



US009843543B2

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

(10) **Patent No.:** **US 9,843,543 B2**
(45) **Date of Patent:** **Dec. 12, 2017**

(54) **APPARATUS AND METHOD FOR
MULTI-FORMAT AND MULTI-PROTOCOL
GROUP MESSAGING**

(56) **References Cited**

U.S. PATENT DOCUMENTS

(71) Applicant: **Entefy Inc.**, Palo Alto, CA (US)
(72) Inventors: **Alston Ghafourifar**, Los Altos Hills,
CA (US); **Philip Nathan Greenberg**,
San Diego, CA (US)
(73) Assignee: **Entefy Inc.**, Palo Alto, CA (US)

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

(Continued)

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

FOREIGN PATENT DOCUMENTS

WO	9931575	6/1999
WO	2013112570 A1	8/2013

(21) Appl. No.: **14/985,798**

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

Primary Examiner — Tauqir Hussain

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

(65) **Prior Publication Data**

US 2016/0119261 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)

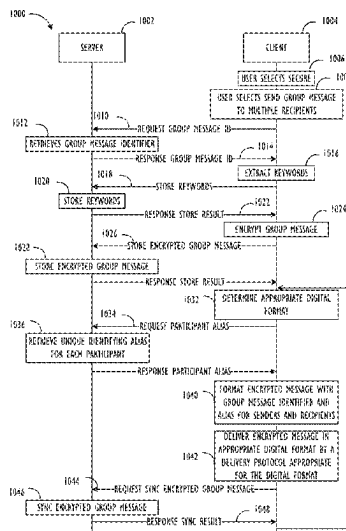
(52) **U.S. Cl.**
CPC **H04L 51/066** (2013.01); **H04L 51/36**
(2013.01)

(58) **Field of Classification Search**
CPC H04L 12/589; H04L 12/5835; H04L
12/5855; H04L 51/35; H04L 51/066;
H04L 51/14; H04M 2201/60; H04M
2203/2044; H04M 2203/4509; H04M
2203/4554; H04W 4/18; G06F 3/04886
USPC 709/206
See application file for complete search history.

(57) **ABSTRACT**

This disclosure relates generally to 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, tablets, laptops, and the like, to send encrypted or unencrypted group communications in a number of pre-determined and/or 'determined-on-the-fly' communications formats and/or protocols. Correlation of messages comprising secure group communications across multi-format multi-protocol may be determined by tracking message groups and mapping aliases across protocols. 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 and security mechanisms to fall away entirely for the sender of the communication.

20 Claims, 16 Drawing Sheets



(56)

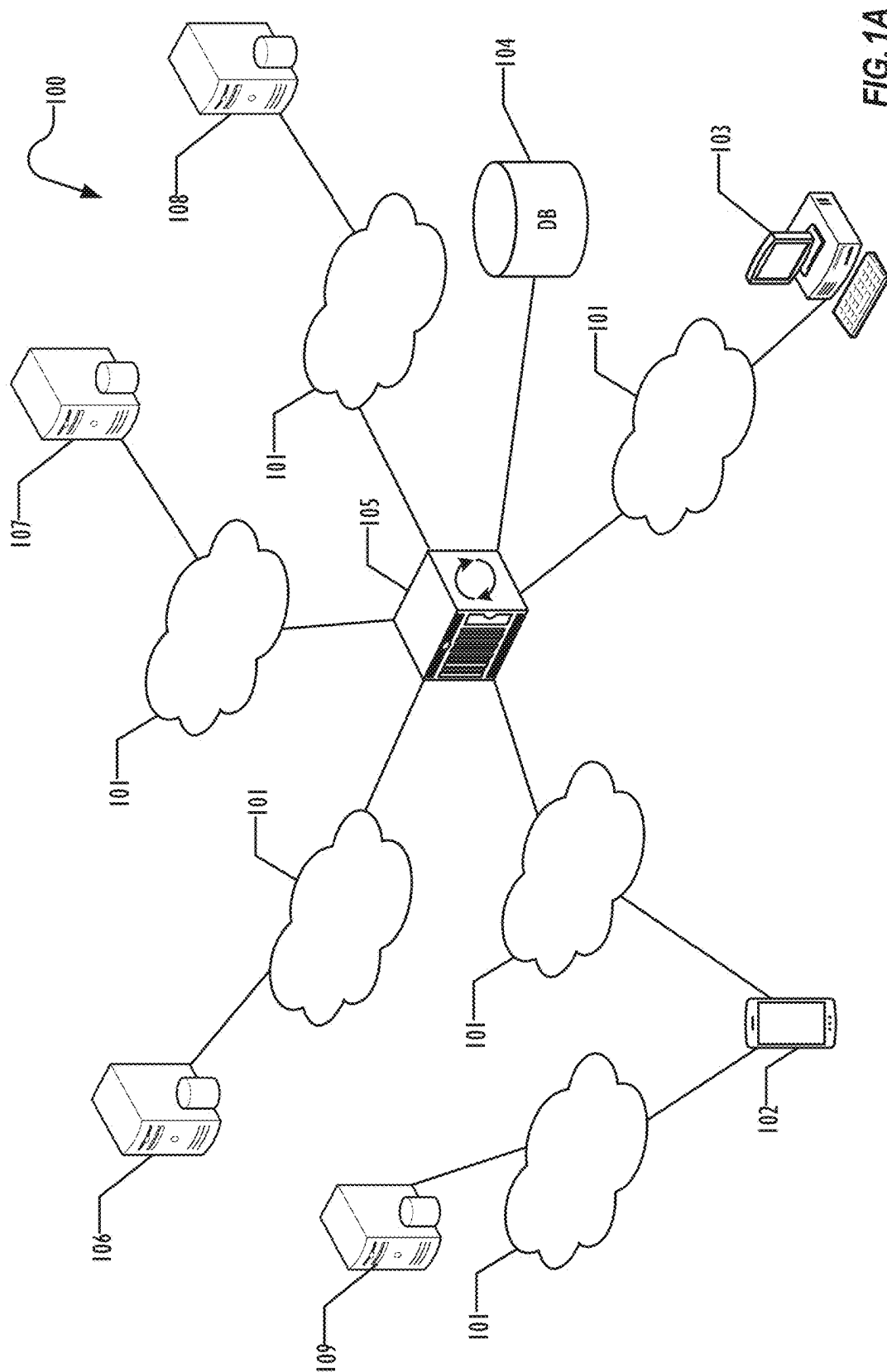
References Cited

U.S. PATENT DOCUMENTS

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	Galicja	
2005/0015443	A1 *	1/2005	Levine	H04L 12/5835 709/204
2006/0193450	A1	8/2006	Flynt	
2007/0130273	A1	6/2007	Huynh	
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	G06F 17/30867
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	H04L 12/581 709/206
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



