EXHIBIT 1

US005883580A

United States Patent [19]

Briancon et al.

[11] **Patent Number: 5,883,580**

[45] **Date of Patent: Mar. 16, 1999**

[54] **GEOGRAPHIC-TEMPORAL SIGNIFICANT MESSAGING**

[75] Inventors: **Alain Charles Louis Briancon**, McKinney; **Terence Edward Sumner**, Azle, both of Tex.

[73] Assignee: **Motorola, Inc.**, Schaumburg, Ill.

[21] Appl. No.: **822,024**

[22] Filed: **Mar. 24, 1997**

[51] **Int. Cl.**[6] **G08B 5/22**

[52] **U.S. Cl.** **340/825.44**; 340/825.49; 340/825.52

[58] **Field of Search** 340/825.44, 825.47, 340/825.46, 309.4, 309.15, 311.1, 825.49, 825.22; 455/33, 34, 31, 32, 38, 432, 433, 440

[56] **References Cited**

U.S. PATENT DOCUMENTS

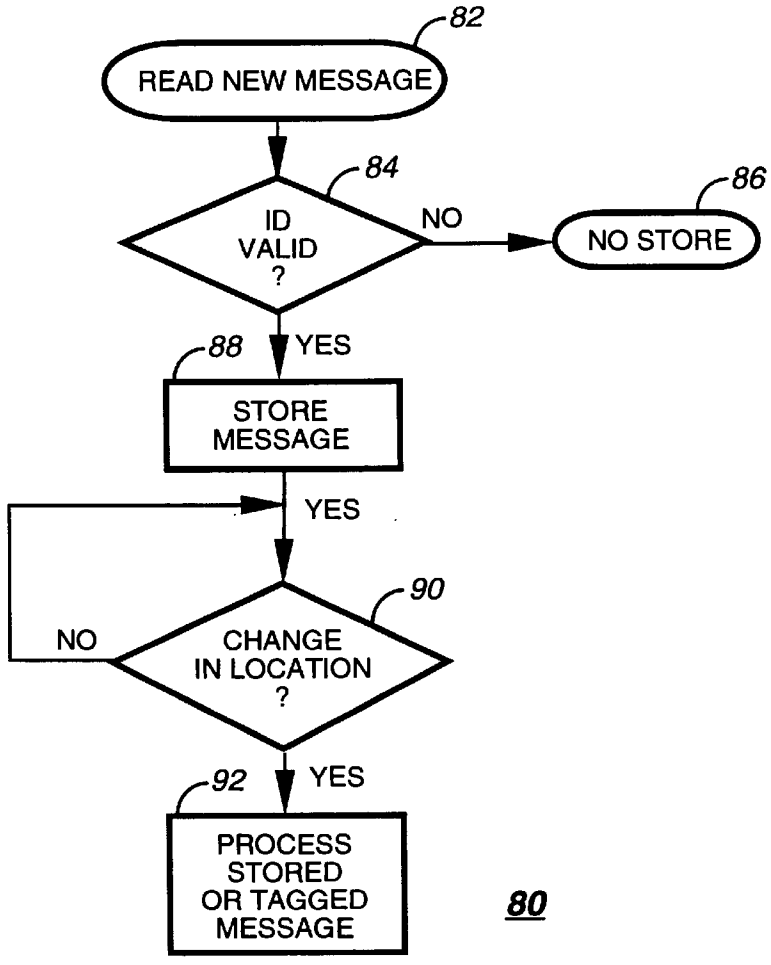
5,225,826	7/1993	DeLuca et al.	340/825.44
5,254,986	10/1993	DeLuca	340/825.44
5,504,476	4/1996	Marrs et al.	340/825.44
5,539,395	7/1996	Buss et al.	340/827
5,600,312	2/1997	Willard et al.	340/825.47

Primary Examiner—Michael Horabik
Assistant Examiner—Jean B. Jeanglaude
Attorney, Agent, or Firm—Pablo Meles; Charles W. Bethards

[57] **ABSTRACT**

A messaging device for receiving a plurality of messages that have a relevancy status includes a selective call receiver **(22)** for receiving the plurality of messages, the relevancy status, and updates to the relevancy status, a memory **(30)** for storing at least a portion of the plurality of messages and the updates to the relevancy status and a processor **(24, 40** and **28)** for processing the portion of the plurality of messages that have received updates to the relevancy status. The relevancy status is ideally a location identifier and could additionally include a valid time indication.

20 Claims, 3 Drawing Sheets



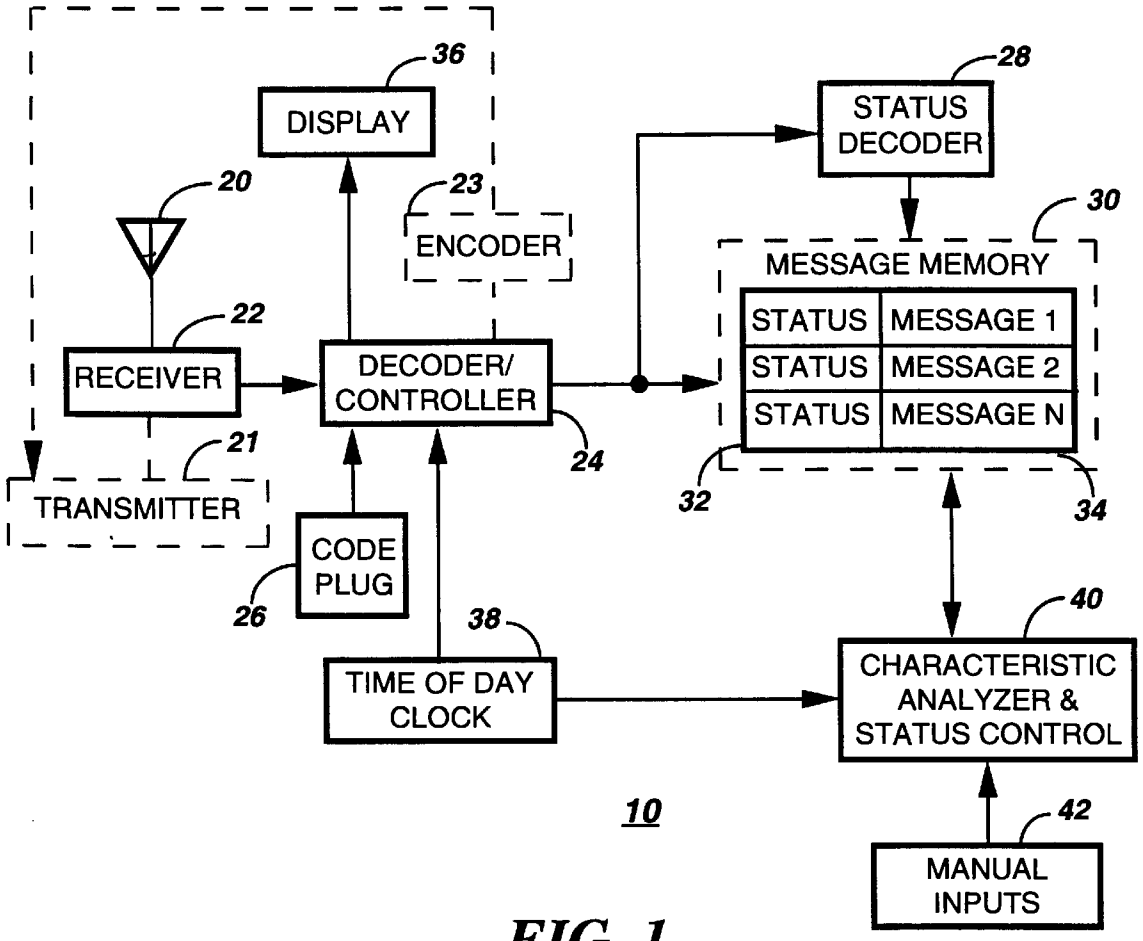

36
DISPLAY
28
STATUS DECODER
20
23
ENCODER
30
MESSAGE MEMORY
22
RECEIVER
DECODER/ CONTROLLER
STATUS
MESSAGE 1
STATUS
MESSAGE 2
STATUS
MESSAGE N
21
24
32
34
TRANSMITTER
CODE PLUG
26
38
40
TIME OF DAY CLOCK
CHARACTERISTIC ANALYZER & STATUS CONTROL
10
42
MANUAL INPUTS

FIG. 1

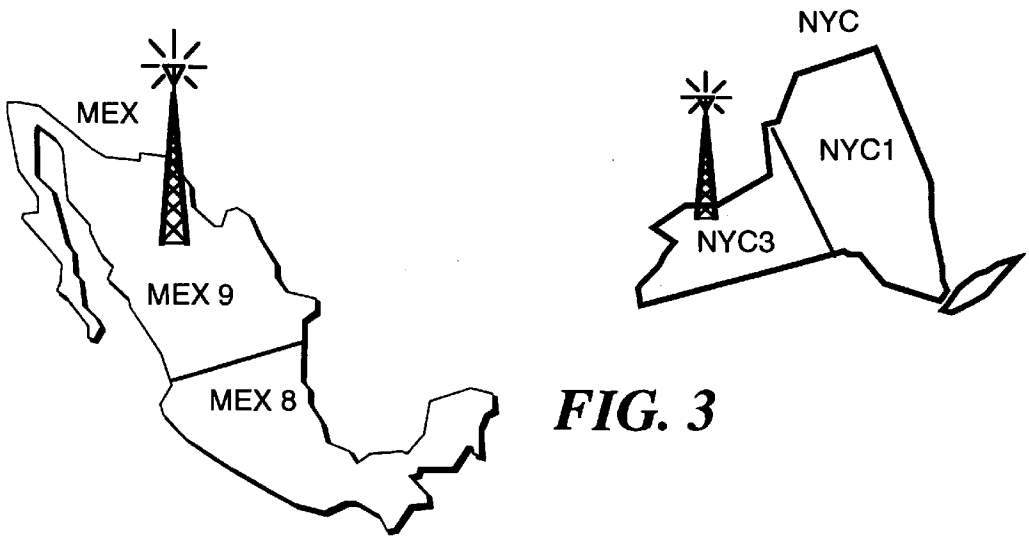
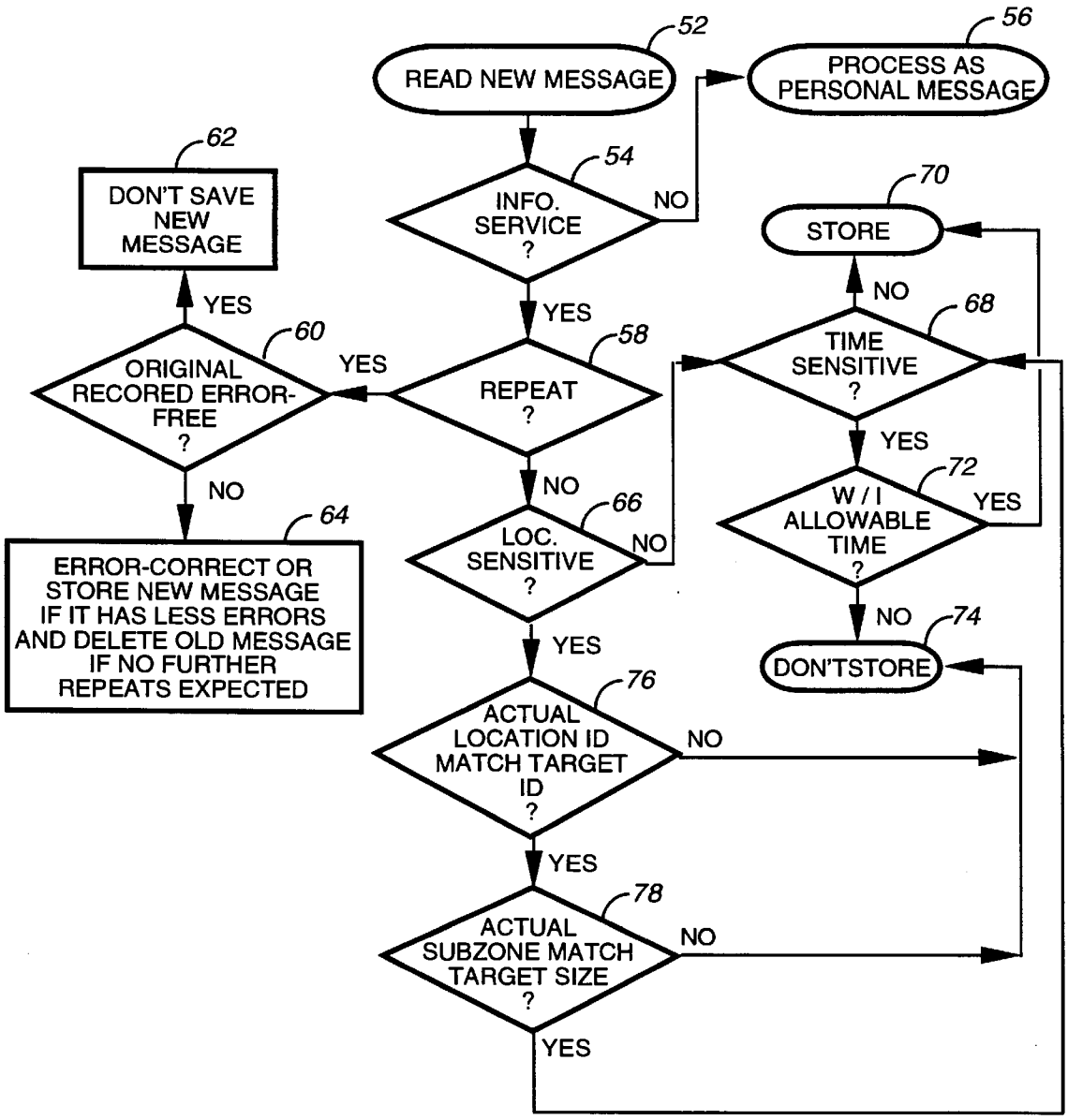

MEX
MEX 9
MEX 8
NYC
NYC1
NYC3

FIG. 3


52
READ NEW MESSAGE
56
PROCESS AS PERSONAL MESSAGE
54
INFO. SERVICE ?
NO
YES
62
DON'T SAVE NEW MESSAGE
YES
60
ORIGINAL RECORED ERROR-FREE ?
YES
58
REPEAT ?
NO
64
ERROR-CORRECT OR STORE NEW MESSAGE IF IT HAS LESS ERRORS AND DELETE OLD MESSAGE IF NO FURTHER REPEATS EXPECTED
66
LOC. SENSITIVE ?
NO
YES
76
ACTUAL LOCATION ID MATCH TARGET ID ?
NO
YES
78
ACTUAL SUBZONE MATCH TARGET SIZE ?
NO
YES
70
STORE
NO
68
TIME SENSITIVE ?
YES
72
W / I ALLOWABLE TIME ?
YES
NO
74
DON'TSTORE

50

FIG. 2

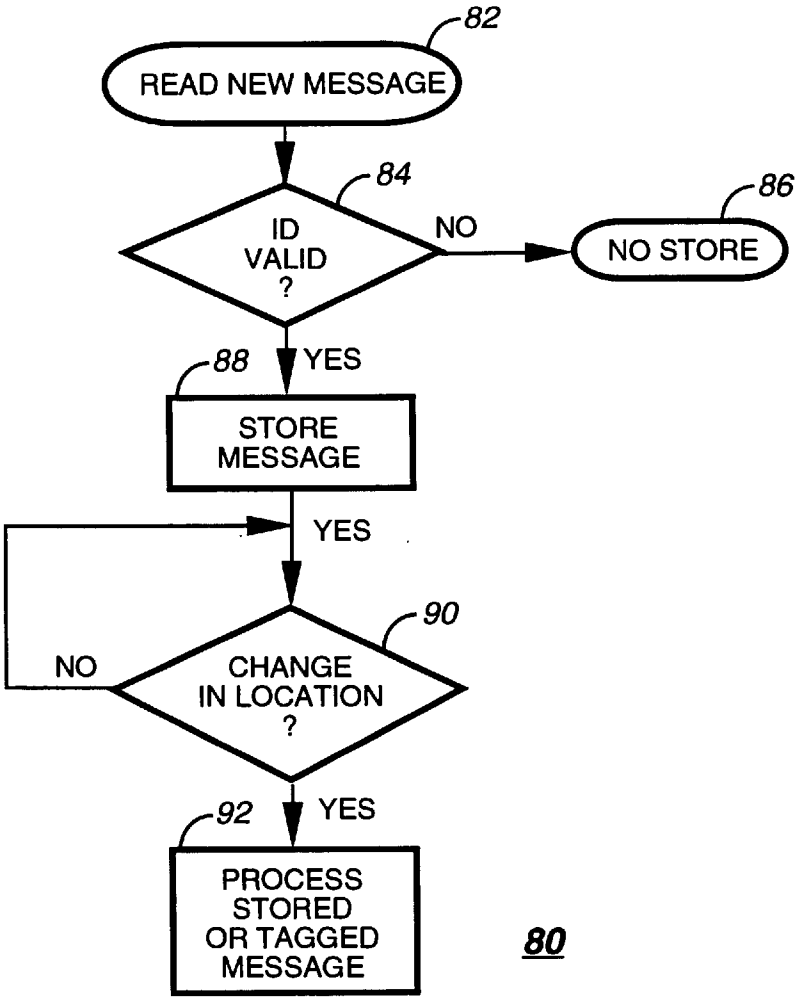

82
READ NEW MESSAGE
84
ID VALID ?
NO
86
NO STORE
YES
88
STORE MESSAGE
YES
90
NO
CHANGE IN LOCATION ?
YES
92
PROCESS STORED OR TAGGED MESSAGE
80

FIG. 4

GEOGRAPHIC-TEMPORAL SIGNIFICANT MESSAGING

FIELD OF THE INVENTION

This invention relates in general to selective call messaging devices and more specifically to messaging devices that process messages logically for a user in the context of space and time.

BACKGROUND OF THE INVENTION

Information services can be temporal as well as spatial in nature. Sending information about highway traffic for example is relevant to a pager user only if the user is in or near the market where the traffic problem exists and if the message is read within several hours of the report of the traffic problem. For instance, if the pager receives the message "635 backed up to stadium" while in Dallas and then the user arrives in Denver, the user reading this information would not obtain any value from this message. Likewise, if the user receives the same message, but reads the message 5 hours later, the message would be of little value and occupy memory space that could be used for other more pertinent messages. A pager that accounts for the temporal nature of messages is described in U.S. Pat. No. 5,225,826, assigned to Motorola, Inc. and entitled "Variable Status Receiver", this patent hereby incorporated by reference. Thus, a need exists for an intelligent messaging device that is able to distinguish messages that are solely spatially sensitive or both time and space sensitive and that further logically processes such messages in accordance with the actions of the user.

SUMMARY OF THE INVENTION

In a first aspect of the present invention, a method processing spatially sensitive information at a selective call receiver comprises the steps of receiving the spatially sensitive information along with an associated identifier and storing the spatially sensitive information if the associated identifier is valid. When a change is detected in the associated identifier as the selective call receiver roams from a first area to a second area, the selective call receiver process the spatially sensitive information.

In a second aspect of the present invention, a messaging device for receiving a plurality of messages with a portion of the messages having a relevancy status comprises a selective call receiver for receiving the plurality of messages, the relevancy status for the portion of the plurality of messages, and updates to the relevancy status, a memory for storing at least a portion of the plurality of messages and the updates to the relevancy status, and a processor for processing the portion of the plurality of messages that have received updates to the relevancy status.

In a third aspect of the present invention, a messaging device for receiving a plurality of messages having at least a portion with a location identifier comprises a selective call receiver for receiving the plurality of messages, the location identifier for the portion of the plurality of messages, and updates to the location identifier, a first memory location for storing a list of existing location identifiers, a second memory location for storing at least a portion of the plurality of messages, the location identifier for the portion of the plurality of messages, and the updates to the location identifiers, and a processor for processing the portion of the plurality of messages that have a location identifier by comparing the first memory location with the second memory location.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a block diagram of a messaging device in accordance with the present invention.

FIG. **2** is a flow chart of an algorithm illustrating a method in accordance with the present invention.

FIG. **3** is a representation of locations used in system in accordance with the present invention.

FIG. **4** is another flow chart of an algorithm illustrating a method in accordance with the present invention.

DETAILED DESCRIPTION

A messaging device for receiving a plurality of messages wherein at least a portion of the plurality of messages have a relevancy status comprises a selective call receiver for receiving the plurality of messages with the relevancy status as well as updates to the relevancy status. The messaging device further preferably comprises a memory for storing at least a portion of the plurality of messages and the updates to the relevancy status and a processor for processing the portion of the plurality of messages that have received updates to the relevancy status. Preferably the relevancy status is a location identifier or a valid time period indicator or a combination of location and time indicators. The location identifier is preferably selected from the group consisting of Simulcast System Identifier (SSID), Network Identifier (NID), Service Provider Identifier (SPID), Zone Identifier, Subzone Identifier, market indicator, information identifier, transmitter colorcode or prohibited location indicator. Some of these identifiers will be used with Motorola's FLEX™ or ReFLEX™ roaming paging protocol. The relevancy status can make up a portion of a receiver's address or can be another part of the message received. In any event, the messaging device must be able to determine its location either explicitly (as with location identifiers) or implicity as can be contemplated within the scope of the present invention. (For example, if only a particular frequency or modulation scheme is used in a particular area, then the location of the received signal can be implicitly determined.) With respect to time and the processing of time-based decisions, the messaging device may have the ability to derive time from the over-the-air protocol it receives (again, such as Motorola's synchronized FLEX™ or ReFLEX™ protocols), but alternatively the messaging device may also derive time from a real time clock. Further, it is contemplated within the present invention that the messaging device can be a one-way or a two-way selective call receiver.

In accordance with the present invention and with reference to FIG. **1**, a messaging device for receiving a plurality of messages that have a location identifier in at least a portion of the messages comprises a selective call receiver for receiving the plurality of messages as well as updates to the location identifiers. The messaging device further comprises a first memory location used for storing a list of existing location identifiers and a second memory location used for storing at least a portion of the plurality of messages, the associated location identifier for the portion of the plurality of messages, and the updates to the location identifiers. The messaging device further comprises a processor for processing the portion of the plurality of messages that have a location identifier by comparing the first memory location with the second memory location. Preferably the messaging device is a one-way selective call receiver or a two-way selective call transceiver and the processor stores a message and enables the display of the message if a match is found between the existing location identifiers and the location identifier associated with the message.

Alternatively, if desired, the messaging device may be programmed to find prohibited location indicators. In such a case, a processor does not store (or does not display or does not display portions as desired of) a message if a match is found between the existing location identifiers and the location identifier associated with the message. In another embodiment, the messaging device's processor can delete an existing message from the second memory if a match is found between the existing location identifiers and the location identifier associated with the message.

Again, FIG. **1** illustrates a selective call receiver **10** or alternatively a transceiver (with receiver **22** and transmitter **21**) operatively constructed in accordance with the present invention. This receiver or transceiver is preferably associated with a paging system having a base station transmitter and terminal. The receiver responds to control data information from the base station terminal and in turn stores and provides data messages to a user during operation.

Radio frequency modulated selective call paging signals are received at antenna **20**, demodulated by receiver **22** and decoded by decoder **24**. Alternatively, the transceiver can further encode messages and/or control signals via encoder **23** and transmit messages or control signals via the transmitter **21** and antenna **20**. Decoder **24** compares address signals within the selective call signal with at least one predetermined address signal stored within the code plug **26**. In response to detecting an address assigned to the pager, a message signal following the address is processed. Status decoder **28** determines if the message signal includes status and timing information and stores the information in a status area **32** within message memory **30**. If no initial status information is included, a default status is assigned to the message. The status decoder **28** may additionally make a decision on whether a message should be stored at all. Thus, the decoder **24** can decode a valid or target address, yet the status decoder **28** can still find an invalid status (such as invalid time or an invalid location) and direct the messaging device to not store the message (or alternatively store and not display the message). If the address and status is valid, then the message information is stored in a message area **34** within message memory where each message has a corresponding status signal. The pager also has a display **36** for displaying message information as well as time of day information as determined by a time of day clock **38**. Alternatively, a pager using Motorola's FLEX™ family of paging protocols can derive the time from the synchronized frame information inherently available from over the air messaging using the protocol itself and store the time information where appropriate. The time of day clock **38** may also provide day and date information and is capable of generating alarms in response to the time of day being equal to a time set by various functions of the pager. The status of messages stored in message memory may be changed by a characteristic analyzer and status controller **40** in response to the time of day clock **38** or in response to geographic information received in memory **30** or in response to a combination of the time and geographic information. Additionally, the status of a message stored in message memory can be changed under manual control from manual inputs **42** which may include a plurality of buttons for manually operating various functions of the pager. The device **40** is preferably a memory location containing at least an algorithm similar to the one outlined in FIG. **2** as will be described. Decoder **24** further analyzes the status assigned to a received message and the status of messages stored in the message memory **30** and determines which messages if any are to be deleted in order to store the message, or if the received message will be stored. It should also be understood that it is within contemplation of the present invention to have location and other identifiers determined by any source, not just from the same channels that provide the messaging information. Thus, if the device **10** further provided a means for a GPS received signal, such information could be used to provide processing of messages within context of location.

Referring to FIG. **2**, an algorithm **50** in accordance with the present invention is shown. A selective call receiver or transceiver receives a new message and decodes or reads the new message at step **52**. With particular reference to paging, receipt of messages are typically treated differently dependent upon whether a message is a personal message or an information services message. More likely than not, information services messages will only have a particular relevancy during a given period or for a particular geographic location. The present invention certainly contemplates that personal messages would be within the scope of the claimed invention, but information services messages are merely used here as a clear example. Thus, at step **54**, the selective call receiver may detect if the message is an information services message. If not, the message can be treated as a personal message at step **56**. If the message is an information services message, then the algorithm may determine if the message is a repeat message at step **58**. If the message is a repeat message, then an inquiry is made to see if the original message was received without errors at step **60**. If no errors existed in the original message, then the new message is not saved as shown in step **62**. If errors existed in the original message at step **60**, then several options could be implemented. A first option could include error correcting the original message with information from the repeat message. Another option could include storing the new message if the new message has less errors than the original message. Further, the original message could be deleted, particularly if no further repeats were expected. If the message is not a repeat message, then it is determined if the message is location sensitive at step **66**. If the message is not location sensitive at step **66**, then it is determined if the message is time sensitive at step **68**. If the message is not time sensitive, then the message is stored at step **70**. If the message is time sensitive, then it is determined if the message received exists within a valid time period at step **72**. If the time appears valid, then the message is stored at step **70**. If the time is invalid, then the message is not stored as shown at step **74**. Alternatively (not shown), the message could be stored, but not displayed. In another alternative, the message is stored and only portions (that were considered valid) of the message is displayed.

If the message is location sensitive at step **66**, then a match for a location identifier is sought at step **76**. Alternatively, a subzone or sub-location match may be sought as well as shown in step **78**. The location identifier and sub-locations can come in many variants including Simulcast System Identifier (SSID), Network Identifier (NID), Service Provider Identifier (SPID), Zone Identifier, Subzone Identifier, market indicator, information identifier, transmitter colorcode as well as prohibited location indicator to name a few. If no location match is indicated, then the message will not be stored at step **74**. If a location match is indicated, then an inquiry into time validity is done at step **68** as previously described above.

Operationally, a method **80** (with reference to FIG. **4**) of processing spatially sensitive information at a selective call receiver in accordance with the present invention preferably comprises the steps of receiving (**82**) the spatially sensitive

information along with an associated identifier and storing **(88)** the spatially sensitive information if the associated identifier is valid. If at decision block **84**, it is determined that the identifier is invalid, then the message is not stored **(86)**. Then, the method continues by detecting a change in the associated identifier at decision block **90** as the selective call receiver roams from a first area to a second area. If a change is detected, then, the spatially sensitive information is processed at step **92**. Preferably, such processing includes deleting at least a portion of the spatially sensitive information which is not relevant to the second area. Alternatively, processing of such messages could include storing a message and deleting such message until a later time when the memory management scheme of the pager determines that the message should be deleted. A memory management scheme could include automatically deleting at least portions of the spatially sensitive information when the memory of the selective call receiver is filled beyond a threshold level and otherwise deleting such portions of the spatially sensitive information when the user prompts the selective call receiver to display a message or delete messages or portions thereof upon other events at the messaging device as could be reasonably contemplated. In another alternative, the selective call receiver could delete at least portions of the spatially sensitive information when a user prompts the selective call receiver to display at least portions of the spatially sensitive information.

