



(12) **United States Patent**  
**Ghafourifar et al.**

(10) **Patent No.:** **US 9,930,002 B2**  
(45) **Date of Patent:** **\*Mar. 27, 2018**

(54) **APPARATUS AND METHOD FOR INTELLIGENT DELIVERY TIME DETERMINATION FOR A MULTI-FORMAT AND/OR MULTI-PROTOCOL COMMUNICATION**

(58) **Field of Classification Search**  
CPC ..... H04L 12/5835; H04L 12/5855; H04L 12/589; H04L 51/066; H04L 51/36;  
(Continued)

(71) Applicant: **Entefy Inc.**, Palo Alto, CA (US)

(56) **References Cited**

U.S. PATENT DOCUMENTS

(72) Inventors: **Mehdi Ghafourifar**, Los Altos Hills, CA (US); **Alston Ghafourifar**, Los Altos Hills, CA (US); **Brienne Ghafourifar**, Los Altos Hills, CA (US)

5,481,597 A 1/1996 Given  
5,951,638 A 9/1999 Hoss  
(Continued)

(73) Assignee: **Entefy Inc.**, Palo Alto, CA (US)

FOREIGN PATENT DOCUMENTS

(\*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

WO 9931575 6/1999  
WO 2013112570 A1 8/2013

This patent is subject to a terminal disclaimer.

*Primary Examiner* — Tauqir Hussain

(74) *Attorney, Agent, or Firm* — Blank Rome LLP

(21) Appl. No.: **14/985,756**

(22) Filed: **Dec. 31, 2015**

(65) **Prior Publication Data**

US 2016/0119274 A1 Apr. 28, 2016

**Related U.S. Application Data**

(63) Continuation-in-part of application No. 14/141,551, filed on Dec. 27, 2013.

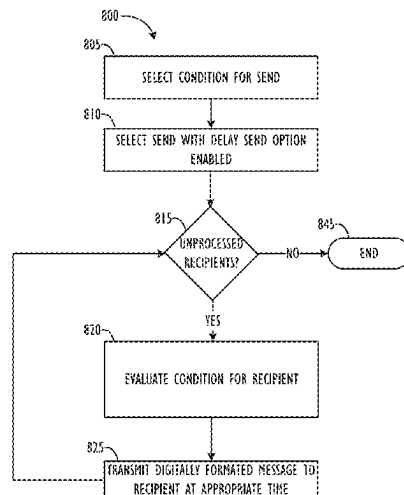
(51) **Int. Cl.**  
**G06F 15/16** (2006.01)  
**H04L 12/58** (2006.01)

(52) **U.S. Cl.**  
CPC ..... **H04L 51/36** (2013.01); **H04L 51/066** (2013.01); **H04L 51/14** (2013.01); **H04L 51/26** (2013.01)

(57) **ABSTRACT**

This disclosure relates generally to apparatus, methods, and computer readable media for composing communications for computing devices across multiple formats and multiple protocols. More particularly, but not by way of limitation, this disclosure relates to apparatus, methods, and computer readable media to permit computing devices, e.g., smartphones, tablets, laptops, and the like, to send communications in a number of pre-determined and/or 'determined-on-the-fly' communications formats and/or protocols at one or more determined delivery times via a single, seamless user interface. The determined delivery time of any message to any recipient may be intelligently determined individually for each recipient, e.g., by using one or more context factors to assess optimal delivery times for each recipient. The techniques disclosed herein allow communications systems to become 'message-centric' or 'people-centric,' as opposed to 'protocol-centric,' allowing the message protocol and delivery time to be abstracted from the sender of the communication, if the sender so desires.

**21 Claims, 16 Drawing Sheets**



**(58) Field of Classification Search**

CPC ..... H04L 51/32; H04L 67/22; H04L 67/36;  
H04L 67/2823; H04L 29/06; H04L 43/08;  
G06F 17/16  
USPC ..... 709/206  
See application file for complete search history.

2005/0015443 A1\* 1/2005 Levine ..... H04L 12/5835  
709/204  
2006/0193450 A1 8/2006 Flynt  
2007/0130273 A1 6/2007 Huynh  
2007/0237135 A1\* 10/2007 Trevallyn-Jones ..... H04L 45/02  
370/354

**(56)**

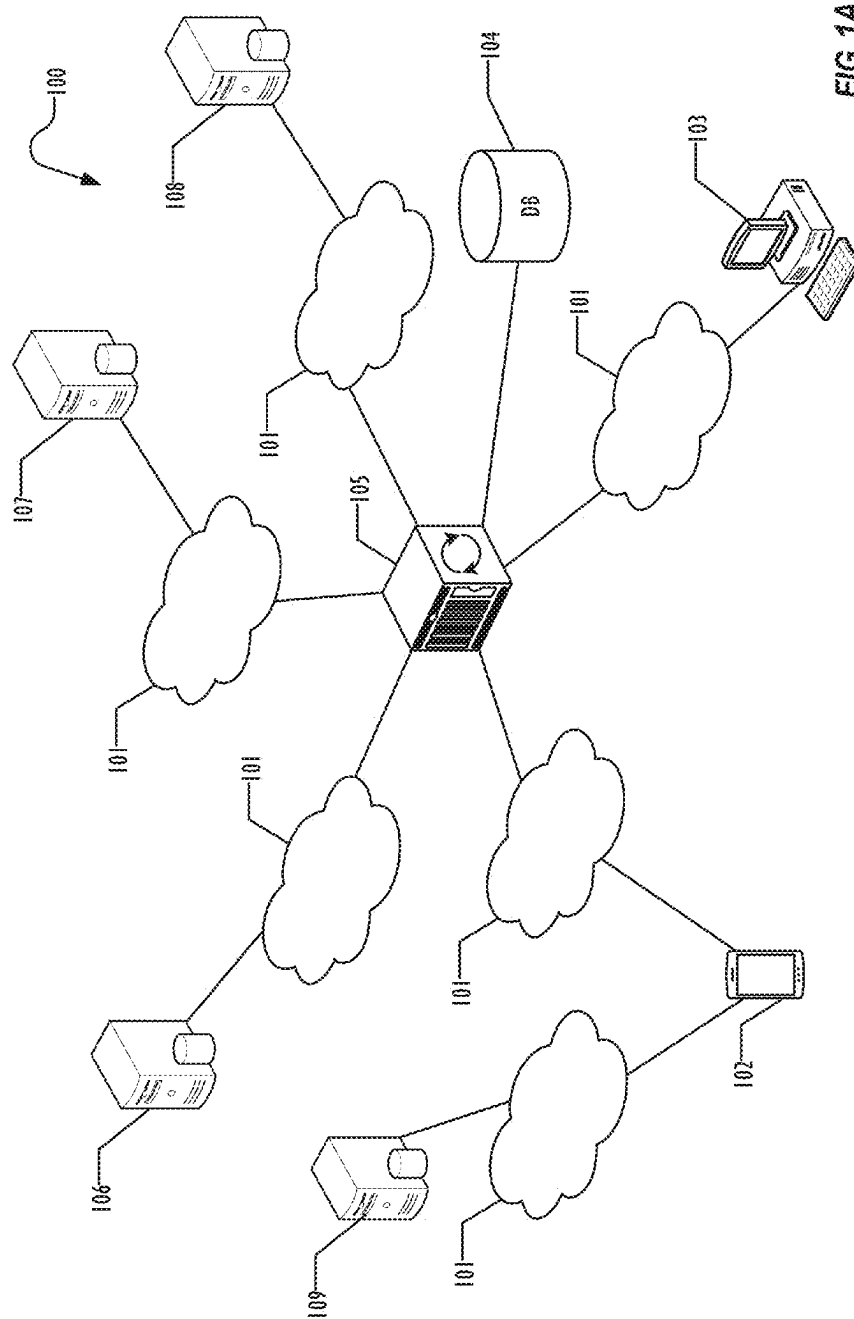
**References Cited**

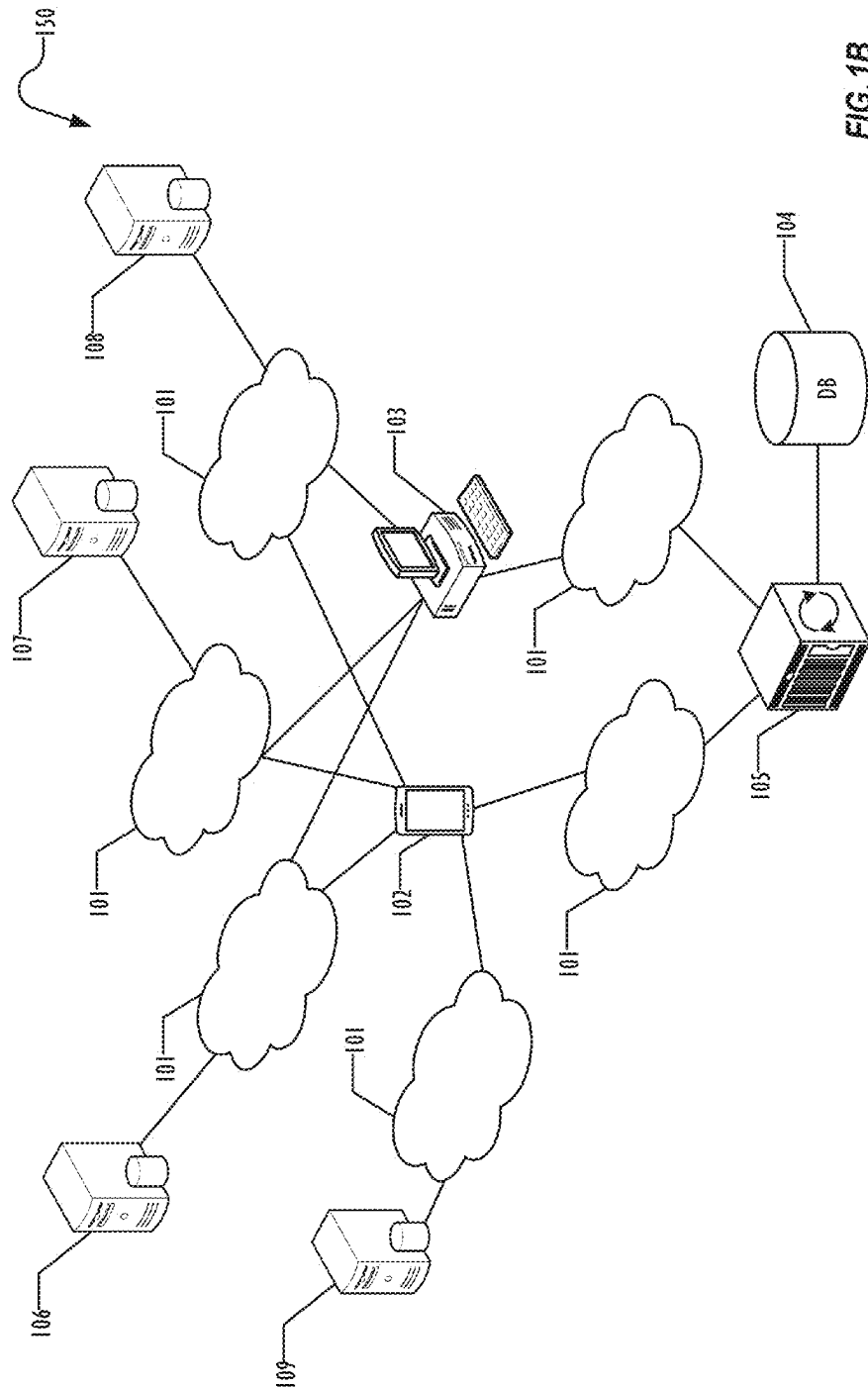
**U.S. PATENT DOCUMENTS**

6,101,320 A 8/2000 Schuetze  
6,950,502 B1 9/2005 Jenkins  
7,673,327 B1 3/2010 Polis  
7,886,000 B1 2/2011 Polis  
7,908,647 B1 3/2011 Polis  
8,090,787 B2 1/2012 Polis  
8,095,592 B2 1/2012 Polis  
8,108,460 B2 1/2012 Polis  
8,112,476 B2 2/2012 Polis  
8,122,080 B2 2/2012 Polis  
8,156,183 B2 4/2012 Polis  
8,281,125 B1 10/2012 Briceno  
8,296,360 B2 10/2012 Polis  
8,433,705 B1 4/2013 Dredze  
8,438,223 B2 5/2013 Polis  
8,458,256 B2 6/2013 Polis  
8,458,292 B2 6/2013 Polis  
8,458,347 B2 6/2013 Polis  
8,468,202 B2 6/2013 Polis  
8,959,156 B2 2/2015 Polis  
2002/0133509 A1 9/2002 Johnston  
2002/0178000 A1 11/2002 Aktas  
2002/0194322 A1 12/2002 Nagata  
2004/0137884 A1 7/2004 Engstrom  
2004/0243719 A1 12/2004 Roselinsky  
2004/0266411 A1 12/2004 Galicia

2008/0062133 A1 3/2008 Wolf  
2008/0088428 A1 4/2008 Pitre  
2008/0261569 A1 10/2008 Britt  
2009/0016504 A1 1/2009 Mantell  
2009/0119370 A1 5/2009 Stern  
2009/0177744 A1 7/2009 Marlow  
2009/0181702 A1 7/2009 Vargas  
2009/0271486 A1 10/2009 Ligh  
2009/0292814 A1 11/2009 Ting  
2009/0299996 A1 12/2009 Yu  
2010/0057872 A1\* 3/2010 Koons ..... G06Q 30/02  
709/206  
2010/0210291 A1 8/2010 Lauer  
2010/0323728 A1 12/2010 Gould  
2010/0325227 A1 12/2010 Novy  
2011/0051913 A1 3/2011 Kesler  
2011/0078247 A1 3/2011 Jackson  
2011/0130168 A1 6/2011 Vendrow  
2011/0194629 A1 8/2011 Bekanich  
2012/0209847 A1 8/2012 Rangan  
2012/0221962 A1 8/2012 Lew  
2013/0151508 A1 6/2013 Kurabayashi  
2013/0262852 A1 10/2013 Roeder  
2013/0267264 A1 10/2013 Abuelsaad  
2013/0304830 A1 11/2013 Olsen  
2013/0332308 A1 12/2013 Linden  
2014/0297807 A1 10/2014 Dasgupta  
2015/0039887 A1 2/2015 Kahol  
2015/0281184 A1 10/2015 Cooley  
2016/0087944 A1 3/2016 Downey

