarily obscure the invention.

Messages and Conversations

[0032] “Media” as used herein is intended to broadly mean virtually any type of media, such as but not limited to, voice, video, text, still pictures, sensor data, GPS data, or just about any other type of media, data or information. Time-based media is intended to mean any type of media that changes over time, such as voice or video. By way of comparison, media such as text or a photo, is not time-based since this type of media does not change over time.

[0033] As used herein, the term “conversation” is also broadly construed. In one embodiment, a conversation is intended to mean a thread of messages, strung together by some common attribute, such as a subject matter or topic, by name, by participants, by a user group, or some other defined criteria. In another embodiment, the messages of a conversation do not necessarily have to be tied together by some common attribute. Rather one or more messages may be arbitrarily assembled into a conversation. Thus a conversation is intended to mean two or more messages, regardless if they are tied together by a common attribute or not.

The Communication System

[0034] Referring to FIG. 1, an exemplary communication system including one or more communication servers **10** and a plurality of client communication devices **12** is shown. A communication services network **14** is used to interconnect the individual client communication devices **12** through the servers **10**.

[0035] The server(s) **10** run an application responsible for routing the metadata used to set up and support conversations as well as the actual media of messages of the conversations between the different client communication devices **12**. In one specific embodiment, the application is the server application described in commonly assigned co-pending US application Ser. No. 12/028,400 (U.S. Patent Publication No. 2009/0003558), Ser. No. 12/192,890 (U.S. Patent Publication No. 2009/0103521), and Ser. No. 12/253,833 (U.S. Patent Publication No. 2009/0168760), each incorporated by reference herein for all purposes.

[0036] One or more of the server(s) **10** may also be configured as a web server. Alternatively, one or more

separate web servers may be provided or accessible over the network **14**. The web servers are responsible for serving web content to the client communication devices **12**.

[0037] The client communication devices **12** may be a wide variety of different types of communication devices, such as desktop computers, mobile or laptop computers, tablet-PCs, notebooks, e-readers, WiFi devices such as the iPod by Apple, mobile or cellular phones, Push To Talk (PTT) devices, PTT over Cellular (PoC) devices, radios, satellite phones or radios, VoIP phones, or conventional telephones designed for use over the Public Switched Telephone Network (PSTN). The above list should be construed as exemplary and should not be considered as exhaustive or limiting. Any type of communication device may be used.

[0038] The network **14** may in various embodiments be the Internet, PSTN, a circuit-based network, a mobile communication network, a cellular network based on CDMA or GSM for example, a wired network, a wireless network, a tactical radio network, a satellite communication network, any other type of communication network, or any combination thereof. The network **14** may also be either heterogeneous or homogeneous network.

The Communication Application

[0039] The server(s) **10** are also responsible for downloading a communication application to the client communication devices **12**. The downloaded communication application is very similar to the above-mentioned application running on the servers **10**, but differs in several regards. First, the downloaded communication application is written in a programming language so that it will run within the context of the web page appearing within the browser of the communication device. Second, the communication application is configured to create a user interface that appears within the web page appearing within by a web browser running on the client communication device **12**. Third, the downloaded communication application is configured to cooperate with a multi-media platform, such as Flash by Adobe Systems, to support various input and output functions on the client communication device **12**, such as a microphone, speaker, display, touch-screen display, camera, video camera, keyboard, etc. Accordingly when the application is downloaded, the user has the experience that the user interface is an integral part of a web page running within a browser on the client communication device **12**.

[0040] Referring to FIG. 2, a block diagram of a communication application **20** is illustrated. The communication application **20** includes a Multiple Conversation Management System (MCMS) module **22**, a Store and Stream module **24**, and an interface **26** provided between the two modules. The key features and elements of the communication application **20** are briefly described below. For a more detailed explanation, see U.S. application Ser. Nos. 12/028,400, 12/253,833, 12/192,890, and 12/253,820 (U.S. Patent Publication Nos. 2009/0003558, 2009/0168760, 2009/0103521, and 2009/0168759), all incorporated by reference herein.