In another embodiment, a method of processing a message among a plurality of messages received over the air at a selective call receiver comprises the steps of receiving the plurality of messages wherein at least a portion of the messages have a relevancy status that includes a space/time indicator. The messages are stored if the relevancy status matches a stored relevancy status at the selective call receiver. When the space/time indicator changes, the method then tags the messages for processing and then the tagged messages are processed. Such change in the space/time indicator can be a change in location or a change in time beyond a predefined threshold period. Such processing could include the step of deleting messages that have an invalid space/time indicator.

Ten messages are provided below in a table as an example of how a selective call receiver might operate in accordance with an embodiment of the present invention. Each message in the table provides one or more bits of status information: Address (whether personal or information services); location identifiers, sub-location identifiers, time (either derived from the protocol or a real time clock); a spatial or time sensitive message or a message that is both time and space sensitive; and a stale time indicator. The target location identifier and the target sub-location identifier could be location identifiers already stored in memory (such as a look-up table) in the messaging device. The process column describes what process the messaging device performs based on the relevancy status received.

	ADDR	LOC ID	SUB LOC ID	TIME	MESSAGE	TARGET LOC. ID	TARGET SUB LOC ID	STALE TIME	PROCESS
1	INFO	A	A1	1 PM	10% off by showing pager before closing	A	A1	5 PM	STORE/ ALERT
2	INFO	A	A1	10 AM	sales force meeting @ noon	A	A2	12:30	STORE/ ALERT
3	INFO	A	A1	10 AM	sales force meeting @ noon	B	B5	12:30	NO STORE/ NO ALERT
4	INFO	A	A1	10 AM	sales force meeting @ noon	A	A1	12:30	STORE/ ALERT 5 MINS B/F MEETING
5	INFO	A	A1	1 PM	sales force meeting @ noon	A	A1	12:30	DELETE OLD PAGE
6	INFO	NYC	NYC1	3 AM 7/4/99	Avoid NYC, Riot 48 hr curfew	NYC	NYC1	3 AM 7/6/99	STORE/ ALERT
7	INFO	NYC	NYC3	3 AM 7/4/99	Avoid NYC, Riot 48 hr curfew	NYC	NYC1	3 AM 7/6/99	STORE/ ALERT
8	INFO	MEX	MEX9	7 AM 7/4/99	Avoid NYC, Riot 48 hr curfew	NYC	NYC1	3 AM 7/6/99	DO NOT DISPLAY & DELETE
9	OVERRIDE 1	MEX	MEX9	3 AM 7/4/99	Avoid NYC, Riot 48 hr curfew	NYC	NYC1	3 AM 7/6/99	STORE/ AUTO DISPLAY
10	OVERRIDE 2	MEX	MEX9	3 AM 7/4/99	Avoid NYC, Riot 48 hr curfew	NYC	NYC1	3 AM 7/6/99	STORE/ NO AUTO DISPLAY

In message **1**, the location and sub-location identifiers match and the message received is during a valid time period, therefore the message is stored and the user is alerted. Message **1** represents an example of a message that a retail store could send to encourage customers to buy within a certain time period. In message **2**, the sub-location does not match, but the messaging device can still be programmed to store and alert the user, particularly if the sub-location identifier appears to be adjacent to the target sub-location identifier. In message **3**, the location identifiers do not match and therefore the message is preferably not stored. In message **4**, the location identifiers match and the

time is valid, but the pager is alternatively programmed to store the message and alert the user 5 minutes before the pertinent time as derived from the message. In this case, a meeting at 12am is indicated and the user would be alerted at 11:55am. In message **5**, the time period is invalid. Thus, the pager could either delete the new message or alternatively (if the new message is a repeat) the old page could be deleted.

Message **6** has matching location and sub-location identifiers and is received during a valid time period. Thus, the message would be stored and the user alerted. Message **7**, as shown in message **2**, fails to show a matching sub-location identifier, but the messaging device can still be programmed to store and alert the user, particularly if the sub-location identifier appears to be adjacent to the target sub-location identifier. Message **8**, as shown in message **3** fails to show matching location and sub-location identifiers, therefore the message is not retained if the message was previously received and stored. In fact, the messaging device could be specifically programmed to delete messages if the "NYC" location identifier is not indicated or specifically programmed to delete messages if the "MEX" is indicated. Message **9** and **10** merely show examples of "override" type messages that would store and automatically display (or alternatively store and not display in the case of message **10**) regardless of what location identifiers and time indicators are received at the messaging device.

In another aspect of the present invention, an address list containing a target address and corresponding valid location indication could be stored within a memory location in a messaging device. The memory location could contain the exemplary look-up table found below:

TARGET/ADDRESS	WHERE VALID
123	A
456	A,B
987654	everywhere
789	ALL but E

Preferably, the messaging device would first determine its current location and then enable its target addresses. Subsequently, the messaging device would store messages for the enabled targets and otherwise ignore messages for its disabled targets. This look-up table could handle the storage mechanism of the device.

As an example, a messaging device that recognizes its location as being location "A" or location "B" which further receives a message with a target address of "456" would store this particular message. If the messaging device recognizes its location as being something other than location "A" or "B", then the message is not stored. As another example, if the messaging device is in location "A" and receives a message with a target address of "789", then the message is stored. If the same message with the target address of "789" were received in a location the messaging device recognizes as location "E", then the message would not be stored.

In addition to the storage scheme described above, the messaging device would preferably have a display algorithm for messages already stored that could be controlled by a separate look up table such as the following:

LOCATION	MESSAGE	DISCARD	ALERT	DISPLAY
A	1		✓	✓
B	2		X	✓
C	3			
D	4		X	X
E	5	✓		
NOT F	6	✓		

Thus, in accordance with the look-up table above, a messaging device entering location "A" would alert the user and display message **1**. The user of a messaging device entering location "B" can see message **2**, but the user would not be alerted. In another instance, a user of a messaging device entering location "D" will not be alerted and will not be able to display message **4**. Depending upon the algorithm, message **4** may then subsequently be displayed when the messaging device recognizes that it is not within location "D". Further in accordance with the table above, the messaging device would discard message **5** upon recognition by the messaging device of its presence in location "E". Likewise, message **6** would be discarded upon recognition by the messaging device of a location other than location "F".

It should be understood that the disclosed embodiments are merely examples and the invention is not restricted thereto. It will be understood by those skilled in the art that variations and modifications can be made within the scope and spirit of the present invention as defined by the appended claims.

What is claimed is:

1. A method of processing spatially sensitive information at a selective call receiver, comprising the steps of:

receiving the spatially sensitive information along with an associated identifier;

storing the spatially sensitive information if the associated identifier is valid;

detecting a change in the associated identifier as the selective call receiver roams from a first area to a second area; and

processing the spatially sensitive information.

2. The method of claim **1**, wherein the method further comprises the step of deleting at least portions of the spatially sensitive information.

3. The method of claim **1**, wherein the method further comprises the step of at least displaying portions of the spatially sensitive information.

4. The method of claim **1**, wherein the method further comprises the step of deleting at least portions of the spatially sensitive information when the user prompts the selective call receiver to display at least portions of the spatially sensitive information.

5. The method of claim **1**, wherein the method further comprises the step of automatically deleting at least portions of the spatially sensitive information when the memory of the selective call receiver is filled beyond a threshold level and otherwise deleting said at least portions of the spatially sensitive information when the user prompts the selective call receiver to display a message.

6. A method of processing a message among a plurality of messages received over the air at a selective call receiver, comprising the steps of:

receiving the plurality of messages, at least a portion of the messages being received with a relevancy status that includes a space/time indicator,

storing the messages if the relevancy status matches a stored relevancy status at the selective call receiver;

tagging the messages for processing when the space/time indicator changes; and

processing the messages that were tagged, the step of processing further comprises the step of deleting e that have an invalid space/time indicator.

7. The method of claim **6**, wherein the step of tagging further comprises the step of tagging the messages when the space/time indicator indicates a change in location.

8. The method of claim **6**, wherein the step of tagging further comprises the step of tagging the messages when the space/time indicator indicates a change in time beyond a threshold period.

9. The method of claim **6**, wherein the step of processing further comprises the step of displaying at least portions of the messages that were tagged.

10. A messaging device for receiving a plurality of messages, at least a portion of the plurality of messages having a relevancy status, the messaging device comprising:

a selective call receiver for receiving the plurality of messages, the relevancy status for the portion of the plurality of messages, and updates to the relevancy status;

a memory for storing at least a portion of the plurality of messages and the updates to the relevancy status; and

a processor for processing the portion of the plurality of messages that have received updates to the relevancy status.

11. The messaging device of claim **10**, wherein the relevancy status is a location identifier.

12. The messaging device of claim **11**, wherein the location identifier is selected from the group consisting of Simulcast System Identifier (SSID), Network Identifier (NID), Service Provider Identifier (SPID), Zone Identifier, Subzone Identifier, market indicator, information identifier, transmitter colorcode, and a prohibited location identifier.

13. The messaging device of claim **10**, wherein the messaging device further comprises a real time clock.

14. The messaging device of claim **13**, wherein the messaging device receives an address that includes the relevancy status, wherein the relevancy status is selected from the group consisting of Simulcast System Identifier (SSID), Network Identifier (NID), Service Provider Identifier (SPID), Zone Identifier, Subzone Identifier, market indicator, information identifier, transmitter colorcode, prohibited location identifier, and timeout indicator.

15. The messaging device of claim **10**, wherein the messaging device further comprises a transmitter.

16. A messaging device for receiving a plurality of messages, at least a portion of the plurality of messages having a location identifier, the messaging device comprising:

a selective call receiver for receiving the plurality of messages, the location identifier for the portion of the plurality of messages, and updates to the location identifier;

a first memory location for storing a list of existing location identifiers;

a second memory location for storing at least a portion of the plurality of messages, the location identifier for the portion of the plurality of messages, and the updates to the location identifiers; and

a processor for processing the portion of the plurality of messages that have a location identifier by comparing the first memory location with the second memory location.

17. The messaging device of claim **16**, wherein the messaging device is a one-way selective call receiver and the processor stores a message and enables the display of the message if a match is found between the existing location identifiers and the location identifier associated with the message.

18. The messaging device of claims **16**, wherein the messaging device is a one-way selective call and the processor does not store a message if a match is found between the existing location identifiers and the location identifier associated with the message.

19. The messaging device of claim wherein the messaging device is a one-way selective call and the processor deletes an existing message from the second memory if a match is found between the existing location identifiers and the location identifier associated with the message.

20. The messaging device of claim **16**, wherein the messaging device further comprises a transmitter.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 5,883,580
DATED : March 16, 1999
INVENTOR(S) : Briancon et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 9,
Line 6, please change "deleting e that" to -- deleting messages that --.

Column 10,
Line 36, please add "16" after the word claim and before wherein.

Signed and Sealed this

Seventh Day of May, 2002

Attest:

Attesting Officer

JAMES E. ROGAN
Director of the United States Patent and Trademark Office

EXHIBIT 2

US005922047A

United States Patent [19]

Newlin et al.

[11] Patent Number: 5,922,047

[45] Date of Patent: Jul. 13, 1999

[54] **APPARATUS, METHOD AND SYSTEM FOR MULTIMEDIA CONTROL AND COMMUNICATION**

[75] Inventors: **Douglas J. Newlin**, Geneva; **Timothy M. Burke**, Algonquin, both of Ill.

[73] Assignee: **Motorola, Inc.**, Schaumburg, Ill.

[21] Appl. No.: **08/735,295**

[22] Filed: **Oct. 22, 1996**

[51] **Int. Cl.**[6] **G06F 17/00**

[52] **U.S. Cl.** **709/217**

[58] **Field of Search** 345/200.47; 364/131, 364/133, 188, 185, 138; 340/825.08; 709/217

[56] **References Cited**

U.S. PATENT DOCUMENTS

5,621,662	4/1997	Humphries et al.	364/185
5,774,357	6/1998	Hoffberg et al.	364/188

OTHER PUBLICATIONS

X10 FAQ version May 16, 1996.

International Organization for Standarization Organization Internationale de Normalization MPEG–2 FAQ's; Leonardo Chiariglione Jul. 1997.

MCSI OmniBand White Paper; Jim Esserman, 1998.

MCSI News Release; Motorola Multimedia Invests in MCSI; Ron Katznelson; Mar. 8, 1996.

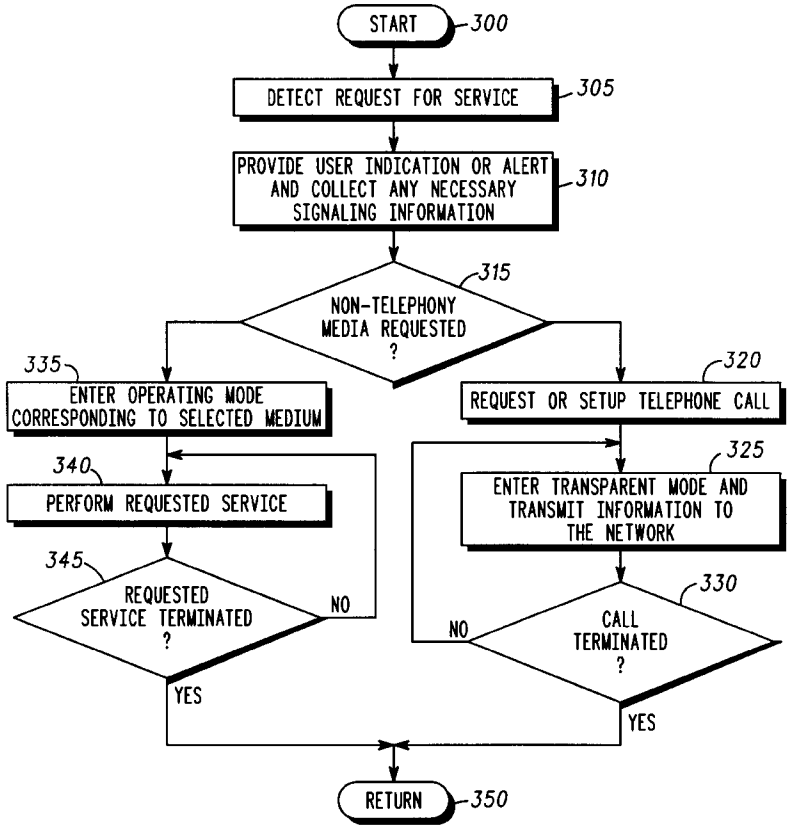
Primary Examiner—Ellis B. Ramirez

Attorney, Agent, or Firm—Nancy R. Gamburd; Gregory B. Gulliver

[57] **ABSTRACT**

A multimedia control apparatus (**101**, **201**) provides for control over a plurality of media (**145**), including telephony, video conferencing, video display, home automation and control, and other communication systems such as voice mail. The multimedia control apparatus includes a network interface (**110**) coupleable to a network communication channel (**103**) for communication with a network (**104**); a user interface (**120**) responsive to receive a first control signal of a plurality of control signals; a plurality of media application modules (**140**), each media application module of the plurality of media application modules coupleable to a corresponding medium of a plurality of corresponding media (**145**); and a processor arrangement (**130**) having a plurality of operating modes. The processor arrangement (**130**) is responsive, through a set of program instructions and in response to the first control signal, to select a first media application module of the plurality of media application modules, to enter a first operating mode of the plurality of operating modes, the first operating mode corresponding to the first media application module, and with the processor arrangement being further responsive, in conjunction with the first media application module, to control the corresponding medium.

33 Claims, 3 Drawing Sheets



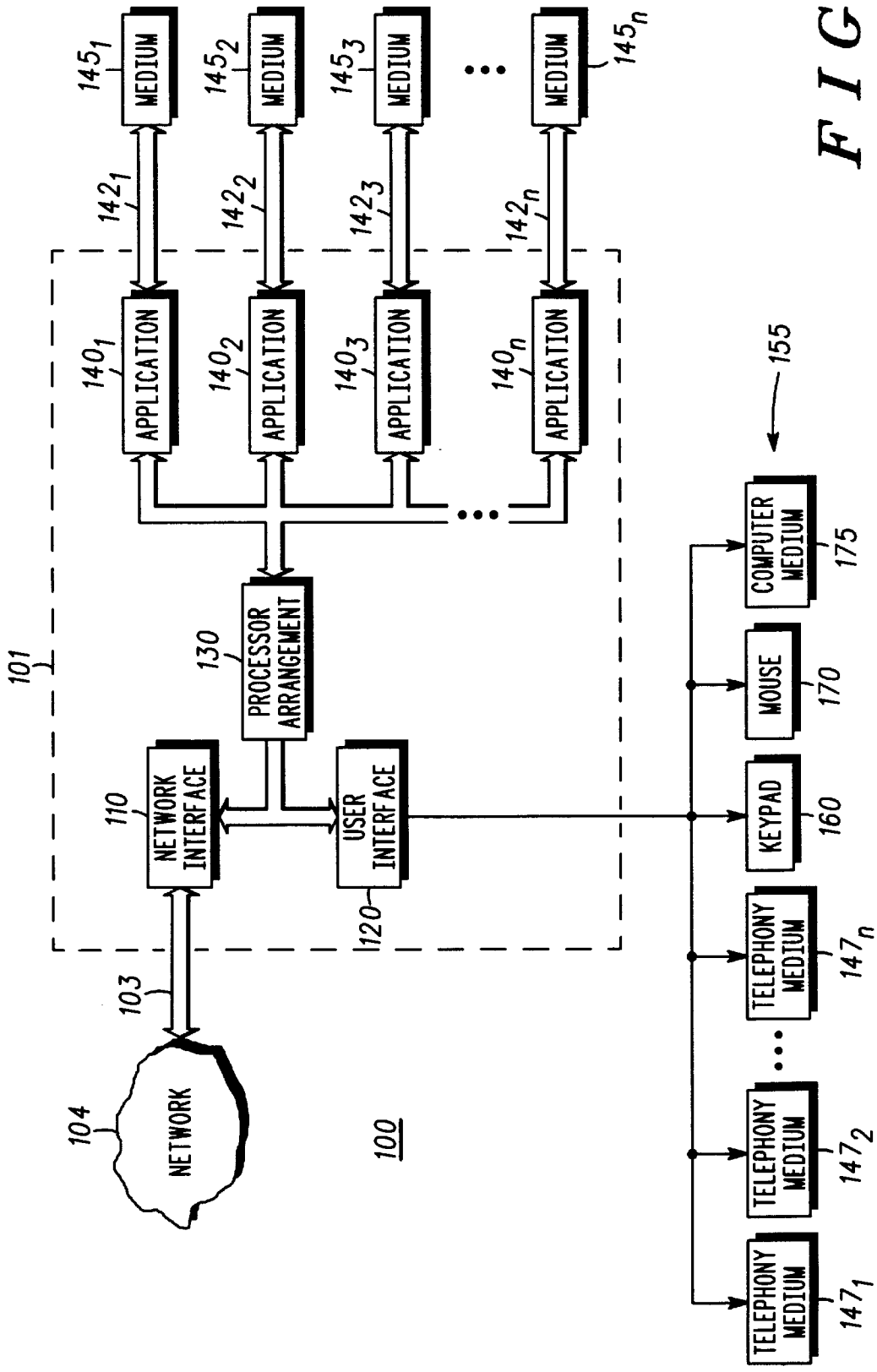

MEDIUM
APPLICATION
PROCESSOR ARRANGEMENT
NETWORK INTERFACE
USER INTERFACE
NETWORK
COMPUTER MEDIUM
MOUSE
KEYPAD
TELEPHONY MEDIUM
100
101
103
104
110
120
130
140
142
145
147
155
160
170
175

FIG. 1

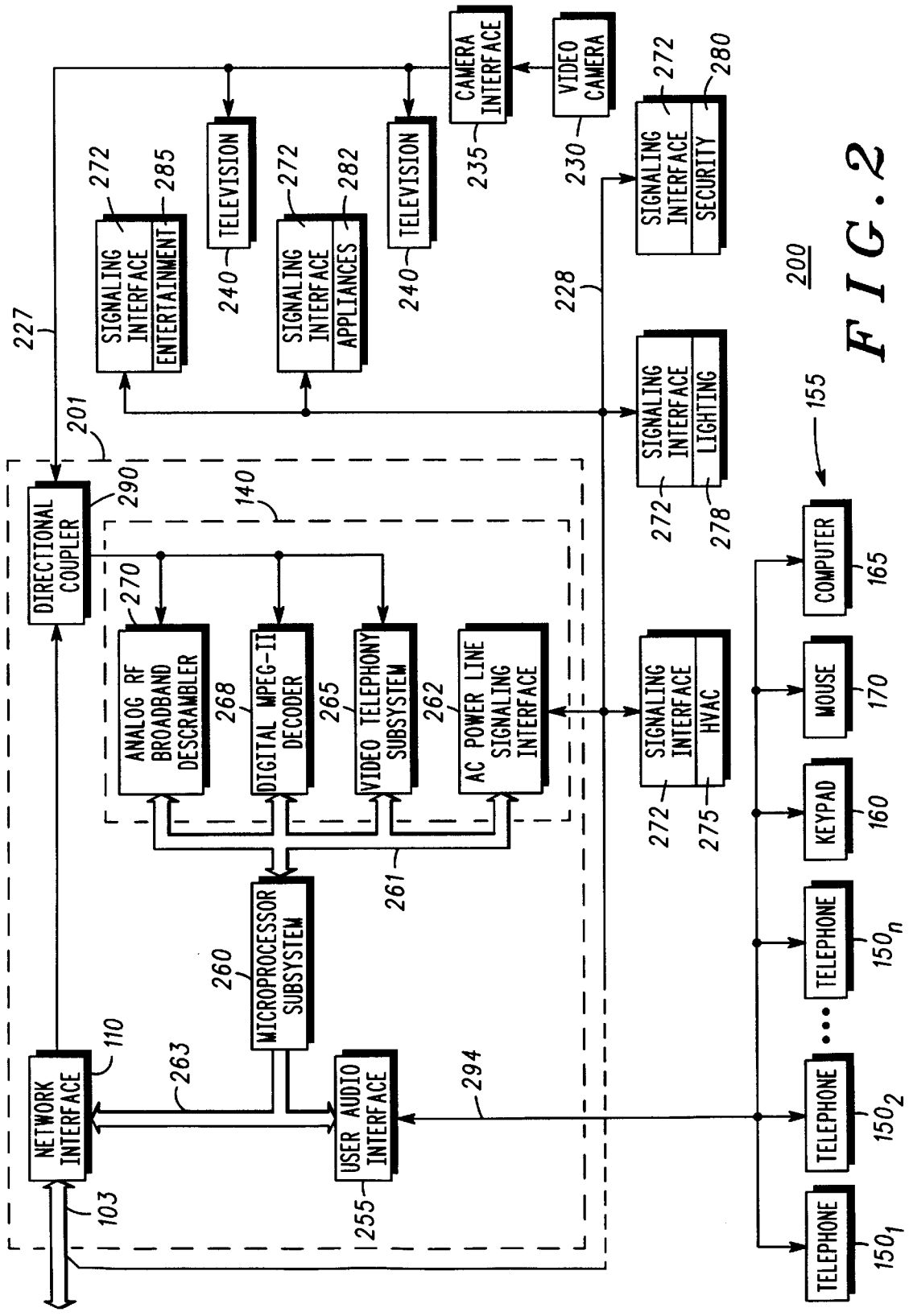

200
227
201
290
140
DIRECTIONAL COUPLER
270
ANALOG RF BROADBAND DESCRAMBLER
268
DIGITAL MPEG-II DECODER
265
VIDEO TELEPHONY SUBSYSTEM
262
AC POWER LINE SIGNALING INTERFACE
261
MICROPROCESSOR SUBSYSTEM
260
NETWORK INTERFACE
110
263
103
USER AUDIO INTERFACE
255
294
SIGNALING INTERFACE ENTERTAINMENT
272
285
SIGNALING INTERFACE APPLIANCES
272
282
TELEVISION
240
TELEVISION
240
CAMERA INTERFACE
235
VIDEO CAMERA
230
228
SIGNALING INTERFACE SECURITY
272
280
SIGNALING INTERFACE LIGHTING
272
278
155
SIGNALING INTERFACE HVAC
272
275
COMPUTER
165
MOUSE
170
KEYPAD
160
TELEPHONE
150n
TELEPHONE
1502
TELEPHONE
1501

FIG. 2

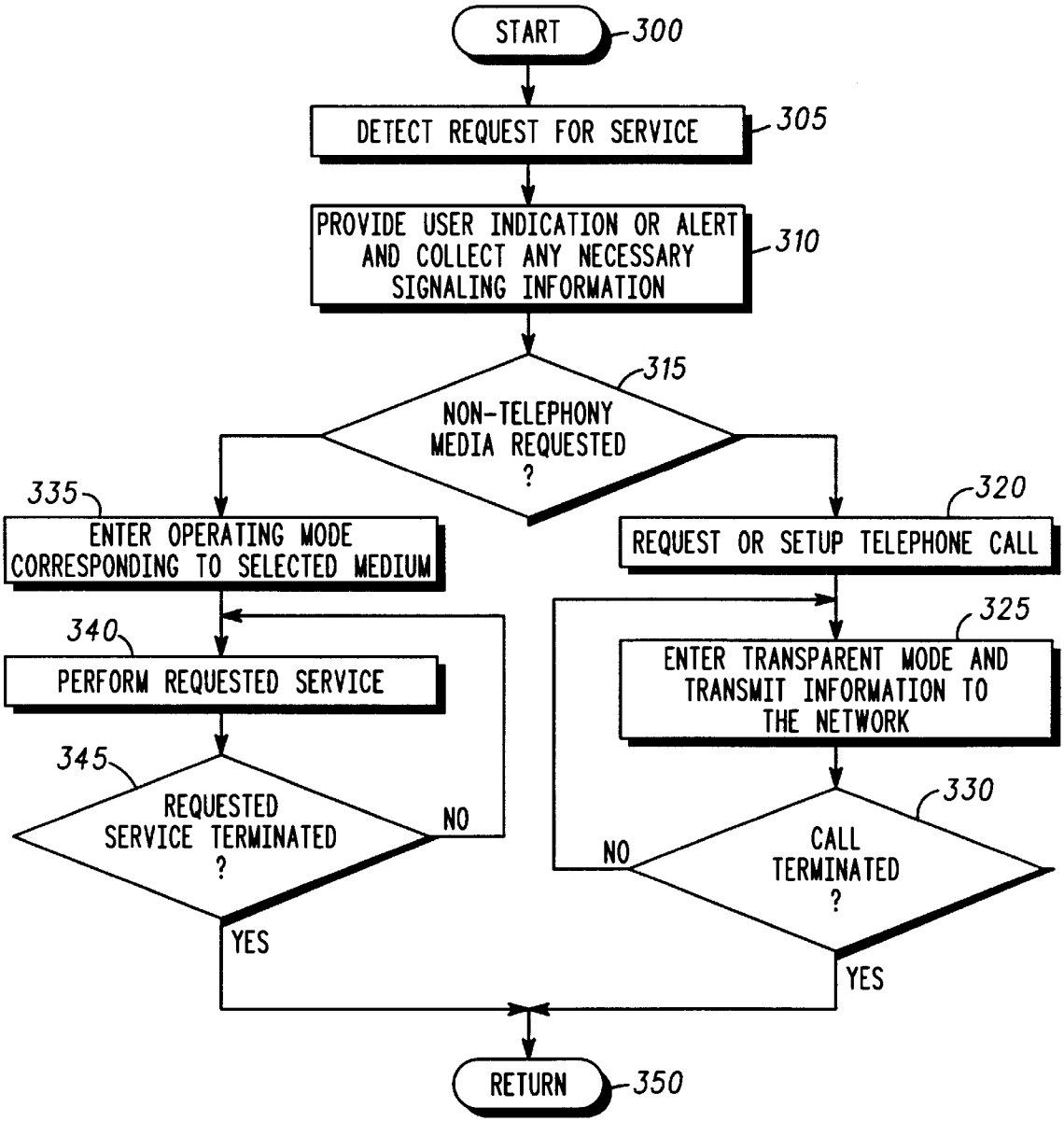

START
300
DETECT REQUEST FOR SERVICE
305
PROVIDE USER INDICATION OR ALERT AND COLLECT ANY NECESSARY SIGNALING INFORMATION
310
315
NON-TELEPHONY MEDIA REQUESTED ?
335
ENTER OPERATING MODE CORRESPONDING TO SELECTED MEDIUM
340
PERFORM REQUESTED SERVICE
345
REQUESTED SERVICE TERMINATED ?
NO
YES
320
REQUEST OR SETUP TELEPHONE CALL
325
ENTER TRANSPARENT MODE AND TRANSMIT INFORMATION TO THE NETWORK
330
CALL TERMINATED ?
NO
YES
RETURN
350

FIG. 3

APPARATUS, METHOD AND SYSTEM FOR MULTIMEDIA CONTROL AND COMMUNICATION

CROSS-REFERENCE TO RELATED APPLICATIONS

This application is related to the following U.S. Patent Applications (collectively referred to as the "related applications"), each incorporated by reference herein, with priority claimed for all commonly disclosed subject matter:

Newlin et al., U.S. patent application Ser. No. 08/658,792, filed Jun. 5, 1996, entitled "Audio/Visual Communication System and Method Thereof", Motorola Docket No. PD05634AM (the "first related application");

Burke et al., U.S. patent application Ser. No. 08/706,100, filed Aug. 30, 1996, entitled "Apparatus, Method And System For Audio And Video Conferencing And Telephony", Motorola Docket No. PD05686AM (the "second related application");

Burke et al., U.S. patent application Ser. No. 08/715,887, filed Sep. 18, 1996, entitled "Videophone Apparatus, Method And System For Audio And Video Conferencing And Telephony", Motorola Docket No. PD05689AM (the "third related application");

Burke et al., U.S. patent application Ser. No. 08/725,602, filed Oct. 3, 1996, entitled "Apparatus, Method And System For Wireline Audio And Video Conferencing And Telephony", Motorola Docket No. PD05703AM (the "fourth related application");

Burke et al., U.S. patent application Ser. No. 08/726,320, filed Oct. 3, 1996, entitled "Videophone Apparatus, Method And System For Wireline Audio And Video Conferencing And Telephony", Motorola Docket No. PD05725AM (the "fifth related application"); and

Newlin et al., U.S. patent application Ser. No. 08/672,819, filed Jun. 28, 1996, entitled "ISDN Communication System and Method Thereof", Motorola Docket No. PD05677AM (the "sixth related application").