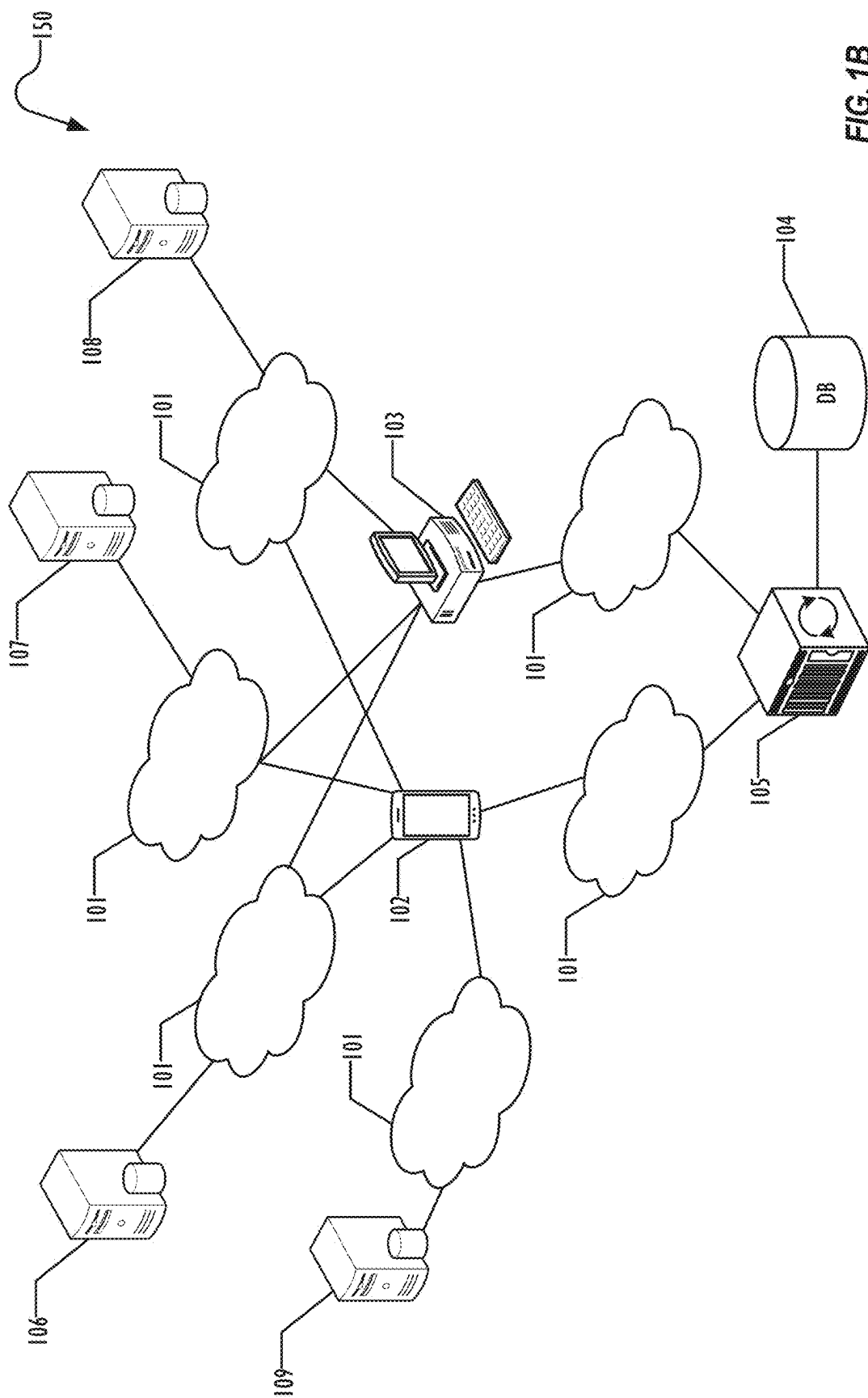


FIG. 1B

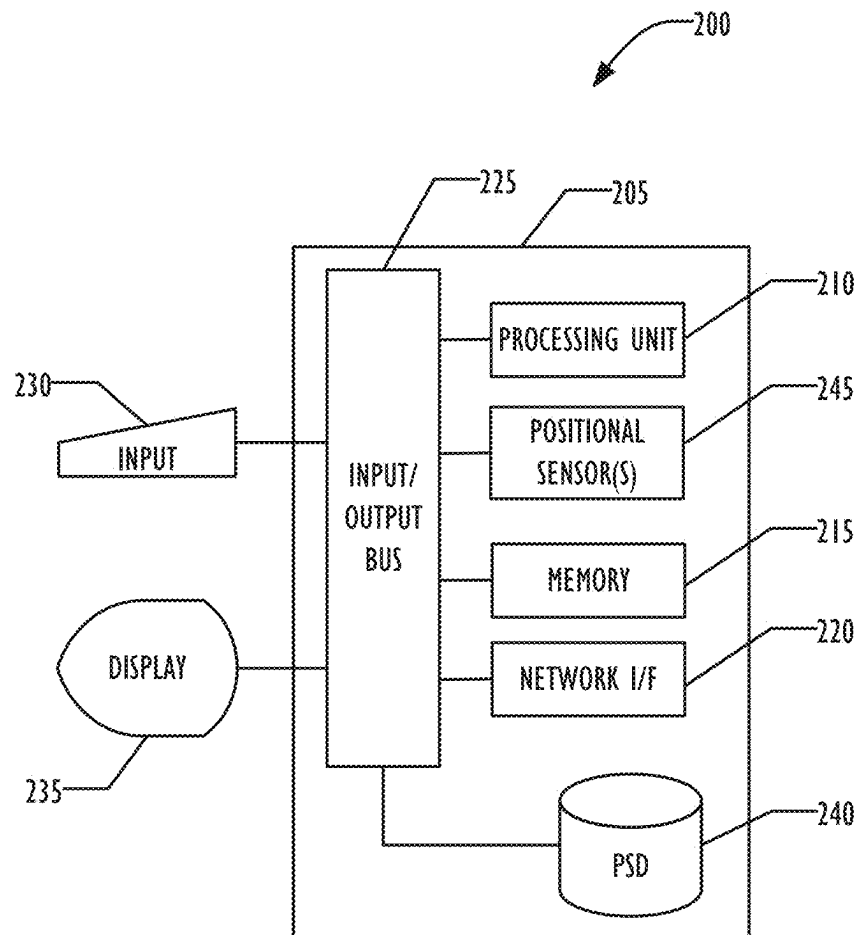


FIG. 2A

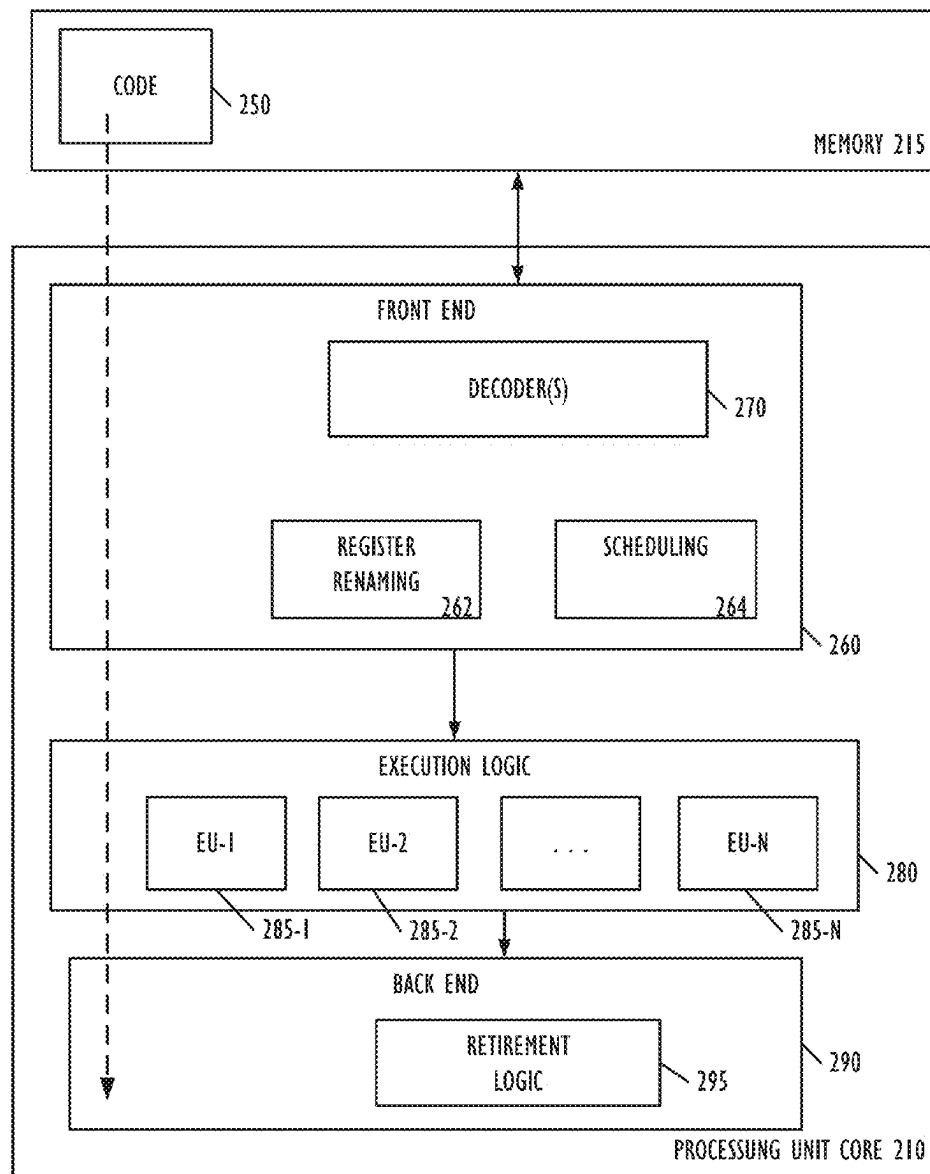


FIG. 2B

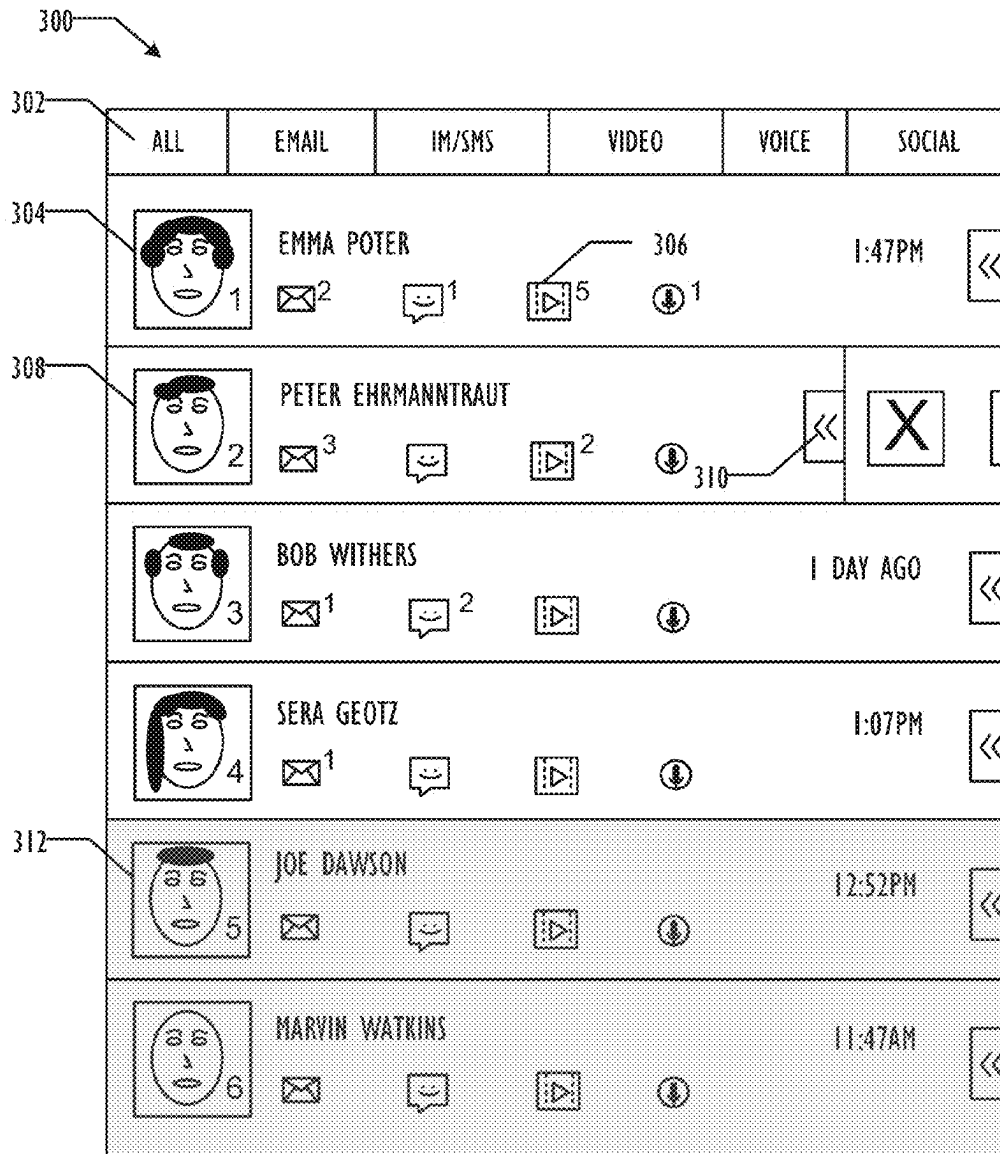


FIG. 3A

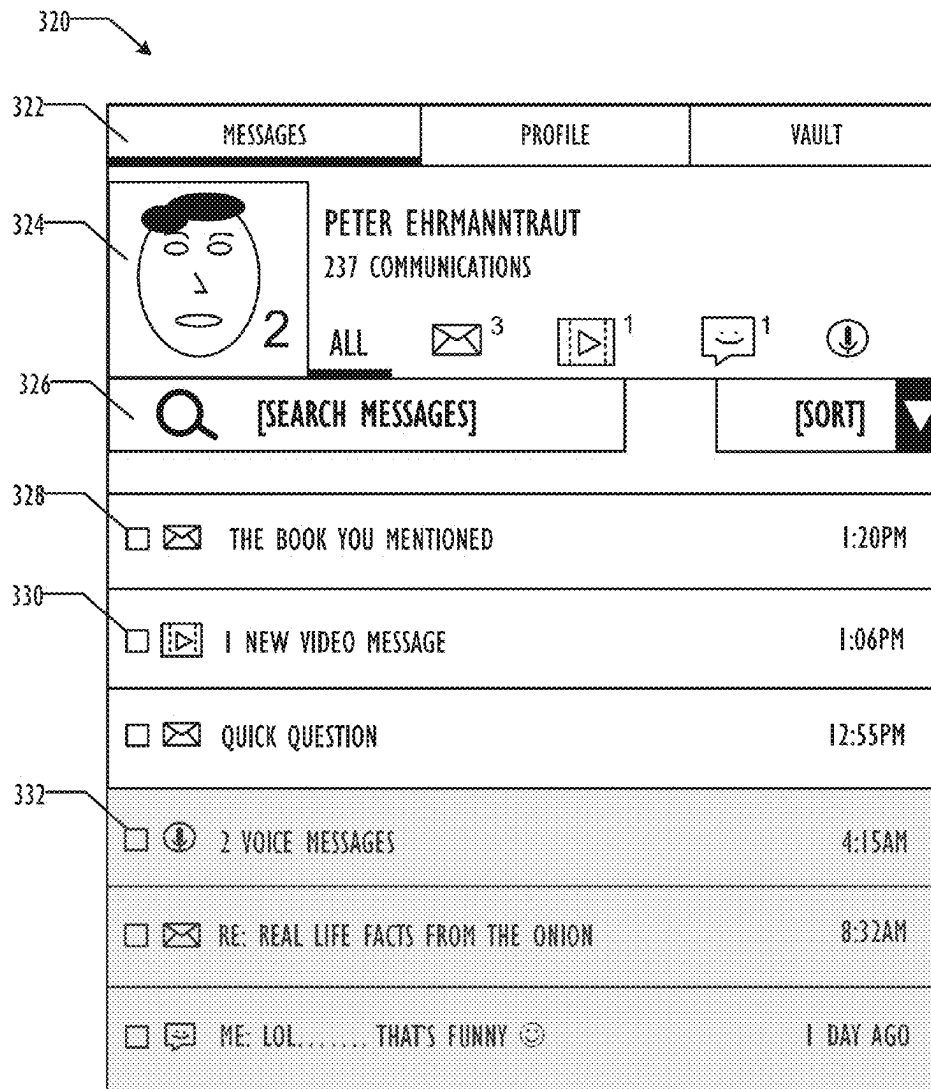


FIG. 3B

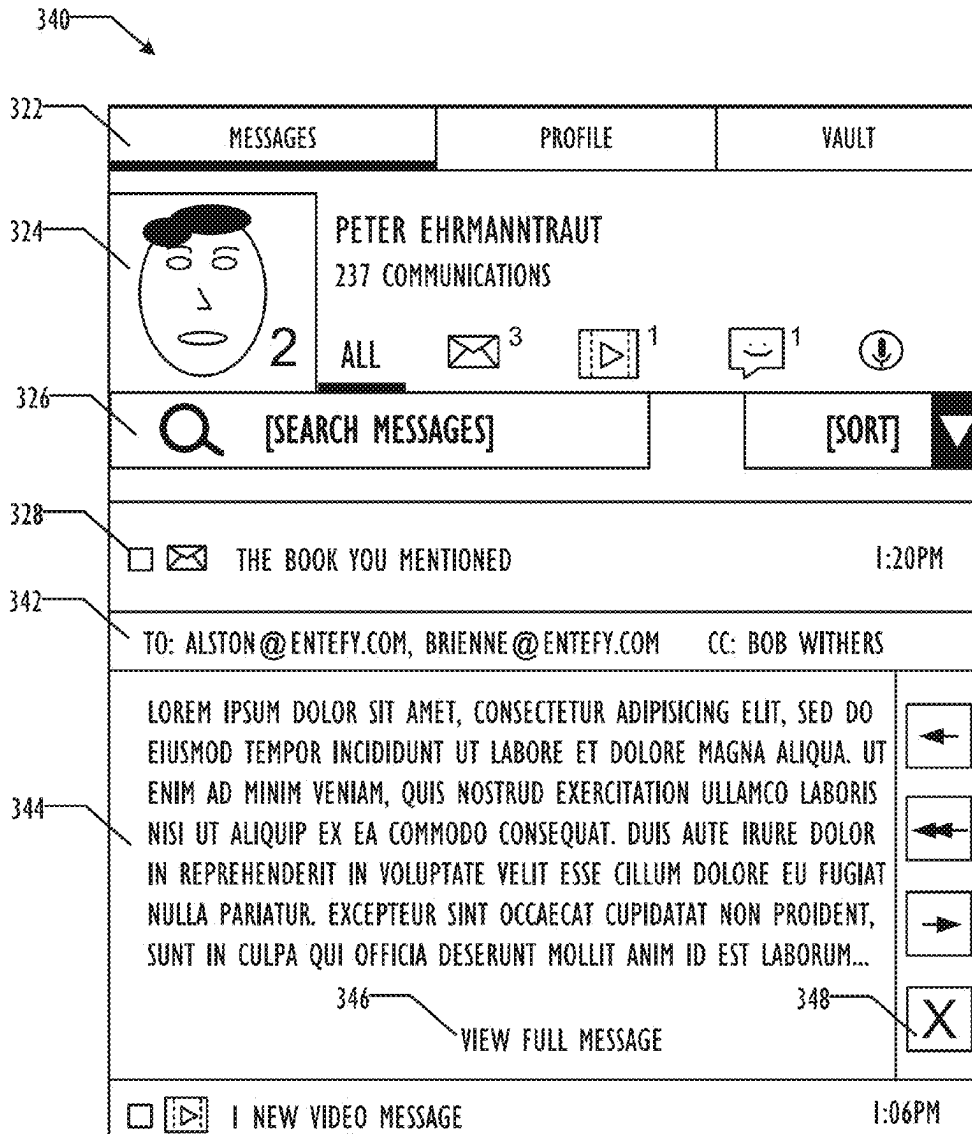


FIG. 3C

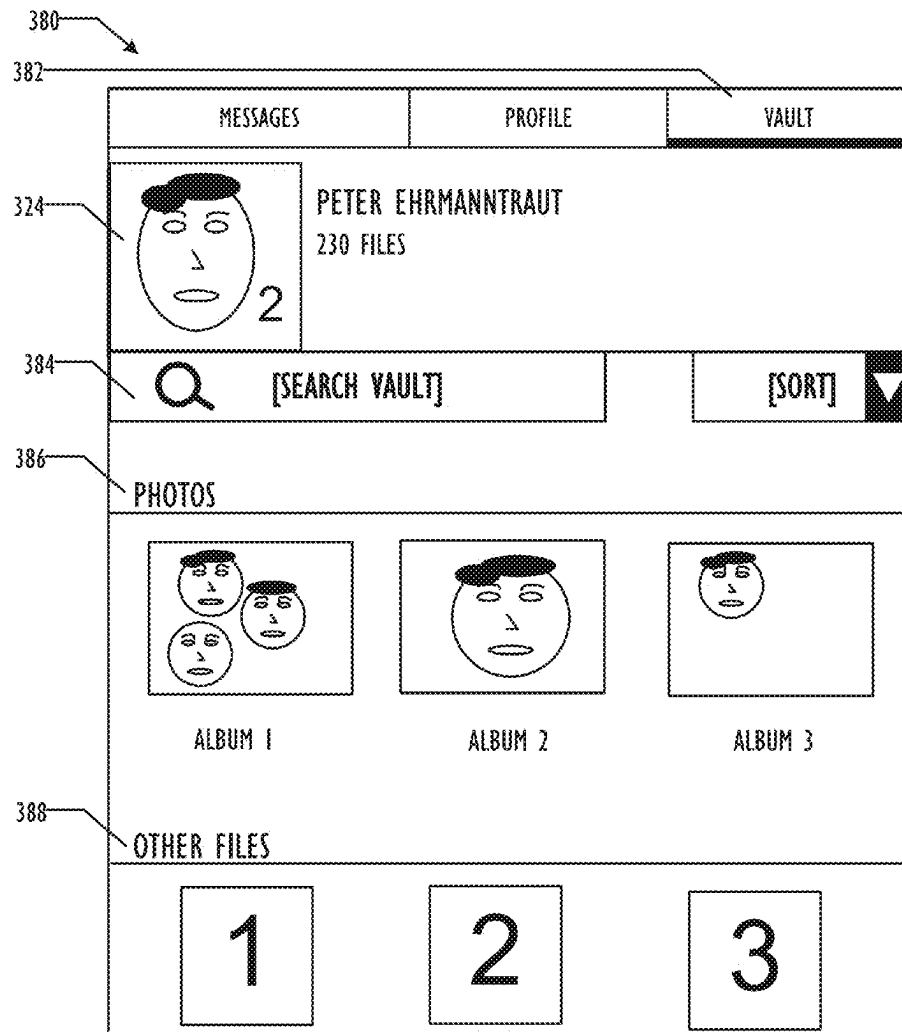


FIG. 3D

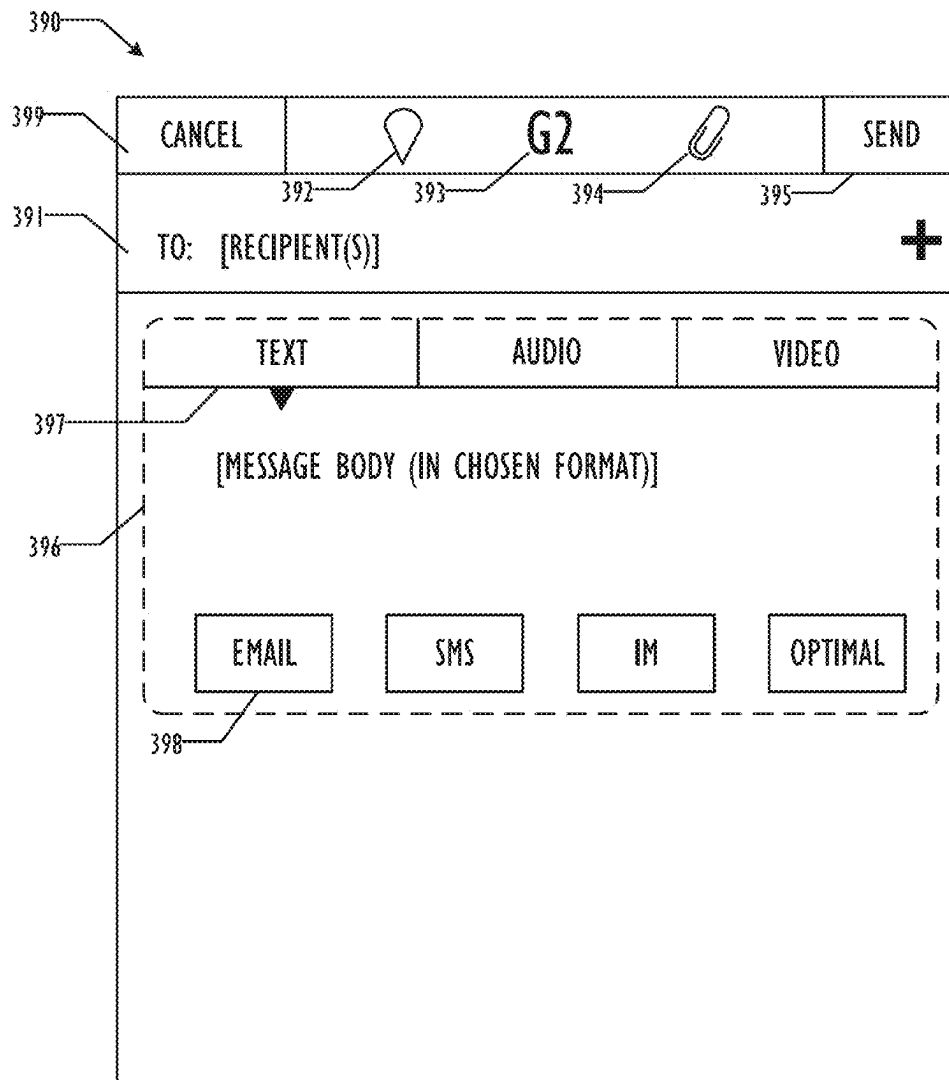


FIG. 3E

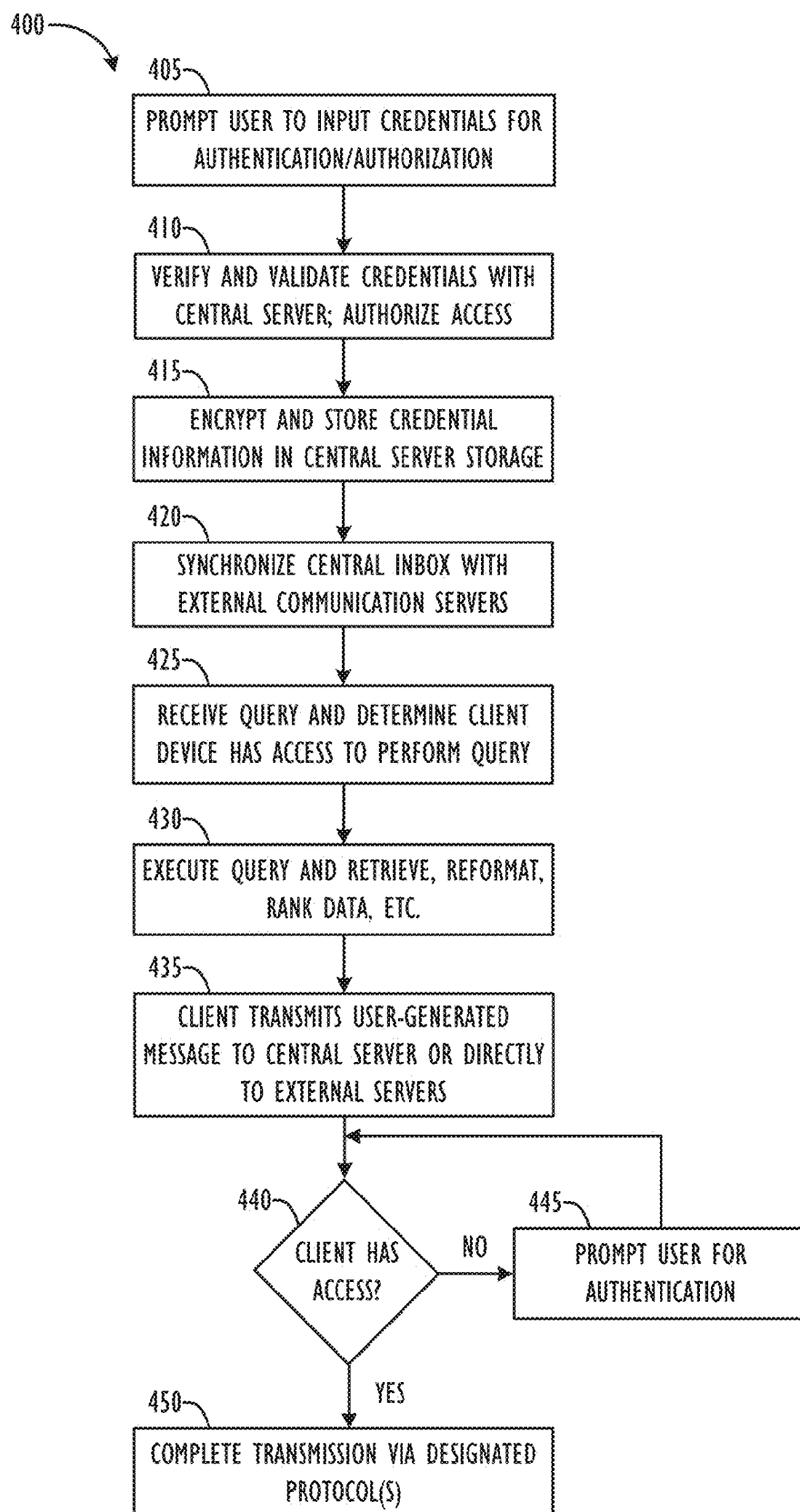


FIG. 4

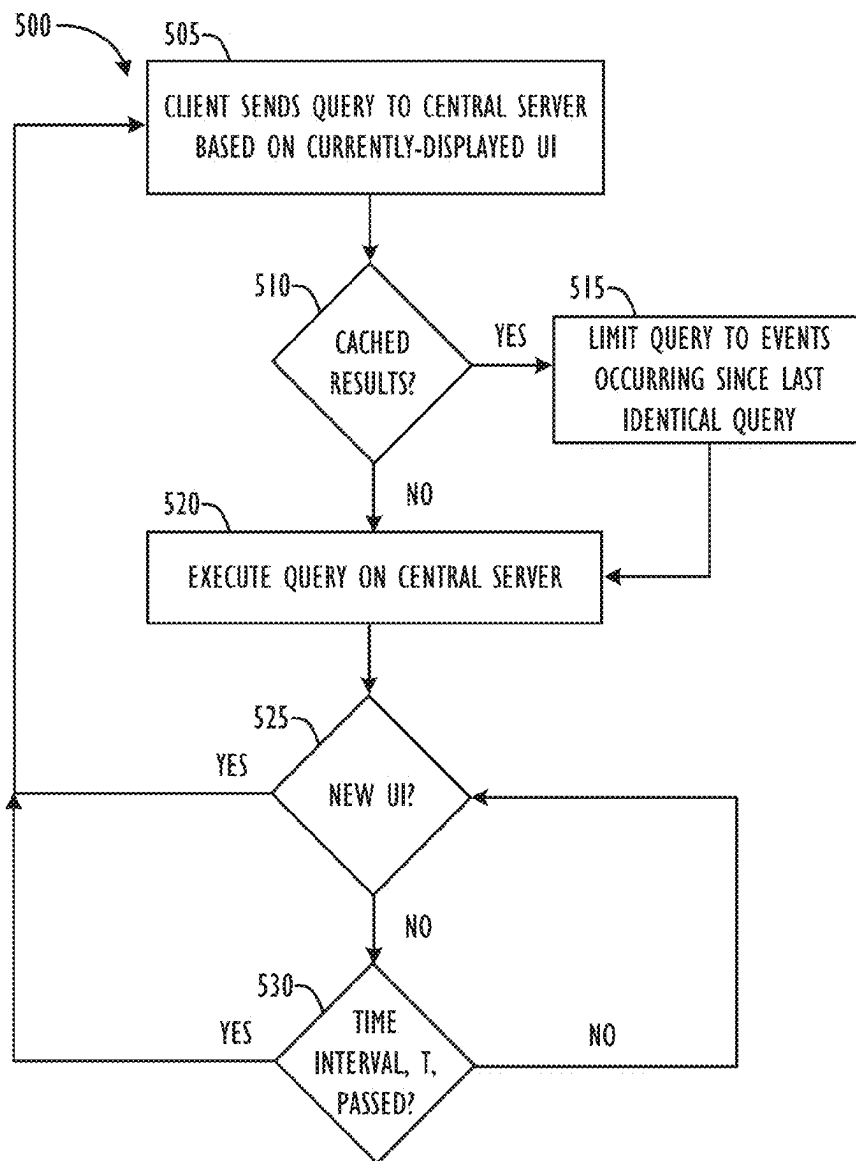


FIG. 5

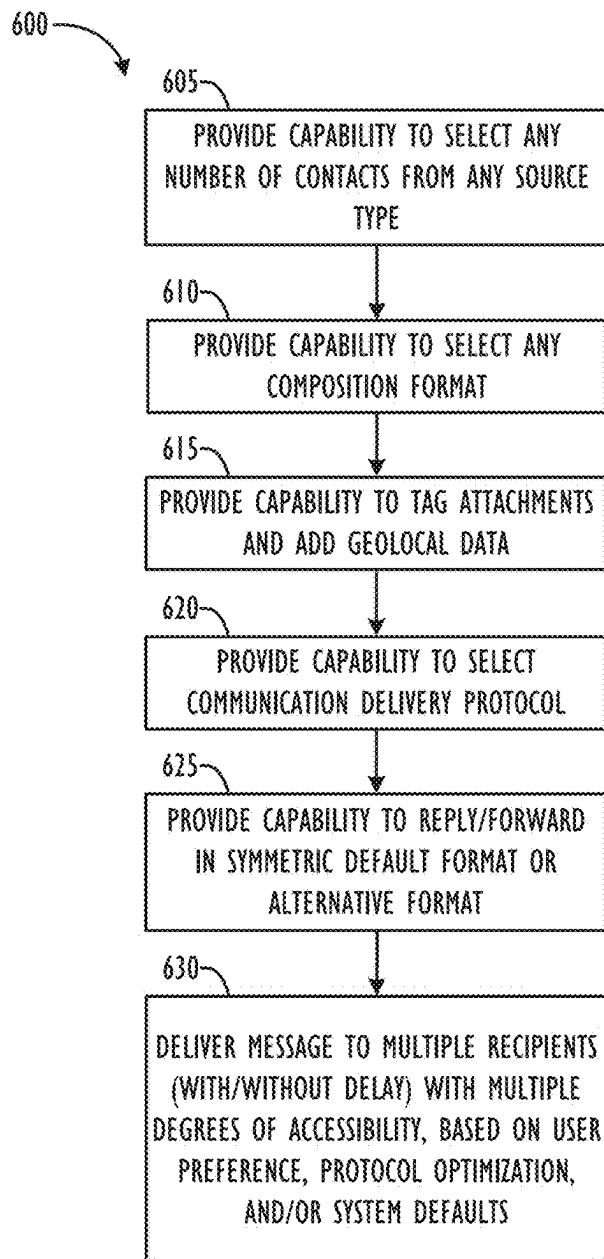


FIG. 6

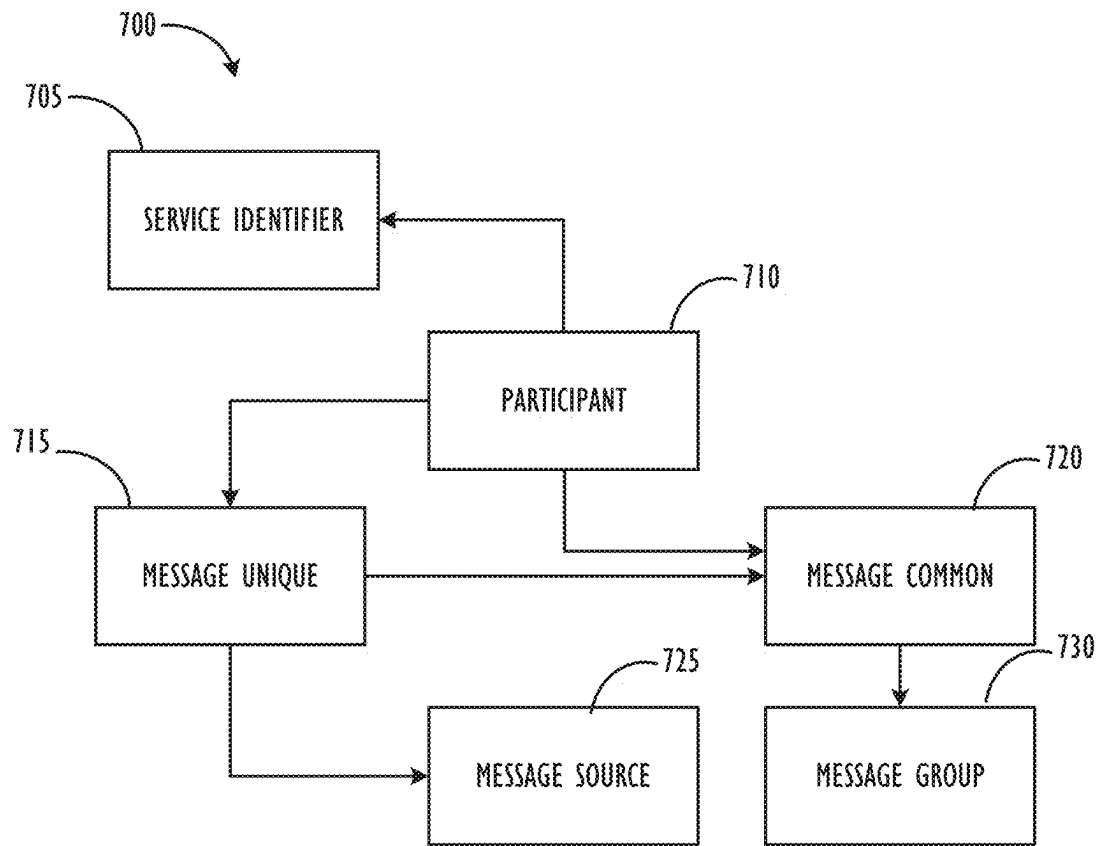


FIG. 7

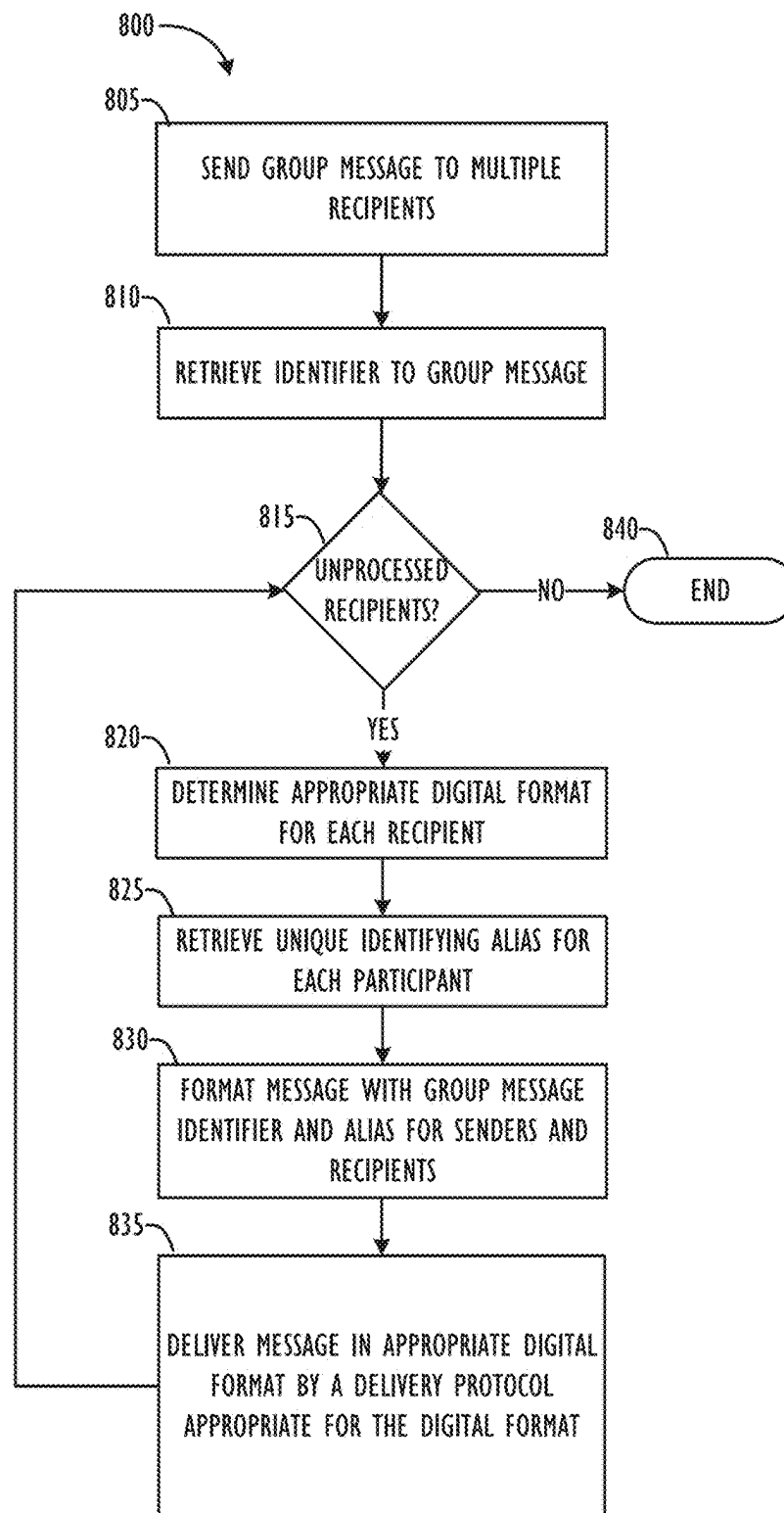


FIG. 8

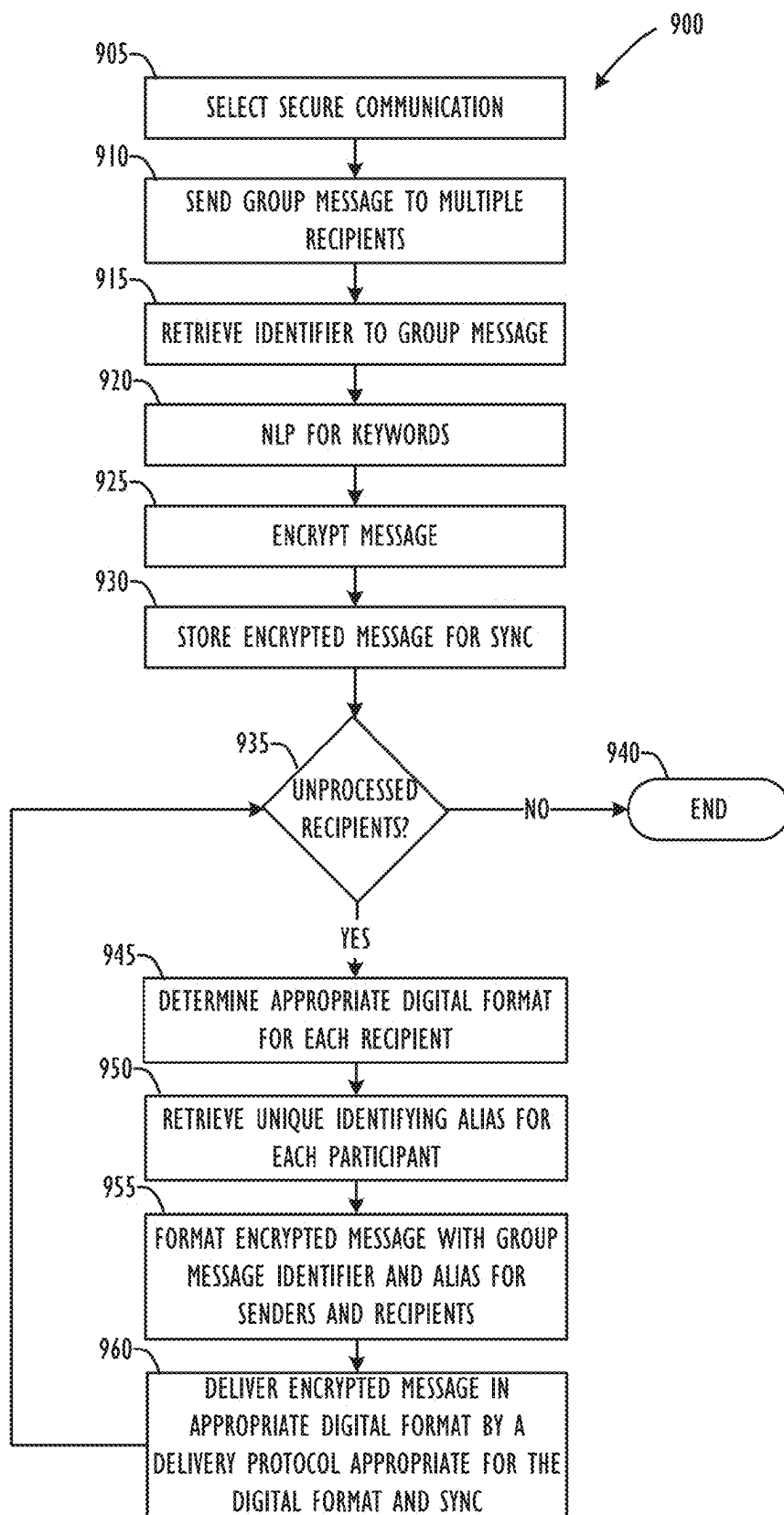


FIG. 9

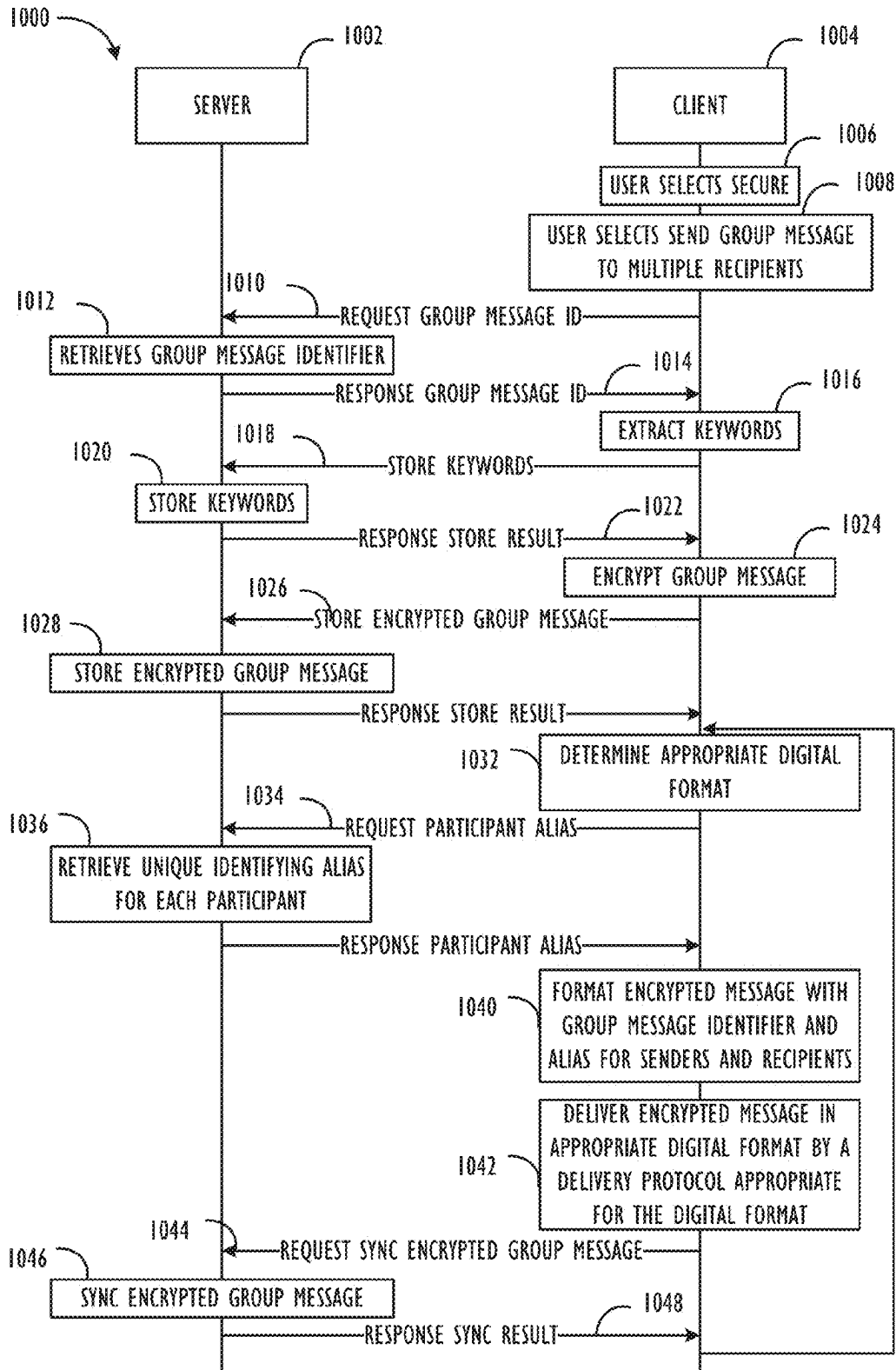


FIG. 10

1

APPARATUS AND METHOD FOR MULTI-FORMAT AND MULTI-PROTOCOL GROUP MESSAGING

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.

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 that 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"—such ease of communication is not possible.

In the past, users of communications systems first had to choose a communication format before composing a message or selecting desired recipient(s). For example, a user must pick up a telephone before calling someone, or a user must launch a text or email application