[0041] The MCMS module **22** includes a number of modules and services for creating, managing, and conducting multiple conversations. The MCMS module **22** includes a user interface module **22A** for supporting the audio and video functions on the client communication device **12**, rendering/encoding module **22B** for performing rendering and encoding tasks, a contacts service module **22C** for

managing and maintaining information needed for creating and maintaining contact lists (e.g., telephone numbers, email addresses or other unique identifiers), a presence status service module 22D for sharing the online status of the user of the client communication device 12 and which indicates the online status of the other users and the MCMS data base 22E, which stores and manages the metadata for conversations conducted using the client communication device 12.

[0042] The Store and Stream module 24 includes a Persistent Infinite Memory Buffer or PIMB 28 for storing in a time-indexed format the time-based media of received and sent messages. The store and stream module 24 also includes four modules for encode receive 26A, transmit 26C, net receive 26B and render 26D. The function of each module is described below.

[0043] The encode receive module 26A performs the function of progressively encoding and persistently storing in the PIMB 28 in a time-indexed format the media created using the client communication device 12 as the media is created.

[0044] The transmit module 26C progressively transmits the media created using the client communication device 12 to other recipients over the network 14 as the media is created and progressively stored in the PIMB 28.

[0045] The encode receive module 26A and the transmit module 26C perform their respective functions at approximately the same time. For example, as a person speaks into their client communication device 12 during a conversation, the voice media is simultaneously and progressively encoded, persistently stored and transmitted as the voice media is created.

[0046] The net receive module 26B is responsible for progressively storing media received from others in the PIMB 28 in a time-indexed format as the media is received.

[0047] The render module 24D enables the rendering of persistently stored media either synchronously in the near real-time mode or asynchronously in the time-shifted mode by retrieving media stored in the PIMB 28. In the real-time mode, the render module 24D renders media simultaneously as it received and persistently stored by the net received module 26B. In the time-shifted mode, the render module 24D renders media previously stored in the PIMB at an arbitrary time after the media was stored. The rendered media could be either received media, transmitted media, or both received and transmitted media. Synchronous and asynchronous communication should be broadly construed herein and generally mean the sender and receiver are concurrently or not concurrently engaged in communication respectively.

[0048] The version of the application running on the server(s) 10 will typically not include the encode receive module 24A and render module 24D since encoding and rendering functions are typically not performed on the server(s) 10.

[0049] The PIMB 28 located on the communication application 20 may not be physically large enough to indefinitely store all of the media transmitted and received by a user. The PIMB 28 is therefore configured like a cache, and stores only the most relevant media, while the PIMB located on a server 10 acts as backup or main storage. As physical space in the memory used for the PIMB 28 runs out, certain media on the client 12 may be replaced using any well-known algorithm, such as least recently used or first-in, first-out. In the event the user wishes to review replaced media, then the

media is retrieved from the server 10 and locally stored in the PIMB 28. Thereafter, the media may be rendered out of the PIMB 28. The retrieval time is ideally minimal so as to be transparent to the user.

Client Communication Devices

[0050] Referring to FIG. 3A, a block diagram of a client communication device 12 according to a non-exclusive embodiment of the invention is shown. The client communication device 12 includes a network connection 30 for connecting the client communication device 12 to the network 14, a number of input/output devices 31 including a speaker 31A for rendering voice and other audio based media, a mouse 31B for cursor control and data entry, a microphone 31C for voice and other audio based media entry, a keyboard or keypad 31D for text and data entry, a display 31E for rendering image or video based media, and a camera 31F for capturing either still photos or video. It should be noted that elements 31A through 31F are each optional and are not necessarily included on all implementations of a client communication device 12. In addition, the display 31E may be a touch-sensitive display capable of receiving inputs using a pointing element, such as a pen, stylus or finger. In yet other embodiments, client communication devices 12 may optionally further include other media generating devices (not illustrated), such as sensor data (e.g., temperature, pressure), GPS data, etc.

[0051] The client communication device 12 also includes a web browser 32 configured to generate and display HTML/Web content 33 on the display 31E. An optional multi-media platform 34, such as the Adobe Flash player, provides audio, video, animation, and other interactivity features within the Web browser 33. In various embodiments, the multi-media platform 34 may be a plug-in application or may already reside on the device 12.