FIELD OF THE INVENTION

This invention relates in general to communications and control systems and, more specifically, to an apparatus and method for multimedia control and communication.

BACKGROUND OF THE INVENTION

The inventions disclosed in the various related applications provide for both telephony and for audio/video conferencing, utilizing a video access apparatus coupleable via a communications channel to a telecommunications network. In the second and third related applications, the preferred embodiment of the video access apparatus provides for both telephony and for audio/video conferencing utilizing a CACS protocol for communication with a primary station via hybrid fiber coaxial cable, and the primary station, in turn, provides connectivity to a telecommunications network and a cable television services infrastructure. In the fourth and fifth related applications, the video access apparatus provides for both telephony and for audio/video conferencing with direct wireline connectivity to a telecommunications network, utilizing a wireline interface suitable, for example, for connection to an Integrated Services Digital Network ("ISDN") and/or to a Public Switched Telephone Network ("PSTN").

In the preferred embodiments of the second and fourth related applications, video conferencing capability is provided utilizing ordinary or common equipment and devices typically found in consumer or subscriber premises, such as telephones, televisions, and video cameras (video camcorders). In the third and fifth related applications, such video conferencing capability is provided utilizing one or more videophone apparatuses. Common to the inventions of each of the related applications, however, is the use of one user interface to select and control the various media applications, such as selecting telephony or video conferencing. In the preferred embodiments, one or more telephones may be used to provide for entry of various control signals, into an audio user interface of the video access apparatus, to select the operating mode of the video access apparatus. For example, in the preferred embodiment, entry of a predetermined sequence (such as "**" of the DTMF tones) is utilized to select a video conferencing mode, with a telephony mode automatically and transparently selected in the absence of the predetermined sequence.

In addition to various media applications such as telephony, stereo or other audio, and video conferencing, other media (or multimedia) applications are being introduced into consumer or subscriber environments. For example, current technology includes AC (alternating current) power line signaling capability to provide various control functions for typical consumer or subscriber premise equipment, such as heating, ventilating and air conditioning equipment ("HVAC"), and appliances such as dishwashers, clothes washers, ovens, ranges, and coffee makers. Current technology, however, has provided only for separate and application specific control systems, such as a remote control for audio stereo, another remote control for video (television and VCR), separate telephones, a separate thermostat, and separate interfaces to control AC power line signaling. For example, current technology has not provided for a singular interface which controls digital video, analog video, video conferencing, and telephony functions. In addition, each of these application specific control systems often include features redundantly found in other control systems, such as each having a separate information display, such as separate thermostat LCDs and separate audio stereo LCDs.

Other prior art multimedia systems have largely been computer based, requiring either personal computers or workstations which are configured additionally with multimedia computer processing boards, multimedia software or other programming, and specified audio/visual conferencing equipment, such as a video camera and microphone. In these computer-based multimedia systems, multimedia control is provided only for the computer-based multimedia applications via, for example, a mouse or keyboard, and do not provide for control over other, more diverse media applications. In addition, such multimedia control is typically limited to the specific location of the designated computer, and cannot operate from more than one designated node or location within the user premises.

Accordingly, a need has remained for a singular or integrated apparatus and corresponding method capable of providing control functions over multiple and diverse media applications. Such a multimedia control apparatus should also preferably operate at more than one designated node or location within the user premises, or may be mobile, or may be configured as needed for additional locations. In addition, such a singular or integrated apparatus should be user friendly, relatively or comparatively less expensive, and should avoid device or equipment redundancy.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram illustrating an embodiment of a multimedia control apparatus and multimedia control system in accordance with the present invention.

FIG. **2** is a block diagram illustrating a preferred embodiment of a multimedia control apparatus and multimedia control system in accordance with the present invention.

FIG. **3** is a flow diagram illustrating a multimedia control method in accordance with the present invention.

DETAILED DESCRIPTION OF THE INVENTION

As mentioned above, a need has remained for a singular or integrated apparatus capable of providing control functions over multiple and diverse media applications (also referred to as "multimedia"). The present invention provides for such a singular (or integrated) apparatus and method capable of providing such diverse multimedia control functions. In the preferred embodiment, the multimedia control apparatus provides for control over a plurality of media applications including telephony, video conferencing, analog video, digital video, and AC power line signaling (for control over premise or subscriber equipment such as HVAC, lighting, security, and entertainment). Also in the preferred embodiment, any connected telephone becomes a multimode telephone, providing the physical interface for telephony functions and for multimedia control functions. In addition, the multimedia control apparatus in accordance with the present invention is user friendly, is relatively or comparatively less expensive, and avoids device or equipment redundancy.

FIG. **1** is a block diagram illustrating an embodiment of a multimedia control apparatus **101** and multimedia control system **100** in accordance with the present invention. As illustrated in FIG. **1**, the multimedia control system **100** includes the multimedia control apparatus **101** and one or more media **145** (such as medium $\mathbf{145}_1$ through medium $\mathbf{145}_n$ (collectively media **145**)) or telephony media **147** (such as telephony medium $\mathbf{147}_1$ through telephony medium $\mathbf{147}_n$ (collectively telephony media **147**)). The various media **145** include, for example, video conferencing, video programming and video access control, home or premise automation and control, and voice mail and other messaging systems, with telephony media **147** being a special case or specific subset of media **145**. When the media **145** are limited to video conferencing and to telephony, the multimedia control apparatus **101** is substantially and effectively identical to the various video access apparatuses disclosed in the related applications.

Continuing to refer to FIG. **1**, the multimedia control apparatus **101** is coupleable, via a network interface **110**, to a first, network communication channel **103** for communication with a network **104**, and referred to as a network communication channel **103** to distinguish it from other communication channels of the multimedia control system **100**, such as those utilized to communicate with the various media **145**, which are referred to as media communication channels **142**. The network communication channel **103** may be wireline, such as one or more twisted pair wires, may be cable, such as hybrid fiber coaxial cable, may be wireless, such as for cellular or other radio frequency transmissions, or may be of any other appropriate communications medium. The network **104**, as disclosed in the fourth and fifth related applications, for example, may be a public switched telephone network ("PSTN") for ordinary telephony and analog data communications (also referred to as POTS, for plain old telephone service), or may be an Integrated Services Digital Network ("ISDN") for digital voice and data communications, or any combination of such existing or future telecommunications networks. For such wireline networks, the network communication channel **103** will usually be coupled to the network **104** through a local digital or analog switch (not illustrated). In addition, as disclosed in the second and third related applications, when a CACS communication protocol is implemented, the network communication channel **103** may be coupled to the network **104** via a primary station, which among other things provides one or more network interfaces utilizing other or additional protocols, such as various ISDN protocols, and also provides for connection to CATV (cable television) services infrastructure. The network **104** may also be an electric power network, such as that provided by an electric utility, which may also utilize various forms of communication and signaling such as PLC (power line carrier) for purposes such as variable rate, time of day billing. In such case, the network communication channel **103** may also include alternating current (AC) power lines, as discussed in greater detail below with reference to FIG. **2**.

The multimedia control apparatus **101** illustrated in FIG. **1** includes various features or components disclosed and discussed in detail in the related applications, including the network interface **110**, the processor arrangement **130**, and the user interface **120**. Depending upon the implementation of the multimedia control system **100**, such as wireline, cable or wireless, the network interface **110** of the multimedia control apparatus **101** will vary accordingly. For example, for cable applications, as disclosed in the second and third related applications, the network interface **110** be a cable network interface and will include a CATV transceiver and a communications ASIC (application specific integrated circuit), which perform various functions, respectively, such as radio frequency (RF) modulation and demodulation and CACS protocol encoding and decoding. For wireline applications, as disclosed in the fourth and fifth related applications, the network interface **110** will include a telephony interface and/or an ISDN interface, which perform various functions, respectively, such as analog telephony and analog modem functions (e.g., International Telecommunications Union (ITU) V.34 and V.34bis protocols), and implementation of various digital (ISDN) protocols for voice and data transmission (e.g., ITU Q.921 LAPD data link and Q.910 physical layer (interface) digital protocols). As discussed in the related applications, the network interface **110** is utilized to transmit and receive analog or digital video, audio, and other information and data (generally referred to as data), in any given format, protocol, or modulation scheme compatible with the network **104** and with any particular network connections or switches; such analog or digital signals are referred to herein as protocol signals, such as V.34 protocol signals or ISDN protocol signals. For example, for an analog POTS transmission, audio/video data (having been compressed and formatted utilizing ITU H.323 or H.324 protocols), is then further encoded and modulated utilizing ITU V.34 or V.34bis protocols. Also for example, when coupled to an ISDN via the network communication channel **103**, the network interface **110** will transmit and receive data in accordance with the ISDN series of protocols (such as the Q.x series).

Continuing to refer to FIG. **1**, the multimedia control apparatus **101** is coupleable to a plurality of media $\mathbf{145}_1$ through $\mathbf{145}_n$, via a corresponding media communication channel **142** (such as a first media communication channel $\mathbf{142}_1$ through a n^{th} media communication channel $\mathbf{142}_n$.

For example, medium $\mathbf{145}_1$ may be a television, with a corresponding first media communication channel $\mathbf{142}_1$ being, for example, a typical television cable, such as a 75 Ohm coaxial cable. As discussed in greater detail below, the

various media 145_1 through 145_n are operated and controlled through corresponding media application modules 140_1 through 140_n (collectively media application modules **140**). In addition, the correspondence between the various media **145**, the corresponding media communication channel **142**, and the media application modules **140**, may be, but in most instances will not be, a one-to-one correspondence. For example, one or more televisions, as media **145**, may be coupled through one or more branches or terminations of a coaxial cable, as a first media communication channel 142_1, to the multimedia control apparatus, and operated and controlled by a plurality of media application modules **140**, such as a video conferencing module, a digital video module, and an analog broadband video module. Other media **145**, such as home or premise automation and control, e.g., lighting, HVAC, premise entertainment and premise security, may be coupled via a premise AC power line, as a second media communication channel 142_2, to the multimedia control apparatus, and operated and controlled by a single media application module **140**, such as an AC power line signaling interface.

The various media application modules **140** are connected or coupled to a processor arrangement **130**. The processor arrangement **130**, in turn, is connected or coupled to the network interface **110** and to the user interface **120**. The functions of each of these components is also discussed in detail in the related applications concerning telephony and video conferencing media applications. Various functions of each of these components is also discussed in greater detail below with reference to FIGS. **2** and **3**. The user interface **120** is utilized for reception of a control signal of a plurality of control signals, such as a request to place a telephony call, a request to place an audio and video conference call, a request to perform AC line signaling, a request to display a digital video signal, a request to perform a video loop back function, and other control signals such as alerting signals of incoming telephony or audio and video conference calls. In the preferred embodiment, the user interface is implemented as a user audio interface **255** illustrated in FIG. **3** and as illustrated in the related applications. The processor arrangement **130**, in conjunction with the media application modules **140**, provide control functionality over the various media **145**, and as explained in the related applications and in greater detail below, may be comprised of a single integrated circuit ("IC"), or may include a plurality of integrated circuits or other components connected or grouped together, such as microprocessors, digital signal processors, ASICs, associated memory (such as RAM and ROM), and other ICs and components. As a consequence, as used herein, the term processor arrangement should be understood to equivalently mean and include a single processor, or arrangement of processors, microprocessors, controllers, or some other grouping of integrated circuits which perform the functions discussed in greater detail below. For example, in the preferred embodiment, the processor arrangement **130** is implemented as illustrated in FIG. **2** and in the related applications as a microprocessor subsystem **260**.

Continuing to refer to FIG. **1**, through the user interface **120**, the multimedia control apparatus **101** is coupled to one or more physical interfaces **155** (for a user to physically access the multimedia control apparatus for the input of one or more control signals), such as a plurality of telephony media 147_1 through 147_n (collectively referred to as telephony media **147**), to a keyboard **160**, to a computer mouse **170**, and to a computer medium **175**. The user interface **120** is implemented preferably as the user audio interface **255** illustrated in FIG. **2** and in the related applications. The telephony media **147** are implemented preferably as telephones or as videophones, with physical input of the plurality of control signals provided through a telephone keypad as DTMF (dual tone multifrequency) or pulse dialing signals, and with audio input and output provided through a microphone and speaker of the various telephones (or videophones). In addition to or in lieu of the telephony media **147**, the keyboard **160**, the mouse **170**, and/or the computer medium **175**, may also be utilized to input the plurality of control signals.

As discussed in greater detail below, the methodology of the present invention may be programmed and stored, as a set of program instructions for subsequent execution, in the processor arrangement **130** and its associated memory and other equivalent components. The set of program instructions may also be stored in any storage device, such as a memory IC, a floppy disk, a CD ROM, or any other readable or executable media. In the preferred embodiment, the processor arrangement **130** is utilized, in conjunction with a stored set of program instructions and in response to any control signals entered by the user or received from the network **104**, for a variety of functions. As a consequence, the processor arrangement **130** has a plurality of operating modes such as, for example, operating modes for ISDN data and voice transmissions, analog or digital video transmissions, video conferencing, POTS telephony, analog data transmission, or AC power line signaling. For example, for audio and video conferencing, the processor arrangement **130** is utilized (embodied as the microprocessor subsystem **260** and video telephony subsystem **265**), as described in the related applications, first, to convert a received protocol signal (from the network interface **110**) both to a baseband output video signal (to be modulated and transmitted to a television or other video display, as one or more of the media **145**) and to an output audio signal; and second, to convert both a baseband input video signal (a demodulated input video signal having originated from a camera interface **235** illustrated in FIG. **2** and the related applications) and an input audio signal, to a second protocol signal (to be modulated or formatted and transmitted by the network interface **110** to the network **104**). For other applications, the processor arrangement **130** may receive, via the user interface **120**, a control signal entered by a user through, for example, one or more of the telephony media **147**, and in response to the control signal and the set of (stored) program instructions, the processor arrangement **130** selects a first media application module of the plurality of media application modules **140**, such as video conferencing, enters a first operating mode of the plurality of processor operating modes, with the first operating mode corresponding to the first media application module, such as a video conferencing operating mode, and in conjunction with the first media application module, the processor arrangement **130** controls the corresponding medium, such as a video display and video camera utilized for the videoconferencing.

FIG. **2** is a block diagram illustrating a preferred embodiment of a multimedia control apparatus **201** and a multimedia control system **200** in accordance with the present invention. As discussed above, and as disclosed in detail in the related applications, the multimedia control apparatus **201** includes a network interface **110** (coupleable to the network communication channel **103** for communication with a network **104**); a user audio interface **255** (as a user interface **120**); and a microprocessor subsystem **260** (as or as part of the processor arrangement **130**). As one of the media application modules **140**, the multimedia control apparatus

201 includes a video telephony subsystem 265, for audio/video conferencing. In this preferred embodiment, the multimedia control apparatus 201 is substantially similar to and may include all of the components of the various video access apparatuses disclosed in the related applications, such as the network interface 110, the user audio interface 255, the microprocessor subsystem 260, and the video telephony subsystem 265. The multimedia control apparatus 201 differs from the various video access apparatuses disclosed in the related applications in that the multimedia control apparatus 201 has additional media application modules 140, namely, an analog RF broadband descrambler 270, a digital MPEG-II decoder 268, and an AC power line signaling interface 262.

As disclosed in detail in the related applications, the video telephony subsystem 265 further includes one or more RF modulators and demodulators, and an audio/video compression and decompression subsystem (preferably utilizing the ITU H.32x series of protocols). The video telephony subsystem 265, coupled through a directional coupler 290 to a second communication channel 227 (as one of the media communication channels 142), transmits an RF output video (or audio/video) signal for display on one or more televisions 240 (as media 145), and receives an RF input video (or audio/video) signal from the camera interface 235 and video camera 230, for subsequent transmission to the network 104.

As illustrated in the related applications, the microprocessor subsystem 260 consists of a microprocessor or other processing unit, such as the Motorola MC68LC302, and memory, which includes random access memory (RAM) and read-only memory (ROM), and in the preferred embodiment, also includes flash programmable memory (such as flash EPROM or E^2PROM), with communication provided over the busses 261 and 263 to the network interface 110, the user audio interface 255, the video telephony subsystem 265, an analog RF broadband descrambler 270, a digital MPEG-II decoder 268, and an AC power line signaling interface 262. The read only memory also utilizes flash programmable memory, such that the memory contents may be downloaded from the network 104. As a consequence, different versions of operating software (program instructions), such as upgrades, may be implemented without modifications to the multimedia control apparatus 201 and without user intervention. The microprocessor subsystem 260 provides control and configuration of the various media application modules 140, ordinary telephony call processing, digital telephony call processing, and is also used to implement an ISDN or other protocol stack when required for analog or digital video calls, such as ITU Q.931 message signaling.

Continuing to refer to FIG. 2, the multimedia control apparatus 201 also includes, as additional media application modules 140 connected between the microprocessor subsystem 260 and the second communication channel 227, an analog RF broadband descrambler 270 and a digital MPEG-II decoder 268. The analog RF broadband descrambler 270 and a digital MPEG-II decoder 268 are utilized, respectively, to descramble analog video or decode and decompress digital video, which may have been transmitted via the network communication channel 103 from, for example, a CATV services infrastructure, for display on the televisions 240. The multimedia control apparatus 201 also includes, as another media application module 140, an AC power line signaling interface 262, coupled to a third communication channel 228, such as an AC power line in a consumer or customer premise. As discussed above, because such an AC power line may also communicate with the network 104, the third communication channel 228 may also be considered to be part of the first, network communication channel 103. In turn, the third communication channel 228 is coupled, through the signaling interfaces 272, to HVAC 275, lighting 278, premise security 280, appliances 282, and premise entertainment 285, as instances of media 145. In the preferred embodiment, the AC power line signaling interface 262 and the signaling interfaces 272 are programmable, and are or include an Echelon LonWorks system incorporating Motorola MC143120 or MC143150 Neuron ICs and Echelon power line transceivers PLT 20 or PLT 21, which provide signaling and control over the media connected to the third communication channel 228 (which is preferably an AC power line), such as turning lighting on and off, opening and locking doors, turning HVAC on or off, changing volume settings of a home entertainment system, setting a thermostat to a desired temperature, turning a sprinkler system on or off, etc. Similar or equivalent AC power line signaling interfaces 262 and signaling interfaces 272 are also available from other manufacturers.

As disclosed in detail in the related applications, the user audio interface 255 is designed to interface with standard household telephone sets, including wireless devices and speaker phones, such as telephones 150_1 through 150_n. The user audio interface 255 is intended to support both audio POTS calls and video calls and, in conjunction with the network interface 110, also supports analog modem functions. In addition, in conjunction with one of the physical interfaces 155, such as a telephone 150, keyboard 160, mouse 170, or computer 165, the user audio interface provides for entry of the various control signals utilized, for example, to select media applications or place telephony or video calls. In the preferred embodiment, any of the telephones 150 are utilized for entry of the various control signals, and POTS calls are processed in a "transparent" mode, such that placing and receiving telephone calls occur as if no video conferencing or other multimedia functions were present. Also in the preferred embodiment, video calls and multimedia functions are processed as exceptions, requiring a designated or predetermined dialing sequence entered by the user to invoke a video call or another media function.

The various telephones 150 utilized in the preferred embodiment may be any type of ordinary telephone, including cordless (portable) telephones, corded telephones, DTMF or pulse dialing telephones, videophones, or speaker phones. A particularly innovative feature of the present invention, as discussed in greater detail below, is that when connected to a multimedia control apparatus, an ordinary telephone becomes a multimode telephone having a plurality of operating modes, such as for telephony, the audio portion of video conferencing, and multimedia control functionality. Also as discussed in greater detail below, use of such a multimode telephone also provides for mobility of control, such that any multimode telephone, including portable telephones, may provide multimedia control. For example, a user in an upstairs bedroom of a home may utilizing a telephone to control the home thermostat located in a downstairs hallway.

As disclosed in the related applications, the user audio interface 255 preferably includes a SLIC circuit (Subscriber Loop Interface Circuit) which provides "BORSHT" functions for telephone service within the user premises; ring generation; an audio codec for the audio portion of a video call or telephony call, providing analog-to-digital conversions for voice digitizing of the input (voice) audio signal originating from the microphone portion of one or more of

the telephones **150**$_1$ through **150**$_n$, and digital-to-analog conversion for voice recovery from an output digital voice data stream or signal (to create the output audio signal to the speaker portion of the telephones **150**$_1$ through **150**$_n$); and a programmable digital signal processor (DSP) and associated memory (referred to as the voice processing DSP in the related applications, to distinguish another DSP referred to as a video processing DSP). The DSP of the user audio interface **255** contains program memory and data memory to perform signal processing functions such as DTMF/dial pulse detection and generation, analog modem functions, call progress tone (dial tone, busy tone) generation, PCM-to-linear and linear-to-PCM (pulse code modulation) conversion, and speech prompt playback. The memory associated with the DSP, in the preferred embodiment, includes high density read only memory (referred to as speech ROM) containing PCM encoded (or compressed) speech segments used for interaction with the user, such as in prompting the user for keypad DTMF or dial pulse entry when in the video calling or other multimedia modes. In addition, optional speech random access memory may be used for user voice storage functions, and electrically alterable, programmable non-volatile (flash) memory for storage of programs (and updates) or algorithms.

The multimedia control apparatus **201** and, more specifically, the microprocessor subsystem **260** and the user audio interface **255** in the preferred embodiment, operate in a plurality of modes, which may vary according to the types of media applications supported. In the preferred embodiment, the plurality of operating modes may be divided into two groups, first, for POTS or other ordinary telephony, and second, for all other multimedia applications (such as video conferencing), which for ease of reference are referred to herein as non-telephony modes (with the understanding that such non-telephony modes include all non-ordinary POTS functionalities, such as videoconferencing (video telephony) or the audio portions of video conferencing). The telephony (POTS) mode is user transparent, as a default mode which is entered whenever the user goes off hook, for example, utilizing a telephone **150**. As discussed in greater detail below, the video conferencing and other media application modes are entered as an exception, through the user entering (dialing) a specific, predetermined sequence which, in the preferred embodiment, is not recognized as a telephony sequence. The various modes may be entered by or through the user entering a control signal locally, such as through a telephone **150**, or remotely, through control signals entered via a connection from the network **104** through the network interface **110**. In the telephony (POTS) mode, the DSP of the user audio interface **255** generates the customary "dial" tone when the user telephone (of the telephones **150**$_1$ through **150**$_n$) goes off hook. The user then enters the dialing sequence via the keypad of a telephone **150**, just as in known or customary telephone dialing. The DSP decodes the dialing digits and stores them in a calling memory buffer. Upon decoding the first two digits entered (which are not the first two digits of a specific predetermined non-telephony mode sequence), the DSP recognizes that the requested call is not for a non-telephony media application and, as a consequence, signals the microprocessor subsystem **260** to initiate a POTS call through the network interface **110**. When the call is granted (by the network **104**) and the audio link is established, the DSP forwards the stored digits to the local digital or analog switch and connects the audio paths between the user's telephone(s) **150** and the network **104**. From this point on, the DSP will not decode any dialed digits and will simply pass through the input and output PCM digital voice data stream, until the user's telephone goes on hook and the call is terminated.

Alternatively for a telephony session, the user audio interface **255** may create or maintain a connection to a central office of a network **104**, to provide transparency for telephony. Once the entry of the specific predetermined sequence for a non-telephony mode is detected, the user audio interface **255** breaks or terminates the central office connection, and enters the selected non-telephony mode, such as a video conferencing mode, under local control of the multimedia control apparatus **201**.

As indicated above, in the preferred embodiment, the user initiates a non-telephony, multimedia application mode, such as a video conferencing mode, as an exception to the normal telephony mode, by entering a specific predetermined sequence which is recognized by the DSP of the user audio interface **255** as a non-telephony multimedia application mode sequence. Alternatively, a plurality of non-telephony mode sequences may be utilized, with each predetermined sequence specific to a selected non-telephony multimedia application mode, such as a video mode. This methodology is also discussed below with reference to the flow chart of FIG. **3**. For example, for a non-telephony mode of the preferred embodiment, the first two digits of the specific, predetermined sequence are unique and specifically unused in a standard POTS call, such as "* *", and as a consequence, may specifically signal the DSP to enter a non-telephony multimedia application mode. Alternatively, other specific, predetermined sequences could be programmed by the user for recognition as a non-telephony mode by the DSP. Again, the various media application modes may be entered locally through one of the physical interfaces **155** or remotely via a connection through the network **104** and the network interface **110**. Immediately after decoding the two special digits or other specific predetermined sequence indicating a non-telephony mode, the multimedia control apparatus **201** initializes the multimedia control process, such as through the DSP generating, playing or displaying a speech or video prompt sequence, such as "Please select a call option or press the '#' key for help", which is stored in the speech ROM portion of memory of the user audio interface **255**. The action taken by the DSP will then be responsive to and depend upon the sequence entered or key pressed by the user following the initial prompt. For example, if the '#' key is pressed, the user may hear or see a menu of commands such as, for example, the following:

"To enter video conference mode, press 1"

"To enter video programming mode, press 2"

"To enter home or premise automation mode, press 3"

"To enter voice mail, press 4"

"To hear this menu again, press #'

Following the user's selection of the particular or specific non-telephony media application mode, such as a video conference mode, a sub-menu of commands is generated or displayed by the multimedia control system **201**. For example, if the user selected a video conference mode, the user may hear or see a sub-menu of commands such as the following:

"To place a Directory call, press *"

"To update the call Directory, press 2"

"To place a manual video call, press 3"

"To mute the camera, press 4"

"To view the camera on your television, press 5"

"To hear this menu again, press #'Also for example, if the user selected a home automation mode, the user may

hear or see a sub-menu of commands (or prompts) such as the following:

"To turn off all lights, press 1"

"To lower heat for sleeping, press 2"

"To set all home comfort settings for the parents, press 3"

"To arm the security system, press 4"

"To start the dishwasher, press 5"

"To start the lawn sprinklers, press 6"

Thus, in the preferred embodiment, an automated and user friendly prompting sequence is used to guide the user through the multimedia control operation or sequence, through a single (or integrated) physical interface, such as a telephone **150**, rather than through multiple and different (and often confusing) interfaces. It should be noted that in the preferred embodiment, various directory features may be used to simplify the multimedia control process. For example, after the user goes off hook and presses the '*' key three times followed by a single digit '1', '2'. . . '9', a video conference call may be placed automatically using a sequence of numbers stored in a directory for that digit. This feature may be necessary or desirable under a variety of circumstances, for example, when an ISDN call may require the entry of two separate 10-digit numbers to connect the call through the network **104**. Also as an option in the preferred embodiment, a more sophisticated system may store a simple name tag or other alphanumeric entry associated with the directory entry, created by the user, and played back to the user by the DSP of the user audio interface **255**. For example, a prompt in response to making a directory call may be: "To call 'grandma', press 1"; "To call 'mother', press 2"; "To call 'work', press 3"; in which the speech segments "grandma", "mother", and "work" are spoken by the user, and recorded and stored in memory of the multimedia control apparatus **201**. More sophisticated systems may include speaker/voice recognition techniques, to recognize the user selection, eliminating the need to press any keys on a telephone keypad or other manual entry of information into the user interface **120** or user audio interface **255**. It should also be noted that video call control functions, such as camera muting, unmuting, and local playback (loop back), also may be selected with the same user interface.