before composing the text or email, etc. And, while text might be the most convenient format at the time for the composer, text may not be convenient for the receiver—resulting in a delayed receipt of 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 user can select the desired recipient(s). Then, the user may compose the outgoing message (in any format). Next, the system (or the user, in some embodiments) chooses the 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 user's client device or a central communications system server) and sent to the desired recipient(s) via the chosen delivery protocol(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, which may be employed to deliver the outgoing communication to the

2

desired recipient(s) in the most preferred way, e.g., either through preferences that the recipient 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. This system 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 format or protocol.

Messages sent through the multi-format communications network described herein may reach participants of the group in traditional formats or protocols or, with 'on-network' recipients, in their preferred format or protocol. With group messages in particular, users desire a system wherein the incoming messages adapt seamlessly to the preferred format and/or protocol of the respective recipient, while maintaining desired levels of security and/or encryption.

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 secure group 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 for implementing a multi-protocol multi-format communication transmission, according to one or more disclosed embodiments.

FIG. 9 is a flowchart of one embodiment of a method for implementing a secured protocol multi-format communication transmission, according to one or more disclosed embodiments.

FIG. 10 is a sequence diagram of one embodiment of a method for implementing a secured multi-protocol multi-format communication transmission, according to one or more disclosed embodiments.

DETAILED DESCRIPTION

Disclosed are apparatuses, methods, and computer readable media for composing secure group messages 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 group communications, including secure communications, in a number of pre-determined and/or 'determined-on-the-fly' communications formats and/or protocols via a single, seamless user interface.

Determinations of outgoing communication formats and/or protocols 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, 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,' 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 illustrated as mobile phone 102. Sync server 105, in connection