[0052] The web browser 32 may be any well-known software application for retrieving, presenting, and traversing information resources on the Web. In various embodiments, well known browsers such as Internet Explorer by Microsoft, Firefox by the Mozilla Foundation, Safari by Apple, Chrome by Google, Opera by Opera Software for desktop, mobile, embedded or gaming systems, or any other browser may be used. Although the browser 32 is primarily intended to access the world-wide-web, in alternative embodiments, the browser 32 can also be used to access information provided by servers in private networks or content in file systems.

[0053] The input/output devices 31A through 31F, the browser 32 and multi-media platform 34 are all intended to run on an underlying hardware platform 35. In various embodiments, the hardware platform may be any microprocessor or microcontroller platform, such as but not limited to those offered by Intel Corporation or ARM Holdings, Cambridge, United Kingdom, or equivalents thereof.

[0054] Referring to FIG. 3B, the same client communication device 12 after the communication application 20 has been downloaded is illustrated. After the download, the client communication device 12 includes a web browser plug-in application 36 with a browser interface layer 37. The multi-media platform 34 communicates with an underlying communication application 20 using remote Application Programming Interfaces or APIs, as is well known in the art. The web browser plug-in application 36 takes advantage of the multi-media platform 34 and the functionality and ser-

vices offered by the browser 32. The browser interface layer 37 acts as an interface between the web browser 32 and the communication application 20. The browser interface layer 37 is responsible for (i) invoking the various user interface functions implemented by the communication application 20 and presenting the appropriate user interface within the content presented through browser 32 to the user of client communication device 12 and (ii) receiving inputs from the user through the browser 32 and other inputs on the client communication device 12, such as microphone 31C, mouse 31B, keyboard 31D, or touch display 31E and providing these inputs to the communication application 20. As a result, the user of the client communication device 12 may control the operation of the communication application 20 when setting up, participating in, or terminating conversations through the web browser 32 and the other input/output devices optionally provided on the client communication device 12.

[0055] It should be noted that the emerging next generation HTTP5 standard, as currently proposed, supports some of the multimedia functions performed by the multi-media platform 34, web-browser plug-in 36, and/or browser interface layer 37. To the extent the functionality performed by 34, 36 and 37 is supported by the native HTTP in the future, it may be possible to eliminate the need of some or all of these elements on the client communication devices 12 respectively. Consequently, FIG. 3B should not be construed as limiting in any regard. Rather it should be anticipated that the elements 34, 36 and 37 be fully or partially removed from the device 12 as their functionality is replaced by native HTTP in the future.

[0056] Referring to FIG. 3C, a diagram 100 illustrating a non-exclusive embodiment of a sequence for implementing the principles of the present invention is shown. In the initial step 102, a web server is maintained on a network. As noted above, one or more of the servers 10 may be configured as a web server or one or more separate web servers on may be accessed. In the next step 104, a user of a communication device 12 accesses one of the web servers over the network 14 and requests, as needed, the multi-media platform 34, the communication application 20, the browser plug-in application 36, and browser interface layer 37. In reply, these software plug-in modules are downloaded, as needed, in step 106 to the client device 12 of the user. In step 108, web content is served to the client communication device 12. The downloaded communication application 20 and multi-media platform 34 cooperate along with the served content to create a user interface within the web pages appearing within the browser 32. In step 112, the user participates in one or more conversations through the user interface. The server(s) 10 route the transmitted and received media among the participants of the conversation in step 114.

[0057] The communication application 20 enables the user of the client communication device 12 to set up and engage in conversations with other client communication devices 12 (i) synchronously in the real-time mode, (ii) asynchronously in the time-shifted mode and to (iii) seamlessly transition the conversation between the two modes (i) and (ii). The conversations may also include multiple types of media besides voice, including text, video, sensor data, etc. The user participates in the conversations through the user interface appearing within the browser 32, the details of which are described in more detail below.

The User Interface

[0058] FIG. 4 is a diagram of an exemplary user interface 40, rendered by the browser 32 on the display 31E of a client communication device 12. The interface 40 enables or facilitates the participation of the user in one or more conversations on the client device 12 using the communication application 20.