Other sophisticated systems for user interaction may also include use of the television **240** or other video display for on-screen visual display of a menu of options, with corresponding entry of user control signals, such as call control and placement information, occurring in a variety of ways, such as through the keypad of the telephones **150**, through a infrared remote control link with the multimedia control apparatus **201** (or **101**), or through the input video path via the second communication channel **227**. In this manner, in conjunction with a multimedia control apparatus **101** or **201**, any of the physical interfaces **155**, such as the telephone keypad or remote control link or mouse, coupled with a video display such as a television **240**, may effectively form a distributed graphical user interface for multimedia control. For example, a keypad of a telephone **150** may operate as a cursor displayed on a television **240**, selecting menu items displayed or navigating a series or sequence of windows. As a consequence, an embodiment of the present invention includes distributed graphical user interface, in which the distributed graphical user interface comprises, first, a physical interface **155**, such as a telephone **150** or a mouse **170**, with the physical interface responsive to an entry of a plurality of control signals; second, a video display such as a television **240**, in which the video display is responsive to generate a first graphical display (such as a first window or a first menu), of a plurality of graphical displays, from a first video signal of a plurality of video signals; and third, a multimedia control apparatus (**101** or **201**) coupled to the physical interface **155** and to the video display via at least one communication channel (such as media communication channel **227**), the multimedia control apparatus having a processor arrangement **130** responsive, through a set of program instructions and in response to a first control signal of the plurality of control signals, to generate the first video signal, the multimedia control apparatus (**101** or **201**) further having a modulator (such as the modulator included within the video telephony subsystem **265** and as shown in the related applications) to transmit the first video signal to the video display via the at least one communication channel. The graphical display of the distributed graphical user interface may also be altered or changed, for example, with the processor arrangement **130** being further responsive, through the set of program instructions and in response to a second control signal of the plurality of control signals, to generate a second video signal of the plurality of video signals, and with the modulator also being responsive to transmit the second video signal to the video display, and wherein the video display generates a second graphical display, of the plurality of graphical displays, from the second video signal.

These various methods of user prompting, on-screen display, and user feedback are especially useful to guide the user through the process of placing a video call, and help to make audio video conferencing and other multimedia applications especially user-friendly. In addition, these various methods also illustrate the "tri-ality" of the modes of a telephone **150** in the preferred embodiment, for telephony, for audio input and output, and for multimedia control.

Network configuration is yet another function which may be performed via a telephone **150** (or other physical interface **155**) and the user audio interface **255**, especially utilizing menu options displayed utilizing an on screen display (on a television **240** or other video display). For example, as disclosed in the sixth related application, automatic ISDN configuration capabilities, for example, for ISDN parameters such as switch type and SPID, may be implemented within the microprocessor subsystem **260** (or processor arrangement **130**) and executed by the user via control functionality (as options entered by the user via the telephone **150** or other physical interface **155**). In addition, for POTS video conferencing capability, V.x or other modem configuration parameters (such as auto or manual answer) may also be configured as options entered by the user via the telephone **150** or other physical interface **155**.

FIG. **3** is a flow diagram illustrating a multimedia control method in accordance with the present invention. FIG. **3** also illustrates the multiple roles or modes of a telephone, such as telephones $\mathbf{150}_1$ through $\mathbf{150}_n$, in the system of the present invention, including providing telephony (POTS), providing multimedia control, and providing the audio portion of a video conference call. Referring to FIG. **3**, beginning with start step **300**, a request for service is detected, step **305**, such as going off hook or receiving an incoming alert signal. Next, in step **310**, a user indication or alert is provided, such as a dial tone or an incoming ring signal, and signaling information is collected, such as DTMF digits of a phone number or "**". When a non-telephony mode has been requested in step **315**, such as through entry of "**" or receipt of an incoming message from the network **104**, then the method proceeds to step **335**. When a non-telephony mode has not been requested in step **315**, the method

proceeds to request or set up a telephony call, such as generating DTMF tones and connecting an audio path between the user's telephone and the network **104**, step **320**, followed by entering the transparent telephony mode and transmitting audio (typically PCM) data to the network **104**, step **325**. The audio data will have been PCM encoded by the user audio interface **255**, and will have been transformed into an appropriate digital or analog format (e.g., ISDN, POTS, etc.) by the network interface **110** for transmission to the network **104**. When the telephony call is terminated, step **330**, the method may end, return step **350**.

Continuing to refer to FIG. **3**, when a non-telephony mode has been requested in step **315**, the method proceeds to step **335** and initializes and enters an operating mode corresponding to the selected media application. For example, for a selected video conference application, the multimedia control method will initialize and enter the video conference mode, such as by playing an initial speech prompt as discussed above. Next, in step **340**, the method performs the requested service such as, for example, for video conference mode, collecting a video input request type and originating a video conference call using a directory, updating a video conference call directory, manually originating a video conference call, muting an input (audio or video), providing loop back (e.g., local self-view such as monitoring or other surveillance), playing help or error messages or menu options, or exiting the video conferencing control system. For other modes, the requested service may include selecting a digital video channel for viewing, descrambling an analog video channel for viewing, or initiating an AC power line signaling interface command to raise or lower a temperature setting. In step **345**, when the requested media application service is terminated, such as by going on hook in a video call, the method may end, return step **350**.

In summary, FIGS. **1** through **3** disclose an apparatus for multimedia control (**101** or **201**), with the apparatus comprising, first, a network interface **110**, the network interface **110** coupleable to a first communication channel **103** for communication with a network **104**; second, a user interface **120**, the user interface **120** responsive to receive a first control signal of a plurality of control signals; third, a plurality of media application modules **140**, each media application module **140** of the plurality of media application modules **140** coupleable to a corresponding medium **145** of a plurality of corresponding media **145**; and fourth, a processor arrangement **130**, the processor arrangement **130** coupled to the network interface **110**, to the user interface **120** and to the plurality of media application modules **140**, the processor arrangement **130** having a plurality of operating modes, the processor arrangement **130** responsive, through a set of program instructions and in response to the first control signal, to select a first media application module **140** of the plurality of media application modules **140**, to enter a first operating mode of the plurality of operating modes, the first operating mode corresponding to the first media application module **140**, the processor arrangement further responsive, in conjunction with the first media application module, to control the corresponding medium **145**.

Also in summary, the method embodiment of the present invention may also be stored as a set of program instructions. As a consequence, an embodiment of the present invention also includes a data storage device, such as a memory or a disk, readable by a processor arrangement **130** in a multimedia control apparatus (**101** or **201**), the storage device encoding processor executable instructions for multimedia control, with the data storage device comprising, first, instruction means for selecting a first media application of a plurality of media applications (FIG. **3**, steps **305**, **310** and **315**); second, instruction means for entering a first operating mode of a plurality of operating modes, the first operating mode corresponding to the first media application (FIG. **3**, steps **335** or **320**); and third, instruction means for controlling a first medium of a plurality of media, the first medium corresponding to the first media application (FIG. **3**, steps **340** or **325**).

Numerous advantages from the various multimedia control apparatuses **101** and **201**, and from the various multimedia control systems **100** and **200**, are readily apparent. First, the various multimedia control apparatuses, in accordance with the present invention, provide a singular (or integrated) apparatus and method capable of providing diverse multimedia control functions. In the preferred embodiment, the multimedia control apparatus provides for control over a plurality of media applications including telephony, video conferencing, analog video, digital video, voice mail, and AC power line signaling (for control over premise or subscriber equipment such as HVAC, lighting, security, and entertainment). As a consequence, a user does not need to learn many different sets of commands or menus to operate a plurality of disparate multimedia applications; instead, a user may utilize a single device, such as a telephone located anywhere within the user premises, to control all applications throughout the premises.

Another particularly innovative feature of the various apparatus and system embodiments of the present invention is the "tri-ality" of the use of a telephone **150** in the preferred embodiment for telephony (POTS), for audio input and output (for video, CACS, ISDN or POTS), and for multimedia control (for selecting video conferencing, other video modes, other multimedia modes, or telephony modes). Another related significant feature of the present invention is the ability of any connected telephone **150** in the user's premises to have such a "tri-ality" of functionality, by virtue of being connected to the multimedia control apparatus via typical premise telephone wiring. Any connected telephone, including portable telephones and speaker phones, becomes a multimode telephone, providing the physical interface for telephony functions, for audio functions, and for multimedia control functions. Such multimedia control may be both mobile and available from a plurality of locations within a user's premises. As a consequence, the user is not confined to a single location, such as at a PC, a dedicated control panel, or in a dedicated conference room, for multimedia control capability. In addition, the system may be configured as needed for additional locations, for example, simply by adding or removing telephones or other physical interfaces.

Yet another significant feature of the preferred embodiment of the present invention is the transparency of telephony operation, such that a user need not be aware of the multimedia capability to place or receive a telephone call. In addition, in accordance with the preferred embodiment, the multimedia control system utilizes equipment typically found in consumers' homes or premises, such as existing telephones, and for video applications, existing televisions and video cameras or camcorders. The preferred embodiment also avoids device redundancy, eliminating multiple user interfaces, displays, and control systems and panels. As a consequence, the system may be implemented at relatively low cost, especially compared to the currently available PC-based or stand alone multimedia systems. Moreover, the system of the present invention is user friendly, easy to install and use, avoids device or equipment redundancy, and should be relatively less expensive for in-home purchase and use by consumers.

From the foregoing, it will be observed that numerous variations and modifications may be effected without departing from the spirit and scope of the novel concept of the invention. It is to be understood that no limitation with respect to the specific methods and apparatus illustrated herein is intended or should be inferred. It is, of course, intended to cover by the appended claims all such modifications as fall within the scope of the claims.

What is claimed is:

1. An apparatus for multimedia control, the apparatus comprising:

a network interface, the network interface coupleable to a first communication channel for communication with a network;

a user interface, the user interface responsive to selectively receive a first control signal of a plurality of control signals or an audio signal;

a plurality of media application modules, each media application module of the plurality of media application modules coupleable to a corresponding medium of a plurality of corresponding media; and

a processor arrangement, the processor arrangement coupled to the network interface, to the user interface and to the plurality of media application modules, the processor arrangement having a plurality of operating modes, the processor arrangement responsive, through a set of program instructions and in response to the first control signal, to select a first media application module of the plurality of media application modules, to enter a first operating mode of the plurality of operating modes, the first operating mode of the plurality of operating modes, the first operating mode corresponding to the first media application module, the processor arrangement further responsive, in conjunction with the first media application module, to control the corresponding medium.

2. The multimedia control apparatus of claim **1**, wherein the processor arrangement is further responsive to select a telephony media application module, of the plurality of media application modules, in the absence of the first control signal designating a non-telephony media application module of the plurality of media application modules.

3. The multimedia control apparatus of claim **1**, wherein the processor arrangement is further responsive to generate and transmit to the user interface a plurality of user prompt signals, each user prompt signal of the plurality of user prompt signals corresponding to a media application module of the plurality of media application modules.

4. The multimedia control apparatus of claim **3**, wherein the user interface is further responsive to receive the first control signal as a selection responsive to the plurality of user prompt signals.

5. The multimedia control apparatus of claim **1**, wherein the plurality of media application modules includes telephony control, video conferencing control, digital video control, analog video control, audio/video entertainment control, HVAC control, lighting control, security control, and appliance control.

6. The multimedia control apparatus of claim **1**, wherein the user interface is further coupleable to a physical interface for entry of the first control signal.

7. The multimedia control apparatus of claim **6**, wherein the physical interface is a telephone.

8. The multimedia control apparatus of claim **1**, wherein the network interface is an ISDN interface.

9. The multimedia control apparatus of claim **1**, wherein the network interface is an POTS telephony interface.

10. The multimedia control apparatus of claim **1**, wherein the network interface is a cable network interface.

11. A method for multimedia control, the method comprising:

(a) selecting a first media application of a plurality of media applications;

(b) entering a first operating mode of a plurality of operating modes, the first operating mode corresponding to the first media application; and

(c) controlling a first medium of a plurality of media, the first medium corresponding to the first media application.

12. The method for multimedia control of claim **11**, wherein step (a) further comprises selecting a telephony media application in the absence of an express selection of a non-telephony media application of the plurality of media applications.

13. The method for multimedia control of claim **12**, wherein the selection of the telephony media application is user transparent.

14. The method for multimedia control of claim **11**, wherein step (a) further comprises generating a plurality of user prompt signals, each user prompt signal of the plurality of user prompt signals corresponding to a media application of the plurality of media applications.

15. The method for multimedia control of claim **14**, wherein step (a) further comprises receiving a first control signal, of a plurality of control signals, as a selection responsive to the plurality of user prompt signals.

16. The method for multimedia control of claim **11**, wherein the plurality of media applications includes telephony control, video conferencing control, digital video control, analog video control, audio/video entertainment control, HVAC control, lighting control, security control, and appliance control.

17. An apparatus comprising:

a multimode telephone, the multimode telephone coupleable to a multimedia control apparatus for communication with a network via a first communication channel, the multimode telephone having a plurality of operating modes, the plurality of operating modes including:

a first operating mode in which the multimode telephone operates in a first media application, of a plurality of media applications, for audio telephony communication with the network; and

a second operating mode in which the multimode telephone operates for entry of a first control signal of a plurality of control signals for selection and control of a second media application of the plurality of media applications, the second media application being a non-telephony media application.

18. The apparatus of claim **17** wherein the multimode telephone is mobile.

19. A data storage device readable by a processor arrangement in a multimedia control apparatus, the storage device encoding processor executable instructions for multimedia control, the data storage device comprising:

instruction means for selecting a first media application of a plurality of media applications;

instruction means for entering a first operating mode of a plurality of operating modes, the first operating mode corresponding to the first media application; and

instruction means for controlling a first medium of a plurality of media, the first medium corresponding to the first media application.

20. The data storage device of claim 19, further comprising instruction means for selecting a telephony media application, of the plurality of media applications, in the absence of an express selection of a non-telephony media application of the plurality of media applications.

21. The data storage device of claim 19, further comprising instruction means for generating a plurality of user prompt signals, each user prompt signal of the plurality of user prompt signals corresponding to a media application of the plurality of media applications.

22. The data storage device of claim 21, further comprising instruction means for receiving a first control signal, of a plurality of control signals, as a selection responsive to the plurality of user prompt signals.

23. An apparatus comprising:

a processor arrangement having a plurality of operating modes, the plurality of operating modes including:

a first operating mode in which the processor arrangement operates in a first media application of a plurality of media applications, for audio telephony communication with a telecommunications network; and

a second operating mode in which the processor arrangement, in response to a first control signal of a plurality of control signals, operates in a second media application of the plurality of media applications, the second media application being a non-telephony media application.

24. The apparatus of claim 23, wherein the processor arrangement is responsive to select the first operating mode in the absence of the first control signal.

25. The apparatus of claim 23, wherein the processor arrangement is responsive to generate a plurality of user prompt signals, each user prompt signal of the plurality of user prompt signals corresponding to a media application of the plurality of media applications.

26. The apparatus of claim 25, wherein the processor arrangement is further responsive to the first control signal as a selection responsive to the plurality of user prompt signals.

27. The apparatus of claim 23, wherein the plurality of media applications includes telephony control, video conferencing control, digital video control, analog video control, audio/video entertainment control, HVAC control, lighting control, security control, and appliance control.

28. The apparatus of claim 23, wherein the processor arrangement further comprises:

a microprocessor; and

a memory coupled to the microprocessor.

29. A distributed graphical user interface comprising:

a physical audio interface, the physical audio interface responsive to an entry of a plurality of control signals;

a video display, the video display responsive to generate a first graphical display, of a plurality of graphical displays, from a first video signal of a plurality of video signals; and

a multimedia control apparatus coupled to the physical audio interface and to the video display via at least one communication channel, the multimedia control apparatus having a processor arrangement responsive, through a set of program instructions and in response to a first control signal of the plurality of control signals, to generate the first video signal, the multimedia control apparatus further having a modulator to transmit the first video signal to the video display via the at least one communication channel.

30. The distributed graphical user interface of claim 29, wherein the physical interface is a telephone.

31. The distributed graphical user interface of claim 29, wherein the physical interface is mobile.

32. The distributed graphical user interface of claim 29, wherein the video display is a television.

33. The distributed graphical user interface of claim 29, wherein the processor arrangement of the multimedia control apparatus is further responsive, through the set of program instructions and in response to a second control signal of the plurality of control signals, to generate a second video signal of the plurality of video signals, the modulator responsive to transmit the second video signal to the video display, and wherein the video display is further responsive to generate a second graphical display, of the plurality of graphical displays, from the second video signal.

* * * * *

EXHIBIT 3

US006425002B1

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

(10) **Patent No.: US 6,425,002 B1**
(45) **Date of Patent: Jul. 23, 2002**

(54) **APPARATUS AND METHOD FOR HANDLING DISPATCHING MESSAGES FOR VARIOUS APPLICATIONS OF A COMMUNICATION DEVICE**

(75) Inventors: **Rodd Bryan Zurcher**; **David Frank Baum**, both of Lake Zurich; **James Van Peursem**, Hanover Park, all of IL (US)

(73) Assignee: **Motorola, Inc.**, Schaumburg, IL (US)

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

(21) Appl. No.: **09/198,095**

(22) Filed: **Nov. 23, 1998**

(51) **Int. Cl.**[7] **G06F 15/173**
(52) **U.S. Cl.** **709/223**; 709/202; 709/206; 709/224; 709/314
(58) **Field of Search** 709/200–203, 709/204, 205–206, 213, 223–224, 227, 313–315; 455/442, 418–420, 461–463

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,335,347 A * 8/1994 Foss et al. 709/314
5,517,662 A * 5/1996 Coleman 709/201
5,634,005 A * 5/1997 Matsuo 709/206
5,696,902 A * 12/1997 Leclercq et al. 709/223
5,742,905 A * 4/1998 Pepe et al. 455/461
5,758,083 A * 5/1998 Singh et al. 709/223
5,832,221 A * 11/1998 Jones 709/206
5,892,916 A * 4/1999 Gehlhaar et al. 709/223
5,964,830 A * 10/1999 Durrett 709/200
6,073,142 A * 6/2000 Geiger et al. 709/204
6,226,666 B1 * 5/2001 Chang et al. 709/202

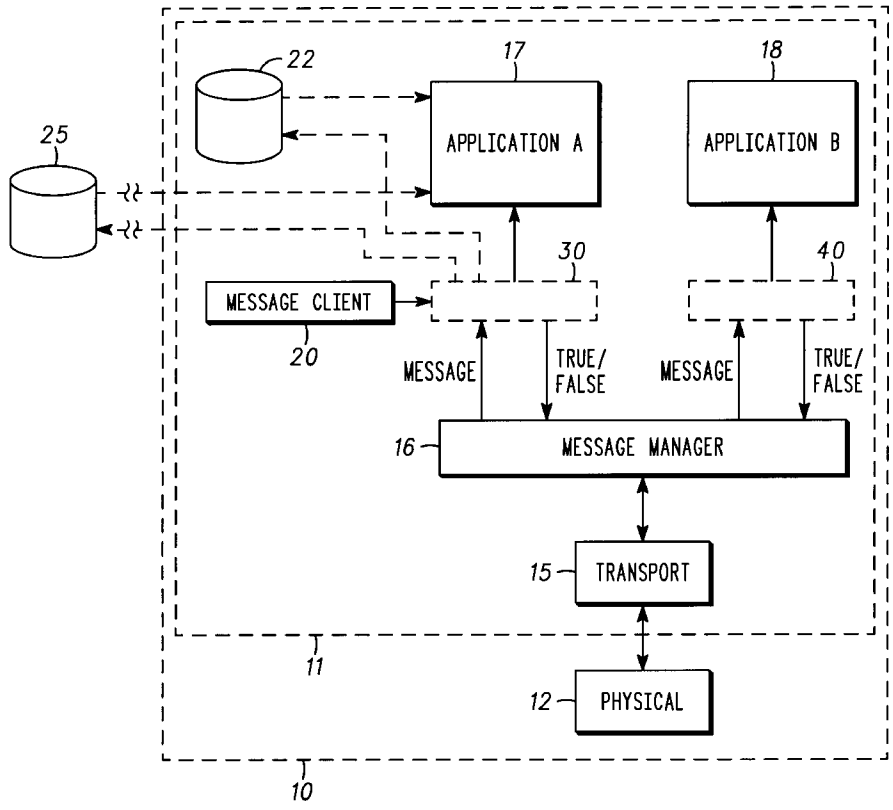
* cited by examiner

Primary Examiner—Bharat Barot
(74) *Attorney, Agent, or Firm*—Hugh C. Dunlop; Romi N. Bose; Hisashi D. Watanabe

(57) **ABSTRACT**

A communications device **(10)** with a processor having instructions that execute on the processor. The programmed instructions include a message manager program **(16)** for accepting and dispatching messages, one or more application programs **(17, 18)** for handling and presenting messages; and one ore more message client programs **(30, 40)** that receive messages from the message manager program **(16)** and provides them to the application program. The message manager program **(16)** accepts a registration from each message client program and sets rules, and message attributes to which the rules are to be applied for new messages.

11 Claims, 4 Drawing Sheets



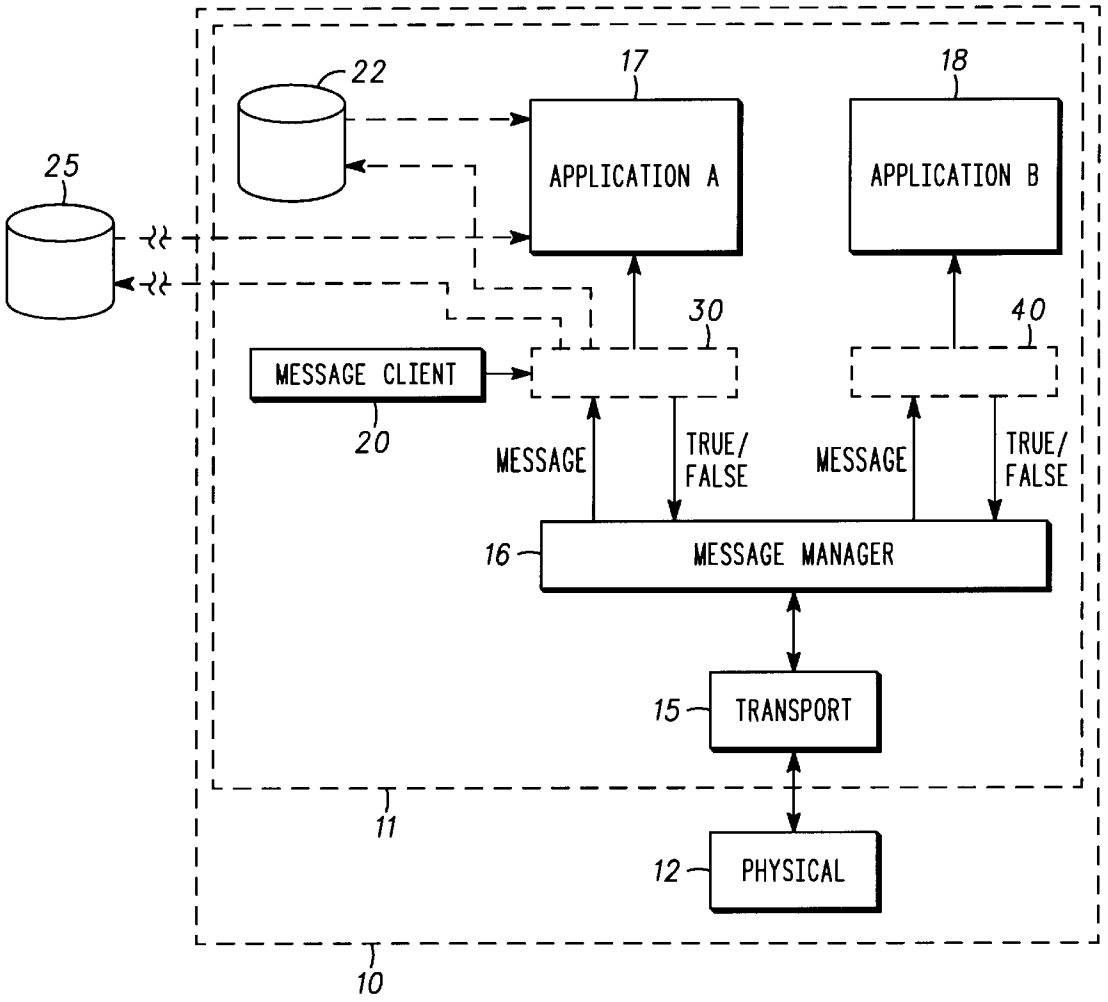

22
17
18
25
APPLICATION A
APPLICATION B
30
40
MESSAGE CLIENT
20
MESSAGE
TRUE/
FALSE
MESSAGE
TRUE/
FALSE
16
MESSAGE MANAGER
15
TRANSPORT
11
12
PHYSICAL
10

FIG. 1

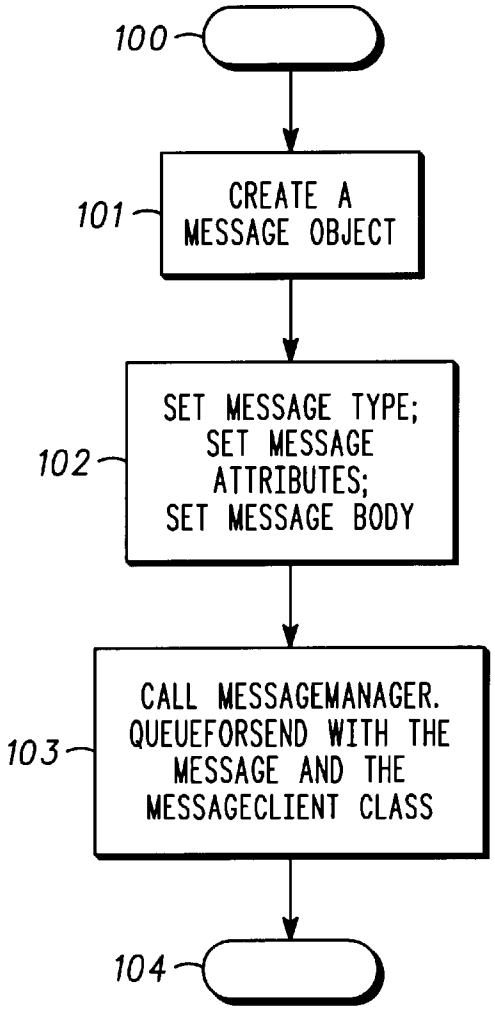

100
101
CREATE A MESSAGE OBJECT
102
SET MESSAGE TYPE; SET MESSAGE ATTRIBUTES; SET MESSAGE BODY
103
CALL MESSAGEMANAGER. QUEUEFORSEND WITH THE MESSAGE AND THE MESSAGECLIENT CLASS
104