4

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 transfer; 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. Similar to infrastructure 100 shown in FIG. 1A, infrastructure

5

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 phone processor, or, as examples, one or more members of

6

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 Ehrmantraut, whereas checkbox **330** represents the first new video message from user Peter Ehrmantraut. 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 mes-

sage 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), 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 time delayed). Likewise, the multi-protocol, multi-format communication composition system also allows the user the ability to send one message to multiple recipients in multiple formats and/or via multiple protocols. The choice of format/

11

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 for the query that the client device is currently trying to send (Step 510). If there are cached results at Step 510, the query

12

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.

Multi-Protocol, Multi-Format Communication Transmission

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 multi-protocol, multi-format communication transmission, as well as other features, including Optimal delivery, possible. It should be appreciated that the UMO facilitates not only the communication between 'on-network' and 'off-network' users, but also facilitates the

13

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. 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

14

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 for implementing a multi-protocol multi-format communication transmission, according to one or more disclosed embodiments.

The flowchart 800 starts with a request to send a group message to multiple recipients (Step 805). In one embodiment, this could be a group email message, where one user fulfills the roll of the sender and the others are recipients, all communicating by email. Alternatively, a group message could comprise a group SMS text message, which may automatically be converted to a MMS message, where one user, likewise, fulfills the roll of sender and the other are recipients. In another embodiment, one user fulfills the roll of the sender and the others are recipients, however, the mechanism, or “protocol,” for delivery of the message may be different for each recipient. For example, one recipient may receive the message by email, another by SMS text message, and another by FACEBOOK® instant message.