[0059] The interface 40 includes a folders window 42, an active conversation list window 44, a window 46 for displaying the history of a conversation selected from the list displayed in window 44, a media controller window 48, and a window 49 displaying the current time and date. Although not illustrated, the interface also includes one or more icons for creating a new conversations and defining the participant(s) of the new conversation.

[0060] The folders window 42 includes a plurality of optional folders, such as an inbox for storing incoming messages, a contact list, a favorites contact list, a conversation list, conversation groups, and an outbox listing outgoing messages. It should be understood that the list provided above is merely exemplary. Individual folders containing a wide variety of lists and other information may be contained within the folders window 42.

[0061] Window 44 displays the active conversations the user of client communication device 12 is currently engaged in. In the example illustrated, the user is currently engaged in three conversations. In the first conversation, a participant named Jane Doe previously left a text message, as designated by the envelope icon, at 3:32 PM on Mar. 28, 2009. In another conversation, a participant named Sam Fairbanks is currently leaving an audio message, as indicated by the voice media bubble icon. The third conversation is entitled "Group 1." In this conversation, the conversation is "live" and a participant named Hank Jones is speaking. The user of the client communication device 12 may select any of the active conversations appearing in the window 44 for participation.

[0062] Further in this example, the user of client communication device 12 has selected the Group 1 conversation for participation. As a result, a visual indicator, such as the shading of the Group 1 conversation in the window 44 different from the other listed conversations, informs the user that he or she is actively engaged in the Group 1 conversation. Had the conversation with Sam Fairbanks been selected, then this conversation would have been highlighted in the window 44. It should be noted that the shading of the selected conversation in the window 44 is just one possible indicator. In various other embodiments, any indicator, either visual, audio, a combination thereof, or no indication may be used.

[0063] Within the selected conversation, a "MUTE" icon and an "END" icon are optionally provided. The mute icon allows the user to disable the microphone 24 of client communication device 12. When the end icon is selected, the user's active participation in the Group 1 conversation is terminated. At this point, any other conversation in the list provided in window 44 may be selected. In this manner, the user may transition from conversation to conversation within the active conversation list. The user may return to the Group 1 conversation at anytime.

[0064] The conversation window 46 shows the history of the currently selected conversation, which in this example again, is the Group 1 conversation. In this example, a sequence of media bubbles each represent the media con-

tributions to the conversation respectively. Each media bubble represents the media contribution of a participant to the conversation in time-sequence order. In this example, Tom Smith left an audio message that is 30 seconds long at 5:02 PM on Mar. 27, 2009. Matt Jones left an audio message 1 minute and 45 seconds in duration at 9:32 AM on Mar. 28, 2009. Tom Smith left a text message, which appears in the media bubble, at 12:00 PM on Mar. 29, 2009. By scrolling up or down through the media bubbles appearing in window 46, the entire history of the Group 1 conversation may be viewed.

[0065] The window 46 further includes a number of icons allowing the user to control his or her participation in the selected Group 1 conversation. A “PLAY” icon allows the user to render the media of a selected media bubble appearing in the window 46. For example, if the Tom Smith media bubble is selected, then the corresponding voice message is accessed and rendered through the speaker 31A on the client communication device 12. With media bubbles containing a text message, the text is typically displayed within the bubble. In either case, when an old message bubble is selected, the media of the conversation is being reviewed in the time-shifted mode.

[0066] The “TEXT” and the “TALK” icons enable the user of the client communication device 12 to participate in the conversation by either typing or speaking a message respectively. The “END” icon removes the user from participation in the conversation.

[0067] When another conversation is selected from the active list appearing in window 44, the history of the newly selected conversation appears in the conversation history window 46. Thus by selecting different conversations from the list in window 44, the user may switch participation among multiple conversations.

[0068] The media controller window 48 enables the user of the client communication device 12 to control the rendering of voice and other media of the selected conversation. The media controller window operates in two modes, the synchronous real-time mode and the asynchronous time shifted mode, and enables the seamless transition between the two modes.

[0069] In the time-shifted mode, the media of a selected message is identified within the window 48. For example (not illustrated), if the previous voice message from Tom Smith sent at 5:02 PM on Mar. 27, 2009, is selected, information identifying this message is displayed in the window 48. The scrubber bar 52 allows the user to quickly traverse a message from start to finish and select a point to start the rendering of the media of the message. As the position of the scrubber bar 52 is adjusted, the timer 54 is updated to reflect the time-position relative to the start time of the message.