\* cited by examiner





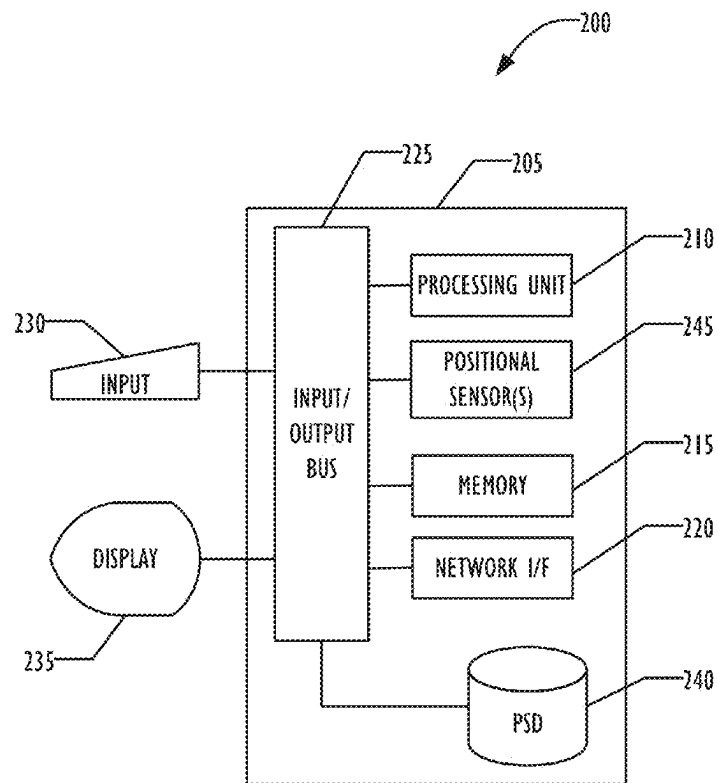


FIG. 2A

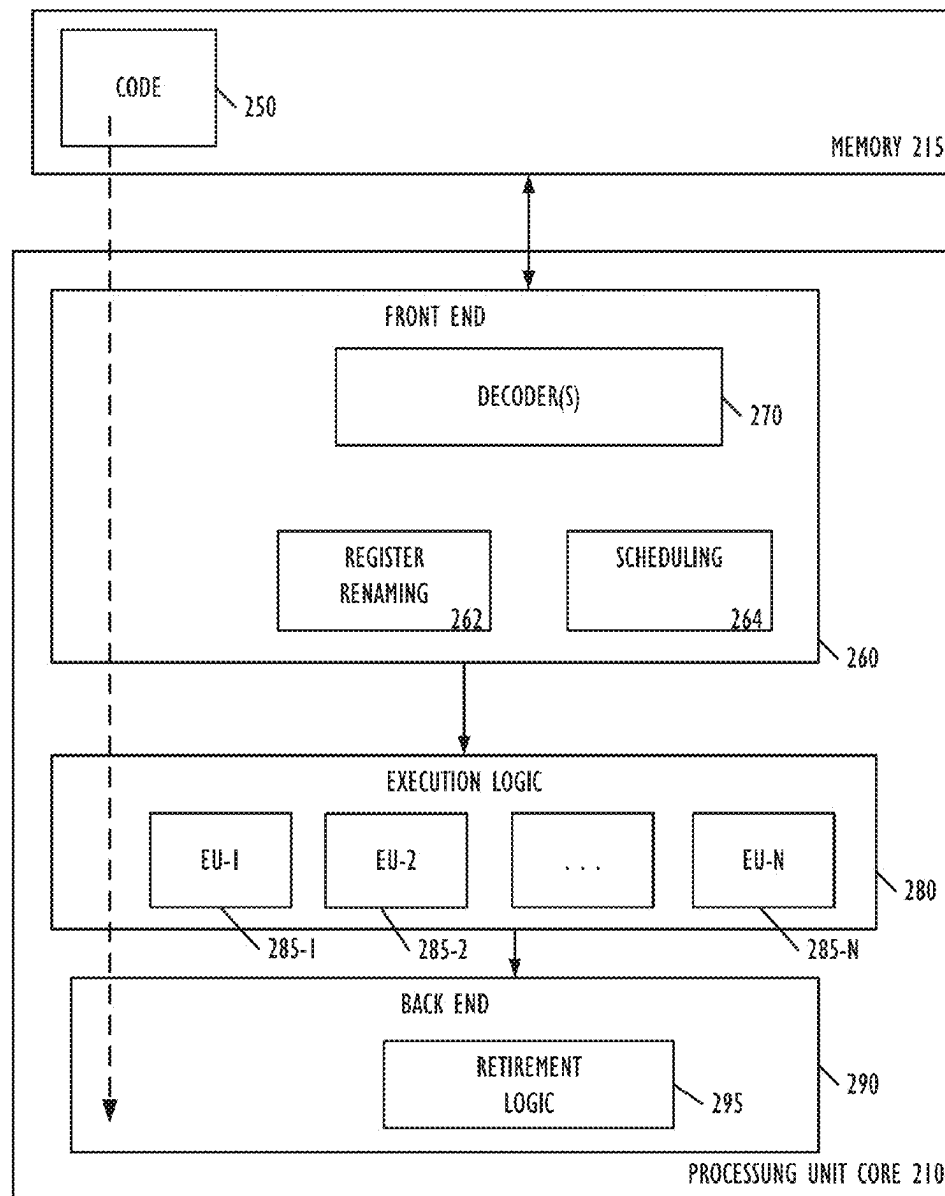


FIG. 2B

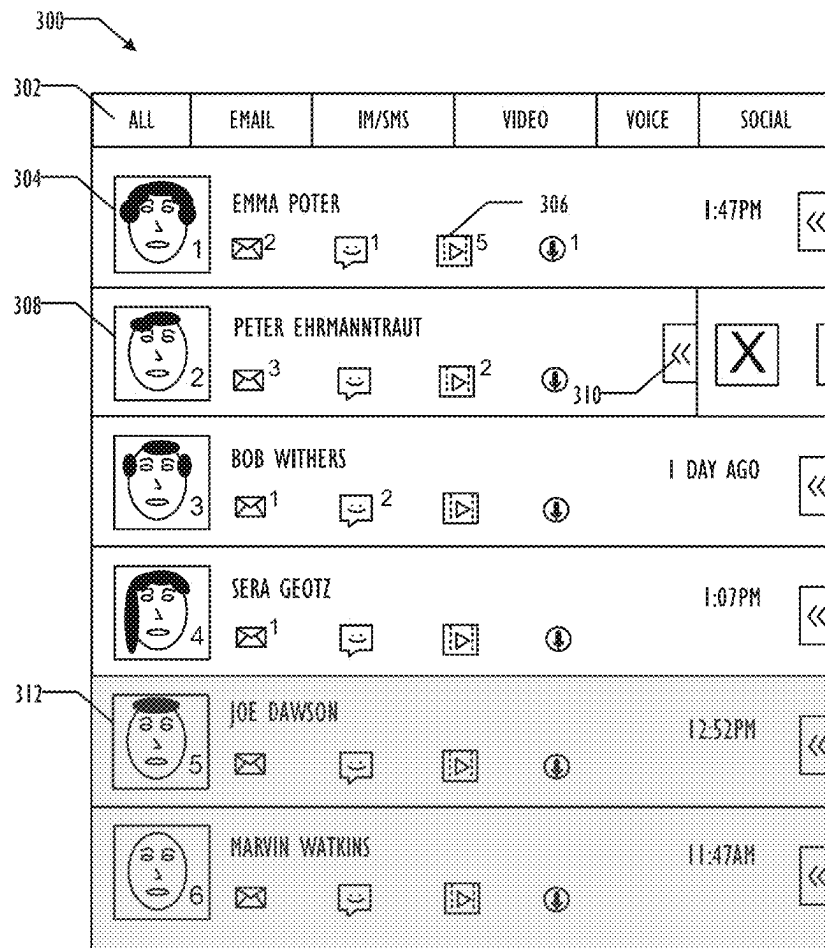


FIG. 3A

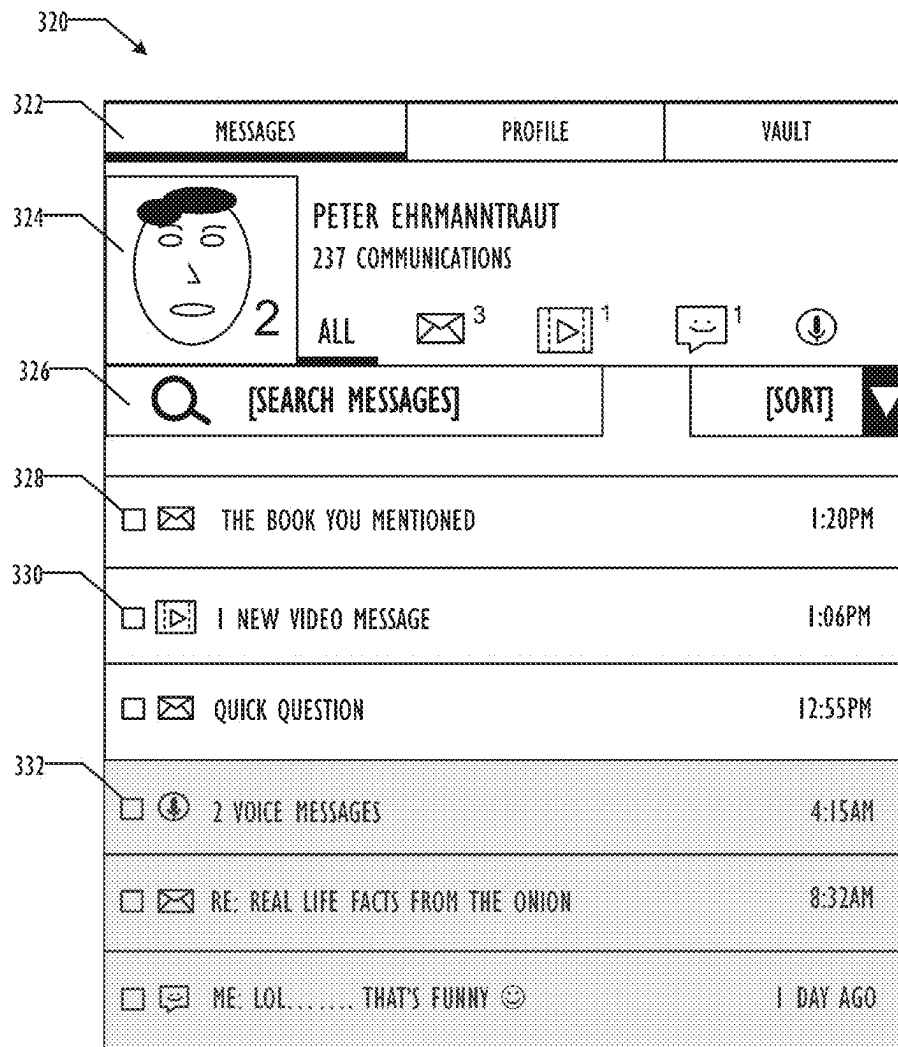


FIG. 3B



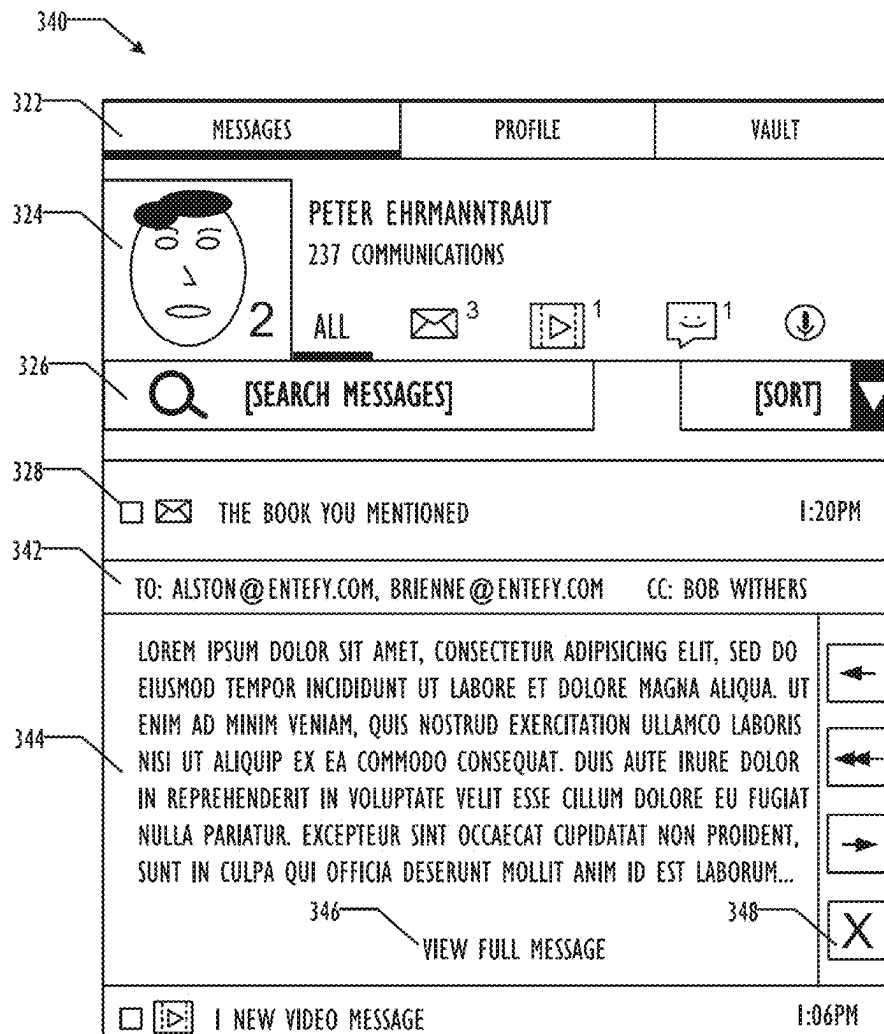


FIG. 3C

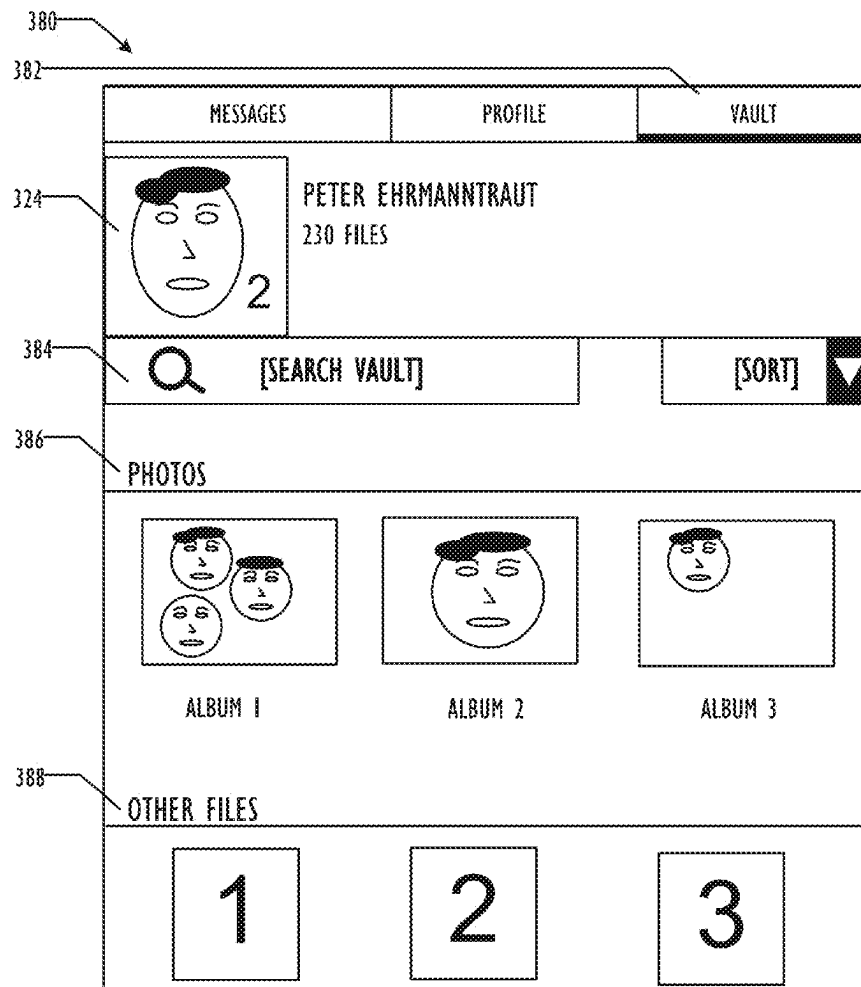


FIG. 3D

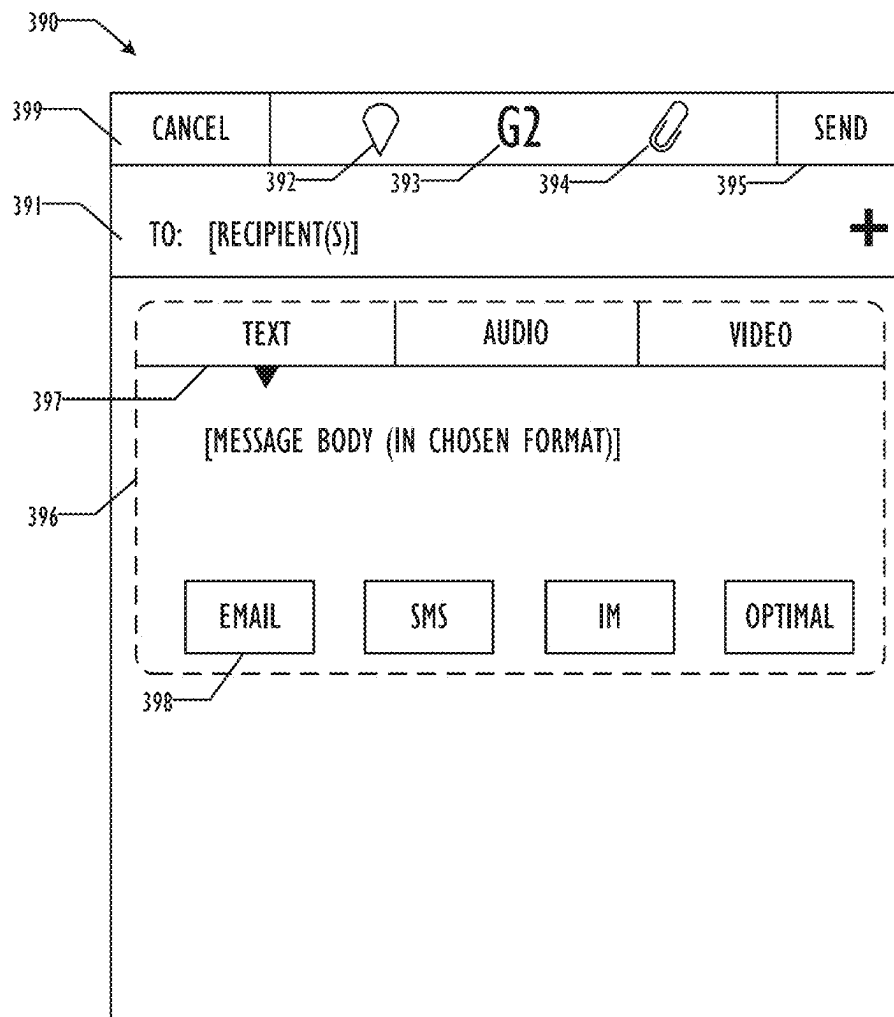


FIG. 3E

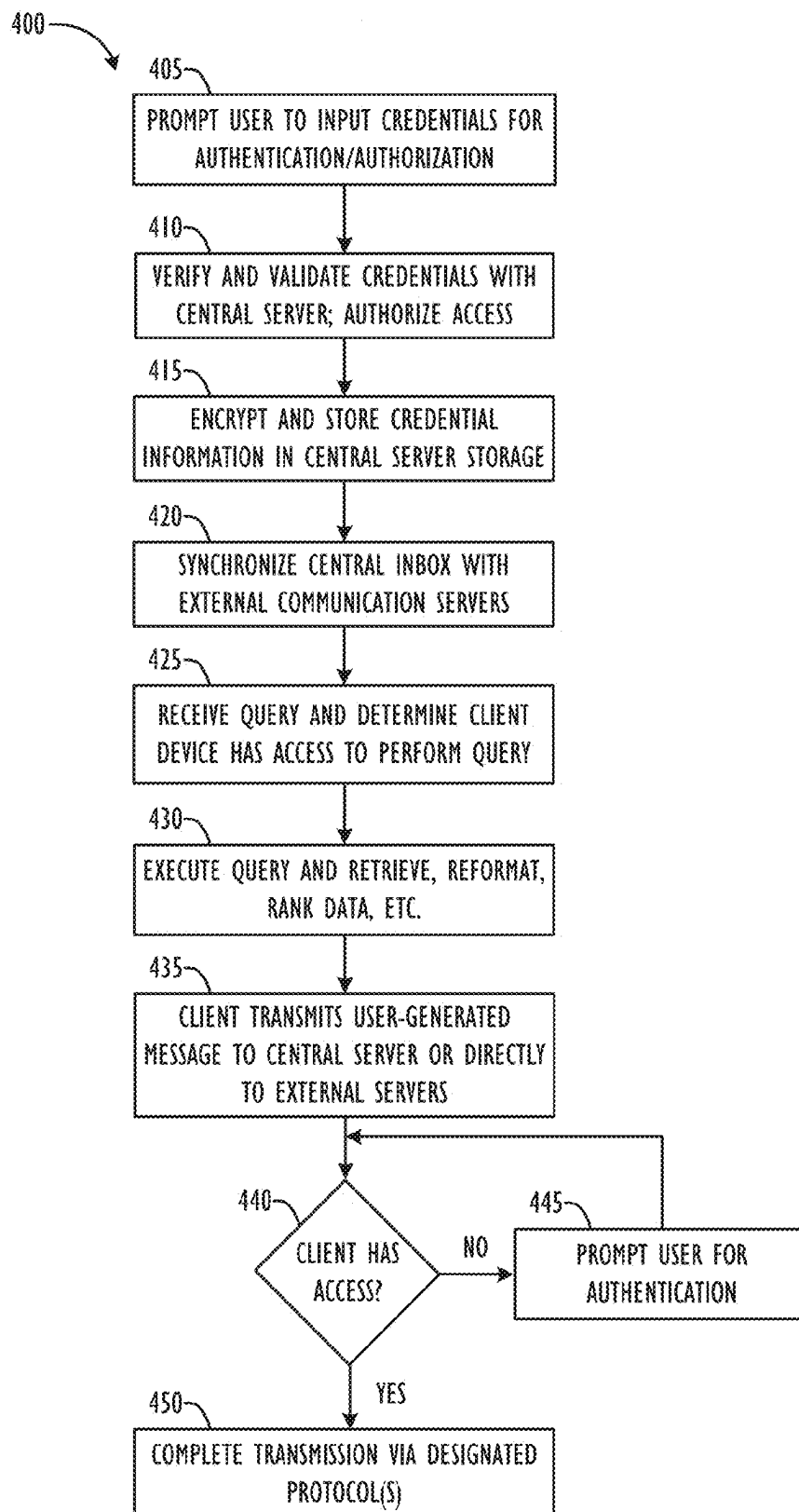


FIG. 4

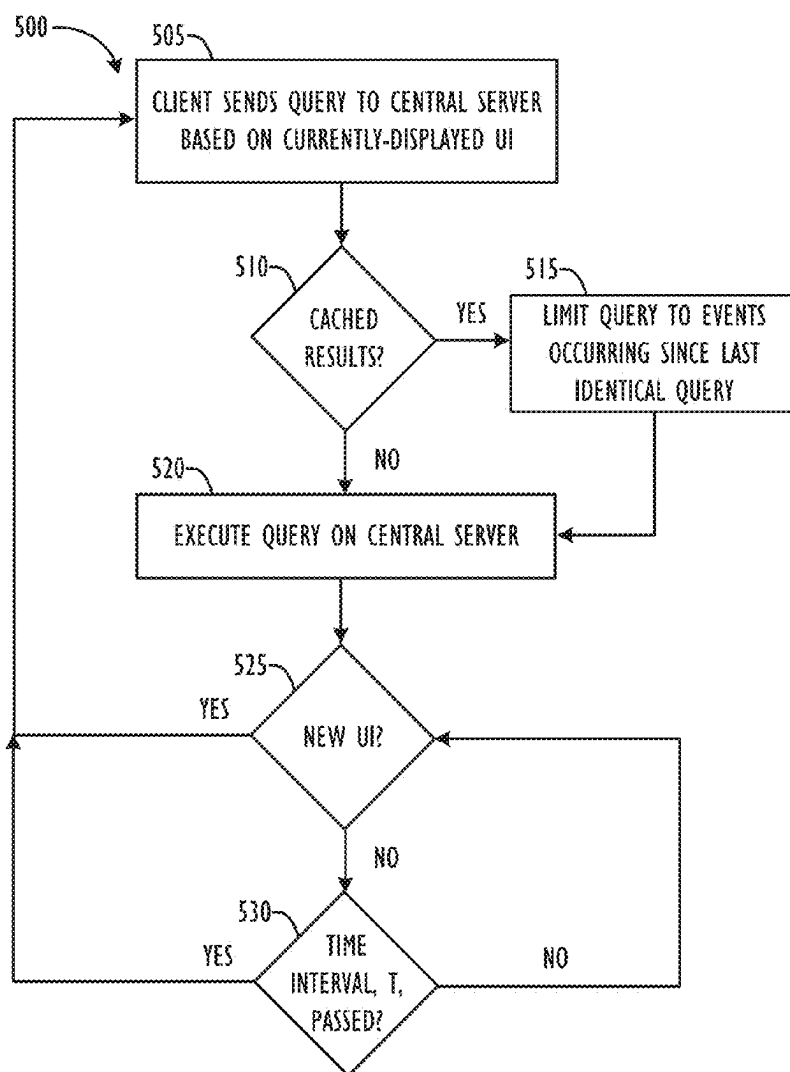


FIG. 5

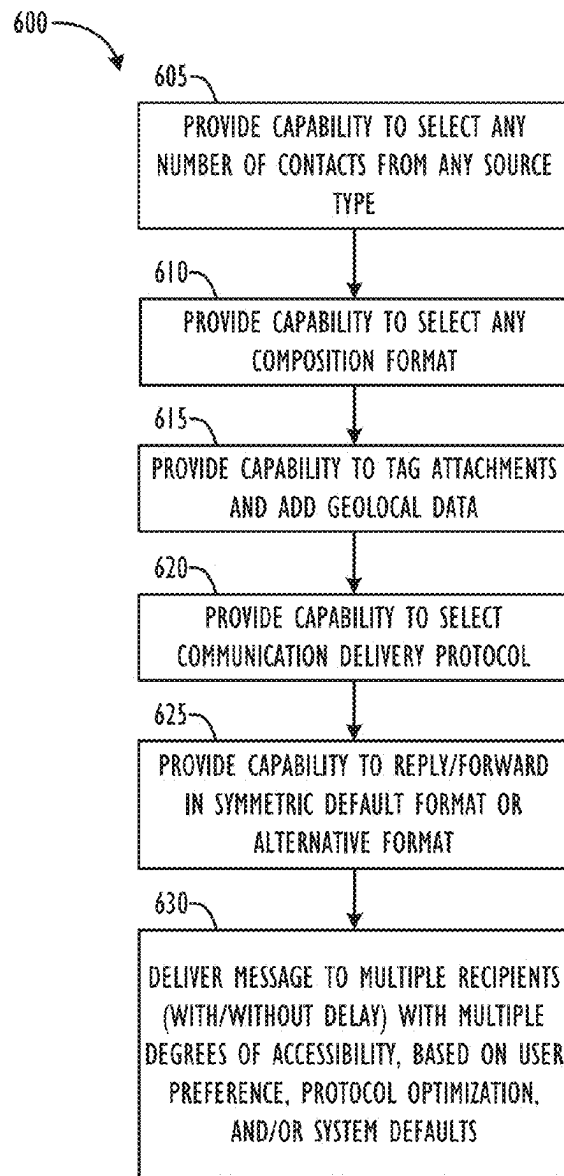


FIG. 6

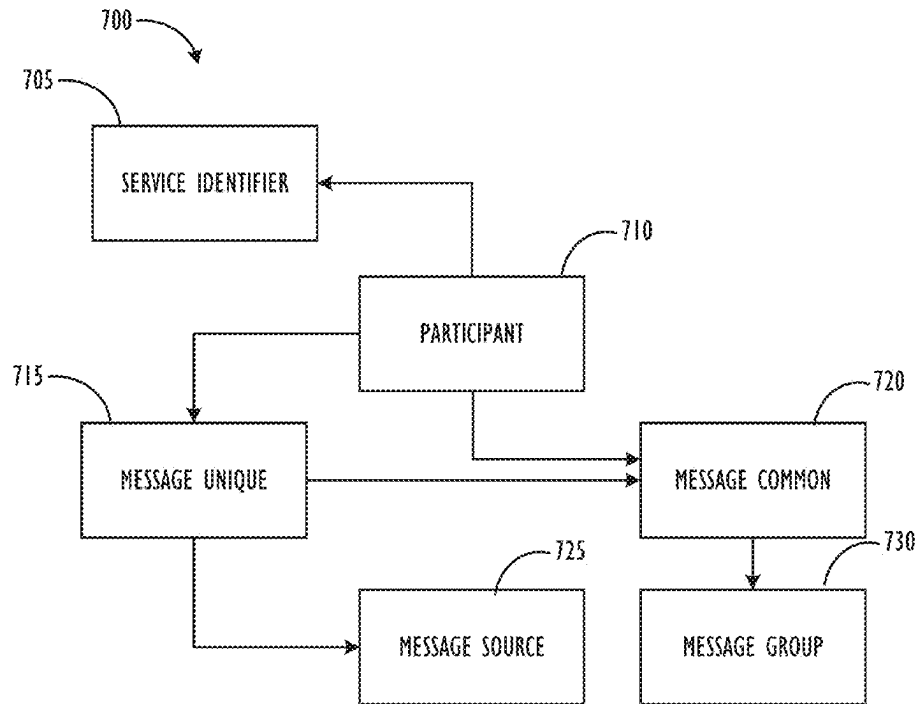


FIG. 7

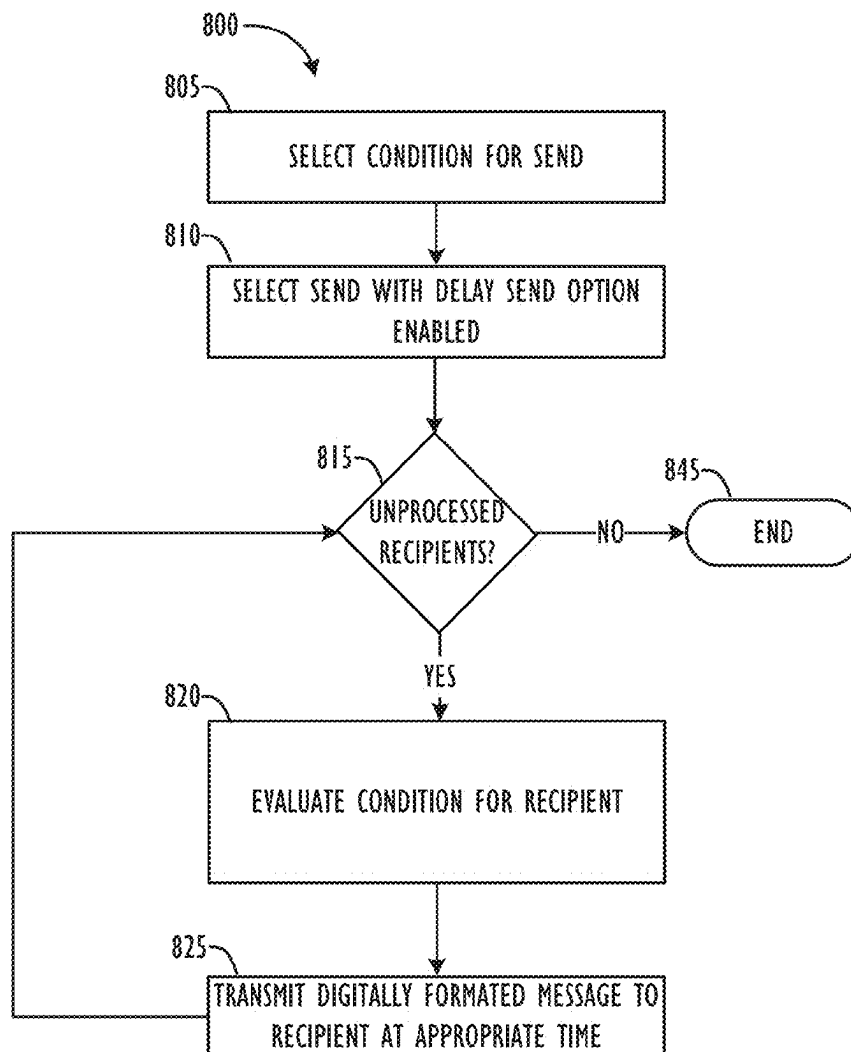


FIG. 8



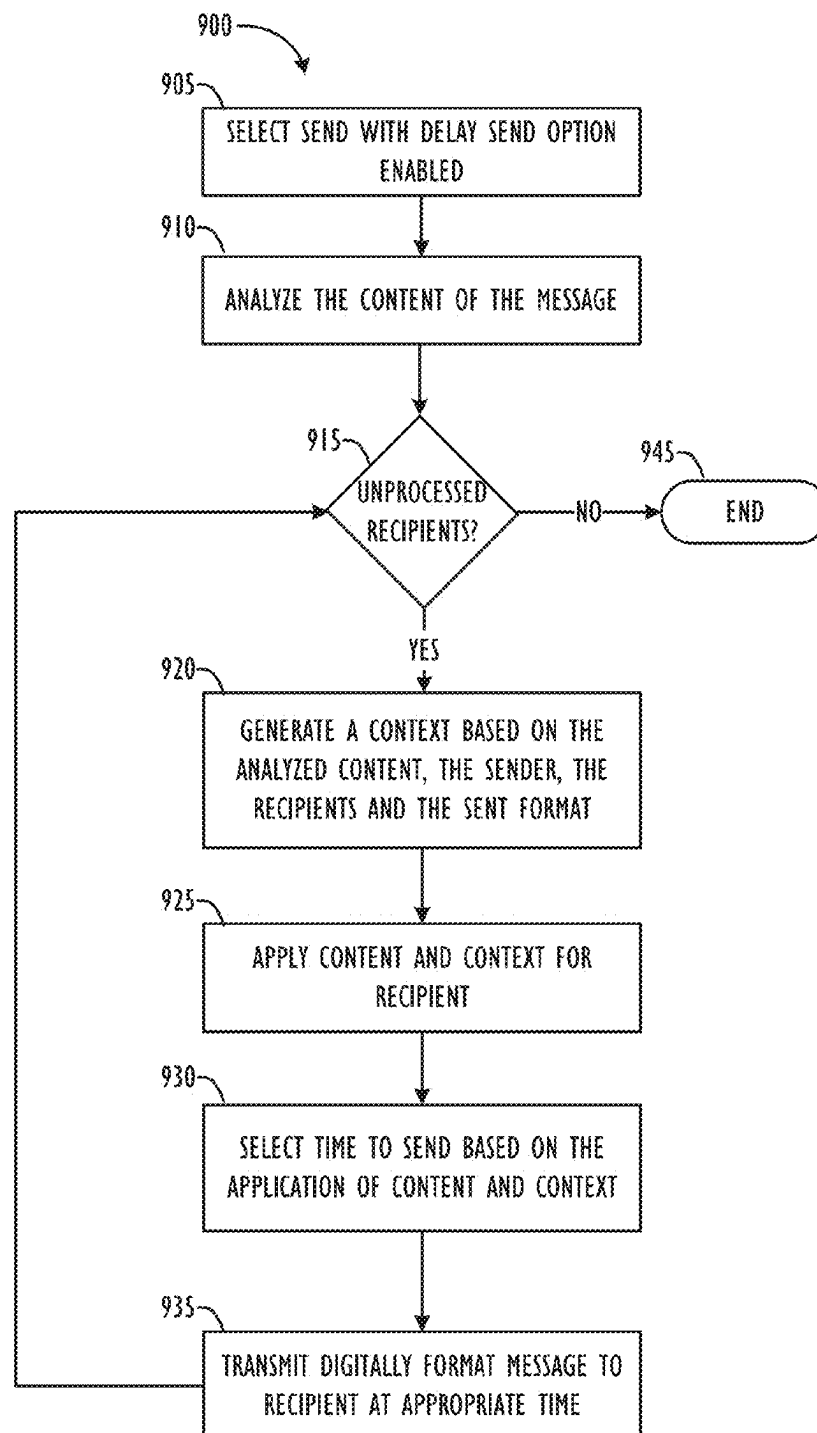


FIG. 9

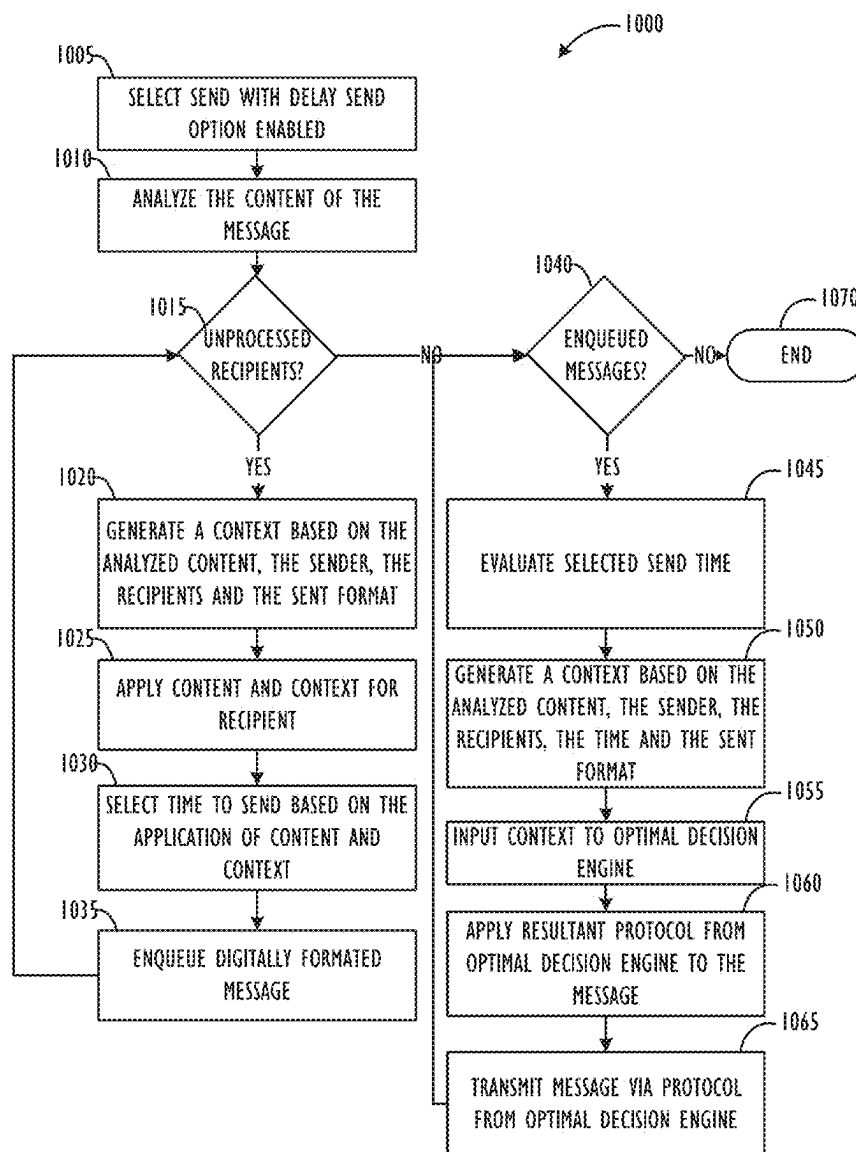


FIG. 10

1