Upon receiving the sent message, the system may create a unique identifier for the group message (Step 810). In UMO, this unique identifier may be a Message Group 730 object. As messages are potentially being converted to different delivery protocols for each recipient, the original recipients must be tracked appropriately. A unique identifier for the group message corresponds to the messages sent between a specific group of participants. The identifier is unique to the participants and the chain of communication. This is used to correlate responses back and forth and deliver the messages to the group participants. If the message is the starting message in a new chain of messages, the system may generate a new unique identifier. In other embodiments, additional participants may added to the group of participants, e.g., via a UMO Participant 705 object, and may have messages tracked with the same Message Group 730 object, as the unique identifier. This may allow for the viewing (or the hiding) of messages in the chain of communication that were delivered “before” the new participant(s) joined the group message thread.

The next step involves the processing of any unprocessed recipients (Step 815). It operates until all recipients are processed and then it ends at Step 840.

The next step involves determining an appropriate digital format for each recipient (Step 820). In one embodiment, this would involve utilizing the digital format for which a previous message with the same unique identifier, or Message Group 730 object, was received. An example of this would be to utilize an email format for an ongoing email group communication. A more complex approach would be to use a predetermined user preference giving guidance as to which digital format to use and under what circumstances to use it. For example, if it was after 11 P.M., a participant may only wish to utilize email format. Even more complex would be to use the “Optimal option” disclosed above. For example, communications between one participant and two others historically takes place in email prior to 5 P.M. local time. After 5 P.M., the communications take place via SMS text message. In this example, the system would factor in the historical patterns of communication into determining the appropriate digital format for each recipient.

The next step is to retrieve a unique identifying alias for each participant (Step 825). As some addressing mechanisms for different communication protocols are incompat-

15

ible, e.g., a phone number for SMS text message is not compatible with email, it is necessary to provide a unique identifying alias that maps the actual participant to an address that is compatible for the determined appropriate digital format. For example, elaborating on the SMS-to-email discussed example above, the telephone number 408-452-2387 maps to john.q.public2387@myaddress.com. This has the added benefit of obscuring personal information, in this case, the personal telephone number, from all recipients. When a response message is received 'on-network,' the alias is always verified by the system, and used to look up the original address of the participant and deliver it accordingly.

The next step is to format the message with group message identifier, Message Group 730 object, and alias for senders and recipients (Step 830). In this embodiment, the message is constructed with the updated aliases for the participants, and the content is converted in to the determined appropriate digital format.

The last step is to deliver the message by appropriate delivery protocol (Step 835). If it was determined that the appropriate digital format is email, the appropriate protocol for delivery may be SMTP.

As mentioned above, this process then iterates over the entirety of the receiving participants, or Participant 705 objects, until this process is applied to every receiving participant in the group message corresponding to the Message Group 730 object.

FIG. 9 shows a flowchart 900 of one embodiment of a method for implementing a secured protocol multi-format communication transmission, according to one or more disclosed embodiment.

The process described in flowchart 900 is similar to that of flowchart 800. However, with the inclusion of security, e.g., in the form of encryption, some additional steps may be taken. Most known and accepted industry standard encryption methods may be applied to any encryption step discussed herein, depending on the preferences of a particular embodiment. For this example, a "zero knowledge" privacy enforcement scheme, wherein the central server does not hold the keys needed to decrypt the content of clients that is stored at or processed by the server, is assumed. Additionally, for this embodiment, the encrypted message may be stored and indexed (e.g., using one or more index-able keywords indicative of the content or presumed-content of the underlying encrypted message) for later searching and retrieval by the user.

The first step is for a user to select an option to utilize a secure group communication protocol (Step 905). This indicates to the system that the user intends to encrypt the content of the message.

Next, the user presents a request to send a secure group message to multiple recipients (Step 910).

The next step is to create an identifier for the group message (Step 915). This is equivalent to Step 810 in flowchart 800.

Prior to encryption, Natural Language Processing (NLP) techniques may be utilized to identify keywords (Step 920). Natural language processors can parse the text of the message, prior to "zero knowledge" privacy enforcement, to create index-able keywords so a user may search past messages without compromising the message content itself. Additionally, keywords extracted in this step may also be used for other purposes, including the determination of Optimal delivery option. The index-able keywords may be stored in a database for later retrieval. The NLP extracted keywords would be closely related to the content of the parsed message.

16

Next, the system encrypts message (Step 925). As mentioned above, this may be accomplished by any desired encryption techniques.

The next step is to store the encrypted message for later synchronization (Step 930). The encrypted message is stored for later synchronization to any applicable third party system and clients for 'on-network' users. The indexable keywords extracted (Step 920) are correlated to the encrypted stored message, and are used to locate the encrypted message.

Like the processing in flowchart 800, the next step involves the processing of any unprocessed recipients (Step 935). It operates until all recipients are processed and then it ends at an End step (Step 940).

The process for determining the format, aliasing the participants, formatting the message, and delivering the message is similar to that in flowchart 800.

As mentioned above, this process then iterates over the entirety of the receiving participants until this process is applied to every receiving participant in the group message.

A difference from flowchart 800, and consistent with the network topology in FIG. 1A, is that the message itself in flowchart 900 may not be modified to adapt to the chosen message format, nor the protocol that the message is being transported on. The entirety of the encrypted message may instead be inserted into a message object as a "payload," just as it was encrypted. If Optimal delivery was selected, the NLP extracted keywords are utilized to determine the appropriate digital format and the respective delivery protocol for the desired recipients.

Upon delivery of the encrypted message (Step 960), the stored encrypted message from Step 930 is then synchronized with applicable third party systems as well as other clients by 'on-network' users, so that the group communication is seamless across all clients.

Alternatively, in a network topology consistent with FIG. 1B, the client processes the messages. As the client is directly attached to the third party computer systems, there is no need to encrypt prior to sending to recipients as there is no "middleman" server. Encryption would only be necessary prior to storing to the centralized sync server. For example, in the context of FIG. 1B, a mobile phone 102 would act as the client attached directly to third party computer systems 106-109. The mobile phone 102 would perform the NLP, determine the appropriate delivery formats for those respective computer systems 106-109 based on the NLP, deliver them by the respective delivery protocol, and then encrypt the message prior to relaying the encrypted message and NLP keywords back to the centralized sync server 105 for storage.

FIG. 10 shows a sequence diagram 1000 of one embodiment of a method for implementing a secured multi-protocol multi-format communication transmission, according to one or more disclosed embodiments.

Sequence diagram 1000 is another embodiment of flowchart 900, demonstrating the steps and the actors involved in implementing a secured multi-protocol multi-format communication transmission.

The sequence diagram 1000 includes two actors: a server 1002 and a client 1004. In other embodiments, the number of servers and clients may vary.

The sequence begins with a user selection for secure 1006 being received by the client 1004. This corresponds to Step 905 flowchart 900. The client 1004 receives a user selection to send a user group message to multiple recipients 1008, which corresponds to 910.

The client 1004 then requests a group message identifier 1010 from the server 1002. The server 1002, then retrieves

17

a group message identifier **1012**, and sends a response **1014** with the group message identifier to the client **1004**. The group message identifier may be generated if the received message is determined to be the first in a new chain, or it may be pre-existing one relating this message to an ongoing chain of messages.

The request for the group message identifier **1010**, the retrieval of a group message identifier **1012**, and the subsequent response **1014** are independent in this sequence and may take place prior to determining the appropriate format **1032** of the participants. The appropriate format in one embodiment is determined by the "Optimal" delivery option. The "Optimal" delivery option utilizes a predictive time-based data model based on "on-network" user preferences and historical communication. A predictive time-based data model may be implemented by a factorization machine, support vector machines, or other machine learning technique.

Next, NLP techniques may be utilized to extract keywords **1016** by the client **1004**, corresponding to Step **920**. By doing the NLP on the client **1004** privacy is maintained. The client **1004** sends the keywords to be stored **1018** by the server **1002**. The server **1002**, then stores the keywords **1020** in an indexable database for searching later. A response **1022** indicating success or failure of the storing of the keywords **1020** is transmitted to the client **1004**. Additionally, the keywords extracted by NLP techniques may be utilized for facilitate an "Optimal" delivery format and/or protocol selection. For example, if words in the message are determined via the NLP to relate to a funeral or the death of a loved one, then a text message via SMS may not be the most appropriate channel to deliver the message over to the recipient(s).

Once the keywords have been stored, the message may be encrypted. The next three steps **1024**, **1026**, **1028**, and **1030** comprise an implementation of steps **925** and **930** from FIG. **9**. The client **1004** encrypts the group message **1024** and sends the encrypted group message **1026** to the server **1002** for synchronization. The server **1002** then synchronizes the encrypted group message **1028**, and sends a response **1030** indicating success or failure to the client **1004**. By synchronizing the encrypted message on the server, the server **1002** interfaces with third party systems, which may provide the communication protocol for the encrypted group message, and inputs the encrypted group message to that system. This allows the third party client applications to remain in sync.

The client **1004** then determines the appropriate digital format **1032** for the message. This corresponds to step **945** in flowchart **900**.

The server **1002** is responsible for maintaining continuity with alias mappings, so the client **1004**, requests a participant alias **1034**. The server **1002** retrieves a unique identifying alias for each participant **1036**, which is for similar purposes as described in steps **825** and **950**. The server **1002** then sends a response with the participant alias **1038**. Any time a message is delivered across protocols, this mapping is used to identify a user, however, it obfuscates personal data like telephone numbers, but maintains an address mapping back to the original participant. When a response message is received 'on-network,' the alias is always verified by the server, and used to look up the original address of the participant and deliver it accordingly.

As with steps **955** and **960**, the encrypted message is formatted with the group message identifier and the alias for the senders and participants **1040**, and subsequently delivered in the appropriate digital format by the appropriate delivery protocol **1042**.

18

Once the encrypted message is successfully sent, the client **1004** sends a request **1044** to the server **1002** to synchronize the encrypted group message for 'on-network' users with any third party system that was used for the message transmission. For example, if FACEBOOK® messaging was utilized as the digital format and the delivery protocol, the server **1002** would synchronize that transmission, in that the encrypted group message would be up to date across other clients as well. The server would sync the encrypted group message **1046**, and provide a response **1048** to the client **1004** indicating success.

The reception of a message encoded and delivered in this manner would thereby be decoded using the correlative decryption scheme to the initial message encryption.

EXAMPLES

The following examples pertain to further embodiments.