[0070] The pause icon 57 allows the user to pause the rendering of the media of the message. The jump backward icon 56 allows the user to jump back to a previous point in time of the message and begin the rendering of the message from that point forward. The jump forward icon 58 enables the user to skip over media to a selected point in time of the message.

[0071] The rabbit icon 55 controls the rate at which the media of the message is rendered. The rendering rate can be either faster, slower, or at the same rendering rate the media of the message was originally encoded.

[0072] In the real-time mode, the participant creating the current message is identified in the window 48. In the example illustrated, the window identifies Hank Jones as speaking. As the message continues, the timer 50 is updated, providing a running time duration of the message. The jump backward and pause icons 56 and 57 operate as mentioned above. By jumping from the head of the conversation in the real-time mode back to a previous point using icon 56, the conversation may be seamlessly transitioned from the live or real-time mode to the time-shifted mode. The jump forward icon 58 is inoperative when at the head of the message since there is no media to skip over when at the head.

[0073] The rabbit icon 55 may also be used to implement a rendering feature referred to as Catch up To Live or “CTL”. This feature allows a recipient to increase the rendering rate of the previously received and persistently stored media of an incoming message until the recipient catches up to the media as it is received. For example, if the user of the client device joins an ongoing conversation, the CTL feature may be used to quickly review the previous media contributions of the unheard message or messages until catching up to the head of the conversation. At this point, the rendering of the media seamlessly merges from the time-shifted mode to the real-time mode.

[0074] By using the render control options, the user may seamlessly transfer a conversation from the time-shifted mode to the real-time mode and vice versa. For example, the user may use the pause or jump backward render options to seamlessly shift a conversation from the real-time to time-shifted modes or the play, jump forward, or CTL options to seamlessly transition from the time-shifted to real-time modes.

[0075] It should be noted that the user interface 40 is merely exemplary. It is just one of many possible implementations for providing a user interface for client communication devices 12. It should be understood that the features and functionality as described herein may be implemented in a wide variety of different ways. Thus the specific interface illustrated herein should not be construed as limiting in any regard.

Web Communities

[0076] With the Internet and world-wide-web becoming pervasive, web sites that create or define communities are become exceedingly popular. For example, Internet users with a common interest tend to aggregate at select web sites where they can converse and interact with others. Social networking sites like Facebook.com, online dating sites like match.com, video game sites like addictivedgames.com, and other forums, such as stock trading, hobbies, etc., have all become very popular. Up to now, members of these various web sites could communicate with each other by either email or instant messaging style interactions. Some sites support the creation of voice and video messaging, and other sites support live voice and video communication. None, however, allow members to participate in conversations either synchronously in the real-time mode or asynchronously in the time-shifted mode or provide the ability to seamlessly transition communication between the two modes.

[0077] By embedding the user interface 40 in one or more web pages of a web site, the members of a web community may participate in conversations with one another. In FIGS. 5A through 5D for example, the user interface 40 is shown

embedded in a social networking site, an online video gaming site, an online dating site, a stock trading forum respectively. When users of client communication devices **12** access these or similar web sites, they may conduct conversations with other members, in either the real-time mode, the time-shifted mode, and have the ability to seamlessly shift between the modes, as described in detail herein.

[0078] Referring to FIG. 6A, a diagram of a browser-enabled display on a mobile client communication device **12** according to the present invention is shown. In this example, the user interface **40** is provided within the browser-enabled display of a mobile client communication device **12**, such as a mobile phone or radio. FIG. 6B is a diagram of the mobile client communication device **12** with a keyboard **85** superimposed onto the browser display. With the keyboard **85**, the user may create text messages during participation in conversations.

[0079] Although a number of popular web-based communities have been mentioned herein, it should be understood that this list is not exhaustive. The number of web sites is virtually unlimited and there are far too many web sites to list herein. In each case, the members of the web community may communicate with one another through the user interface **40** or a similar interface as described herein.

Real-Time Communication Protocols

[0080] In various embodiments, the store and stream module **24** of the communication application **20** may rely on a number of real-time communication protocols.