# APPARATUS AND METHOD FOR INTELLIGENT DELIVERY TIME DETERMINATION FOR A MULTI-FORMAT AND/OR MULTI-PROTOCOL COMMUNICATION

## CROSS REFERENCE TO RELATED APPLICATIONS

This application is a continuation-in-part to the commonly-assigned and co-pending nonprovisional patent application having U.S. patent application Ser. No. 14/141,551 filed Dec. 27, 2013, and entitled, "Apparatus and Method for Multi-Format Communication Composition" ("the '551 application"). The '551 application is also hereby incorporated by reference in its entirety.

## TECHNICAL FIELD

This disclosure relates generally to apparatuses, methods, and computer readable media for composing communications for computing devices across multiple communications formats and protocols wherein the time of send/delivery of any message or group of message to any recipient or group of recipients can be intelligently determined using one or more context factors to assess optimal transaction time for each party involved.

## BACKGROUND

The proliferation of personal computing devices in recent years, especially mobile personal computing devices, combined with a growth in the number of widely-used communications formats (e.g., text, voice, video, image) and protocols (e.g., SMTP, IMAP/POP, SMS/MMS, XMPP, etc.) has led to a communications experience that many users find fragmented and restrictive. Users desire a system that will provide ease of communication by sending an outgoing message created in whatever format was convenient to the composer, with delivery options to one or more receivers in whatever format or protocol (and at whatever time) works best for them—all seamlessly from the composer's and recipient(s)'s perspective. With current communications technologies that remain "protocol-centric"—as opposed to "message-centric" or "people-centric"—such ease of communication is not possible.