Example 1 is a non-transitory computer readable medium comprising computer executable instructions stored thereon to cause one or more processing units to obtain a first message in a first digital format from a first user, the first message directed to one or more desired recipients, wherein at least part of the content of the first message is encrypted; obtain, after obtaining the first message, a unique group identifier based, at least in part, on the combination of: the first message, the first user, and the one or more desired recipients; generate a unique alias for each of the one or more desired recipients and the first user; for each of the one or more desired recipients: determine a second digital format for the respective desired recipient to receive an outgoing second message, wherein the outgoing second message comprises the content of the first message; determine, for the respective desired recipient, a first protocol for the delivery of the outgoing second message to the respective desired recipient; convert the outgoing second message into the second digital format for the respective desired recipient; insert the generated unique aliases into the outgoing second message; insert the encrypted content into the second outgoing message; insert the group identifier into the second outgoing message; and direct the outgoing second message to be sent to the respective desired recipient via the determined first protocol, using the generated unique alias for the respective desired recipient.

Example 2 includes the subject matter of example 1, wherein the encrypted content comprises a body of the first message.

Example 3 includes the subject matter of example 1, wherein the instructions to determine the second digital format comprise instructions to apply a set of parameters based on the respective desired recipient to a predictive time-based data model.

Example 4 includes the subject matter of example 1, wherein the instructions to determine a first protocol further comprise instructions to select a first protocol capable of transporting the second digital format.

Example 5 includes the subject matter of example 3, wherein the predictive time-based data model comprises a factorization machine.

Example 6 includes the subject matter of example 1, further comprising instructions to store the at least part of the content of the first message that is encrypted at a storage location remote to the first user.

Example 7 includes the subject matter of example 1, wherein the instructions to generate the unique aliases comprise instructions to map an appropriate format-specific addresses to each desired recipient and the first user.

19

Example 8 is an apparatus comprising a display; a memory; and one or more processing units, communicatively coupled to the memory, wherein the memory stores instructions to configure the one or more processing units to obtain a first message in a first digital format from a first user, the first message directed to one or more desired recipients, wherein at least part of the content of the first message is encrypted; obtain, after obtaining the first message, a unique group identifier based at least in part on the combination of first message, the first user, and the one or more desired recipients; generate a unique alias for each of the one or more desired recipients and the first user; for each of the one or more desired recipients: determine a second digital format for the respective desired recipient to receive an outgoing second message, wherein the outgoing second message comprises the content of the first message; determine, for the respective desired recipient, a first protocol for the delivery of the outgoing second message to the respective desired recipient; convert the outgoing second message into the second digital format for the respective desired recipient; insert the generated unique aliases into the outgoing second message; insert the encrypted content into the second outgoing message; insert the group identifier into the second outgoing message; and direct the outgoing second message to be sent to the respective desired recipients via the determined first protocol, using the generated unique alias for the respective desired.

Example 9 includes the subject matter of example 8, wherein the encrypted content comprises a body of the first message.

Example 10 includes the subject matter of example 8, wherein the instructions to determine the second digital format comprise instructions to apply a set of parameters based on the respective desired recipient to a predictive time-based data model.

Example 11 includes the subject matter of example 8, wherein the instructions to determine a first protocol further comprise to select a first protocol capable of transporting the second digital format.

Example 12 includes the subject matter of example 10, wherein the instructions wherein the predictive time-based data model comprises a factorization machine.

Example 13 includes the subject matter of example 8, further comprising instructions to store at least part of the content of the first message that is encrypted at a storage location remote to the first user.

Example 14 includes the subject matter of example 8, wherein the instructions to generate the unique aliases comprise instructions to map appropriate format-specific addresses to each desired recipient and the first user.

Example 15 is a computer-implemented method of communicating digital information, comprising obtaining a first message in a first digital format from a first user, the message directed to one or more desired recipients, wherein at least part of the content of the first message is encrypted; obtaining, after obtaining the first message, a unique group identifier based at least in part on the combination of first message, the first user, and the one or more desired recipients; generating a unique alias for each of the one or more desired recipients and the first user; for each of the one or more desired recipients: determining a second digital format for the respective desired recipient to receive an outgoing second message, wherein the outgoing second message comprises the content of the first message; determining, for the respective desired recipient, a first protocol for the delivery of the outgoing second message to the respective desired recipient; converting the outgoing second message

20

into the second digital format for the respective desired recipient; inserting the generated unique aliases into the outgoing second message; inserting the encrypted content into the second outgoing message; inserting the group identifier into the second outgoing message; and directing the outgoing second message to be sent to the respective desired recipient via the determined first protocol, using the generated alias for the respective desired recipient.

Example 16 includes the subject matter of example 15, wherein the encrypted content comprises a body of the first message.

Example 17 includes the subject matter of example 15, wherein determining the second digital format comprises applying a set of parameters based on the respective desired recipient to a predictive time-based data model.

Example 18 includes the subject matter of example 15, wherein determining a first protocol further comprises selecting a first protocol capable of transporting the second digital format.

Example 19 includes the subject matter of example 15, further comprising storing the at least part of the content of the first message that is encrypted at a storage location remote to the first user.

Example 20 includes the subject matter of example 17, wherein the generating the unique aliases comprises mapping an appropriate format-specific address to each desired recipient and the first user.

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 executable instructions stored thereon to cause one or more processing units within a multi-format communication system to:

21

obtain a first message in a first digital format from a first user, the first message directed to one or more desired recipients, wherein at least part of the content of the first message is encrypted;

obtain, after obtaining the first message, a unique group identifier based on a combination of each of: the first message, the first user, and the one or more desired recipients;

generate a unique alias for each of the one or more desired recipients and the first user; and

for each of the one or more desired recipients:

determine a second digital format for the respective desired recipient to receive an outgoing second message, wherein the outgoing second message comprises the content of the first message;

determine, for the respective desired recipient, a first protocol for the delivery of the outgoing second message to the respective desired recipient; convert the outgoing second message into the second digital format for the respective desired recipient;

insert the generated unique aliases into the outgoing second message;

insert the encrypted content into the second outgoing message;

insert the group identifier into the second outgoing message; and

direct the outgoing second message to be sent to the respective desired recipient via the determined first protocol, using the generated unique alias for the respective desired recipient.

2. The non-transitory computer readable medium of claim 1, wherein the encrypted content comprises a body of the first message.

3. The non-transitory computer readable medium of claim 1, wherein the instructions to determine the second digital format comprise instructions to apply a set of parameters based on the respective desired recipient to a predictive time-based data model.

4. The non-transitory computer readable medium of claim 1, wherein the instructions to determine a first protocol further comprise instructions to select a first protocol capable of transporting the second digital format.

5. The non-transitory computer readable medium of claim 3, wherein the predictive time-based data model comprises a factorization machine.

6. The non-transitory computer readable medium of claim 1, further comprising instructions to store the at least part of the content of the first message that is encrypted at a storage location remote to the first user.

7. The non-transitory computer readable medium of claim 1, wherein the instructions to generate the unique aliases comprise instructions to map an appropriate format-specific addresses to each desired recipient and the first user.

8. A multi-format communication apparatus, comprising:

a display;

a memory; and

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

obtain a first message in a first digital format from a first user, the first message directed to one or more desired recipients, wherein at least part of the content of the first message is encrypted;

obtain, after obtaining the first message, a unique group identifier based on a combination of each of: first message, the first user, and the one or more desired recipients;

22

generate a unique alias for each of the one or more desired recipients and the first user; and

for each of the one or more desired recipients:

determine a second digital format for the respective desired recipient to receive an outgoing second message, wherein the outgoing second message comprises the content of the first message;

determine, for the respective desired recipient, a first protocol for the delivery of the outgoing second message to the respective desired recipient;

convert the outgoing second message into the second digital format for the respective desired recipient;

insert the generated unique aliases into the outgoing second message;

insert the encrypted content into the second outgoing message;

insert the group identifier into the second outgoing message; and

direct the outgoing second message to be sent to the respective desired recipients via the determined first protocol, using the generated unique alias for the respective desired.

9. The apparatus of claim 8, wherein the encrypted content comprises a body of the first message.

10. The apparatus of claim 8, wherein the instructions to determine the second digital format comprise instructions to apply a set of parameters based on the respective desired recipient to a predictive time-based data model.

11. The apparatus of claim 8, wherein the instructions to determine a first protocol further comprise to select a first protocol capable of transporting the second digital format.

12. The apparatus of claim 10, wherein the instructions wherein the predictive time-based data model comprises a factorization machine.

13. The apparatus of claim 8, further comprising instructions to store at least part of the content of the first message that is encrypted at a storage location remote to the first user.

14. The apparatus of claim 8, wherein the instructions to generate the unique aliases comprise instructions to map appropriate format-specific addresses to each desired recipient and the first user.

15. A computer-implemented method of communicating digital information within a multi-format communication system, comprising:

obtaining a first message in a first digital format from a first user, the message directed to one or more desired recipients, wherein at least part of the content of the first message is encrypted;

obtaining, after obtaining the first message, a unique group identifier based on a combination of each of: first message, the first user, and the one or more desired recipients;

generating a unique alias for each of the one or more desired recipients and the first user; and

for each of the one or more desired recipients:

determining a second digital format for the respective desired recipient to receive an outgoing second message, wherein the outgoing second message comprises the content of the first message;

determining, for the respective desired recipient, a first protocol for the delivery of the outgoing second message to the respective desired recipient;

converting the outgoing second message into the second digital format for the respective desired recipient;

inserting the generated unique aliases into the outgoing
second message;
inserting the encrypted content into the second outgoing
message;
inserting the group identifier into the second outgoing 5
message; and
directing the outgoing second message to be sent to the
respective desired recipient via the determined first
protocol, using the generated alias for the respective
desired recipient. 10

16. The method of claim 15, wherein the encrypted
content comprises a body of the first message.

17. The method of claim 15, wherein determining the
second digital format comprises applying a set of parameters
based on the respective desired recipient to a predictive 15
time-based data model.

18. The method of claim 15, wherein determining a first
protocol further comprises selecting a first protocol capable
of transporting the second digital format.

19. The method of claim 15, further comprising storing 20
the at least part of the content of the first message that is
encrypted at a storage location remote to the first user.

20. The method of claim 17, wherein the generating the
unique aliases comprises mapping an appropriate format-
specific address to each desired recipient and the first user. 25

* * * * *