FIG.2

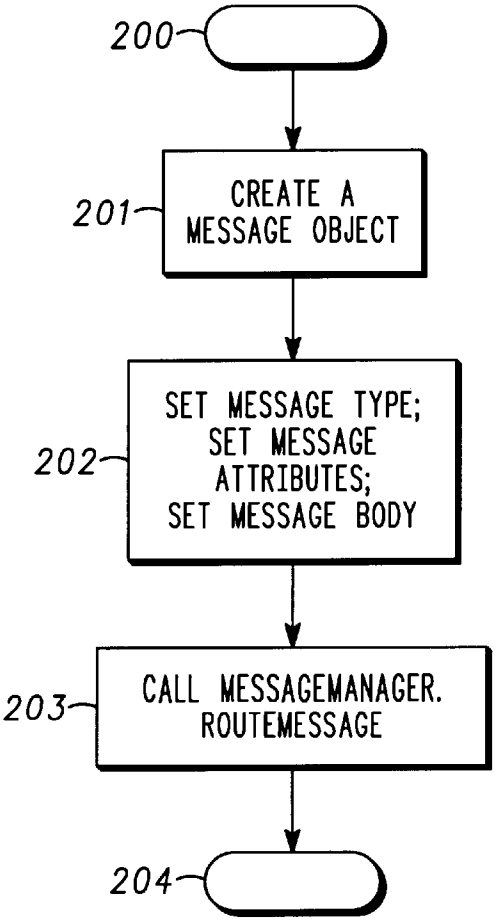

200
201
CREATE A MESSAGE OBJECT
202
SET MESSAGE TYPE; SET MESSAGE ATTRIBUTES; SET MESSAGE BODY
203
CALL MESSAGEMANAGER. ROUTEMESSAGE
204

FIG.3

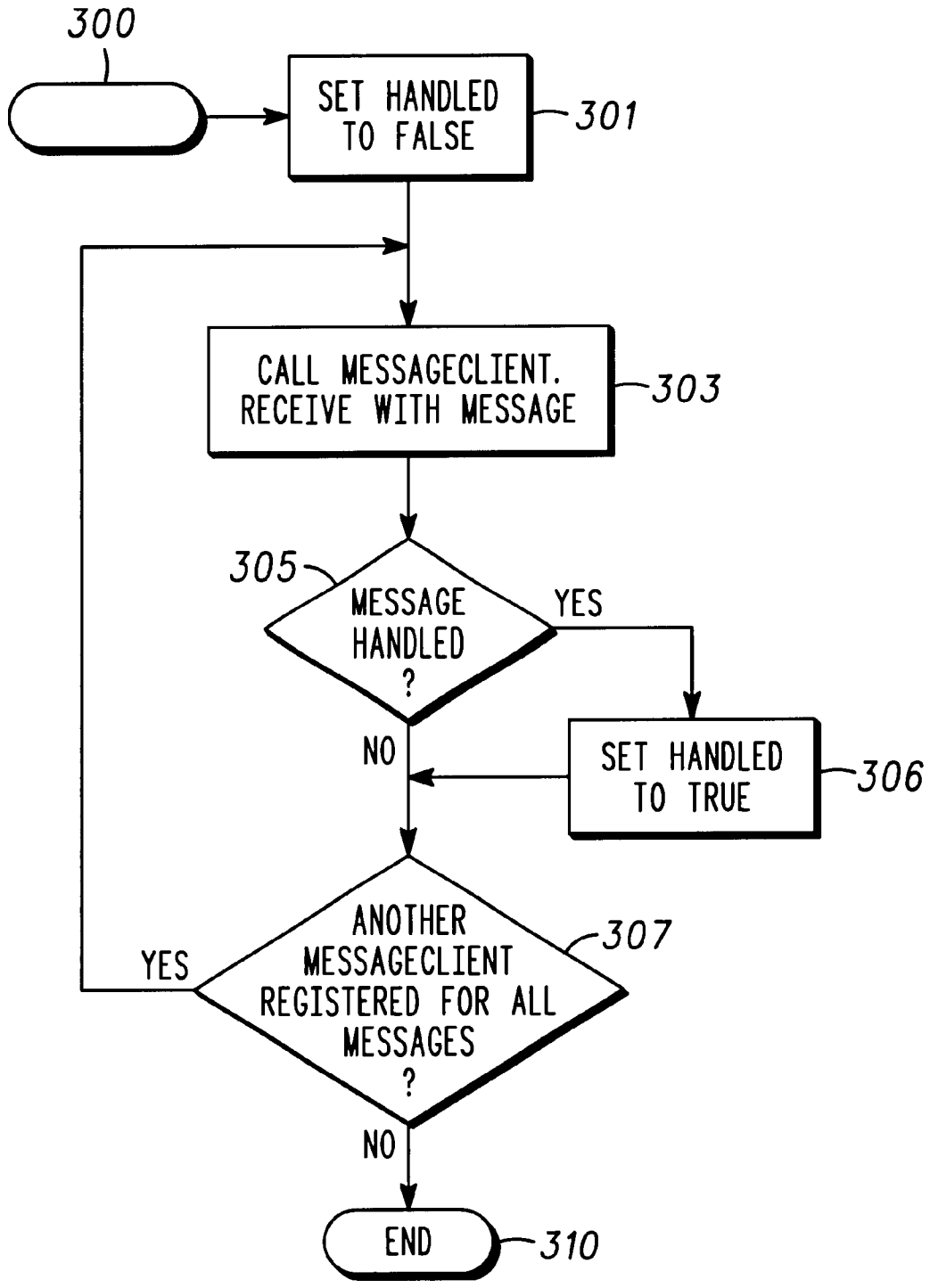

300
SET HANDLED TO FALSE
301
CALL MESSAGECLIENT. RECEIVE WITH MESSAGE
303
305
MESSAGE HANDLED ?
YES
NO
SET HANDLED TO TRUE
306
ANOTHER MESSAGECLIENT REGISTERED FOR ALL MESSAGES ?
307
YES
NO
END
310

FIG. 4

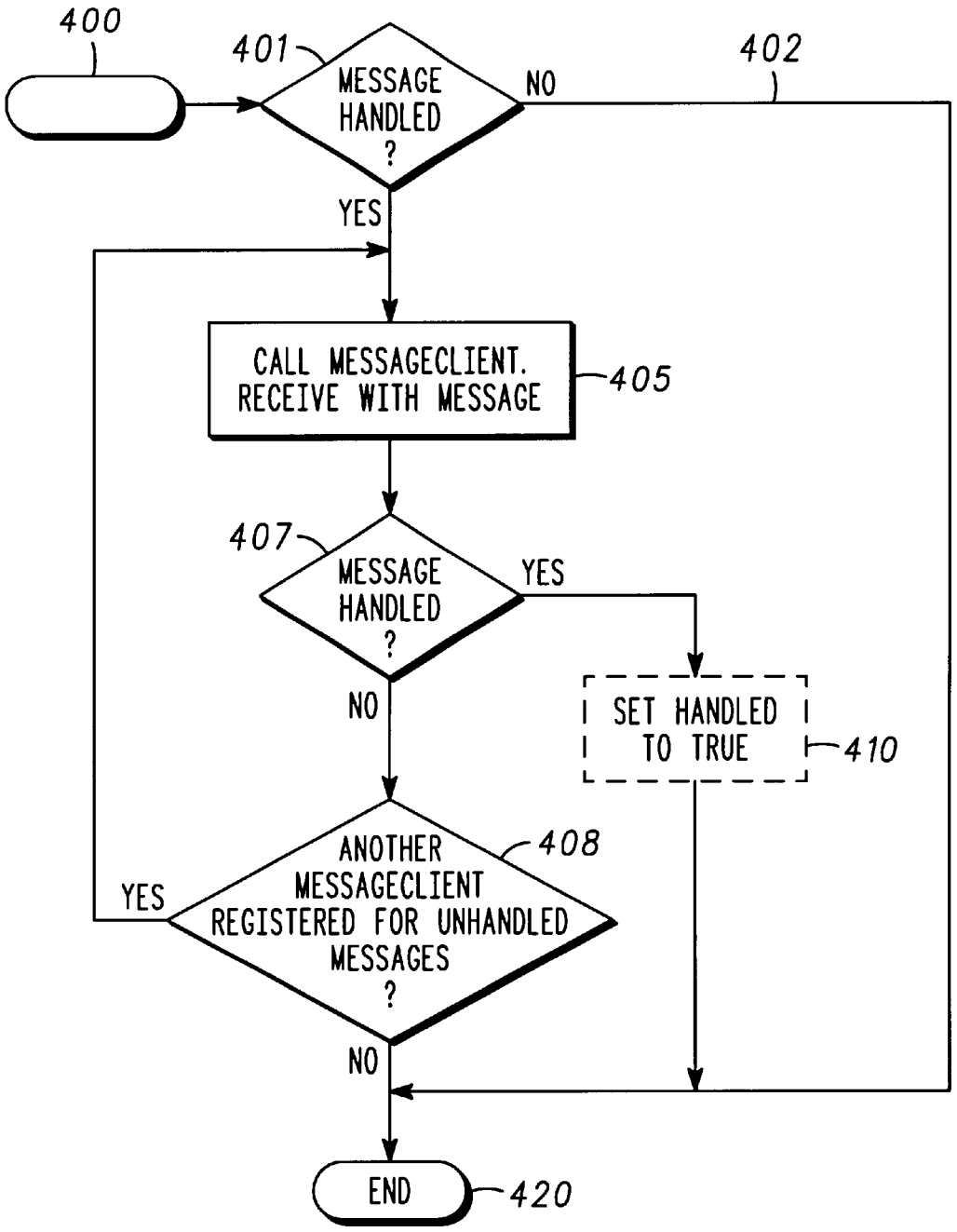

400
401
MESSAGE HANDLED ?
NO
402
YES
CALL MESSAGECLIENT. RECEIVE WITH MESSAGE
405
407
MESSAGE HANDLED ?
YES
NO
SET HANDLED TO TRUE
410
ANOTHER MESSAGECLIENT REGISTERED FOR UNHANDLED MESSAGES ?
408
YES
NO
END
420

FIG. 5

APPARATUS AND METHOD FOR HANDLING DISPATCHING MESSAGES FOR VARIOUS APPLICATIONS OF A COMMUNICATION DEVICE

FIELD OF THE INVENTION

This invention relates to a communications device comprising a processor having instructions that execute the processor in the form of interacting computer programs. The invention relates to such a device with a message manager program and a message client program.

BACKGROUND OF THE INVENTION

In the field of communication devices it has become desirable to make provision for more than one application program for enabling a user of the device to interact with incoming and outgoing messages. Such messages may be conveyed to and from the device over any of a number of bearer services or "transports", as described in U.S. Pat. No. 5,509,000 of Oberlander. Examples of such bearer services are: short messaging service in GSM "global system or mobile" digital cellular radio systems; universal datagram protocol "UDP" packets; and internet protocol "IP" packets.

When there are multiple bearers conveying messages to and from multiple user applications, at least two possible alternative architectures are available. In the first alternative, a universal mailbox is provided. All messages arrive at and are dispatched from the universal mailbox, regardless of the application that creates the message or accepts the message. This a complex arrangement from the point of view of facilitating application development, because careful consideration needs to be given to the interaction to the different applications and the competition of applications for handling of messages in the universal mailbox.

An alternative architecture provides that each application has its own mailbox. This is a more modular approach that is suitable for unlimited expansion by addition of new applications. A message manager is required between the applications and the various transports to manage the receipt and dispatch of messages between the various applications and the various transports.

It would be desirable to provide a message manager and an interface to one or more applications in an object-oriented programming language such as Java (TM), but currently such a language has no defined application programmer interface (API) adequate for such a function. It would be desirable to create an API that facilitates the handling and dispatching of messages between a message manager and various applications in a clear and a convenient manner to ensure the correct messages are delivered to the correct applications and to ensure that applications only receive messages that are of interest to those applications.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a block diagram of a communications device and its software structure.

FIG. **2** is a flow diagram illustrating an outbound message queuing process.

FIG. **3** is a flow diagram illustrating an inbound message routing process.

FIGS. **4** and **5** are flow diagrams illustrating processes performed by the message manager software of FIG. **1**.

DETAILED DESCRIPTION OF THE DRAWINGS

Referring to FIG. **1**, a communications device **10**, such as a cellular telephone or a two-way mobile radio or a two-paging device e.g. a "Pagewriter"™ is shown, which comprises a microprocessor **11** coupled to a radio transceiver **12**. The radio transceiver **12** will be referred to as the "physical" layer of the device, as it encompasses all physical functionality (such as channel selection, power control and time slot allocation or spreading or de-spreading), depending on the particular radio bearer that is being used by the communication device **10**. The physical layer **12** communicates with the microprocessor **11** and with various software modules loaded into the microprocessor **11** and executing on that processor. It will be understood that the microprocessor **11** may also perform processes dedicated to the physical layer **12** but these need not be described in detail.

Within the microprocessor **11** there are various computer programs or software modules, including a transport program **15**, a message manager program **16**, and one or more applications. In the example shown, two applications are shown **17** and **18**, labeled application "A" and application "B". Also shown is a message client **20** and a memory **22**. FIG. **1** further illustrates a remote message store **25**, which is not located within the communications device **10** but is in communication with the communications device **10**. The remote message store **25** is shown as communicating with the message client **20**, but this is shown for simplification. In fact the remote message store **25** uses the transport module **15** and the physical layer **12** to communicate with the message client **20** and also requires its own transport layer and physical layer (for example as described in U.S. Pat. No. 5,771,353 of Eggleston et al.) to establish this communication.

The applications A and B are programs that interact between a user of the communications device **10** and messages being received by and being received from the communications device **10**.

In operation, the user interacts with these applications via any suitable user interface, typically a small screen and a keyboard or a screen and a pen and tablet interface. Examples of applications are an electronic mail application such as "Outlook" (™) by Microsoft Corporation or "Lotus Notes" (™), but there are many applications that may run on the processor **11** for handling and presenting messages and further examples are given below.

The operation will be described in broad outline and will again be described in greater detail. In broad outline, an outbound message is generated on one of the applications **17** and **18** and a message object is created and delivered to the message manager **16**. The message manager **16** queues the message in an outbound message queue. The message manager **16** maintains a single queue or separate queue for each message type. The message manager **16** delivers the messages in a prioritized order to the transport program **15** which attempts to deliver the messages through the physical layer **12** to a destination (typically a server that is reached via radio base station). If the transport successfully delivers the message, it informs the message manager **16** that is ready to accept and deliver the next message. The particular ordering of the delivery of the messages by the message manager **16** to the transport layer **15** is unimportant. The ordering of messages may depend on what transport layer and what physical layer are available to the message manager, and further details can be found in copending patent application Ser. No. 08/720,984 of Pearce et al., filed on Oct. 15, 1996, now U.S. Pat. No. 5,910,951 and assigned to the assignee of the present invention, the specification of which is hereby incorporated by reference.

For inbound messages, the message manager **16** and the applications **17** and **18** require an orderly procedure for

determining whether the inbound message is to be delivered to one application or the other application or both applications. The software language used for construction of the message manager **16** and the applications **17** and **18** is preferably an object oriented language such as Java (™). A class called MessageClient resides in the permanent instructions loaded in microprocessor **11**. Upon receipt of a message, an instance **30** of the class is created for application A (this may be referred to as Message Client A). The message is delivered to Message Client A. Depending on the function of Message Client A, the message may be delivered to the application A. Alternatively, it may be delivered to memory **22** for later delivery to application A or they may be routed to remote storage **25**. Similarly, the message is selectively delivered to Message Client B **40** and is delivered to application B. The selective delivery to Message Client A and Message Client B is described in greater detail below. The detailed description explains that the selective delivery of a message to one or both of the message clients **30** and **40** depends upon pre-registration of the respective message client with the message manager **16** and depends on attributes of messages for which the respective message client is registered with the message manager. It also depends on one or more rules in the message manager **16**. The rule or rules in the message manager **16** is or are applied to attributes of each new message that arrives. Depending on the result of those rules, the message is delivered to one of those applications A or B or to both applications (and indeed any further applications that may exist).

For a more detailed description of the preferred embodiment of the invention, the following terminology will be used:

Client:—A Java application that may both source and sink Messages.

Message:—Data, and the addressing information necessary to deliver the data to another client, and optionally additional attributes.

Transport:—A session layer protocol that may send messages, or receive messages. Examples include SMS, SMTP, POP3 and IMAP4.

Inbound message:—Messages entering the devices via a transport.

Outbound message:—Messages created by a client and leaving the device via a transport.

Handled message:—A message that has had delivery attempted by a transport (outbound case) or a message that has been accepted and processed by a client (inbound case).

Unhandled message:—A message that has been refused a delivery attempt by a transport (outbound case) or a message that has been refused processing by a client (inbound case).

Four scenarios in particular are now described, including a message client **30** or **40** registering with the message manager **16**, a message client queuing an outbound message for delivery, a message client participating in the routing of an inbound message and a transport creating an outbound message delivery session.

For a client to either queue an outbound message for delivery or be able to receive an inbound message, it must first register with the message manager **16**. The process of registering with the message manager does not actually register the client itself, rather a specific implementation **30** or **40** of a subclass of the Message Client object **20** is registered. Thus, an implementation of a subclass of the message client object **20** is registered as "Message Client A". Following registration, the message manager **16** is able to identify the name of the subclass implementation as "Message Client A" and for that subclass implementation, the message manager **16** maintains a record of the rules that are to be applied for that subclass and implementation upon receipt of a new message.

A message client provides two primary methods for the MessageManager:

SendComplete:—This method is called with a Message and a Status, and is used to inform a Client that a previously queued message has been sent.

Receive:—This method is used to give the Client an opportunity to accept a message. If after parsing the attributes or the body of the Message if the MessageClient decides it can process the Message it returns true, declaring the message handled. Otherwise the Client returns false declaring the Message not handled.

When a client registers a MessageClient with the MessageManager it also specifies what class of Messages it wants to be given a chance to accept. There are two classes: all, and unhandled only. MessageClients registered for all, will see every Message routed through the MessageManager, regardless whether another MessageClient has already handled the Message. MessageClients registered for unhandled will only be given an opportunity to accept messages that no other MessageClient has handled.

The message manager **16** maintains a table of messageClients identified by name (Message Client A, Message Client B etc.) and of attributes of messages for which that Message Client is registered (handled/unhandled, type).

The process of queuing a message for outbound delivery is now described, with reference to FIG. **2**. The process starts at step **100** and in step **101** a message object is created. A message object comprises a payload (for example text and/or graphics) and attributes such as subject, recipient, type and handled or unhandled). In step **102**, the message type is set, the message manager **16** is called, that class being "MessageManager.QueueForSend". This class is called with the message and the message client class. The outbound message being processed ends at step **104**.

The above simple process corresponds to the following psuedocode

```
Create a Message object
Set Message type
Set Message attributes
Set Message body
Call MessageManager.QueueForSend with the Message
    and the MessageClient class.
```

MessageManager.QueueForSend will return a unique message ID that may be used by the client to remove the message from the outbound queue.

The Client will be notified by the MessageManager, via the MessageClient class, when a transport has attempted delivery of a queued message.

The message manager **16** does not actually trigger an outbound transport to run. It is the responsibility of the Client to actually trigger or direct a transport to run.

The process followed by a transport **15** to route an inbound message is somewhat similar and is illustrated in FIG. **3**. Starting at step **200**, a message object is created at step **201**. In step **202** the message type is set the message attributes are set and the message body is set. In step **203** a class in the message manager is called, that class being "MessageManager.RouteMmessage". The process message performed by the transport is completed at step **204**. The above described process is described by the following pseudocode:

```
Create a Message object
Set Message type
Set Message attributes
Set Message body
Call MessageManager.RouteMessage
```

Referring now to FIG. **4**, a process or method performed by the message manager **16** following from step **204** is illustrated. This process starts at step **300**. In step **301**, the attribute or flag "handled" is set to "false", and this step simply indicates that this message has not yet been handled by any message client or any application. Then, a loop begins for each message client that is registered for all messages. This loop will be performed upon receipt of any message and it will be performed for any message client (on behalf of its associated application) that desires to receive all messages. In step **303**, the class "MessageClient.Receive" is called with the message.

When MessageClient.Receive is called, that subclass instance (e.g. message client A) delivers a result. The result delivered is a simple "true" or "false". "True" is delivered when the respective message client indicates that it has handled the message. "False" is delivered when the respective subclass indicates when it has not handled the message. As will be explained below, it is entirely within the choice of the programmer as to whether the message client subclass instance delivers "true" or "false". Thereafter, the process determines whether the message has been indicated as having been handled by the respective message client. If the respective message client has delivered "true," step **305** determines that the message has been handled, and in step **306**, the flag "handled" in the message manager **16** is set to "true" for that message. In either case, the process proceeds to step **307**, where the message manager **16** determines if there is another message client that is registered for all messages. If there is another message client that is registered for all messages, the process returns to the beginning of the loop and a message client is created for the next application (e.g. message client **40**). If following step **307**, there are no messaging clients that are registered for all messages, the process ends at step **310** and flow can proceed to the process illustrated in FIG. **5**.

The above RouteMessage method in the message manager **16** is detailed in the following psuedocode:

```
Set handled to False
For each MessageClient registered for all Messages
Begin
    Call MessageClient.Recieve with Message
    if Message handled then
        set handled to True
End
```

It may, incidentally, be noted that the process of FIG. **5** can be modified in the event that more than one type of message exists. Message types may include, SMS messages and internet messages. A message client may be registered for all SMS messages, or all internet messages, or all messages, whether SMS or internet. Thus, a message client may be registered for a first type of message or a second type of message or all messages. In such an example, step **307** can be modified to look for whether there is another message client that is registered for all messages for all message type. It may also be noted that a MessageClient can be created temporarily for the purpose of creating a call to that MessageClient and released immediately following the call.

Referring now to FIG. **5**, a process is described that follows from the process of FIG. **4** and that is executed by the message minder **16**. This process begins at step **400** and is bypassed at step **401** via a bypass path **402** in the event that the message flag for the current message is set at "true". Thus, step **401** determines whether the message has been handled, and if the message has been handled, the process immediately ends at step **420**. In other words, the process at FIG. **5** executes for unhandled messages only, and therefore only executes for message clients that are registered for unhandled messages only. In the event that there is an unhandled message, the processor proceeds to step **404**. In step **405**, the class MessageClient.Receive is called with the message. As before, the message client (e.g. message client B indicated by reference numeral **40** in FIG. **1**) delivers either "true" or "false" to the message manager **16**. Following step **405**, step **407** determines whether the message has been handled. If the message has not been handled, step **408** determines whether there are any other message clients that are registered with the message manager for unhandled messages only. If there are, the process returns to step **404**.

If, following step **407** the indicated as having been handled, the process proceeds to optional step **410**. In this optional step, the "handled" flag is set to "true" for that message. The program will then proceed to step **420**. Step **410** is indicated as optional, because the particular transport that called "route message" may not care whether the message is handled at all by any application or any message client. If the underlying transport needs to know that the message has been handled, then step **410** is necessary.

When there are no further message clients that are registered for unhandled messages or when the message has been handled by one of the message clients, the process ends at step **420**.

The above described process is illustrated by the following psuedocode:

```
if not handled then
Begin
    for each MessageClient registered for Unhandled
    Messages
    Begin
        Call MessageClient.Receive with Message
        if Message handled then
            end for loop
        End
    End
```

There may be many message clients **30**, **40**, etc., that may be registered with the message manager **16**. Message clients register in chronological order and the message manager **16** executes step **404** for the message clients in the chronological order in which they have registered. Thus, if applications A and B are both registered for unhandled messages only, message manager **16** will create subclass instance **30** first and if the instance of message client so created successfully handles a given message, message manager **16** will not create a message client instance **40** for application B, because flow will proceed from step **407** to step **420**. The chronological order of registering of message clients with the message manager can be overridden such that the user or the program writer can devise a specific sequence in which message clients are to be created for a new message.

As before, a MessageClient can be created temporarily for the purpose of creating a call to that MessageClient and released immediately following the call.

It has been explained that the applications A and B may be electronic mail applications. Many other applications large and small can be devised with the structure described.

At a very minimum, there may be a message client created that has no corresponding application. At a very minimum this message client may have no function other than to deliver "true" for all messages. Such a trivial message client would simply consume all messages that arrive with no purpose. Or such a message client could, for example, simply count the incoming messages. Alternatively, it could count messages of a given type. For example it could count all fax messages and give an indication to the user that there are faxes that have arrived, so the user can turn to a connected fax machine to receive those messages. Another example of a simple message client would be a message client that extracts a sender's address and adds it to address book and it turns "false" to the message manager **16**. Such a message client would build up an address book for all messages without interfering with operation of other applications interacting with the message manager **16**. By returning "false" the message client would not interfere with the ability of other message clients to have the opportunity to receive the message. These are all examples of functions that can take place with the message client without necessarily invoking an application.

The message client can deliver a message to memory **22** or to remote storage **25** for later review when application A or application B is launched. Message client can return "true" if it wishes to indicate that it has successfully received the message and it wishes to prevent any other application from receiving that message or it can return "false" if it wishes to receive the message and simultaneously permit another application to receive the message.

It has been described how a message can have two attributes: a message type and a message handled flag. The message manager **16** applies rules to these attributes for each new message. There may, of course, be further attributes to which further rules can be applied, but adding more attributes and more rules adds complexity to the message manager **16** and it is preferred that such complexities are added to the message client **30**. It is believed that the simple true/false delivery of the result by the message client and the provision of a few simple rules of the message manager provides sufficient flexibility for message clients to execute unlimited numbers of rules against a message and deliver true or false according to rules executed within the message client. As an example, a message client may only wish to handle messages during office hours and can return "false" at any time of day. This is an example of a rule that is executed in the message client. The possible rules and attributes executed by the message manager are illustrated in the following table. The table shows attributes as columns as rows.

	Handled	Unhandled	Don't Care
Type A		X	X
Type B		X	X
Don't Care		X	X

The table show six possibilities for rules that may be executed by the message manager. Conceivably a message manager could be devised to create a message client that is registered for handled messages only, but presently there is no apparent purpose for such a message client.

The computer programs illustrated in FIG. **4** and FIG. **5** in effect execute the rules illustrated in Table H, but will be appreciated by oneskilled in the art that there are many other ways of setting out the program that will create a message client in a selected manner depending on attributes of a message or attributes held in the message manager **16** and deliverable in a software call to the message manager.

For completeness, operation of an outbound delivery session is now described. An outbound transport must iterate over all the messages (of a given type) in the MessageManager's queue. Just as in the inbound case the transport can inform the MessageManager of which messages it actually handled and which ones it didn't. It's entirely up to the transport to determine if it can actually handle (deliver) a message.

There is a distinct difference between attempting to deliver a message and failing; and not handling the message at all. Typically a Client will set a special attribute on all of the messages it queues. At a later time when the Client triggers the transport to run, it passes the special attribute to the transport. The transport uses this information to determine which messages it should handle and which messages it should not, by intervention over message and sending messages that match the "special attribute", as set out in the following pseudocode.

```
Call MessageManager.OpenTransportSession with a message type
Save the MessageContext returned
Call MessageManager.GetMessage with the MessageContext
While GetMessage returns a message

Begin
    If the transport can handle this message then
    Begin
        Send the message over the transport
        Call MessageManager.SendComplete,
            with the MessageContext, Message, and the send
                status
    End
    else
        Call MessageManager.ReturnUnHandledMessage,
            with the MessageContext and Message,
        Call MessageManager.GetMessage with MessageContext
    End
    Call MessageManager.CloseTransportSession with
        the MessageContext
```

In summary, a communications device has been described comprising a processor having instructions that execute on the processor and it comprise: a message manager program **16** for accepting and dispatching messages; at least one application program **17**, **18** for handling and presenting messages; and at least one message client program **30**, **40**, that receives messages from the message manager program and provides them to the application program. The message manager program accepts a registration from at least one message client program comprising at least one rule and at least one message attribute to which the rule is to be applied for new messages.