[0081] In one optional embodiment, the store and stream module **24** may use the Cooperative Transmission Protocol (CTP) for near real-time communication, as described in U.S. application Ser. Nos. 12/192,890 and 12/192,899 (U.S. Patent Publication Nos. 2009/0103521 and 2009/0103560), all incorporated by reference herein for all purposes.

[0082] In another optional embodiment, a synchronization protocol may be used that maintains the synchronization of time-based media between a sending and receiving client communication devices **12**, as well as any intermediate server **10** hops on the network **14**. See for example U.S. application Ser. Nos. 12/253,833 and 12/253,837, both incorporated by reference herein for all purposes, for more details.

[0083] In various other embodiments, the communication application **20** may rely on other real-time transmission protocols, including for example SIP, RTP, Skype, UDP and TCP. For details on using both UDP and TCP, see U.S. application Ser. Nos. 12/792,680 and 12/792,668 both filed on Jun. 2, 2010 and both incorporated by reference herein.

Addressing

[0084] If the user of a client **12** wishes to communicate with a particular recipient, the user will either select the recipient from their list of contacts or reply to an already received message from the intended recipient. In either case, an identifier associated with the recipient is defined. Alternatively, the user may manually enter an identifier identifying a recipient. In some embodiments, a globally unique identifier, such as a telephone number, email address, may be used. In other embodiments, non-global identifiers may be used. Within an online web community for example, such as a social networking website, a unique identifier may be issued to each member within the community. This unique

identifier may be used for both authentication and the routing of media among members of the web community. Such identifiers are generally not global because they cannot be used to address the recipient outside of the web community. Accordingly the term “identifier” as used herein is intended to be broadly construed and mean both globally and non-globally unique identifiers.

Early and Late Binding

[0085] In early-binding embodiments, the recipient(s) of conversations and messages may be addressed using telephone numbers and Session Internet Protocol (SIP) for setting up and tearing down communication sessions between client communication devices **12** over the network **14**. In various other optional embodiments, the SIP protocol is used to create, modify and terminate either IP unicasts or multicast sessions. The modifications may include changing addresses or ports, inviting or deleting participants, or adding or deleting media streams. As the SIP protocol and telephony over the Internet and other packet-based networks, and the interface between the VoIP and conventional telephones using the PSTN are all well known, a detailed explanation is not provided herein. In yet another embodiment, SIP can be used to set up sessions between client communication devices **12** using the CTP protocol mentioned above.

[0086] In alternative late-binding embodiments, the communication application **20** may be progressively transmit voice and other time-based media as it is created and as soon as a recipient is identified, without having to first wait for a complete discovery path to the recipient to be fully discovered. The communication application **20** implements late binding by discovering the route for delivering the media associated with a message as soon as the unique identifier used to identify the recipient is defined. The route is typically discovered by a lookup result of the identifier as soon as it is defined. The result can be either an actual lookup or a cached result from a previous lookup. At substantially the same time, the user may begin creating time-based media, for example, by speaking into the microphone, generating video, or both. The time-based media is then simultaneously and progressively transmitted across one or more server **10** hop(s) over the network **14** to the addressed recipient, using any real-time transmission protocol. At each hop, the route to the next hop is immediately discovered either before or as the media arrives, allowing the media to be streamed to the next hop without delay and without the need to wait for a complete route to the recipient to be discovered.

[0087] For all practical purposes, the above-described late-binding steps occur at substantially the same time. A user may select a contact and then immediately begin speaking. As the media is created, the real-time protocol progressively and simultaneously transmits the media across the network **14** to the recipient, without any perceptible delay. Late binding thus solves the problems with current communication systems, including the (i) waiting for a circuit connection to be established before “live” communication may take place, with either the recipient or a voice mail system associated with the recipient, as required with conventional telephony or (ii) waiting for an email to be composed in its entirety before the email may be sent.