In the past, users of communications systems first had to choose a communication format and activate a corresponding application or system prior to composing a message or selecting desired recipient(s). For example, if a person wanted to call someone, then he or she would need to pick up a telephone and enter the required phone number or directory in order to connect. If a person wanted to email a colleague, that person would be required to launch an email application before composing and sending email, etc. Further, while long-form text might be the most convenient format at the time for the composer, long-form text may not be convenient for the receiver—resulting in a delayed receipt of and/or response to the message by the receiver. With the multi-format communication composition techniques described herein, however, the user flow is much more natural and intuitive. First, the 'Sender' (a registered user of the multi-format, multi-protocol communication system), can select the desired recipient(s). Then, the Sender may compose the outgoing message (in any format such as text, video recording, audio recording). Next, the system (or the Sender, in some embodiments) intelligently chooses the

2

delivery protocol for the communication, e.g., whether the communication is going to be sent via email, SMS, IM, or social media, etc. Finally, the outgoing message is converted into the desired outgoing message format (either by the Sender's client device or a central communications system server) and sent to the desired recipient(s) via the chosen delivery protocol(s) at the determined time(s).

According to the multi-format communication composition techniques described herein, the emphasis in the communication interface is on the "who" and the "what" of the communication—but not the "how." The multi-format communication composition system described herein takes care of the "how"—including an 'Optimal' option, as determined by the 'Optimal Decision Engine' which may be employed to deliver the outgoing communication to the desired recipient(s) in the most preferred way, e.g., either through preferences that the recipient(s) has specified via his or her profile in a multi-format communications network or through the communication protocol information regarding the desired recipient that is stored in the sender's contact list, and at a preferred time(s).

The subject matter of the present disclosure is directed to overcoming, or at least reducing the effects of, one or more of the problems set forth above. To address these and other issues, techniques that enable seamless, multi-format and multi-protocol communications via a single user interface are described herein.

## BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1A is a block diagram illustrating a server-entry point network architecture infrastructure, according to one or more disclosed embodiments.

FIG. 1B is a block diagram illustrating a client-entry point network architecture infrastructure, according to one or more disclosed embodiments.

FIG. 2A is a block diagram illustrating a computer which could be used to execute the multi-format/multi-protocol communication optimization approaches described herein according to one or more of disclosed embodiments.

FIG. 2B is a block diagram illustrating a processor core, which may reside on a computer according to one or more of disclosed embodiments.

FIG. 3A shows an example of a multi-protocol, person-centric, multi-format inbox feed, according to one or more disclosed embodiments.

FIG. 3B shows an example of a multi-protocol, multi-format inbox feed for messages to and from a particular user, according to one or more disclosed embodiments.

FIG. 3C shows an example of a preview pane for a multi-protocol, multi-format inbox feed for messages to and from a particular user, according to one or more disclosed embodiments.

FIG. 3D shows an example of a document repository page for a particular user, according to one or more disclosed embodiments.

FIG. 3E shows an example of a multi-protocol, multi-format communication composition user interface, according to one or more disclosed embodiments.

FIG. 4 is a flowchart of one embodiment of a method for populating a multi-protocol, person-centric, multi-format inbox feed, according to one or more disclosed embodiments.

FIG. 5 is a flowchart of one embodiment of a method for processing a user interface-driven query, according to one or more disclosed embodiments.

3

FIG. 6 is a flowchart of one embodiment of a method for creating a multi-protocol, multi-format communication transmission, according to one or more disclosed embodiments.

FIG. 7 is a block diagram of one embodiment of a Universal Message Object, according to one or more disclosed embodiments.

FIG. 8 is a flowchart of one embodiment of a method of delivering a message to one or more recipients based on events specified by the sender, according to one or more disclosed embodiments.

FIG. 9 is a flowchart of one embodiment of a method of determining a suggestive delivery time for a message to one or more recipients, according to one or more disclosed embodiments.

FIG. 10 is a flowchart of one embodiment of a method of determining a suggestive delivery time for a message to one or more recipients with an Optimal delivery option selected, according to one or more disclosed embodiments.

#### DETAILED DESCRIPTION

Disclosed are apparatuses, methods, and computer readable media for composing communications for computing devices across multiple formats and multiple protocols. More particularly, but not by way of limitation, this disclosure relates to apparatuses, methods, and computer readable media to permit computing devices, e.g., smartphones, smart devices, tablets, wearable devices, laptops, and the like, to send communications in a number of pre-determined and/or 'determined-on-the-fly' communications formats and/or protocols at a determined time(s) via a single, seamless user interface.

Determinations of outgoing communication formats and/or protocols, as well as delivery times, may be based on, e.g., the format of the incoming communication, the preferred format of the recipient and/or sender of the communication, an optimal format for a given communication session/message, a generated context, and/or economic considerations of format/protocol choice to the recipient and/or sender. The techniques disclosed herein allow communications systems to become 'message-centric' or 'people-centric' as opposed to 'protocol-centric,' eventually allowing consideration of message protocol to fall away entirely for the sender of the communication. With reference to the figures, embodiments of communication optimization schemes according to this disclosure are provided below.

Referring now to FIG. 1A, a server-entry point network architecture infrastructure 100 is shown schematically. Infrastructure 100 contains computer networks 101. Computer networks 101 include many different types of computer networks available today, such as the Internet, a corporate network, or a Local Area Network (LAN). Each of these networks can contain wired or wireless devices and operate using any number of network protocols (e.g., TCP/IP). Networks 101 may be connected to various gateways and routers, connecting various machines to one another, represented, e.g., by sync server 105, end user computers 103, mobile phones 102, and computer servers 106-109. In some embodiments, end user computers 103 may not be capable of receiving SMS text messages, whereas mobile phones 102 are capable of receiving SMS text messages. Also shown in infrastructure 100 is a cellular network 101 for use with mobile communication devices. As is known in the art, mobile cellular networks support mobile phones and many other types of devices (e.g., tablet computers not shown). Mobile devices in the infrastructure 100 are illus-

4

trated as mobile phone 102. Sync server 105, in connection with database(s) 104, may serve as the central "brains" and data repository, respectively, for the multi-protocol, multi-format communication composition and inbox feed system to be described herein. In the server-entry point network architecture infrastructure 100 of FIG. 1A, centralized sync server 105 may be responsible for querying and obtaining all the messages from the various communication sources for individual users of the system and keeping the multi-protocol, multi-format inbox feed for a particular user of the system synchronized with the data on the various third party communication servers that the system is in communication with. Database(s) 104 may be used to store local copies of messages sent and received by users of the system, as well as individual documents associated with a particular user, which may or may not also be associated with particular communications of the users. As such, the database portion allotted to a particular user will contain a record of all communications in any form to and from the user.

Server 106 in the server-entry point network architecture infrastructure 100 of FIG. 1A represents a third party email server (e.g., a GOOGLE® or YAHOO!® email server). (GOOGLE is a registered service mark of Google Inc. YAHOO! is a registered service mark of Yahoo! Inc.) Third party email server 106 may be periodically pinged by sync server 105 to determine whether particular users of the multi-protocol, multi-format communication composition and inbox feed system described herein have received any new email messages via the particular third-party email services. Server 107 represents a third party instant message server (e.g., a YAHOO!® Messenger or AOL® Instant Messaging server). (AOL is a registered service mark of AOL Inc.) Third party instant messaging server 107 may also be periodically pinged by sync server 105 to determine whether particular users of the multi-protocol, multi-format communication composition and inbox feed system described herein have received any new instant messages via the particular third-party instant messaging services. Similarly, server 108 represents a third party social network server (e.g., a FACEBOOK® or TWITTER® server). (FACEBOOK is a registered trademark of Facebook, Inc. TWITTER is a registered service mark of Twitter, Inc.) Third party social network server 108 may also be periodically pinged by sync server 105 to determine whether particular users of the multi-protocol, multi-format communication composition and inbox feed system described herein have received any new social network messages via the particular third-party social network services. It is to be understood that, in a "push-based" system, third party servers may push notifications to sync server 105 directly, thus eliminating the need for sync server 105 to periodically ping the third party servers. Finally, server 109 represents a cellular service provider's server. Such servers may be used to manage the sending and receiving of messages (e.g., email or SMS text messages) to users of mobile devices on the provider's cellular network. Cellular service provider servers may also be used: 1) to provide geo-fencing for location and movement determination; 2) for data transference; and/or 3) for live telephony (i.e., actually answering and making phone calls with a user's client device). In situations where two 'on-network' users are communicating with one another via the multi-protocol, multi-format communication system itself, such communications may occur entirely via sync server 105, and third party servers 106-109 may not need to be contacted.

Referring now to FIG. 1B, a client-entry point network architecture infrastructure 150 is shown schematically. Simi-

5

lar to infrastructure 100 shown in FIG. 1A, infrastructure 150 contains computer networks 101. Computer networks 101 may again include many different types of computer networks available today, such as the Internet, a corporate network, or a Local Area Network (LAN). However, unlike the server-centric infrastructure 100 shown in FIG. 1A, infrastructure 150 is a client-centric architecture. Thus, individual client devices, such as end user computers 103 and mobile phones 102 may be used to query the various third party computer servers 106-109 to retrieve the various third party email, IM, social network, and other messages for the user of the client device. Such a system has the benefit that there may be less delay in receiving messages than in a system where a central server is responsible for authorizing and pulling communications for many users simultaneously. Also, a client-entry point system may place less storage and processing responsibilities on the central multi-protocol, multi-format communication composition and inbox feed system's server computers since the various tasks may be distributed over a large number of client devices. Further, a client-entry point system may lend itself well to a true, "zero knowledge" privacy enforcement scheme. In infrastructure 150, the client devices may also be connected via the network to the central sync server 105 and database 104. For example, central sync server 105 and database 104 may be used by the client devices to reduce the amount of storage space needed on-board the client devices to store communications-related content and/or to keep all of a user's devices synchronized with the latest communication-related information and content related to the user. It is to be understood that, in a "push-based" system, third party servers may push notifications to end user computers 102 and mobile phones 103 directly, thus eliminating the need for these devices to periodically ping the third party servers.

Referring now to FIG. 2A, an example processing device 200 for use in the communication systems described herein according to one embodiment is illustrated in block diagram form. Processing device 200 may serve in, e.g., a mobile phone 102, end user computer 103, sync server 105, or a server computer 106-109. Example processing device 200 comprises a system unit 205 which may be optionally connected to an input device 230 (e.g., keyboard, mouse, touch screen, etc.) and display 235. A program storage device (PSD) 240 (sometimes referred to as a hard disk, flash memory, or non-transitory computer readable medium) is included with the system unit 205. Also included with system unit 205 may be a network interface 220 for communication via a network (either cellular or computer) with other mobile and/or embedded devices (not shown). Network interface 220 may be included within system unit 205 or be external to system unit 205. In either case, system unit 205 will be communicatively coupled to network interface 220. Program storage device 240 represents any form of non-volatile storage including, but not limited to, all forms of optical and magnetic memory, including solid-state storage elements, including removable media, and may be included within system unit 205 or be external to system unit 205. Program storage device 240 may be used for storage of software to control system unit 205, data for use by the processing device 200, or both.

System unit 205 may be programmed to perform methods in accordance with this disclosure. System unit 205 comprises one or more processing units, input-output (I/O) bus 225 and memory 215. Access to memory 215 can be accomplished using the communication bus 225. Processing unit 210 may include any programmable controller device including, for example, a mainframe processor, a mobile

6

phone processor, or, as examples, one or more members of the INTEL® ATOM™, INTEL® XEON™, and INTEL® CORE™ processor families from Intel Corporation and the Cortex and ARM processor families from ARM. (INTEL, INTEL ATOM, XEON, and CORE are trademarks of the Intel Corporation. CORTEX is a registered trademark of the ARM Limited Corporation. ARM is a registered trademark of the ARM Limited Company). Memory 215 may include one or more memory modules and comprise random access memory (RAM), read only memory (ROM), programmable read only memory (PROM), programmable read-write memory, and solid-state memory. As also shown in FIG. 2A, system unit 205 may also include one or more positional sensors 245, which may comprise an accelerometer, gyrometer, global positioning system (GPS) device, or the like, and which may be used to track the movement of user client devices.

Referring now to FIG. 2B, a processing unit core 210 is illustrated in further detail, according to one embodiment. Processing unit core 210 may be the core for any type of processor, such as a micro-processor, an embedded processor, a digital signal processor (DSP), a network processor, or other device to execute code. Although only one processing unit core 210 is illustrated in FIG. 2B, a processing element may alternatively include more than one of the processing unit core 210 illustrated in FIG. 2B. Processing unit core 210 may be a single-threaded core or, for at least one embodiment, the processing unit core 210 may be multithreaded, in that, it may include more than one hardware thread context (or "logical processor") per core.

FIG. 2B also illustrates a memory 215 coupled to the processing unit core 210. The memory 215 may be any of a wide variety of memories (including various layers of memory hierarchy), as are known or otherwise available to those of skill in the art. The memory 215 may include one or more code instruction(s) 250 to be executed by the processing unit core 210. The processing unit core 210 follows a program sequence of instructions indicated by the code 250. Each instruction enters a front end portion 260 and is processed by one or more decoders 270. The decoder may generate as its output a micro operation such as a fixed width micro operation in a predefined format, or may generate other instructions, microinstructions, or control signals which reflect the original code instruction. The front end 260 may also include register renaming logic 262 and scheduling logic 264, which generally allocate resources and queue the operation corresponding to the convert instruction for execution.

The processing unit core 210 is shown including execution logic 280 having a set of execution units 285-1 through 285-N. Some embodiments may include a number of execution units dedicated to specific functions or sets of functions. Other embodiments may include only one execution unit or one execution unit that can perform a particular function. The execution logic 280 performs the operations specified by code instructions.

After completion of execution of the operations specified by the code instructions, back end logic 290 retires the instructions of the code 250. In one embodiment, the processing unit core 210 allows out of order execution but requires in order retirement of instructions. Retirement logic 295 may take a variety of forms as known to those of skill in the art (e.g., re-order buffers or the like). In this manner, the processing unit core 210 is transformed during execution of the code 250, at least in terms of the output generated by the decoder, the hardware registers and tables utilized by the

register renaming logic **262**, and any registers (not shown) modified by the execution logic **280**.