Similarly, a communications device has been described comprising a processor having first instructions **16** for message management, second instructions **17** or **18** for interacting between a message and the user; and third instructions **30**, **40** for accepting a message from the first instructions and providing a message to the second instructions. The first instructions **16** accept a registration from at least one set of third instructions **30** or **40**. The registration comprises at least one rule and at least on message attribute to which the rule is to be applied to new messages.

The above description has been given by way of example only, and modifications of detail can be made by one of

ordinary skill in the art without departing from the spirit and skill of the invention.

We claim:

1. A communication device including a processor having instructions that execute on the processor and that comprises:

a message manager program for accepting and dispatching messages;

at least one application program for handling and presenting messages; and

at least one message client program, each message client program corresponding to a particular application program, that receives messages from the message manager program and provides them to the corresponding application program,

wherein the message manager program accepts a registration from each message client program including at least one rule and at least one attribute to which the rule is to be applied for new messages, and

wherein the at least one attribute including a handled/not handled attribute of messages, the handled attribute indicates that a transport has attempted deliver of a particular massage or a client has processed the particular message whereas the not handled attribute indicates that the transport has refused delivery of the message or a client has refused processing of the message.

2. The device according to claim **1**, wherein the at least one message client program is created upon receipt of a new message and released following selective delivery of the message.

3. The device according to claim **1**, wherein the message manager selectively delivers the new message to a message client program dependent on the handled/not handled attribute of the message.

4. The device according to claim **3**, wherein the message manager creates the message client program dependent on the handled/not handled attribute of the message.

5. The device according to claim **1**, wherein the message attribute includes a message type attribute.

6. The device according to claim **5**, wherein the message manager selectively delivers the new message to a message client program dependent on the message type attribute of the message.

7. The device according to claim **6**, wherein the message manager creates the message client program dependent on the message type attribute of the message.

8. The device according to claim **1**, wherein the registration comprises at least first and second attributes including a handled/not handled attribute and a message type attribute.

9. The device according to claim **8**, wherein the message manager program, upon receipt of a new message, executes first and second rules applied against the first and second attributes respectively.

10. The device according to claim **9**, wherein the message manager selectively delivers the new message to a message client program dependent on a result of executing the first and second rules against the first and second attributes.

11. A communication device including a processor having instructions that execute on the processor and that comprises:

a set of first instructions for message management;

at least one set of second instructions for interacting between message and a user; and

at least one set of third instructions, each set of third instructions corresponding to a particular set of second instructions, for accepting a message from the first instructions and providing the message to the corresponding set of second instructions, wherein

the first instructions accept a registration from each set of third instructions, the registration including at least one rule and at least one attribute to which the rule is to be applied for new messages, the at least one attribute including a handled/not handled attribute of messages, and the handled attribute indicates that a transport has attempted deliver of a particular massage or a client has processed the particular message whereas the not handled attribute indicates that the transport has refused delivery of the message or a client has refused processing of the message.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 6,425,002 B1
DATED : July 23, 2002
INVENTOR(S) : Zurcher et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 9,
Line 22, please change "deliver" to -- delivery --
Line 23, please change "massage" to -- message --

Column 10,
Line 37, please change "massage" to -- message --

Signed and Sealed this

Thirty-first Day of December, 2002

JAMES E. ROGAN
Director of the United States Patent and Trademark Office

EXHIBIT 4

US006493673B1

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

(10) **Patent No.: US 6,493,673 B1**
(45) **Date of Patent: *Dec. 10, 2002**

(54) **MARKUP LANGUAGE FOR INTERACTIVE SERVICES AND METHODS THEREOF**

(75) Inventors: **David Ladd**, Downers Grove, IL (US); **Gregory Johnson**, Carol Stream, IL (US)

(73) Assignee: **Motorola, Inc.**, Schaumburg, IL (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 48 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **09/644,638**

(22) Filed: **Aug. 23, 2000**

Related U.S. Application Data

(63) Continuation of application No. 09/165,487, filed on Oct. 2, 1998.
(60) Provisional application No. 60/094,131, filed on Jul. 24, 1998, and provisional application No. 60/094,032, filed on Jul. 24, 1998.

(51) **Int. Cl.**[7] **G10L 21/00**; G06F 15/00
(52) **U.S. Cl.** **704/275**; 704/270; 704/270.1; 379/88.01
(58) **Field of Search** 704/260, 270, 704/275; 379/88.13, 88.17, 88.22, 88.01

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,649,117	A	7/1997	Landry	395/240
5,655,008	A	8/1997	Futch et al.	379/91.01
5,860,073	A	1/1999	Ferrel et al.	707/522
5,877,766	A	3/1999	Bates et al.	345/357
5,899,975	A	5/1999	Nielsen	704/260
5,915,001	A	6/1999	Uppaluru	379/88.22
5,953,392	A	9/1999	Rhie et al.	379/88.13
6,240,391	B1 *	5/2001	Ball et al.	704/270
6,269,336	B1 *	7/2001	Ladd et al.	704/270
6,349,132	B1 *	2/2002	Wesemann et al.	379/88.17
6,385,583	B1 *	5/2002	Ladd et al.	704/270

OTHER PUBLICATIONS

Wong, W., Dialing into Computer telephony, Network VAR, Mar. 1998, vol. 6, pp. 34–39, see p. 36.
Nava Air Federal Credit Union Call 24 Voice Response Brochure, May 1994, pp. 1–2, see p. 2.

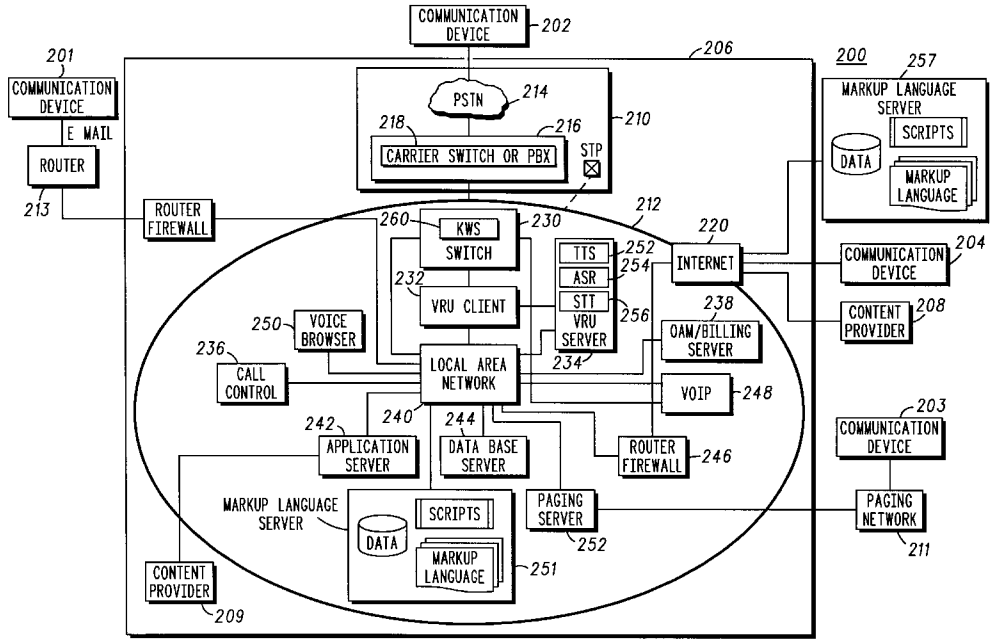
* cited by examiner

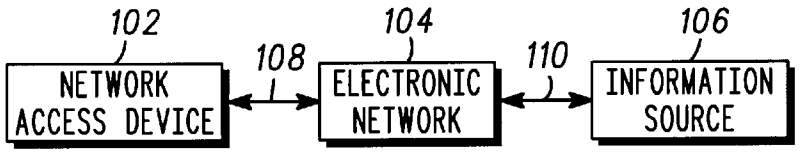
Primary Examiner—Susan McFadden
(74) *Attorney, Agent, or Firm*—Charles W. Bethards; Hisashi D. Watanabe

(57) **ABSTRACT**

The present invention relates to a markup language document stored on a computer-readable medium to provide interactive services and methods thereof. The markup language comprises a dialog element and a step element. The dialog element includes markup language elements in which each markup language element is identifiable by one or more markup tags. The step element is contained within the dialog element and defines a state within the dialog element. The step element includes a prompt element, an input element, and a first attribute. The prompt element includes an announcement to be read to the user, and the input element includes an attribute to allow an audible user input to be converted to a text string.

51 Claims, 10 Drawing Sheets





100

FIG. 1

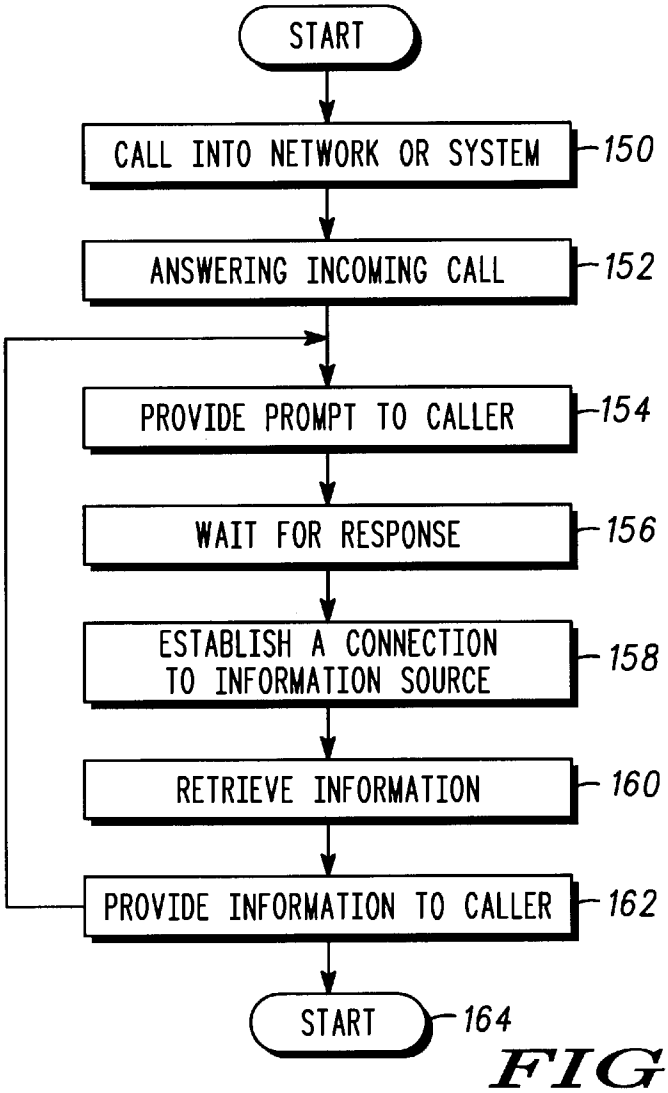
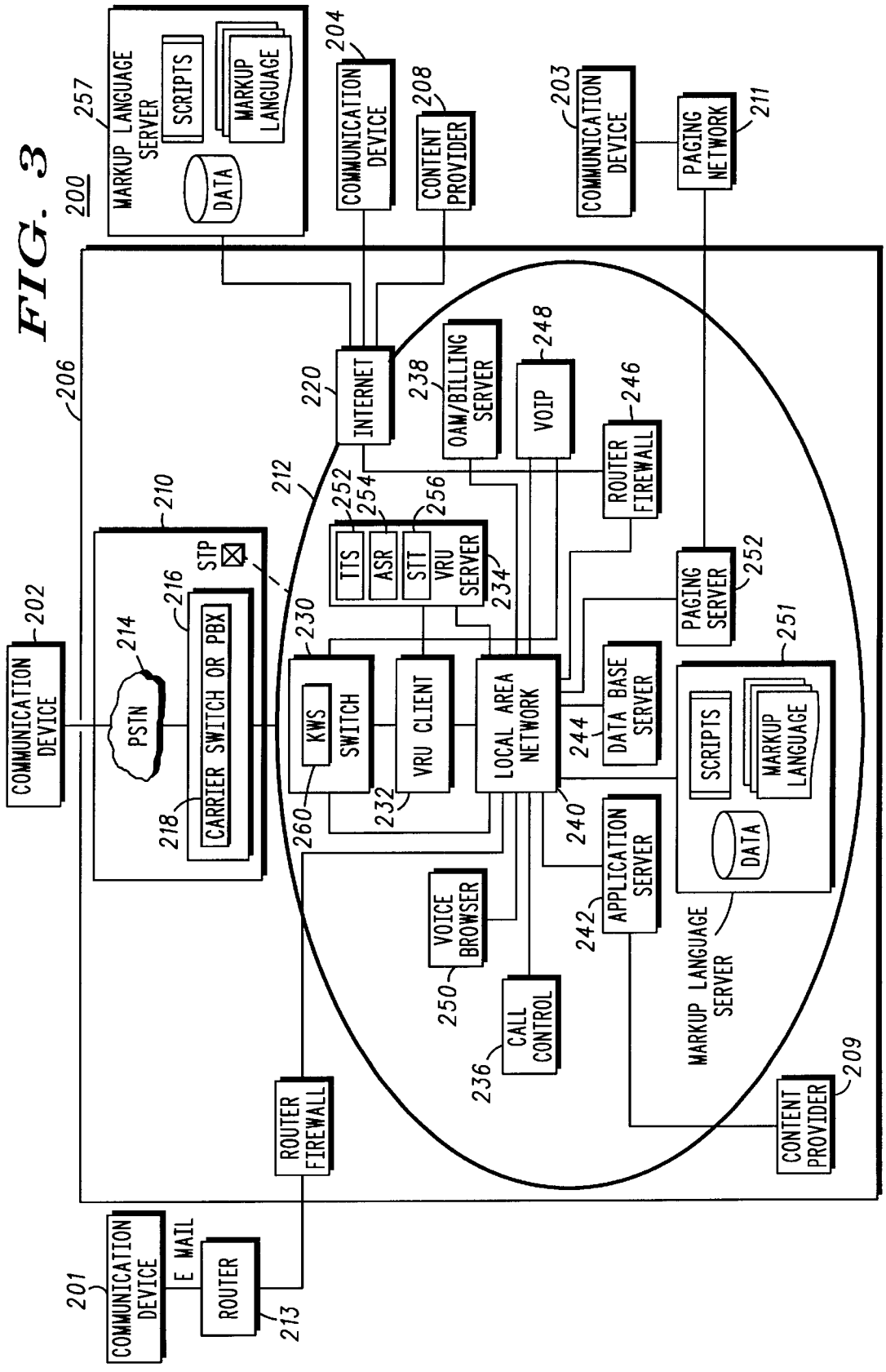


FIG. 2

FIG. 3


200
MARKUP LANGUAGE SERVER
257
SCRIPTS
MARKUP LANGUAGE
DATA
COMMUNICATION DEVICE
204
CONTENT PROVIDER
208
COMMUNICATION DEVICE
203
PAGING NETWORK
211
206
INTERNET
220
OAM/BILLING SERVER
238
VOIP
248
ROUTER FIREWALL
246
212
210
252
254
256
TTS
ASR
STT
VRU SERVER
234
STP
216
214
PSTN
CARRIER SWITCH OR PBX
218
202
COMMUNICATION DEVICE
230
KWS
SWITCH
VRU CLIENT
232
LOCAL AREA NETWORK
244
DATA BASE SERVER
PAGING SERVER
252
251
SCRIPTS
MARKUP LANGUAGE
DATA
260
240
APPLICATION SERVER
242
VOICE BROWSER
250
CALL CONTROL
236
MARKUP LANGUAGE SERVER
ROUTER FIREWALL
CONTENT PROVIDER
209
201
COMMUNICATION DEVICE
E MAIL
ROUTER
213

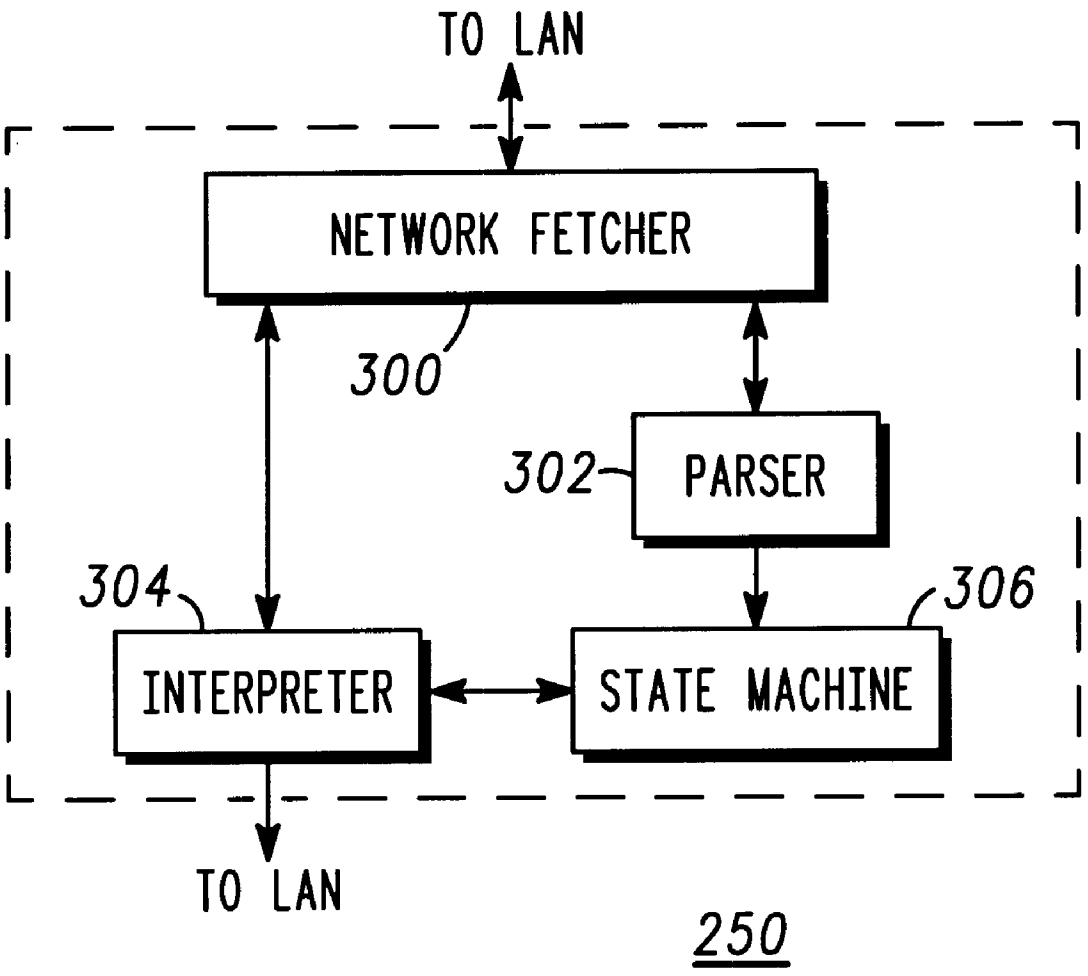

TO LAN
NETWORK FETCHER
300
302
PARSER
304
306
INTERPRETER
STATE MACHINE
TO LAN
250

FIG. 4

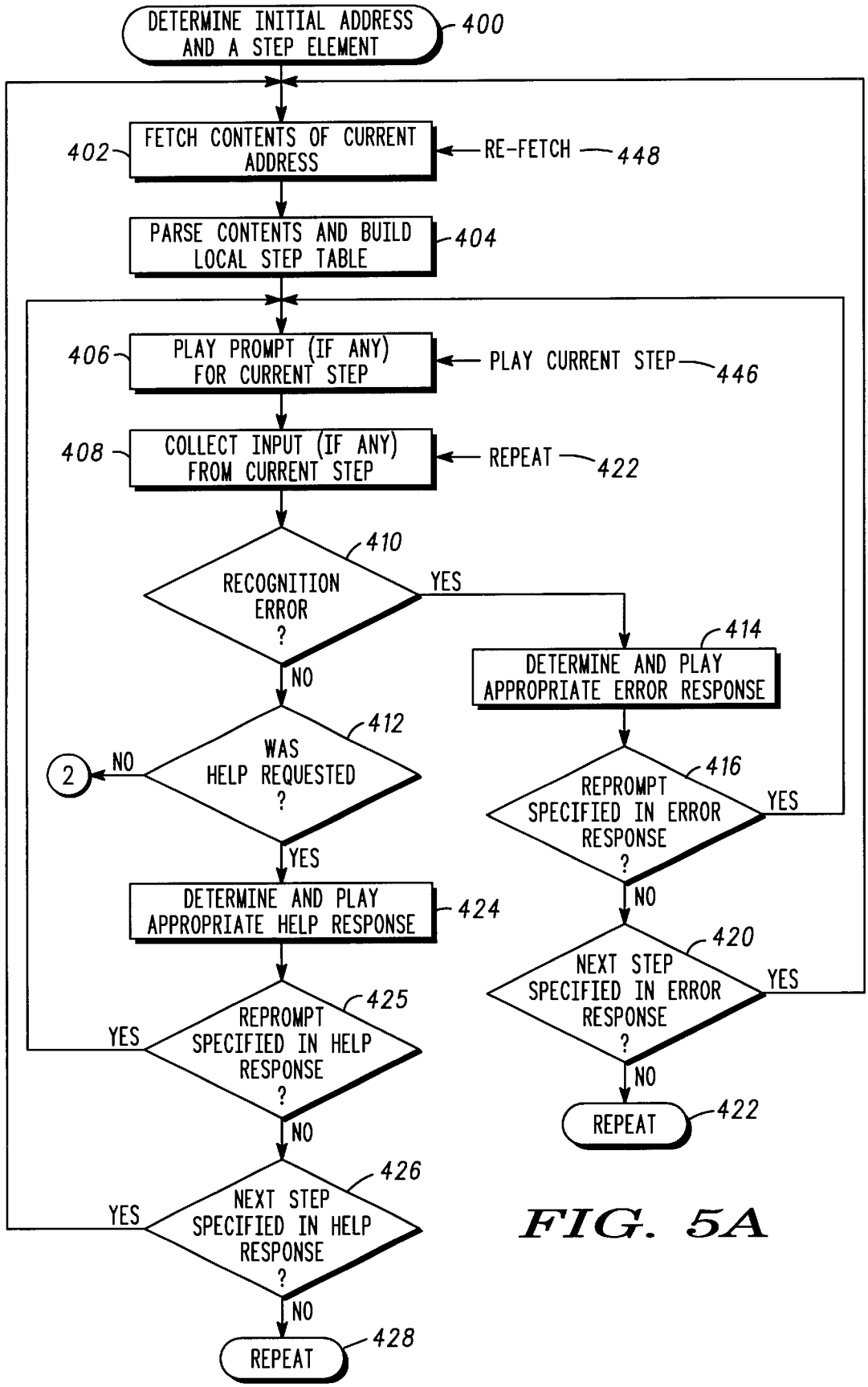

DETERMINE INITIAL ADDRESS AND A STEP ELEMENT
400
FETCH CONTENTS OF CURRENT ADDRESS
402
RE-FETCH
448
PARSE CONTENTS AND BUILD LOCAL STEP TABLE
404
PLAY PROMPT (IF ANY) FOR CURRENT STEP
406
PLAY CURRENT STEP
446
COLLECT INPUT (IF ANY) FROM CURRENT STEP
408
REPEAT
422
RECOGNITION ERROR ?
410
YES
NO
DETERMINE AND PLAY APPROPRIATE ERROR RESPONSE
414
WAS HELP REQUESTED ?
412
2
NO
YES
REPROMPT SPECIFIED IN ERROR RESPONSE ?
416
YES
NO
DETERMINE AND PLAY APPROPRIATE HELP RESPONSE
424
NEXT STEP SPECIFIED IN ERROR RESPONSE ?
420
YES
NO
REPROMPT SPECIFIED IN HELP RESPONSE ?
425
YES
NO
REPEAT
422
NEXT STEP SPECIFIED IN HELP RESPONSE ?
426
YES
NO
REPEAT
428

FIG. 5A

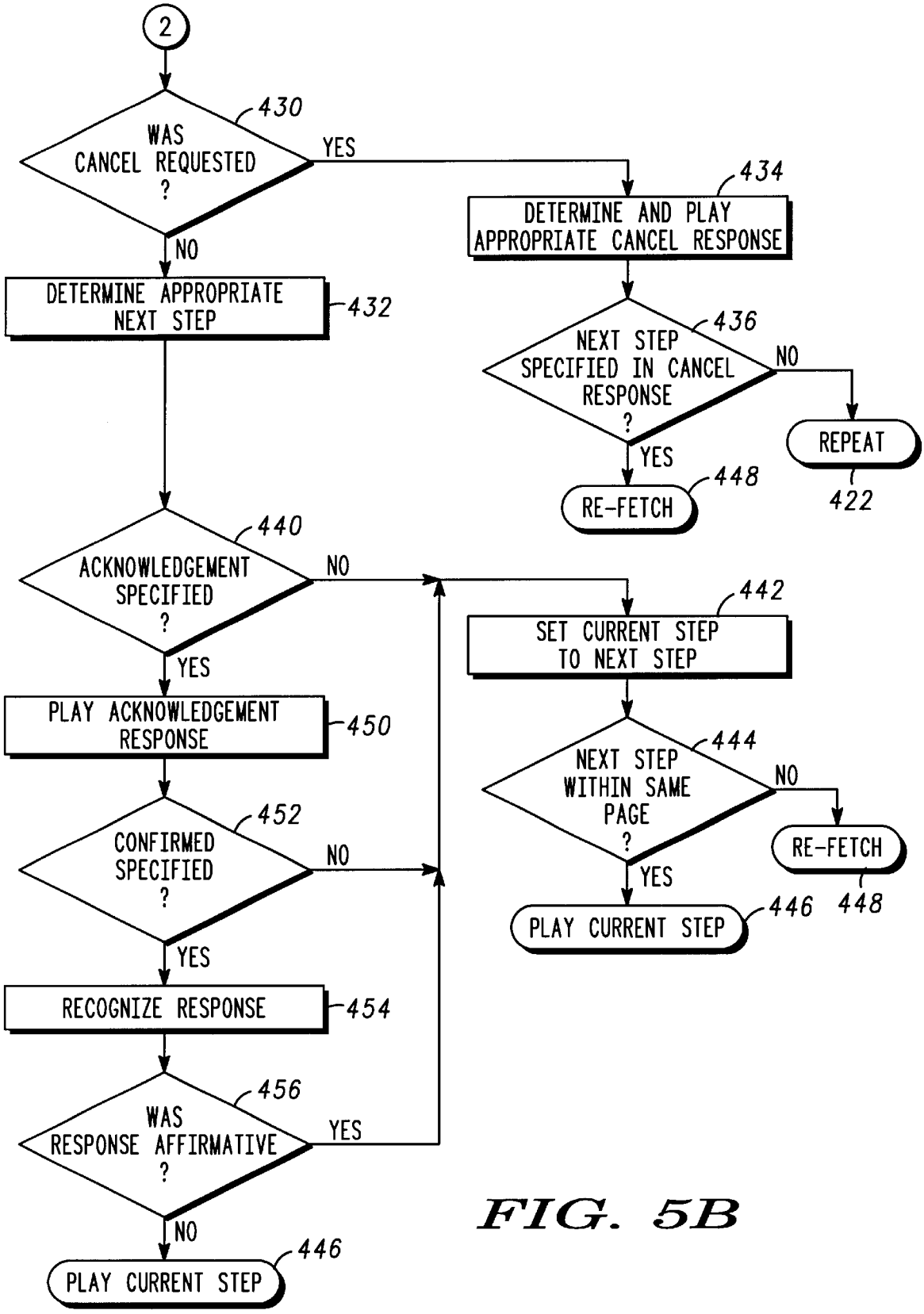

2
430
WAS CANCEL REQUESTED ?
YES
NO
434
DETERMINE AND PLAY APPROPRIATE CANCEL RESPONSE
DETERMINE APPROPRIATE NEXT STEP
432
436
NEXT STEP SPECIFIED IN CANCEL RESPONSE ?
NO
REPEAT
422
YES
448
RE-FETCH
440
ACKNOWLEDGEMENT SPECIFIED ?
NO
442
SET CURRENT STEP TO NEXT STEP
YES
PLAY ACKNOWLEDGEMENT RESPONSE
450
444
NEXT STEP WITHIN SAME PAGE ?
NO
RE-FETCH
448
452
CONFIRMED SPECIFIED ?
NO
YES
PLAY CURRENT STEP
446
YES
RECOGNIZE RESPONSE
454
456
WAS RESPONSE AFFIRMATIVE ?
YES
NO
446
PLAY CURRENT STEP