Progressive Emails

[0088] In one non-exclusive late-binding embodiment, the communication application **20** may rely on “progressive

emails” to support real-time communication. With this embodiment, a sender defines the email address of a recipient in the header of a message (i.e., either the “To”, “CC”, or “BCC” field). As soon as the email address is defined, it is provided to a server **10**, where a delivery route to the recipient is discovered from a DNS lookup result. Time-based media of the message may then be progressively transmitted, from hop to hop to the recipient, as the media is created and the delivery path is discovered. The time-based media of a “progressive email” can be delivered progressively, as it is being created, using standard SMTP or other proprietary or non-proprietary email protocols. Conventional email is typically delivered to user devices through an access protocol like POP or IMAP. These protocols do not support the progressive delivery of messages as they are arriving. However, by making simple modifications to these access protocols, the media of a progressive email may be progressively delivered to a recipient as the media of the message is arriving over the network. Such modifications include the removal of the current requirement that the email server know the full size of the email message before the message can be downloaded to the client communication device **12**. By removing this restriction, the time-based media of a “progressive email” may be rendered as the time-based media of the email message is received. For more details on the above-described embodiments including late-binding and using identifiers, email addresses, DNS, and the existing email infrastructure, see co-pending U.S. application Ser. Nos. 12/419,861, 12/552,979 and 12/857,486, each commonly assigned to the assignee of the present invention and each incorporated herein by reference for all purposes.

Full and Half Duplex Communication

[0089] The communication application **20**, regardless of the real-time protocol, addressing scheme, early or late binding, or if progressive emails are used, is capable of both transmitting and receiving voice and other media at the same time or at times within relative close proximity to one another. Consequently, the communication application is capable of supporting full-duplex communication, providing a user experience similar to a conventional telephone conversation. Alternatively, the communication application is also capable of sending and receiving messages at discrete times, similar to a messaging or half-duplex communication system.

[0090] While the invention has been particularly shown and described with reference to specific embodiments thereof, it will be understood by those skilled in the art that changes in the form and details of the disclosed embodiments may be made without departing from the spirit or scope of the invention. For example, embodiments of the invention may be employed with a variety of components and methods and should not be restricted to the ones mentioned above. It is therefore intended that the invention be interpreted to include all variations and equivalents that fall within the true spirit and scope of the invention.

What is claimed is:

1. A method, comprising:

providing at a first communication device access to a video messaging system through a web browser interfacing with a web site, a user of the first communication

device able to selectively render an incoming video message sent via the video messaging system through the web browser in both:

- (i) a real-time mode by rendering video media of the incoming video message simultaneously as the video media is transmitted by a second communication device, routed over a network to the first communication device, and received at the first communication device,

the incoming video message including a message header containing information for routing the video media of the incoming video message, as a delivery route over the network to the first communication device is discovered using the information contained in the message header, with the discovery of the delivery route simultaneously occurring as the video media is transmitted by the second communication device and rendered at the first communication device, resulting in real-time communication between the first and second communication devices without first establishing a connection over the network between the first and second communication respectively;

- (ii) a time-shifted mode by rendering video media of the incoming message out of storage; and

- (iii) providing the ability for the user to selectively render the video messaging transmission in either of the two modes (i) and (ii).

2. The method of claim **1**, wherein providing access to the video messaging system through the web site further comprises:

receiving a request by the first communication device to download a video messaging code to the first communication device;

downloading the video messaging code to the first communication device in response to the request, the video messaging code configured to create a user interface appearing within the web browser running on the first communication device; and

enabling the user of the first communication device to render the video message through the user interface.

3. The method of claim **1**, further comprising creating the video messaging code in a programming language so that it will run within the web browser.

4. The method of claim **1**, further comprising serving web content that appears within the web browser.

5. The method of claim **1**, further downloading a multimedia platform and a web-browser plug-in as needed to the first communication device.

6. The method of claim **1**, wherein the information contained in the message header is a unique identifier identifying the recipient among a plurality of users.

7. The method of claim **1**, wherein the delivery route over the network to the first communication device is discovered using a DNS lookup result of the information contained in the message header.

8. The method of claim **1**, wherein the web site is a social media web site.

9. The method of claim **1**, further comprising locally storing the video media of the video message at the first communication device.

10. The method of claim **1**, further comprising storing the video media of the video message at a server affiliated with the web site.

11. The method of claim **10**, further comprising serving the stored video media of the video message to the first communication device upon request so that the video media can be viewed in the time-shifted mode.

* * * * *