Although not illustrated in FIG. **2B**, a processing element may include other elements on chip with the processing unit core **210**. For example, a processing element may include memory control logic along with the processing unit core **210**. The processing element may include I/O control logic and/or may include I/O control logic integrated with memory control logic. The processing element may also include one or more caches.

#### Multi-Protocol, Multi-Format Inbox Feed

FIG. **3A** shows an example of a multi-protocol, person-centric, multi-format inbox feed **300**, according to one or more disclosed embodiments. The inbox feed **300** shown in FIG. **3A** may, e.g., be displayed on the display of a mobile phone, laptop computer, or other computing device. In certain embodiments, elements of inbox feed **300** may be interacted with by a user utilizing a touchscreen interface or any other suitable input interface.

As is shown across the top row of the interface **302**, the multi-format, multi-protocol messages received by a user of the system may be grouped by format (e.g., Email, IM/SMS, Video, Voice, etc.), or all formats may be combined together into a single, unified inbox feed, as is shown in FIG. **3A**. Row **304** in the example of FIG. **3A** represents the first “person-centric” message row in the user’s unified inbox feed. As shown in FIG. **3A**, the pictorial icon and name of the sender whose messages are listed in row **304** appear at the beginning of the row. The pictorial icon and sender name indicate to the user of the system that all messages that have been aggregated in row **304** are from exemplary user ‘Emma Poter.’ Note that any indication of sender may be used. Also present in row **304** are several graphical icons **306** that represent links to messages of different types that have been received from Emma Poter. For example, Emma Poter has sent the particular user whose inbox feed is shown in FIG. **3A** two email messages, one instant message, five video messages, and one voice message. The user interface may utilize icons, as is shown in FIG. **3A**, or it may use any other suitable form of indication, such as text, grids, charts, or any other form of personalized identification. The types of messages/communication used in the inbox feed may be selected or personalized, as well. The timestamp (e.g., 1:47 pm in row **304**) may be used to indicate the time at which the most recently-received message has been received from a particular sender.

Moving down to row **308** of inbox feed **300**, messages from a second user, Peter Ehrmanntraut, have also been aggregated into a single row of the feed. As is displayed on the right hand side of row **308** is reveal arrow **310**. Selection of reveal arrow **310** may provide additional options to the user such as to reply, delay reply/delay send, forward, return a call, favorite, archive, or delete certain message from a particular sender. Further, the reveal action may conveniently keep the user on the same screen and allows for quick visual filtering of messages. Gestures and icon features may help the user with the decision-making process regarding the choice to reply, delay replying (including the time delaying of response across multiple protocols), delete, mark as spam, see a full message, translate, read, or flag a message as being unread. With respect to the “delay reply/delay send” option, the multi-protocol, multi-format communication system may determine, based on the determined outgoing message format and protocol, that a particular communication in a particular format should be delayed before being sent to the recipient. For example, a video or voice message may not be appropriate to send at midnight,

and so the system may delay sending the message until such time as the recipient is more likely to be awake, e.g., 9:00 am. On the other hand, the outgoing message is in text format and being delivered via the SMS protocol, sending the message at midnight may be more socially-appropriate. Delay reply/delay send may also take into account the time zone of the recipient and choose a more socially-appropriate delivery time for a message based on the recipient’s local time.

Finally, moving down to row **312**, the ‘grayed-out’ characteristic of the row may be used to indicate that there are no remaining unread/unopened messages of any format or protocol type remaining from a particular sender. Alternately, each message type may be individually grayed out, indicating that there are no new messages of a particular type. It is to be understood that the use of a grayed out row is merely exemplary, and that any number of visual indicators may be used to inform the user of the device that no unread messages remain.

As may now be appreciated, the multi-protocol, person-centric, multi-format inbox feed **300** of FIG. **3A** may provide various potential benefits to users of such a system, including: presenting email, text, voice, video, and social messages all grouped/categorized by contact (i.e., ‘person-centric,’ and not subject-people-centric, subject-centric, or format-centric); providing several potential filtering options to allow for traditional sorting of communications (e.g., an ‘email format’ view for displaying only emails); and displaying such information in a screen-optimized feed format. Importantly, centralization of messages by contact may be employed to better help users manage the volume of incoming messages in any format and to save precious screen space on mobile devices (e.g., such a display has empirically been found to be up to six to seven times more efficient than a traditional inbox format). Further, such an inbox feed makes it easier for a user to delete unwanted messages or groups of messages (e.g., spam or graymail). The order of appearance in the inbox feed may be customized as well. The inbox feed may default to showing the most recent messages at the top of the feed. Alternatively, the inbox feed may be configured to bring messages from certain identified “VIPs” to the top of the inbox feed as soon as any message is received from such a VIP in any format and/or via any protocol. The inbox feed may also alert the user, e.g., if an email, voice message, and text have all been received in the last ten minutes from the same person—likely indicating that the person has an urgent message for the user. The inbox feed may also identify which companies particular senders are associated with and then organize the inbox feed, e.g., by grouping all communications from particular companies together.

In other embodiments, users may also select their preferred delivery method for incoming messages of all types. For example, they can choose to receive their email messages in voice format or voice messages in text, etc.

Referring now to FIG. **3B**, an example of a multi-protocol, multi-format inbox feed for messages to and from a particular user **320** is shown, according to one or more disclosed embodiments. As is shown across the top row of the interface **322**, the messages from a particular user, in this case ‘Peter Ehrmanntraut’ may be displayed in a single multi-format, multi-protocol message feed. Row **322** in the example of FIG. **3B** also presents the user with the opportunity to select the particular sender’s ‘Messages,’ ‘Profile,’ or ‘Vault’ storage, which is a document repository of files shared between the user and a particular sender (e.g., email attachments, MMS, etc.). As shown in FIG. **3B**, the pictorial

icon **324** and name of the sender whose messages are listed in interface **320** appear at the top of the communications page. Also present in interface **320** is search icon **326**, which may be activated to search across all message formats and protocols (e.g., including voice and video messages) from a particular sender for a particular search term(s) or topic. Message items may also be sorted in the feed by various characteristics such as time of receipt, format, or other content and/or semantic-based ranking schemes. Moving down to the messages portion of interface **320**, checkbox **328** represents the first email message received from user Peter Ehrmanntraut, whereas checkbox **330** represents the first new video message from user Peter Ehrmanntraut. Finally, grayed-out checkbox **332** represents an aggregation of voice messages that have already been listened to by the user.

Referring now to FIG. 3C, an example of a preview pane **340** for a multi-protocol, multi-format inbox feed for messages to and from a particular user is shown, according to one or more disclosed embodiments. As is displayed in FIG. 3C, the message associated with checkbox **328** has been opened to provide a more in-depth preview of the associated email text. According to some embodiments, the recipients **342** are listed out above the body **344** of the email, and a link **346** may be activated that causes the application to retrieve the full email message from either the system's sync server or third party email servers. The interface may also provide a number of preview quick action buttons **348** to be performed on the message that is being previewed, e.g., reply, reply all, forward, delete, etc.

Referring now to FIG. 3D, an example of a document repository page **380** for a particular user is shown, according to one or more disclosed embodiments. Row **382** in the example of FIG. 3D presents the user with the opportunity to select the particular sender's 'Vault' page, which is a document repository of files shared between user and the particular sender (e.g., email attachments, MMS, etc.). As with the messages interface, a searching functionality **384** may be provided, which searches the documents associated with the particular user's Vault. A user's Vault may include multimedia files **386**, such as photos, in addition to other files **388**, such as word processing and presentation documents.

#### Multi-Protocol, Multi-Format Communication Composition User Interface

Referring now to FIG. 3E, an example of a multi-protocol, multi-format communication composition user interface **390** is shown, according to one or more disclosed embodiments. The top row of interface **390** in the example of FIG. 3E presents the user with several options related to the composition of a given communication. For instance, icon **392** may provide the user with the ability to geo-tag his or her location onto the message being sent. Icon **393** may be used to indicate that a message has a special status, such as a 'poll question' or other 'request for recommendation' with a response requested by the sender. Such special status messages may optionally be sent to 'tiers' of contacts (e.g., first-tier relationship, second-tier relationships, etc.) or even the general public, as opposed to particular contacts. Icon **394** may be used to attach one or more file attachments to the message being composed, button **399** may be used to cancel the message being composed, and button **395** may be used to send off the message to the one or more recipients specified in the "To:" field **391**.

Message box **396** may be used by the user to enter his or her message any desired communications format or protocol that the system is capable of handling. For example, a text

message may be entered by activating icon **397** and using an on-screen keyboard or the like. Alternately, an audio message or a video message may be recorded by activating the other icons across the top row of message box **396**. Once the message has been composed in the desired format, the user may utilize the row of icons **399** across the bottom of message box **396** to select the desired delivery protocol for the outgoing communication. As shown in FIG. 3E, those protocols may include, e.g., email, SMS/MMS/IM, or Optimal. As may be understood, the selection of desired delivery protocol may necessitate a conversion of the format of the composed message. For example, if a message is entered in audio format, but is to be sent out in a text format, such as via the SMS protocol, the audio from the message would be digitized, analyzed, and converted to text format before sending via SMS (i.e., a speech-to-text conversion). Likewise, if a message is entered in textual format, but is to be sent in voice format, the text from the message will need to be run through a text-to-speech conversion program so that an audio recording of the entered text may be sent to the desired recipients in the selected voice format via the appropriate protocol, e.g., via an email message.

The selection of the "Optimal" delivery option may have several possible implementations. The selection of output message format and protocol may be based on, e.g., the format of the incoming communication, the preferred format or protocol of the recipient and/or sender of the communication (e.g., if the recipient is an 'on-network' user who has set up a user profile specifying preferred communications formats and/or protocols), an optimal format or protocol for a given communication session/message (e.g., if the recipient is in an area with a poor service signal, lower bit-rate communication formats, such as text, may be favored over higher bit-rate communications formats, such as video or voice), a generated context, and/or economic considerations of format/protocol choice to the recipient and/or sender (e.g., if SMS messages would charge the recipient an additional fee from his or her provider, other protocols, such as email, may be chosen instead).

Other considerations may also go into the determination of an optimal delivery option, such as analysis of recent communication volume, analysis of past communication patterns with a particular recipient, analysis of recipient calendar entries, and/or geo-position analysis. Other embodiments of the system may employ a 'content-based' determination of delivery format and/or protocol. For example, if an outgoing message is recorded as a video message, SMS may be de-prioritized as a sending protocol, given that text is not an ideal protocol for transmitting video content. Further, natural language processing (NLP) techniques may be employed to determine the overall nature of the message (e.g., a condolence note) and, thereby, assess an appropriate delivery format and/or protocol. For example, the system may determine that a condolence note should not be sent via SMS, but rather translated into email or converted into a voice message. Thus, the techniques disclosed herein allow communications systems to become 'message-centric' or 'people-centric' as opposed to 'protocol-centric,' eventually allowing consideration of message protocol to fall away entirely for the sender of the communication.

Another beneficial aspect of the multi-protocol, multi-format communication composition system described herein is the ability to allow the user to send one message to the same recipient in multiple formats and/or via multiple protocols at the same time (or, with certain formats/protocols, in a time-delayed fashion). Likewise, the multi-protocol, multi-format communication composition system also

## 11

allows the user the ability to send one message to multiple recipients in multiple formats and/or via multiple protocols. The choice of format/protocol for the outgoing message may be made by either the system (i.e., programmatically) or by the user, e.g., by selecting the desired formats/protocols via the user interface of the multi-protocol, multi-format communication composition system.

FIG. 4 shows a flowchart 400 of one embodiment of a method for populating a multi-protocol, person-centric, multi-format inbox feed, according to one or more disclosed embodiments. First, the system may prompt the user to input his or her credentials so that he or she may be authenticated and authorized (Step 405). Next, the sync server 105 and/or third-party servers 106-109 may verify and validate the user's credentials as being authorized to receive communications associated with a particular account(s) tied to a particular messaging service(s) (Step 410). Next, the user's credentials are encrypted and stored at the sync server 105 so that the user's messages may continue to be retrieved by the system (Step 415). Once the user's credentials have been verified and stored, the system may attempt to synchronize the user's multi-protocol, person-centric, multi-format unified messaging inbox feed with the various external communication servers hosting the user's messages from the various third-party messaging services, e.g., by using one or more third-party credentials of the first user stored at the sync server (Step 420). Next, the system may receive a query from a particular user's client device (e.g., to pull new communications directed to the user) and determine that the client device has access to perform the query (Step 425). Assuming the client device has access, the query will be executed, and the results will be retrieved and optionally reformatted, ranked, etc., according to the user's and/or system's preferences (Step 430). One example of a formatted and sorted query result set is shown in the exemplary user interface of FIG. 3A.

When the user desires to transmit a user-generated message, e.g., via the exemplary user interface of FIG. 3E, the process may resume at Step 435 by the client device transmitting the user-generated message either to the system's sync server or directly to the third-party communications servers. At that point, it may again be verified that the client device has access to send the message(s) (Step 440). If the client device does not have access, the user will again be prompted to enter his or her authentication credentials (Step 445). Once proper authentication has been established, the transmission of the user-generated message may be completed via the designated protocol(s). The nature and type of the protocols may be determined, e.g., in accordance with one or more of the various rules and preferences discussed above with reference to FIG. 3E.

#### User Interface-Driven Search Query Generation

FIG. 5 shows a flowchart 500 of one embodiment of a method for processing a user interface-driven query, according to one or more disclosed embodiments. First, a client device may send a query to a central communications system server, such as sync server 105, based on the status of the currently-displayed user interface (UI) on the client device (Step 505). For example, with respect to the user interface 300 shown in FIG. 3A, the selection of a row in the currently-displayed UI for sender 'Emma Poter' could be associated with one or more system-defined "tags" that would be used by the system to generate a query for messages from user 'Emma Poter.' Likewise, changing the UI to the 'Video' tab in row 302 of user interface 300 would generate a query for only messages in a video format, etc. Next, the system may determine if there are cached results

## 12

for the query that the client device is currently trying to send (Step 510). If there are cached results at Step 510, the query may be limited to events occurring since the last identical query was issued by the client device (Step 515), and then the limited query may be executed by the central communication system server (Step 520). If there are no cached results at Step 510, then the full query may simply be executed by the central communication system server (Step 520).