FIG. 5B

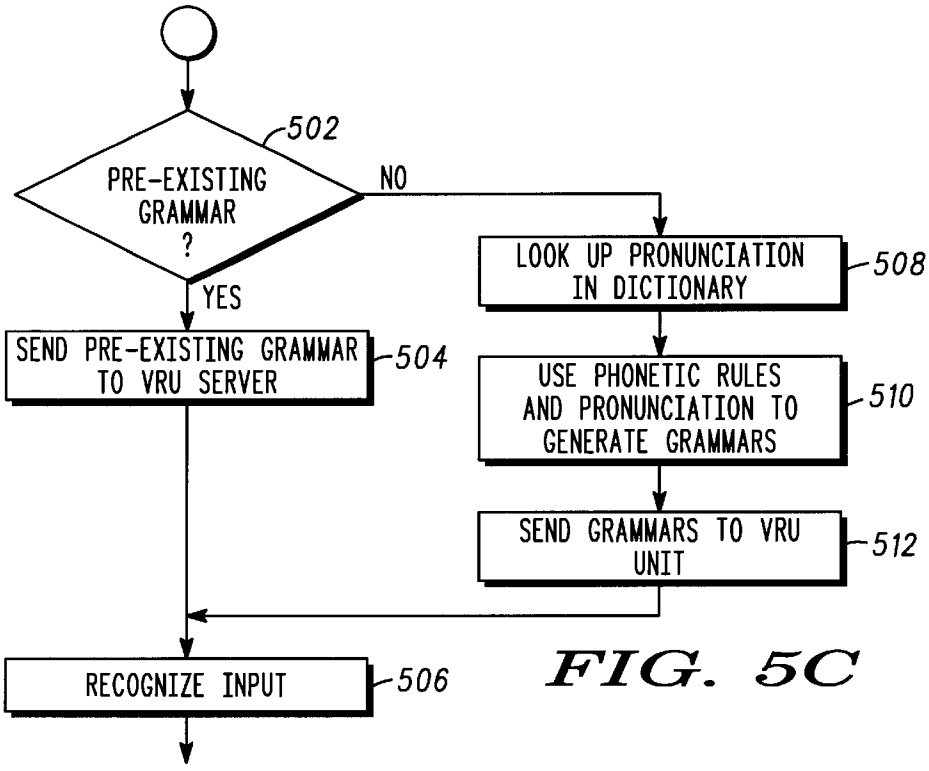


FIG. 5C

```
<? XML VERSION="1.0"?>
<DIALOG>
    <STEP NAME="INIT">
        <PROMPT>WHAT MEAL WOULD LIKE TO HEAR THE SPECIALS
FOR?</PROMPT>
        <INPUT TYPE="OPTIONLIST">
            <OPTION NEXT="#BKFST"> BREAKFAST </OPTION>
            <OPTION NEXT="#LUNCH"> LUNCH </OPTION>
            <OPTION NEXT="#DINNER"> DINNER </OPTION>
        </INPUT>
    </STEP>

    <STEP NAME="BKFST">
        <PROMPT> OUR BREAKFAST SPECIAL IS GREEN EGGS AND HAM </PROMPT>
    </STEP>

    <STEP NAME="LUNCH">
        <PROMPT> OUR LUNCH SPECIAL IS A BACON, LETTUCE, AND TOMATO
SANDWICH. </PROMPT>
    </STEP>

    <STEP NAME="DINNER">
        <PROMPT> OUR DINNER SPECIAL TODAY IS ROAST BEEF AND MASHED
POTATOES. </PROMPT>
    </STEP>
</DIALOG>
```

FIG. 6

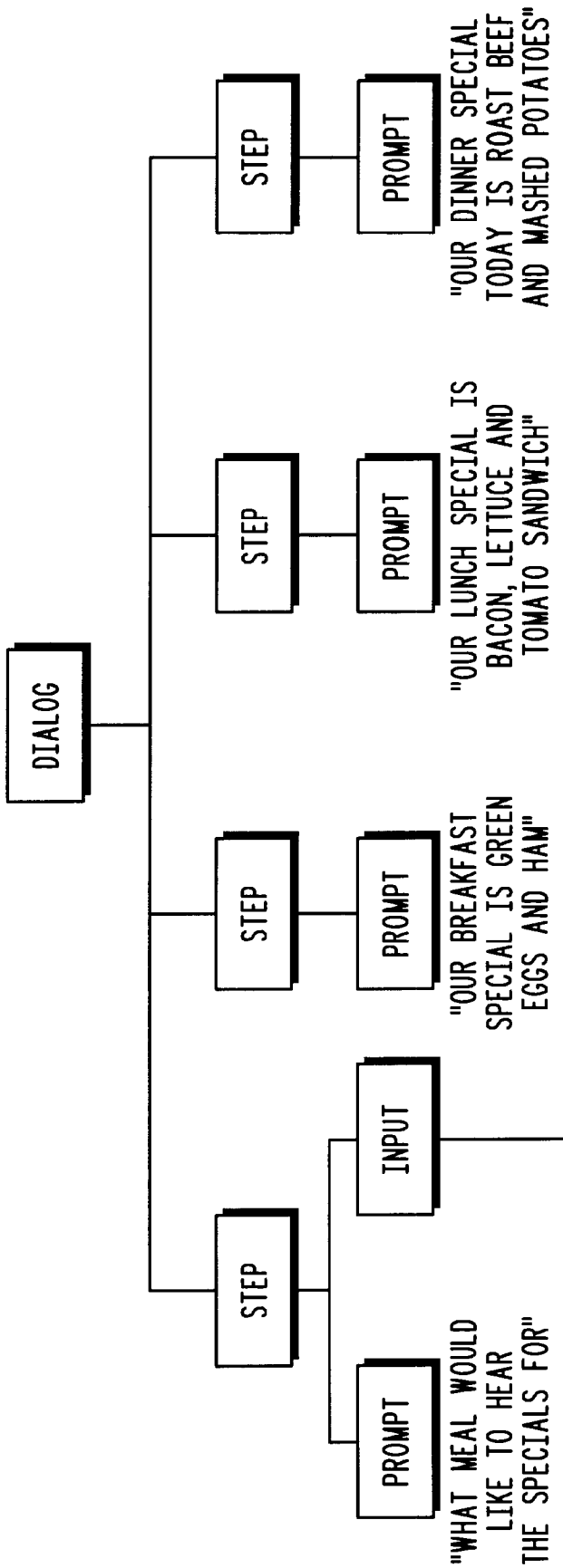

DIALOG
STEP
PROMPT
"WHAT MEAL WOULD LIKE TO HEAR THE SPECIALS FOR"
INPUT
OPTION
"BREAKFAST"
OPTION
"LUNCH"
OPTION
"DINNER"
STEP
PROMPT
"OUR BREAKFAST SPECIAL IS GREEN EGGS AND HAM"
STEP
PROMPT
"OUR LUNCH SPECIAL IS BACON, LETTUCE AND TOMATO SANDWICH"
STEP
PROMPT
"OUR DINNER SPECIAL TODAY IS ROAST BEEF AND MASHED POTATOES"

FIG. 7

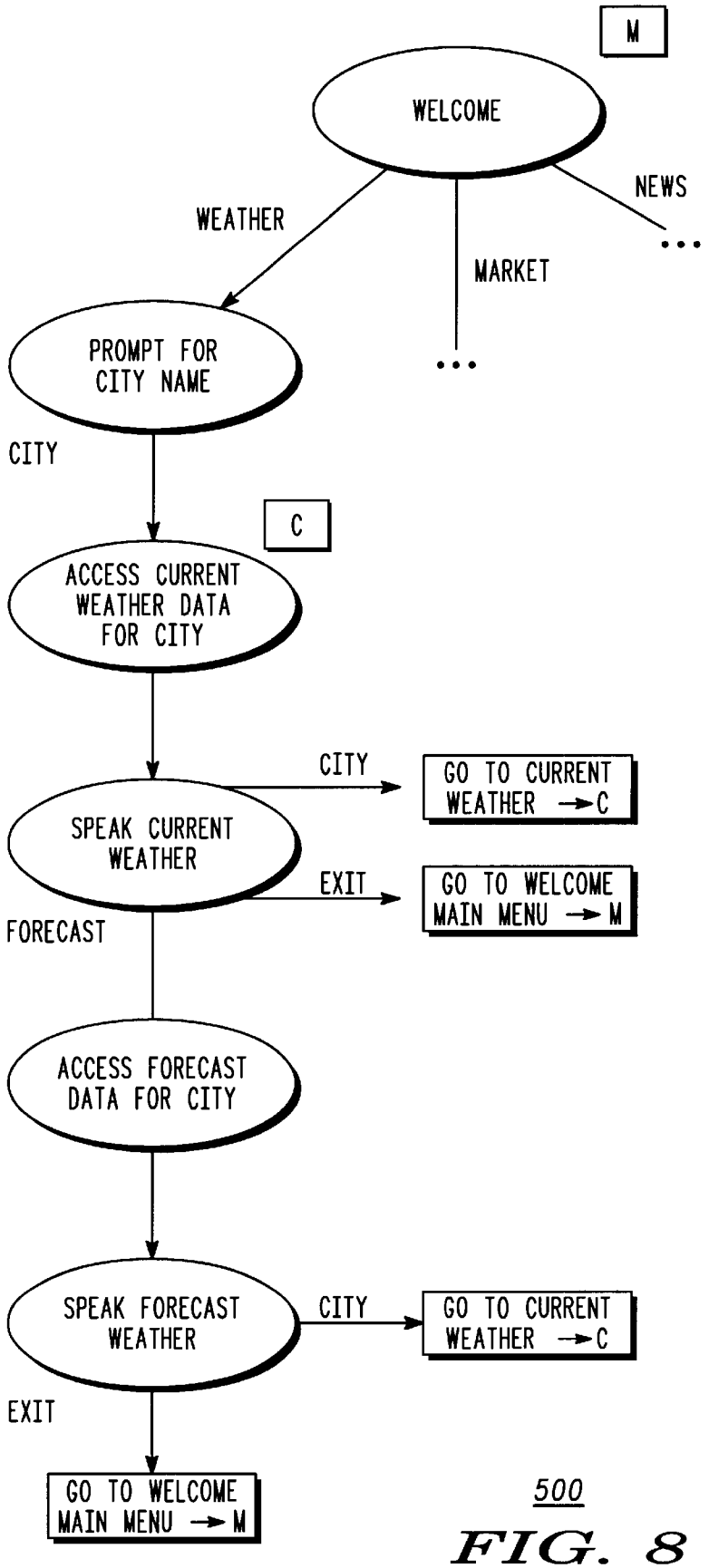

M
WELCOME
NEWS
WEATHER
MARKET
PROMPT FOR CITY NAME
CITY
C
ACCESS CURRENT WEATHER DATA FOR CITY
CITY
GO TO CURRENT WEATHER →C
SPEAK CURRENT WEATHER
EXIT
GO TO WELCOME MAIN MENU →M
FORECAST
ACCESS FORECAST DATA FOR CITY
SPEAK FORECAST WEATHER
CITY
GO TO CURRENT WEATHER →C
EXIT
GO TO WELCOME MAIN MENU →M
500

FIG. 8

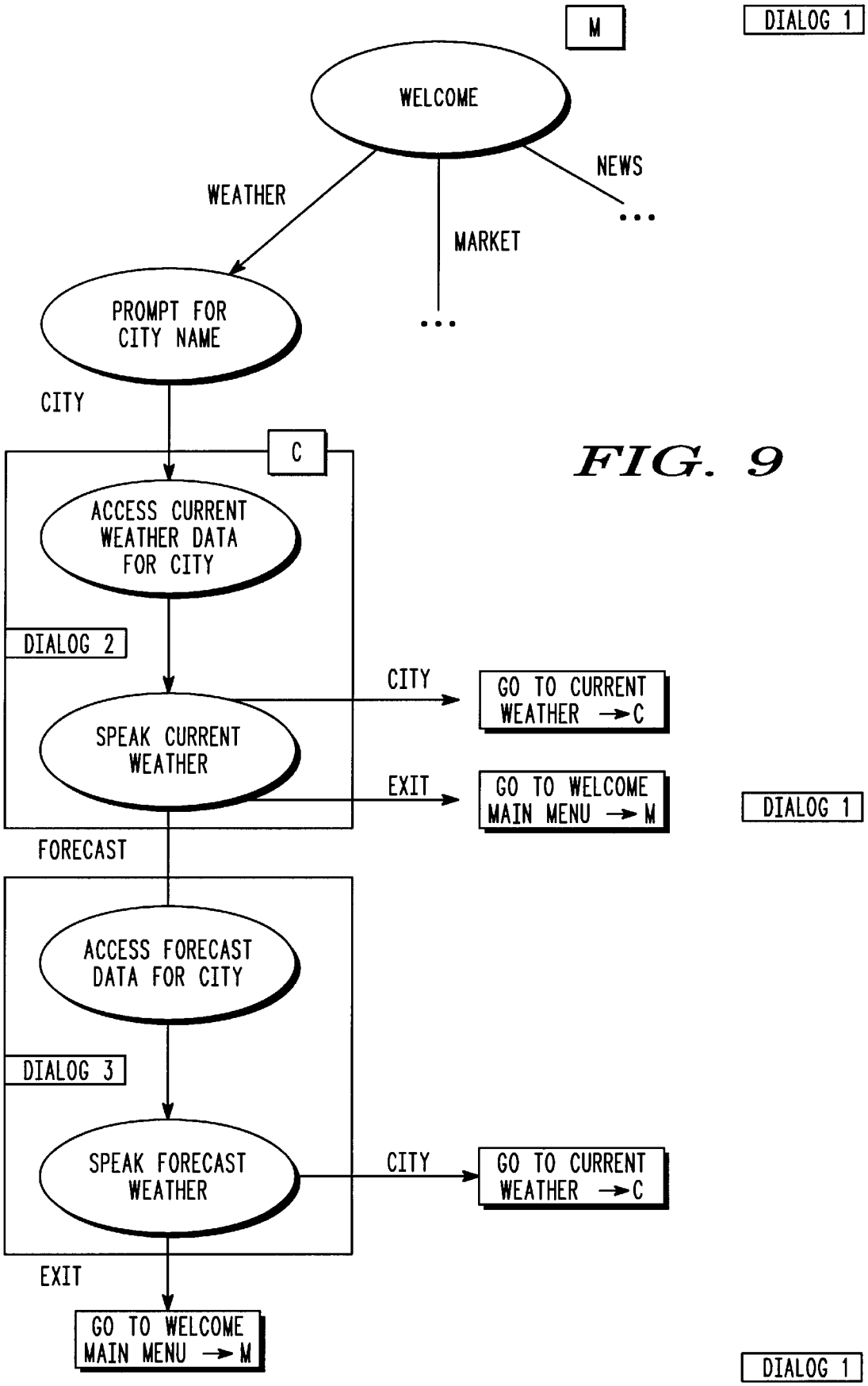

M
DIALOG 1
WELCOME
WEATHER
NEWS
...
MARKET
...
PROMPT FOR CITY NAME
CITY
C
ACCESS CURRENT WEATHER DATA FOR CITY
DIALOG 2
CITY
GO TO CURRENT WEATHER →C
SPEAK CURRENT WEATHER
EXIT
GO TO WELCOME MAIN MENU →M
DIALOG 1
FORECAST
ACCESS FORECAST DATA FOR CITY
DIALOG 3
SPEAK FORECAST WEATHER
CITY
GO TO CURRENT WEATHER →C
EXIT
GO TO WELCOME MAIN MENU →M
DIALOG 1

FIG. 9

```
<STEP NAME="order form">
    <PROMPT> What you like to order? </PROMPT>
    <INPUT TYPE="form" NAME="order" NEXT="#next
    order" METHOD="post"
    ACTION="http://www.test.com/cgi-bin/post-query"
    TIMEOUT="200" />
</STEP>
```

FIG. 10

MARKUP LANGUAGE FOR INTERACTIVE SERVICES AND METHODS THEREOF

The present application is a continuation of U.S. application Ser. No. 09/165,487 filed on Oct. 2, 1998. which is based on prior U.S. application Ser. Nos. 60/094,131 and 60/094,032, filed on Jul. 24, 1998 which are hereby incorporated by reference, and priority thereto for common subject matter is hereby claimed.

FIELD OF THE INVENTION

The present invention generally relates to information retrieval, and more particularity, to methods and systems to allow a user to access information from an information source.

BACKGROUND OF THE INVENTION

On-line electronic information services are being increasingly utilized by individuals having personal computers to retrieve various types of information. Typically, a user having a personal computer equipped with a modem dials into a service provider, such as an Internet gateway, an on-line service (such an America On-line, CompuServer, or Prodigy), or an electronic bulletin board to download data representative of the information desired by the user.

The information from the service provider is typically downloaded in real-time (i.e., the information is downloaded contemporaneously with a request for the information). Examples of information downloaded in this manner include electronic versions of newspapers, books (i.e., an encyclopedia), articles, financial information, etc. The information can include both text and graphical in any of these examples.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention is pointed out with particularity in the appended claims. However, other features of the invention will become more apparent and the invention will be best understood by referring to the following detailed description in conjunction with the accompanying drawings in which:

FIG. **1** is a block diagram of an embodiment of a system in accordance with the present invention;

FIG. **2** is a flow diagram of a method of retrieving information from an information source;

FIG. **3** is an exemplary block diagram of another embodiment of a system in accordance with the present invention;

FIG. **4** is a block diagram of a voice browser of the system of FIG. **3**;

FIGS. **5***a*–**5***c* are flow diagrams of a routine carried out by the voice browser of FIG. **4**;

FIG. **6** is an exemplary markup language document;

FIG. **7** is a diagrammatic illustration of a hierarchical structure of the markup language document of FIG. **6**;

FIG. **8** is an exemplary state diagram of a markup language document; and

FIG. **9** is another an exemplary state diagram of an exemplary application of a markup language document.

FIG. **10** is an exemplary attribute of a markup language document that allows an audible user input to be converted to a text string.

The following is an example, also shown in FIG. **10**, of the user of the FORM input in a markup language document.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

Before explaining the present embodiments in detail, it should be understood that the invention is not limited in its application or use to the details of construction and arrangement of parts illustrated in the accompanying drawings and description. It will be recognized that the illustrative embodiments of the invention may be implemented or incorporated in other embodiments, variations and modifications, and may be practiced or carried out in various ways. Furthermore, unless otherwise indicated, the terms and expressions employed herein have been chosen for the purpose of describing the illustrative embodiments of the present invention for the convenience of the reader and are not for the purpose of limitation.

Referring now to the drawings, and more particularly to FIG. **1**, a block diagram of a system **100** is illustrated to enable a user to access information. The system **100** generally includes one or more network access apparatus **102** (one being shown), an electronic network **104**, and one or more information sources or content providers **106** (one being shown).

The electronic network **104** is connected to the network access apparatus **102** via a line **108**, and the electronic network **102** is connected to the information source **106** via a line **110**. The lines **108** and **110** can include, but are not limited to, a telephone line or link, an ISDN line, a coaxail line, a cable television line, a fiber optic line, a computer network line, a digital subscriber line, or the like. Alternatively, the network access apparatus **102** and the information source **106** can wirelessly communicate with the electronic network. For example, the electronic network **104** can provide information to the network access apparatus **102** by a satellite communication system, a wireline communication system, or a wireless communication system.

The system **100** enables users to access information from any location in the world via any suitable network access device. The users can include, but are not limited to, cellular subscribers, wireline subscribers, paging subscribers, satellite subscribers, mobile or portable phone subscribers, trunked radio subscribers, computer network subscribers (i.e., internet subscribers, intranet subscribers, etc.), branch office users, and the like.

The users can preferably access information from the information source **106** using voice inputs or commands. For example, the users can access up-to-date information, such as, news updates, designated city weather, traffic conditions, stock quotes, calendar information, user information, address information, and stock market indicators. The system also allows the users to perform various transactions (i.e., order flowers, place orders from restaurants, place buy and sell stock orders, obtain bank account balances, obtain telephone numbers, receive directions to various destinations, etc.).

As shown in FIG. **1**, a user utilizes the network access apparatus **102** of the system **100** to communicate and/or connect with the electronic network **104**. The electronic network **104** retrieves information from the information source **106** based upon speech commands or DTMF tones from the user. The information is preferably stored in a database or storage device (not shown) of the information source **106**. The information source **106** can include one or more server computers (not shown). The information source can be integrated into the electronic network **104** or can be remote from the electronic network (i.e., at a content providers facilities). It will also be recognized that the network

access apparatus **102**, the electronic network **104**, and the information source **106** can be integrated in a single system or device.

The information of the information source **106** can be accessed over any suitable communication medium. The information source **106** can be identified by an electronic address using at least a portion of a URL (Uniform Resource Locator), a URN (Uniform Resource Name), an IP (Internet Protocol) address, an electronic mail address, a device address (i.e. a pager number), a direct point to point connection, a memory address, etc. It is noted that a URL can include: a protocol, a domain name, a path, and a filename. URL protocols include: "file:" for accessing a file stored on a local storage medium; "ftp:" for accessing a file from an FTP (file transfer protocol) server; "http:" for accessing an HTML (hypertext marking language) document; "gopher:" for accessing a Gopher server; "mailto:" for sending an e-mail message; "news:" for linking to a Usenet newsgroup; "telnet:" for opening a telnet session; and "wais:" for accessing a WAIS server.

Once the electronic network **104** of the system **100** receives the information from the information source **106**, the electronic network sends the information to the network access apparatus **102**. The electronic network **104** can include an open, wide area network such as the Internet, the World Wide Web (WWW), and/or an on-line service. The electronic network **104** can also include, but is not limited to, an intranet, an extranet, a local area network, a telephone network, (i.e., a public switched telephone network), a cellular telephone network, a personal communication system (PCS) network, a television network (i.e., a cable television system), a paging network (i.e., a local paging network), a regional paging network, a national or a global paging network, an email system, a wireless data network (i.e., a satellite data network or a local wireless data network), and/or a telecommunication node.

The network access apparatus **102** of the system **100** allows the user to access (i.e., view and/or hear) the information retrieved from the information source. The network access apparatus can provided the information to the user as machine readable data, human readable data, audio or speech communications, textual information, graphical or image data, etc. The network access apparatus can have a variety of forms, including but not limited to, a telephone, a mobile phone, an office phone, a home phone, a pay phone, a paging unit, a radio unit, a web phone, a personal information manager (PIM), a personal digital assistant (PDA), a general purpose computer, a network television, an Internet television, an Internet telephone, a portable wireless device, a workstation, or any other suitable communication device. It is contemplated that the network access device can be integrated with the electronic network. For example, the network access device, the electronic network, and/or the information source can reside in a personal computer.

The network access apparatus **102** may also include a voice or web browser, such as, a Netscape Navigator® web browser, a Microsoft Internet Explorer® web browser, a Mosaic® web browser, etc. It is also contemplated that the network access apparatus **102** can include an optical scanner or bar code reader to read machine readable data, magnetic data, optical data, or the like, and transmit the data to the electronic network **104**. For example, the network access apparatus could read or scan a bar code and then provide the scanned data to the electronic network **104** to access the information from the information source (i.e., a menu of a restaurant, banking information, a web page, weather information, etc.).

FIG. **2** illustrates a flow diagram of a method of retrieving information from a destination or database of the information source **106**. At block **150**, a user calls into the electronic network **104** from a network access apparatus. After the electronic network answers the incoming calls at block **152**, the electronic network can attempt to verify that the user is a subscriber of the system and/or the type of network access apparatus the user is calling from. For example, the system may read and decode the automatic number identification (ANI) or caller line identification (CLI) of the call and then determine whether the CLI of the call is found in a stored ANI or CLI list of subscribers. The system may also identify the user by detecting a unique speech pattern from the user (i.e., speaker verification) or a PIN entered using voice commands or DTMF tones.

After the electronic network answers the call, the electronic network provides a prompt or announcement to the caller at block **154** (i.e., "Hi. This is your personal agent. How may I help you"). The electronic network can also set grammars (i.e., vocabulary) and personalities (i.e., male or female voices) for the call. The electronic network can load the grammars and personalities based upon the CLI, the network access apparatus, or the identity of the user. For example, the grammars and personalities can be set or loaded depending upon the type of device (i.e., a wireless phone), the gender of the caller (i.e., male or female), the type of language (i.e., English, Spanish, etc.), and the accent of the caller (i.e., a New York accent, a southern accent, an English accent, etc.). It is also contemplated that the personalities and grammars may be changed by the user or changed by the electronic network based upon the speech communications detected by the electronic network.

At block **156**, the electronic network waits for an input or command from the user that corresponds to a destination of the information source desired by the user. The input can be audio commands (i.e., speech) or DTMF tones. After the electronic network receives the input from the user, the electronic network establishes a connection or a link to the information source at block **158**. The electronic network preferably determines an electronic address of the information source (i.e., URL, a URN, an IP address, or an electronic mail address) based upon the inputs from the user (i.e., speech or DTMF tones). The electronic address can be retrieved from a database using a look-up operation based upon at least a portion of the input.

At block **160**, the electronic network retrieves at least a portion of the information from the destination of the information source at block **160**. The electronic network processes the information and then provides an output to the user based upon the retrieved information at block **162**. The output can include a speech communication, textual information, and/or graphical information. For example, the electronic network can provide a speech communication using speech-to-text technology or human recorded speech. The process then proceeds to block **164** or block **154** as described above. It will be recognized that the above described method can be carried out by a computer.

Referring now to FIG. **3**, an exemplary block diagram of an embodiment of a system **200** to enable a user to access information is shown. The system **200** enables a user to access information from any location in the world via a suitable communication device. The system **200** can provide access to yellow pages, directions, traffic, addresses, movies, concerts, airline information, weather information, new reports, financial information, flowers, personal data, calendar data, address data, gifts, books, etc. The user can also perform a series of transactions without having to terminate

the original call to the system. For example, the user can access a news update and obtain weather information, all without having to dial additional numbers or terminate the original call. The system **200** also enables application developers to build applications for interactive speech applications using a markup language, such as VoxML™ voice markup language developed by Motorola, Inc.

The system **200** generally includes one or more communication devices or network access apparatus **201**, **202**, **203** and **204** (four being shown), an electronic network **206**, and one or more information sources, such as content providers **208** and **209** (two being shown) and markup language servers. The user can retrieve the information from the information sources using speech commands or DTMF tones.

The user can access the electronic network **206** by dialing a single direct access telephone number (i.e., a foreign exchange number, a local number, or a toll-free number or PBX) from the communication device **202**. The user can also access the electronic network **206** from the communication device **204** via the internet, from the communication device **203** via a paging network **211**, and from the communication device **201** via a local area network (LAN), a wide area network (WAN), or an email connection.

The communication devices can include, but are not limited to, landline or wireline devices (i.e., home phones, work phones, computers, facsimile machines, pay phones), wireless devices (i.e., mobile phones, trunked radios, handheld devices, PIMs, PDAs, etc.), network access devices (i.e. computers), pagers, etc. The communication devices can include a microphone, a speaker, and/or a display.