After some amount of time, the client device may poll the inbox feed application to determine whether there is a new UI displaying on the client device (Step 525). If there is a new UI being displayed on the client device, the process 500 may return to Step 505 so that the client application may create and send a new query to the central communications system server based on the currently-displayed UI. If, instead, there is not a new UI being displayed on the client device, the client application may determine whether a given time interval, *t*, has passed since the last query that was sent to the central communications system server (Step 530). If the time interval, *t*, has not passed since the last time the UI was updated, the client application may simply return to Step 525 and continue to poll the inbox feed application to determine whether there is a new UI displaying on the client device. If, instead, the time interval, *t*, has passed since the last time the UI was updated, the client application may simply return to Step 505 so that the client application may create and send a new query to the central communications system server based on the currently-displayed UI. It is to be understood that the exemplary method shown in flowchart 500 may also be achieved by use of a "push-based" system, too, wherein the inbox feed application may push information to the client device periodically without the need for the client device to poll the server.

FIG. 6 shows a flowchart 600 of one embodiment of a method for creating a multi-protocol, multi-format communication transmission, according to one or more disclosed embodiments. First, the user interface of the client application may present the user with the capability to select any number of contacts from any source type (Step 605). Next, the user interface of the client application may present the user with the capability to select any composition format (Step 610). Next, the user interface of the client application may present the user with the capability to tag any desired attachments and/or geo-local data with the outgoing message (Step 615). Next, the user interface of the client application may present the user with the capability to select the desired communication delivery protocol (Step 620). Next, the user interface of the client application may present the user with the capability to reply/forward a given message in symmetric default format (i.e., the same format that the message was received in) or an alternative format (Step 625). Finally, the system may deliver the message to the selected recipient(s) in the selected/determined format(s). As described above in reference to FIG. 3E, the outgoing message format may be sent with or without delay, may have multiple degrees of accessibility, may be based on user preference, protocol optimization, and/or system defaults.

#### Intelligent Delayed Send

According to one or more embodiments described herein, the aforementioned "Optimal Decision Engine" does not require that all 'Participants' (sender(s) & receiver(s)) in any given message or conversation be registered users of the multi-format communication network described herein in order to provide the initial registered Sender (also known as the 'Composer') with an intelligent prediction as to the optimal delivery path/protocol for each recipient(s). The



Optimal Decision Engine could use information such as calendar information showing whether the recipient is in a meeting, recipient position or motion information (e.g., whether the recipient is driving, walking, sleeping, etc.), or historic communication patterns as a way to determine an optimal format or protocol for the communication(s).

In cases where a message or conversation contains more than one intended recipients, and the Sender activates the Optimal option, then the Optimal Decision Engine may be instantiated to evaluate the best delivery method of the given message for each recipient—as individual parties, as well as in a group-based relationship context. For example, based on historical patterns of communication, it can be determined by the Optimal Decision Engine that Sender frequently communicates with recipients A, B, and C, each using personal email addresses. In addition, the Optimal Decision Engine can leverage similar historical transaction patterns to assess that Sender frequently uses another protocol/address when communicating with recipients A, B, and C as a single group. Therefore, according to at least some embodiments (e.g., those based primarily on historic data patterns), the Optimal Decision Engine is able to provide intelligent, context-aware suggestions on preferred message delivery method with only Sender as a registered user of the multi-format, multi-protocol communication network.

When one Sender and at least one recipient are registered users of the multi-format, multi-protocol communications network, then, in addition to using traditional formats or protocols, messages sent through the multi-format, multi-protocol communication composition system described herein may also reach recipients via an established ‘on-network’ format or protocol.

Often a Sender wants to send a message to a recipient or a group of recipients, where the message is not delivered immediately. The Sender desires that the message is sent in manner where the recipients are in the best position to accept the message. Recipients may be “on-network” or “off-network.” “On-network” recipients may have a more robust experience, as their usage will provide more data points to be utilized to intelligently delay reception of sent messages. However, “off-network” recipients may realize the benefits of the intelligent delay sending system, as their history with “on-network” users can be factored into intelligent delay sending.

“On-network” embodiments may use a number of mechanisms to facilitate the intelligent delayed send. For example, a Sender may delay send a message to “pick up eggs” to a recipient. The user may specify that the message should be delayed until the recipient “leaves the office.” As both Sender and recipient are “on-network,” the message can be processed internally to the network, which data collected from both the Sender and the recipient.

In the embodiment of an “on-network” Sender with multiple “on-network” recipients, the intelligent delay send may utilize both Sender and recipient data points to deliver messages at appropriate times and appropriate places. For example, a Sender utilizes intelligent delay send for each recipient to receive when they “arrive home from work.” Utilizing geo-fencing from all of the “on-network” recipients, the intelligent delay send may deliver the Sender’s message to the each of the recipients when they come within a proximity of their homes. This indicates that the send takes place at multiple times, as each recipient may reach their homes at different times.

Similarly, an “on-network” Sender may intelligent delay send a message to multiple “on-network” recipients, with the requirement that all the recipients are at the workplace.

Distinguishing from the previous embodiment, each recipient simultaneously receives the message when each and every recipient has arrived at the workplace.

“Off-network” embodiments may utilize fewer mechanisms to achieve the intelligent delay send, however may still provide a new and useful experience. An “on-network” Sender may intelligent delay send a message to a recipient based on a context of the Sender. For example, the “on-network” Sender may tie the intelligent delay send of a message based on that user’s calendar appointments.

Additionally, in another embodiment, the “on-network” Sender may intelligent delay send a message with a 12 hour delay to an “off-network” recipient. In conjunction with the Optimal delivery option above, the intelligent delay send would deliver the message at the appropriate time, and via a protocol and digital formatting appropriate for the message.

The intelligent delay send may deliver messages “on-network” or “off-network” at time specified by the Sender or suggested by the system, and when used in conjunction with the Optimal delivery option, even across protocols.

As the determination of when to the send the Sender’s message may be dependent on the passage of time, if used in conjunction with an Optimal delivery option, the system may intelligently adapt the digital format as well as the delivery protocol to that which is the most “optimal” for each given recipient at the time of the sending, rather than time of composing.

FIG. 7 shows a block diagram 700 of one embodiment of a Universal Message Object (UMO), according to one or more disclosed embodiments. The block diagram 700 describes the relationship between various components of data required to make the optimal delivery option possible. It should be appreciated that the UMO facilitates not only the communication between ‘on-network’ and ‘off-network’ users, but also facilitates the backflow of updating relevant conversation histories based on the message format and communication protocol utilized.

Participant 710 objects represent an “on-network” or “off-network” user. Participant 710 objects correspond to any people identified in the traditional email format fields, e.g., “To,” “From,” “Cc,” and “Bcc.” However, the Participant 710 objects are not limited to this, as a Participant 710 may be any user engaged in the conversation, and is relational to the service being used as the underlying communication protocol.

Service Identifier 705 object represents the service utilized by a single Participant 710 object in the delivery of a format over a communication protocol. For each “To,” “From,” “Cc,” and “Bcc” associated with a message, there is a Participant 705 object containing a Service Identifier 705 indicating which service was used as the underlying format and communication protocol. The Service Identifier includes data related to the delivery of the message, including the type of the service and the address. In the case of an SMS text message, a Service Identifier 705 object would have the type of “SMS” and the address would be respective telephone number. The Service Identifier 705 object implies a format and communication protocol unique to that indicated service.

Message Unique 715 is the representation format and communication protocol specific format for a message. For every message sent using the Optimal delivery method, one or more Message Unique 715 objects may be instantiated. Message Unique 715 objects contain the format and communication protocol specific data gathered during the Optimal delivery process. For example, time stamps of sent and

received, based on the communication protocol are stored in the object. Additionally, in instances where the format and communication protocol are limited in some fundamental way, e.g. TWITTER® messages limited to 140 characters and SMS text message 160 character limit, it may be necessary to send multiple messages across these communication protocols to fully convey the Sender's intended message. For this purpose, multiple Message Unique **715** objects would be instantiated to track the transmitted content.

The Message Common **720** object is the message that an "on-network" user views in their inbox feed. For every user message sent, there are common components present in all formats and communication protocols. For efficiency, these common components are extracted and contained in one object. Because of this efficiency, there is one Message Common **720** object for every message sent by the Sender. For example, the Message Common **720** object may store the body of the message, as well as the time sent at the moment the Sender selects 'send,' not the 'sent time' as reported by the underlying communication protocol. This has the advantage of presenting one view to the Sender and recipient(s), while resolving minor discrepancies from the underlying communication protocol.

The Message Source **725** object is a representation of the Message Unique **715** object in a Javascript object notation (JSON) format. The Message Source **725** object has a one to one relationship with the Message Unique **715** object.

Message Group **730** object is a representative identifier that coordinates a Message Common **720** object. The purpose of a Message Group **730** object is to enable multi-protocol communication and establish a relationship between those messages. There is a one to one relationship between the Message Group **730** object and the Message Common **720** object.

As multiple multi-protocol communication messages are being represented in this data model, it enables the system to truly facilitate a multi-protocol multi-format communication system. The system tracks each conversation by the Message Group **730** object relating to the Message Common **720** and then all the individual Message Unique **715** objects that relate to the Message Common **720**.

FIG. **8** shows a flowchart **800** of one embodiment of a method of delivering a message to one or more recipients based on events specified by the sender, according to one or more disclosed embodiments.

The flow begins with the user selecting a condition for the sending of a message (**805**). The condition may be based on the user's location, the recipients' locations, time, recipient time zone, the sender's calendar, or the recipients' calendar. The condition may be a single condition or an aggregate of multiple conditions.

Next, the user sends the message with the intelligent delay send option enabled (Step **810**). This step may be independent of the protocol being used to deliver the message.

Next, the system iterates through each of the unprocessed recipients (Step **815**). Since each recipient, either "on-network" or "off-network", is unique, they are processed individually. While this process is depicted as a serial operation here, it may also be processed simultaneously, i.e., in parallel. Once there are no more recipients to be processed, the flowchart ends (Step **845**).

Next, the system evaluates the condition(s) specified by the user (Step **820**). Due to the time-based nature of intelligent delay send, the evaluation of the condition may need to be repeated. For example, if the intelligent delay send is scheduled for delivery far into the future, it may be neces-

sary to evaluate this condition more than once. In this instance, polling may be appropriate including the implementation of an interval based processing loop, which re-evaluates the optimal protocol over a set interval of time.

Additionally, in the case of multiple conditions in a parallel processing implementation, it may be necessary to evaluate conditions across multiple threads. This may be implementing using a signaling feature of a modern programming language. For example, C++11 offers slots and signals to handle signal processing across multiple threads.

Finally, the system transmits the digitally formatted message to recipient at appropriate time (Step **825**). Since multiple recipients are receiving the same message, potentially at different times, the system may utilize an event queue to store the messages until the appropriate delivery time.

FIG. **9** shows a flowchart **900** of one embodiment of a method of determining suggestive delivery time for a message to one or more recipients, according to one or more disclosed embodiments.

The flow begins with a user sending a message with a suggestive intelligent delay send option enabled (Step **905**). This step is independent of the protocol being used to deliver the message.

Next, the system analyzes the content of the message (Step **910**). Natural Language Processing (NLP) or other techniques may be used to extract details of the message. For example, time and date references may be extracted, as well as locational landmarks for purposes of determining recipient availability. In analyzing the message, keywords may be identified through the NLP and those keywords may then be indexed. The indexed keywords may then be stored in the memory of the system.

Next, the system iterates through each of the unprocessed recipients (Step **915**). Since each recipient, either "on-network" or "off-network", is unique, they are processed individually. While this process is depicted as a serial operation here, it may also be processed simultaneously, i.e., in parallel. Once there are no further recipients to be processed, the flowchart ends (Step **945**).

Next for each recipient, the system generates a context based on the analyzed content, the sender, the recipients and the sent format (Step **920**). The context involves the parsed content of the message itself either directly or through the indexed keywords, the user's calendar, the recipient's calendar if available, the user's preferences, the recipient's preferences, and historical communication data between the user and the recipient. The historical communication data may include but not be limited to the parsed content of any historical message between the user and the recipient, the time of day of any historical message between the user and the recipient, favored communication protocols (i.e. FACEBOOK® instant messages) between the user and the recipient, and favored communication devices between the user and the recipient. Additionally, the NLP communication history may provide insight missed by calendars. For example, past discussions about the user's flight arrival may be indexed, when a calendar appointment was not present.

As previously mentioned in relation to the NLP of the message content, since all historical communication passes through this flow, all historical communication has all been indexed and stored. Additionally, the user preferences and sender preferences are stored as well.

Next the content and context is applied for each recipient (Step **925**). This involves utilizing all the contextual data points that are indexed as inputs into a predictive time-based data model. In addition to indexed inputs, non-indexed real

17

time inputs may be included as well. Geo location data of the recipient is one such non-indexed real time input. Factorization machines, support vector machines, or other machine learning techniques may be utilized to effectuate the predictive time-based data model.

Next the appropriate time is selected based on the analysis of the content and context for each recipient (Step 930). Based on the input and predictive time-based data model, the system will select the appropriate time for transmitting the message to the user. Event triggers, particularly timers, can be utilized for transmitting time dependent messages. Additionally, with non-indexed real time input, this approach can be adapted to implement geo-fencing applications, as well.

Finally, the system transmits the digitally formatted message to recipient at appropriate time (Step 935). Since multiple recipients are receiving the same message, potentially at different times, the system may utilize an event queue to store the messages until the appropriate delivery time.

Similar to FIG. 9, FIG. 10 shows a flowchart 1000 of one embodiment of a method of determining a suggestive delivery time for a message to one or more recipients, however, unlike FIG. 9, in flowchart 1000 of FIG. 10, the Optimal delivery option is selected. Steps 1005-1030 correspond directly with Steps 905-930. With the Optimal delivery option enabled, instead of sending a message determined based on how the Sender originally composed it at the time or condition specified by the sender, the digital message is enqueued (Step 1035).

Once all recipients have been processed, the flow may then begin to process the enqueued messages. While there are still enqueued messages (Step 1040), the flow processing the following additional steps.

First, the selected send time from Step 1030 is evaluated (Step 1045). This is to determine if the correct time for the sending of the message has occurred. This can be the literal comparison of time date stamps, or the evaluation of a condition.

Next, similar to the generation of a context at Step 1020, the flow generates a context based on the analyzed content, the sender, the recipients, the time, and the sent format (Step 1050). A context, as described here, may include data related to the content but not necessarily included in the content. Context may include relational data extracted from previous messages as well. These variables may remain the same as those variables utilized at Step 1020 or they may change. For example, over time, content in the message, while textually remaining static, may have different contextual meaning.

The context generated in Step 1050 is then provided to an Optimal Decision Engine (Step 1055), as an input. The Optimal Decision Engine suggests a predictive optimal protocol for the context that was input.

Next, the flow involves applying formatting to the message for transmitting across the resultant protocol that the Optimal Decision Engine produced. More simply stated, the flow applies the resultant protocol from the Optimal Decision Engine to the message (Step 1060).

Finally, the flow transmits the message via the protocol from the Optimal Decision Engine (Step 1065). The flow loops until all enqueued messages are processed, and ends (Step 1070).

## EXAMPLES

The following examples pertain to further embodiments.

Example 1 is a non-transitory computer readable medium comprising computer instructions stored thereon to cause

18

one or more processing units to receive a first message in a digital format from a first user; receive a selection of one or more desired recipients for the first message; analyze, upon receiving the selection, a content of the first message; generate, upon receiving the selection, a context of the first message based, at least in part, on: the analyzed content of the first message, the selection of the one or more desired recipients, and the digital format of the first message; and for each of the one or more desired recipients: apply the context to a predictive time-based data model based, at least in part, on the respective desired recipient; select, based, at least in part, on the application of the context, a first time to send the first message to the respective desired recipient; and transmit, at the selected first time, the first message to the respective desired recipient.

Example 2 includes the subject matter of example 1, further comprising instructions to generate, upon passage of a first amount of time, an updated context of the first message based, at least in part, on: the content of the first message, the selection of the one or more desired recipients, the digital format of the first message, and the passage of the first amount of time.

Example 3 includes the subject matter of example 1, wherein the instructions to apply the context further comprise instructions to evaluate at least one of the following: the respective desired recipient's frequency of usage of a first communication protocol, a time of day of past communications between the first user and the respective desired recipient, historic rates of response of the respective desired recipient, a pattern of group communications including the respective desired recipient, and a current time.

Example 4 includes the subject matter of example 1, wherein the instructions to transmit the first message further comprise instructions to transmit the first message to a first desired recipient using a first protocol operable to carry the digital format of the first message.

Example 5 includes the subject matter of example 1, wherein the instructions to analyze the content of the first message further comprise instructions to convert an audio or video message to a text message.

Example 6 includes the subject matter of example 1, wherein the first time comprises a determined optimal time for the respective desired recipient to receive the first message.

Example 7 includes the subject matter of example 4, wherein the instructions to transmit the first message further comprise instructions to convert the first message to a second digital format and transmit the first message to a second desired recipient using a second protocol operable to carry the second digital format.

Example 8 is a computer-implemented method for selecting optimal transmission of digital messages comprising: receiving a first message in a digital format from a first user; receiving a selection of one or more desired recipients for the first message; analyzing, upon receiving the selection, a content of the first message; generating, upon receiving the selection, a context of the first message, at least in part, on: the analyzed content of the first message, the selection of one or more desired recipients and the digital format; and for each of the one or more desired recipients: applying the context to a predictive time-based data model, at least in part, to the desired recipient; selecting, based, at least in part, on the application the context, a first time to send the first message to the respective desired recipient; and transmitting, at the selected first time, the first message to the desired recipient.

19

Example 9 includes the subject matter of example 8, further comprising generating, upon passage of a first amount of time, an updated context of the first message based, at least in part, on: the content of the first message, the selection of the one or more desired recipients, the digital format of the first message, and the passage of the first amount of time.

Example 10 includes the subject matter of example 8, wherein the applying the context further comprise evaluating at least one of the following: the respective desired recipient's frequency of usage of a first communication protocol, a time of day of past communications between the first user and the respective desired recipient, historic rates of response of the respective desired recipient, a pattern of group communications including the respective desired recipient, and a current time.

Example 11 includes the subject matter of example 8, wherein the transmitting the first message further comprises transmitting the first message to a first desired recipient using a first protocol operable to carry the digital format of the first message.

Example 12 includes the subject matter of example 8, wherein the analyzing the content of the first message further comprises converting an audio or video message to a text message.

Example 13 includes the subject matter of example 8 wherein the first time comprises a determined optimal time for the respective desired recipient to receive the first message.

Example 14 includes the subject matter of example 8, wherein the transmitting the first message further comprises converting the first message to a second digital format and transmitting the first message to a second desired recipient using a second protocol operable to carry the second digital format.

Example 15 is an apparatus comprising: a memory; one or more processing units, communicatively coupled to the memory wherein the memory stores instructions to configure the one or more processing units to receive a first message in a digital format from a first user; receive a selection of one or more desired recipients for the first message; analyze, upon receiving the selection, a content of the first message; generate, upon receiving the selection, a context of the first message based, at least in part, on: the analyzed content of the first message, the selection of the one or more desired recipients, and the digital format of the first message; and for each of the one or more desired recipients: apply the context to a predictive time-based data model based, at least in part, on the respective desired recipient; select, based, at least in part, on the application of the context, a first time to send the first message to the respective desired recipient; and transmit, at the selected first time, the first message to the respective desired recipient.

Example 16 includes the subject matter of example 15, further comprising instructions to generate, upon passage of a first amount of time, an updated context of the first message based, at least in part, on: the content of the first message, the selection of the one or more desired recipients, the digital format of the first message, and the passage of the first amount of time.

Example 17 includes the subject matter of example 15, wherein the instructions to apply the context further comprise instructions to evaluate at least one of the following: the respective desired recipient's frequency of usage of a first communication protocol, a time of day of past communications between the first user and the respective desired recipient, historic rates of response of the respective desired

20

recipient, a pattern of group communications including the respective desired recipient, and a current time.

Example 18 includes the subject matter of example 15, wherein the instructions to transmit the first message further comprise instructions to transmit the first message to a first desired recipient using a first protocol operable to carry the digital format of the first message.

Example 19 includes the subject matter of example 15, wherein the instructions to analyze the content of the first message further comprise instructions to convert an audio or video message to a text message.

Example 20 includes the subject matter of example 15, wherein the first time comprises a determined optimal time for the respective desired recipient to receive the first message.

Example 21 includes the subject matter of example 15, wherein the instructions to transmit the first message further comprise instructions to convert the first message to a second digital format and transmit the first message to a second desired recipient using a second protocol operable to carry the second digital format.

In the foregoing description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the disclosed embodiments. It will be apparent, however, to one skilled in the art that the disclosed embodiments may be practiced without these specific details. In other instances, structure and devices are shown in block diagram form in order to avoid obscuring the disclosed embodiments. References to numbers without subscripts or suffixes are understood to reference all instance of subscripts and suffixes corresponding to the referenced number. Moreover, the language used in this disclosure has been principally selected for readability and instructional purposes, and may not have been selected to delineate or circumscribe the inventive subject matter, resort to the claims being necessary to determine such inventive subject matter. Reference in the specification to "one embodiment" or to "an embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiments is included in at least one disclosed embodiment, and multiple references to "one embodiment" or "an embodiment" should not be understood as necessarily all referring to the same embodiment.

It is also to be understood that the above description is intended to be illustrative, and not restrictive. For example, above-described embodiments may be used in combination with each other and illustrative process steps may be performed in an order different than shown. Many other embodiments will be apparent to those of skill in the art upon reviewing the above description. The scope of the invention therefore should be determined with reference to the appended claims, along with the full scope of equivalents to which such claims are entitled. In the appended claims, terms "including" and "in which" are used as plain-English equivalents of the respective terms "comprising" and "wherein."

What is claimed is:

1. A non-transitory computer readable medium comprising computer instructions stored thereon to cause one or more processing units to:

receive a first message in a digital format from a first user; receive a selection of desired recipients for the first message, wherein the desired recipients of the first message comprise at least a first desired recipient and a second desired recipient; analyze, upon receiving the selection, a content of the first message;

21

generate, upon receiving the selection, a context of the first message based, at least in part, on: the analyzed content of the first message, the selection of the desired recipients for the first message, and the digital format of the first message;

apply the context to a predictive time-based data model, wherein applying the context to the predictive time-based data model comprises evaluating each of the following: each desired recipient's frequency of usage of each of one or more communication protocols; a time of day of past communications between the first user and each desired recipient; and

historic rates of response of each desired recipient to the first user via each of the one or more communication protocols;

select, based, at least in part, on the application of the context, a first time and a first communication protocol from among the one or more communication protocols to use to send the first message to the first desired recipient, wherein the first communication protocol is operable to carry the digital format of the first message; transmit, at the selected first time, the first message to the respective desired recipient;

convert the first message to a second digital format; and transmit the first message to the second desired recipient using a second communication protocol that is operable to carry the second digital format.

2. The non-transitory computer readable medium of claim 1, further comprising instructions to generate, upon passage of a first amount of time, an updated context of the first message based, at least in part, on: the content of the first message, the selection of the desired recipients, the digital format of the first message, and the passage of the first amount of time.

3. The non-transitory computer readable medium of claim 1, wherein the instructions to apply the context further comprise instructions to evaluate: a pattern of group communications including the respective desired recipient.

4. The non-transitory computer readable medium of claim 1, wherein the instructions further comprise instructions to: select, based, at least in part, on the application of the context, a second time to send the first message to the second desired recipient.

5. The non-transitory computer readable medium of claim 1, wherein the instructions to analyze the content of the first message further comprise instructions to convert an audio or video message to a text message.

6. The non-transitory computer readable medium of claim 1, wherein the first time comprises a determined optimal time for the respective desired recipient to receive the first message, based on the predictive time-based data model.

7. The non-transitory computer readable medium of claim 4, wherein the instructions to transmit the first message to the second desired recipient are further configured to transmit the first message to the second desired recipient at the second time.

8. A computer-implemented method for selecting optimal transmission of digital messages comprising:

receiving a first message in a digital format from a first user;

receiving a selection of desired recipients for the first message, wherein the desired recipients of the first message comprise at least a first desired recipient and a second desired recipient;

analyzing, upon receiving the selection, a content of the first message;

22

generating, upon receiving the selection, a context of the first message, at least in part, on: the analyzed content of the first message, the selection of desired recipients for the first message, and the digital format;

applying the context to a predictive time-based data model, wherein applying the context to the predictive time-based data model comprises evaluating each of the following: each desired recipient's frequency of usage of each of one or more communication protocols; a time of day of past communications between the first user and each desired recipient; and historic rates of response of each desired recipient to the first user via each of the one or more communication protocols;

selecting, based, at least in part, on the application of the context, a first time and a first communication protocol from among the one or more communication protocols to use to send the first message to the first desired recipient, wherein the first communication protocol is operable to carry the digital format of the first message; transmitting, at the selected first time, the first message to the desired recipient;

converting the first message to a second digital format; and

transmitting the first message to the second desired recipient using a second communication protocol that is operable to carry the second digital format.

9. The computer-implemented method of claim 8, further comprising generating, upon passage of a first amount of time, an updated context of the first message based, at least in part, on: the content of the first message, the selection of the desired recipients, the digital format of the first message, and the passage of the first amount of time.

10. The computer-implemented method of claim 8, wherein the applying the context further comprise evaluating: a pattern of group communications including the respective desired recipient.

11. The computer-implemented method of claim 8, further comprising: selecting, based, at least in part, on the application of the context, a second time to send the first message to the second desired recipient.

12. The computer-implemented method of claim 8, wherein analyzing the content of the first message further comprises converting an audio or video message to a text message.

13. The computer-implemented method of claim 8, wherein the first time comprises a determined optimal time for the respective desired recipient to receive the first message, based on the predictive time-based data model.

14. The computer-implemented method of claim 8, wherein the transmitting the first message to a second desired recipient further comprises transmitting the first message to the second desired recipient at the second time.

15. An apparatus comprising:

a memory;

one or more processing units, communicatively coupled to the memory wherein the memory stores instructions to configure the one or more processing units to:

receive a first message in a digital format from a first user; receive a selection of desired recipients for the first message, wherein the desired recipients of the first message comprise at least a first desired recipient and a second desired recipient;

analyze, upon receiving the selection, a content of the first message;

generate, upon receiving the selection, a context of the first message based, at least in part, on: the analyzed

23

content of the first message, the selection of desired recipients for the first message, and the digital format of the first message;

apply the context to a predictive time-based data model, wherein applying the context to the predictive time-based data model comprises evaluating each of the following: each desired recipient's frequency of usage of each of one or more communication protocols; a time of day of past communications between the first user and each desired recipient; and historic rates of response of each desired recipient to the first user via each of the one or more communication protocols;

select, based, at least in part, on the application of the context, a first time and a first communication protocol from among the one or more communication protocols to use to send the first message to the first desired recipient, wherein the first communication protocol is operable to carry the digital format of the first message;

transmit, at the selected first time, the first message to the respective desired recipient;

convert the first message to a second digital format; and transmit the first message to the second desired recipient using a second communication protocol that is operable to carry the second digital format.

16. The apparatus of claim 15, further comprising instructions to generate, upon passage of a first amount of time, an

24

updated context of the first message based, at least in part, on: the content of the first message, the selection of the desired recipients, the digital format of the first message, and the passage of the first amount of time.

17. The apparatus of claim 15, wherein the instructions to apply the context further comprise instructions to evaluate: a pattern of group communications including the respective desired recipient.

18. The apparatus of claim 15, wherein the instructions further comprise instructions to: select, based, at least in part, on the application of the context, a second time to send the first message to the second desired recipient.

19. The apparatus of claim 15, wherein the instructions to analyze the content of the first message further comprise instructions to convert an audio or video message to a text message.

20. The apparatus of claim 15, wherein the first time comprises a determined optimal time for the respective desired recipient to receive the first message, based on the predictive time-based data model.

21. The apparatus of claim 15, wherein the instructions to transmit the first message to a second desired recipient are further configured to transmit the first message to the second desired recipient at the second time.

\* \* \* \* \*