As shown in FIG. **3**, the electronic network **206** of the system **200** includes a telecommunication network **210** and a communication node **212**. The telecommunication network **210** is preferably connected to the communication node **212** via a high-speed data link, such as, a T1 telephone line, a local area network (LAN), or a wide area network (WAN). The telecommunication network **210** preferably includes a public switched network (PSTN) **214** and a carrier network **216**. The telecommunication network **210** can also include international or local exchange networks, cable television network, interexchange carrier networks (IXC) or long distance carrier networks, cellular networks (i.e., mobile switching centers (MSC)), PBXs, satellite systems, and other switching centers such as conventional or trunked radio systems (not shown), etc.

The PSTN **214** of the telecommunication network **210** can include various types of communication equipment or apparatus, such as ATM networks, Fiber Distributed data networks (FDDI), T1 lines, cable television networks and the like. The carrier network **216** of the telecommunication network **210** generally includes a telephone switching system or central office **218**. It will be recognized that the carrier network **216** can be any suitable system that can route calls to the communication node **212**, and the telephone switching system **218** can be any suitable wireline or wireless switching system.

The communication node **212** the system **200** is preferably configured to receive and process incoming calls from the carrier network **216** and the internet **220**, such as the WWW. The communication node can receive and process pages from the paging network **211** and can also receive and process messages (i.e., emails) from the LAN, WAN or email connection **213**.

When a user dials into the electronic network **206** from the communication device **202**, the carrier network **216** routes the incoming call from the PSTN **214** to the communication node **212** over one or more telephone lines or trunks. The incoming calls preferably enters the carrier network **216** through one or more "888"[or "800" INWATS trunk lines, local exchange trunk lines, or long distance trunk lines. It is also contemplated that the incoming calls can be received from a cable network, a cellular system, or any other suitable system.

The communication node **212** answers the incoming call from the carrier network **216** and retrieves an appropriate announcement (i.e., a welcome greeting) from a database, server, or browser. The node **212** then plays the announcement to the caller. In response to audio inputs from the user, the communication node **212** retrieves information from a destination or database of one or more of the information sources, such as the content providers **208** and **209** or the markup language servers. After the communication node **212** receives the information, the communication node provides a response to the user based upon the retrieved information.

The node **212** can provide various dialog voice personalities (i.e., a female voice, a male voice, etc.) and can implement various grammars (i.e., vocabulary) to detect and respond to the audio inputs from the user. In addition, the communication node can automatically select various speech recognition models (i.e., an English model, a Spanish model, an English accent model, etc.) based upon a user profile, the user's communication device, and/or the user's speech patterns. The communication node **212** can also allow the user to select a particular speech recognition model.

When a user accesses the electronic network **206** from a communication device registered with the system (i.e., a user's home phone, work phone, cellular phone, etc.), the communication node **212** can by-pass a user screening option and automatically identify the user (or the type of the user's communication device) through the use of automatic number identification (ANI) or caller line identification (CLI). After the communication node verifies the call, the node provides a greeting to the user (i.e., "Hi, this is your personal agent, Maya. Welcome Bob. How may I help you?"). The communication node then enters into a dialogue with the user, and the user can select a variety of information offered by the communication node.

When the user accesses the electronic network **206** from a communication device not registered with the system (i.e., a payphone, a phone of a non-subscriber, etc.), the node answers the call and prompts the user to enter his or her name and/or a personal identification number (PIN) using speech commands or DTMF tones. The node can also utilize speaker verification to identify a particular speech pattern of the user. If the node authorizes the user to access the system, the node provides a personal greeting to the user (i.e., "Hi, this is your personal agent, Maya. Welcome Ann. How may I help you?"). The node then enters into a dialogue with the user, and the user can select various information offered by the node. If the name and/or PIN of the user cannot be recognized or verified by the node, the user will be routed to a customer service representative.

As shown in FIG. **3**, the communication node **212** preferably includes a telephone switch **230**, a voice or audio recognition (VRU) client **232**, a voice recognition (VRU) server **234**, a controller or call control unit **236**, an Operation and Maintenance Office (OAM) or a billing server unit **238**, a local area network (LAN) **240**, an application server unit **242**, a database server unit **244**, a gateway server or router

firewall server **246**, a voice over internet protocol (VOIP) unit **248**, a voice browser **250**, a markup language server **251**, and a paging server **252**. Although the communication node **206** is shown as being constructed with various types of independent and separate units or devices, the communication node **212** can be implemented by one or more integrated circuits, microprocessors, microcontrollers, or computers which may be programmed to execute the operations or functions equivalent to those performed by the device or units shown. It will also be recognized that the communication node **212** can be carried out in the form of hardware components and circuit designs, software or computer programming, or a combination thereof.

The communication node **212** can be located in various geographic locations throughout the world or the United States (i.e., Chicago, Ill.). The communication node **212** can be operated by one or more carriers (i.e., Sprint PCS, Qwest Communications, MCI, etc.) or independent service providers, such as, for example, Motorola, Inc.

The communication node **212** can be co-located or integrated with the carrier network **216** (i.e., an integral part of the network) or can be located at a remote site from the carrier network **216**. It is also contemplated that the communication node **212** may be integrated into a communication device, such as, a wireline or wireless phone, a radio device, a personal computer, a PDA, a PIM, etc. In this arrangement, the communication device can be programmed to connect or link directly into an information source.

The communication node **212** can also be configured as a standalone system to allow users to dial directly into the communication node via a toll free number or a direct access number. In addition, the communication node **212** may comprise a telephony switch (i.e., a PBX or Centrix unit), an enterprise network, or a local area network. In this configuration, the system **200** can be implemented to automatically connect a user to the communication node **212** when the user picks a communication device, such as, the phone.

When the telephone switch **230** of the communication node **212** receives an incoming call from the carrier network **216**, the call control unit **236** sets up a connection in the switch **230** to the VRU client **232**. The communication node **212** then enters into a dialog with the user regarding various services and functions. The VRU client **232** preferably generates pre-recorded voice announcements and/or messages to prompt the user to provide inputs to the communication node using speech commands or DTMF tones. In response to the inputs from the user, the node **212** retrieves information from a destination of one of the information sources and provides outputs to the user based upon the information.

The telephone switch **230** of the telecommunication node **212** is preferably connected to the VRU client **232**, the VOIP unit **248**, and the LAN **240**. The telephone switch **230** receives incoming calls from the carrier switch **216**. The telephone switch **230** also receives incoming calls from the communication device **204** routed over the internet **220** via the VOIP unit **248**. The switch **230** also receives messages and pages from the communication devices **201** and **203**, respectively. The telephone switch **230** is preferably a digital cross-connect switch, Model No. LNX, available from Excel Switching Corporation, **255** Independence Drive, Hyannis, Mass. 02601. It will be recognized that the telephone switch **230** can be any suitable telephone switch.

The VRU client **232** of the communication node **212** is preferably connected to the VRU server **234** and the LAN **240**. The VRU client **232** processes speech communications, DTMF tones, pages, and messages (i.e., emails) from the user. Upon receiving speech communications from the user, the VRU client **232** routes the speech communications to the VRU server **234**. When the VRU client **232** detects DTMF tones, the VRU client **232** sends a command to the call control unit **236**. it will be recognized that the VRU client **232** can be integrated with the VRU server.

The VRU client **232** preferably comprises a computer, such as, a Windows NT compatible computer with hardware capable of connecting individual telephone lines directly to the switch **230**. The VRU client preferably includes a microprocessor, random access memory, read-only memory, a T1 or ISDN interface board, and one or more voice communication processing board (not shown). The voice communication processing boards of the VRU client **232** are preferably Dialogic boards, Model No. Antares, available from Dialogic Corporation, 1515 Route 10, Parsippany, N.J. 07054. The voice communication boards may include a voice recognition engine having a vocabulary for detecting a speech pattern (i.e., a key word or phrase). The voice recognition engine is preferably a RecServer software package, available from Nuance Communications, **1380** Willow Road, Menlo Park, Calif. 94025.

The VRU client **232** can also include an echo canceler (not shown) to reduce or cancel text-to-speech or playback echoes transmitted from the PSTN **214** due to hybrid impedance mismatches. The echo canceler is preferably included in an Antares Board Support Package, available from Dialogic.

The call control unit **236** of the communication node **212** is preferably connected to the LAN **240**. The call control unit **236** sets up the telephone switch **230** to connect incoming calls to the VRU client **232**. The call control unit also sets up incoming calls or pages into the node **212** over the internet **220** and pages and messages sent from the communication devices **201** and **203** via the paging network **203** and email system **213**. The control call unit **236** preferably comprises a computer, such as, a Window NT compatible computer.

The LAN **240** of the communication node **212** allows the various components and devices of the node **212** to communicate with each other via a twisted pair, a fiber optic cable, a coaxial cable, or the like. The LAN **240** may use Ethernet, Token Ring, or other suitable types of protocols. The LAN **240** is preferably a 100 Megabit per second Ethernet switch, available from Cisco Systems, San Jose, Calif. It will be recognized that the LAN **240** can comprise any suitable network system, and the communication node **212** may include a plurality of LANS.

The VRU server **234** of the communication node **212** is connected to the VRU client **232** and the LAN **240**. The VRU server **234** receives speech communications from the user via the VRU client **232**. The VRU server **234** processes the speech communications and compares the speech communications against a vocabulary or grammar stored in the database server unit **244** or a memory device. The VRU server **234** provides output signals, representing the result of the speech processing, to the LAN **240**. The LAN **240** routes the output signal to the call control unit **236**, the application server **242**, and/or the voice browser **250**. The communication node **212** then performs a specific function associated with the output signals.

The VRU server **234** preferably includes a text-to-speech (TTS) unit **252**, an automatic speech recognition (ASR) unit **254**, and a speech-to-text (STT) unit **256**. The TTS unit **252**

of the VRU server **234** receives textual data or information (i.e., e-mail, web pages, documents, files, etc.) from the application server unit **242**, the database server unit **244**, the call control unit **236**, the gateway server **246**, the application server **242**, and the voice browser **250**. The TTS unit **252** processes the textual data and converts the data to voice data or information.

The TTS unit **252** can provide data to the VRU client **232** which reads or plays the data to the user. For example, when the user requests information (i.e., news updates, stock information, traffic conditions, etc.), the communication node **212** retrieves the desired data (i.e., textual information) from a destination of the one or more of the information sources and converts the data via the TTS unit **252** into a response.

The response is then sent to the VRU client **232**. The VRU client processes the response and reads an audio message to the user based upon the response. It is contemplated that the VRU server **234** can read the audio message to the user using human recorded speech or synthesized speech. The TTS unit **252** is preferably a TTS **2000** software package, available from Lernout and Hauspie Speech Product NV, **52** Third Avenue, Burlington, Mass. 01803.

The ASR unit **254** of the VRU server **234** provides speaker independent automatic speech recognition of speech inputs or communications from the user. It is contemplated that the ASR unit **254** can include speaker dependent speech recognition. The ASR unit **254** processes the speech inputs from the user to determine whether a word or a speech pattern matches any of the grammars or vocabulary stored in the database server unit **244** or downloaded from the voice browser. When the ASR unit **254** identifies a selected speech pattern of the speech inputs, the ASR unit **254** sends an output signal to implement the specific function associated with the recognized voice pattern. The ASR unit **254** is preferably a speaker independent speech recognition software package, Model No. RecServer, available from Nuance Communications. It is contemplated that the ASR unit **254** can be any suitable speech recognition unit to detect voice communications from a user.

The STT unit **256** of the VRU server **234** receives speech inputs or communications from the user and converts the speech inputs to textual information (i.e., a text message). The textual information can be sent or routed to the communication devices **201**, **202**, **203** and **204**, the content providers **208** and **209**, the markup language servers, the voice browser, and the application server **242**. The STT unit **256** is preferably a Naturally Speaking software package, available from Dragon Systems, **320** Nevada Street, Newton, Mass. 02160–9803.

The VOIP unit **248** of the telecommunication node **212** is preferably connected to the telephone switch **230** and the LAN **240**. The VOIP unit **248** allows a user to access the node **212** via the internet **220** using voice commands. The VOIP unit **240** can receive VOIP protocols (i.e., H. **323** protocols) transmitted over the internet **220** and can convert the VOIP protocols to speech information or data. The speech information can then be read to the user via the VRU client **232**. The VOIP unit **248** can also receive speech inputs or communications from the user and convert the speech inputs to a VOIP protocol that can be transmitted over the internet **220**. The VOIP unit **248** is preferably a Voice Net software package, available from Dialogic Corporation. It will be recognized that the VOIP device can be incorporated into a communication device.

The telecommunication node **212** also includes a detection unit **260**. The detection unit **260** is preferably a phrase or key word spotter unit to detect incoming audio inputs or communications or DTMF tones from the user. The detector unit **260** is preferably incorporated into the switch **230**, but can be incorporated into the VRU client **232**, the carrier switch **216**, or the VRU server **256**. The detection unit **260** is preferably included in a RecServer software package, available from Nuance Communications.

The detection unit **260** records the audio inputs from the user and compares the audio inputs to the vocabulary or grammar stored in the database server unit **244**. The detector unit continuously monitors the user's audio inputs for a key phase or word after the user is connected to the node **212**. When the key phrase or word is detected by the detection unit **260**, the VRU client **232** plays a pre-recorded message to the user. The VRU client **232** then responds to the audio inputs provided by the user.

The billing server unit **238** of the communication node **212** is preferably connected to the LAN **240**. The billing server unit **238** can record data about the use of the communication node by a user (i.e., length of calls, features accessed by the user, etc.). Upon completion of a call by a user, the call control unit **236** sends data to the billing server unit **238**. The data can be subsequently processed by the billing server unit in order to prepare customer bills. The billing server unit **238** can use the ANI or CLI of the communication device to properly bill the user. The billing server unit **238** preferably comprises a Windows NT compatible computer.

The gateway server unit **246** of the communication node **212** is preferably connected to the LAN **240** and the internet **220**. The gateway server unit **246** provides access to the content provider **208** and the markup language server **257** via the internet **220**. The gateway unit **246** also allows users to access the communication node **212** from the communication device **204** via the internet **220**. The gateway unit **246** can further function as a firewall to control access to the communication node **212** to authorized users. The gateway unit **246** is preferably a Cisco Router, available from Cisco Systems.

The database server unit **244** of the communication node **212** is preferably connected to the LAN **240**. The database server unit **244** preferably includes a plurality of storage areas to store data relating to users, speech vocabularies, dialogs, personalities, user entered data, and other information. Preferably, the database server unit **244** stores a personal file or address book. The personal address book can contain information required for the operation of the system, including user reference numbers, personal access codes, personal account information, contact's addresses, and phone numbers, etc. The database server unit **244** is preferably a computer, such as an NT Window compatible computer.

The application server **242** of the communication node **212** is preferably connected to the LAN **240** and the content provider **209**. The application server **242** allows the communication node **212** to access information from a destination of the information sources, such as the content providers and markup language servers. For example, the application server can retrieve information (i.e., weather reports, stock information, traffic reports, restaurants, flower shops, banks, etc.) from a destination of the information sources. The application server **242** processes the retrieved information and provides the information to the VRU server **234** and the voice browser **250**. The VRU server **234** can provide an audio announcement to the user based upon the information using text-to-speech synthesizing or human recorded voice.

The application server **242** can also send tasks or requests (i.e., transactional information) received from the user to the information sources (i.e., a request to place an order for a pizza). The application server **242** can further receive user inputs from the VRU server **234** based upon a speech recognition output. The application server is preferably a computer, such as an NT Windows compatible computer.

The markup language server **251** of the communication node **212** is preferably connected to the LAN **240**. The markup language server **251** can include a database, scripts, and markup language documents or pages. The markup language server **251** is preferably a computer, such as an NT Window Compatible Computer. It will also be recognized that the markup language server **251** can be an internet server (i.e., a Sun Microsystems server).

The paging server **252** of the communication node **212** is preferably connected to the LAN **240** and the paging network **211**. The paging server **252** routes pages between the LAN **240** and the paging network. The paging server **252** is preferably a computer, such as a NT compatible computer.

The voice browser **250** of the system **200** is preferably connected to the LAN **240**. The voice browser **250** preferably receives information from the information sources, such as the content provider **209** via the application server **242**, the markup language servers **251** and **257**, the database **244**, and the content provider **208**. In response to voice inputs from the user or DTMF tones, the voice browser **250** generates a content request (i.e., an electronic address) to navigate to a destination of one or more of the information sources. The content request can use at least a portion of a URL, a URN, an IP, a page request, or an electronic email.

After the voice browser is connected to an information source, the voice browser preferably uses a TCP/IP connect to pass requests to the information source. The information source responds to the requests, sending at least a portion of the requested information, represented in electronic form, to the voice browser. The information can be stored in a database of the information source and can include text content, markup language document or pages, non-text content, dialogs, audio sample data, recognition grammars, etc. The voice browser then parses and interprets the information as further described below. It will be recognized that the voice browser can be integrated into the communication devices **201**, **202**, **203**, and **204**.

As shown in FIG. **3**, the content provider **209** is connected to the application server **244** of the communication node **212**, and the content provider **208** is connected to the gateway server **246** of the communication node **212** via the internet **220**. The content providers can store various content information, such as news, weather, traffic conditions, etc. The content providers **208** and **209** can include a server to operate web pages or documents in the form of a markup language. The content providers **208** and **209** can also include a database, scripts, and/or markup language documents or pages. The scripts can include images, audio, grammars, computer programs, etc. The content providers execute suitable server software to send requested information to the voice browser.

Referring now to FIG. **4**, a block diagram of the voice browser **250** of the communication node **212** is illustrated. The voice browser **250** generally includes a network fetcher unit **300**, a parser unit **302**, an interpreter unit **304**, and a state machine unit **306**. Although the voice browser is shown as being constructed with various types of independent and separate units or devices, it will be recognized that the voice browser **250** can be carried out in the form of hardware components and circuit designs, software or computer programming, or a combination thereof.

The network fetcher **300** of the voice browser **250** is connected to the parser **302** and the interpreter **304**. The network fetcher **300** is also connected to the LAN **240** of the communication node **212**. The network fetcher unit **304** retrieves information, including markup language documents, audio samples and grammars from the information sources.

The parser unit **302** of the voice browser **250** is connected to the network fetcher unit **300** and the state machine unit **306**. The parser unit **302** receives the information from the network fetcher unit **300** and parses the information according to the syntax rules of the markup language as further described below (i.e., extensible markup language syntax). The parser unit **302** generates a tree or heirarchial structure representing the markup language that is stored in memory of the state machine unit **306**. A tree structure of an exemplary markup language document is shown in FIG. **7**.

The following text defines the syntax and grammar that the parser unit of the voice browser utilizes to build a tree structure of the markup language document.

```
    <!ELEMENT dialog (step|class)*>
    <!ATTLIST dialog bargein (Y|N) "Y">
    <!ELEMENT step (prompt|input|help|error|cancel|ack)*>
    <!ATTLIST step name ID #REQUIRED
            parent IDREF #IMPLIED
            bargein (Y|N) "Y"
            cost CDATA #IMPLIED>
    <!ELEMENT class (prompt|help|error|cancel|ack)*>
    <!ATTLIST class name ID #REQUIRED
            parent IDREF #IMPLIED
            bargein (Y|N) "Y"
            cost CDATA #IMPLIED>
    <!ELEMENT prompt
(#PCDATA|options|value|emp|break|pros|audio)*>
    <!ELEMENT emp
(#PCDATA|options|value|emp|break|pros|audio)*>
    <!ATTLIST emp level (strong|moderate|none|reduced)
"moderate">
    <!ELEMENT pros
(#PCDATA|options|value|emp|break|pros|audio)*>
    <!ATTLIST pros rate CDATA #IMPLIED
            vol CDATA #IMPLIED
            pitch CDATA #IMPLIED
            range CDATA #IMPLIED>
    <!ELEMENT help
(#PCDATA|options|value|emp|break|pros|audio)*>
    <!ATTLIST help ordinal CDATA #IMPLIED
            reprompt (Y|N) "N"
            next CDATA #IMPLIED
            nextmethod (get|post) "get">
    <!ELEMENT error
(#PCDATA|options|value|emp|break|pros|audio)*>
    <!(ATTLIST error type NMTOKENS "ALL"
            ordinal CDATA #IMPLIED
            reprompt (Y|N) "N"
            next CDATA #IMPLIED
            nextmethod (get|post) "get">
    <!ELEMENT cancel
(#PCDATA|value|emp|break|pros|audio)*>
    <!ATTLIST cancel next CDATA #REQUIRED
            nextmethod (get|post) "get">
    <!ELEMENT audio EMPTY>
    <!ATTLIST audio src CDATA #REQUIRED>
    <!ELEMENT ack
    (#PCDATA|options|value|emp|break|pros|audio)*>
    <!ATTLIST ack confirm NMTOKEN "YORN"
            background (Y|N) "N"
            reprompt (Y|N) "N">
    <!ELEMENT input
(option|response|rename|switch|case)*>
```

-continued

```
    <!ATTLIST input type
(none|optionlist|record|grammar|profile|hidden|
    yorn|digits|number|time|date|money|phone) #REQUIRED
            name ID #IMPLIED
            next CDATA #IMPLIED
            nextmethod (get|post) "get"
            timeout CDATA #IMPLIED
            min CDATA #IMPLIED
            max CDATA #IMPLIED
            profname NMTOKEN #IMPLIED
            subtype NMTOKEN #IMPLIED
            src CDATA #IMPLIED
            value CDATA #IMPLIED
            msecs CDATA #IMPLIED
            storage (file|request) #REQUIRED
            format CDATA #IMPLIED>
    <!ELEMENT switch (case|switch)*>
    <!ATTLIST switch field NMTOKEN #REQUIRED>
    <!ELEMENT response (switch)*>
    <!ATTLIST response next CDATA #IMPLIED
            nextmethod (get|post) "get"
            fields NMTOKENS #REQUIRED>
    <!ELEMENT rename EMPTY>
    <!ATTLIST rename varname NMTOKEN #REQUIRED
            recname NMTOKEN #REQUIRED>
    <!ELEMENT case EMPTY>
    <!ATTLIST case value CDATA #REQUIRED
            next CDATA #REQUIRED
            nextmethod (get|post) "get">
    <!ELEMENT value EMPTY>
    <!ATTLIST value name NMTOKEN #REQUIRED>
    <!ELEMENT break EMPTY>
    <!ATTLIST break msecs CDATA #IMPLIED>
            size (none|small|medium|large)
"medium">
    <!ELEMENT options EMPTY>
    <!ELEMENT or EMPTY>
    <!ELEMENT option (#PCDATA|value|or)*>
    <!ATTLIST option value CDATA #IMPLIED
            next CDATA #IMPLIED
            nextmethod (get|post) "get">
```

Referring again to FIG. **4**, the interpreter unit **304** of the voice browser **250** is connected to the state machine unit **306** and the network fetcher unit **300**. The interpreter unit **304** is also connected to the LAN. The interpreter unit **304** carries out a dialog with the user based upon the tree structure representing a markup language document.

The interpreter unit sends data to the TTS **252**. The interpreter unit **304** can also receive data based upon inputs from the user via a VRU server and can send outputs to the information source based upon the user inputs.

The interpreter unit **304** can transition from state to state (i.e., step to step) within a tree structure (i.e., a dialog) of a markup language document or can transition to a new tree structure within the same dialog or another dialog. The interpreter unit determines the next state or step based upon the structure of the dialog and the inputs from the user. When the interpreter unit transitions to a new dialog or page, the address of the new dialog or page is then sent to the network fetcher.

The state machine **306** of the voice browser **250** is connected to the parser unit **302** and the interpreter unit **304**. The state machine **306** stores the tree structure of the markup language and maintains the current state or step that the voice browser is executing.

FIGS. **5***a*–**5***c* illustrate a flow diagram of a software routine executed by the voice browser **250** . The software routine allows interactive voice applications. At block **400**, the voice browser **250** determines an initial address (i.e., a URL) and a step element or name. The voice browser then fetches the contents (i.e., a markup or language document) of the current address from the information sources (i.e., content providers and markup language servers) at block **402**. After the voice browser fetches the address, the voice browser processes the contents and builds a local step table (i.e., a tree structure) at block **404**.

At block **406**, a prompt can be played to the user via the TTS unit of the system **200** for the current element.

The voice browser then waits for an input from the user (i.e., speech or DTMF tones). At block **408**, the voice browser can collect input from the user for the current step element. FIG. **5**c shows an exemplary flow diagram of a routine that is executed by the voice browser to determine the grammar for speech recognition.

At block **502**, the voice browser determines whether a pre-determined grammar exists for the user input and the markup language. For example, the voice browser determines whether the grammar for the user input is found in a predetermined or pre-existing grammar stored in a database or contained in the markup language. If the grammar is found, the voice browser sends the grammar to the VRU server at block **504**. At block **506**, the VRU server compares the user input to the grammar to recognize the user input. After the VRU server recognizes the user input, the process proceeds to block **410** (see FIG. **5***a*) as described below.

If a pre-existing grammar is not found at block **502**, the voice browser dynamically generates the grammar for the user input. At block **508**, the voice browser looks up the pronunciations for the user in a dictionary at block **508**. The dictionary can be stored in a database of the system or stored on an external database (i.e., the voice browser can fetch a dictionary from the processor or from the internet).

At block **510**, the voice browser generates the grammar for the user inputs based upon the pronunciations from the dictionary and phonetic rules. A software routine available from Nuance Communication, Model No. RecServer, can be used to generate the grammar. At block **512**, the grammar is sent to the VRU server. The voice browser then attempts to match the grammar to the user input at block **506**.

After the voice browser detects or collects an input from the user at block **408**, the voice browser determines whether there is an error at block **410**. If the voice browser is having difficulty recognizing inputs from the user or detects a recognition error, a timeout error, etc., an appropriate error message is played to the user at block **414**. For example, if the voice browser detected too much speech from the user or the recognition is too slow, a prompt is played (i.e., "Sorry, I didn't understand you") to the user via the VRU server. If the voice browser receives unexpected DTMF tones, a prompt is played (i.e., "I heard tones. Please speak your response") to the user via the VRU server. If the voice browser does not detect any speech from the user, a prompt is read to the user (i.e., "I am having difficulty hearing you").

At block **416**, the voice browser determines whether a re-prompt was specified in the error response or element. If a re-prompt is to be played to the user at block **416**, the process proceeds to block **406** as described above. If a re-prompt is not to be played to the user at block **416**, the voice browser determines whether there is a next step element specified in the error response at block **420**. If another step element is specified in the error response at block **420**, the process proceed to block **402** as described above. If another step element is not specific in the error response at block **420**, the process proceeds to block **422**.

If the voice browser does not detect a recognition error at block **410**, the voice browser determines whether the user

requested help at block **412**. If the user requested help, an appropriate help response is played to the user (i.e., "please enter or speak your pin") at block **424**.

At block **425**, the voice browser determines whether a re-prompt was specified in the help response or step. If a re-prompt is specified in the help response at block **425**, the process proceeds to block **406** as described above. If a re-prompt is not specified in the help response at block **425**, the voice browser determines whether a next step element is specified in the help response at block **426**. If another step element is specified in the help response at block **426**, the process proceeds to block **402** as described above. If another step element is not specific in the help response at block **426**, the process proceeds to block **428**.

At block **430**, the voice browser determines whether a cancel request has been indicated by the user. If the voice browser detects a cancel request from the user at block **430**, an appropriate cancel message is played to the user at block **434** (i.e.,"Do you wish to exit and return to the Main Menu?").

At block **436**, the voice browser then determines whether there a next step element is specified in the cancel response or element. If another step element is specified in the cancel response at block **436**, the process proceeds to block **448**. If another step element is not specified in the error response at block **436**, the process proceeds to block **422**.

If a cancel request was not detected at block **430**, the voice browser determines the next step element at block **432**. At block **440**, the voice browser determines whether there is an acknowledgement specified in the next step element. If there is no acknowledgement specified in the step element at block **440**, the voice browser sets the current step element to the next step element at block **442** and then determines whether the next step element is within the same page at block **444**.

If the next step element is within the same page as the current step element at block **444**, the process proceeds to block **446**. If the next step element is not within the same page as the current page at block **444**, the process proceeds to block **448**.

If an acknowledgement is specified in the next step element at block **440**, an acknowledgement response is played to the user at block **450**. The voice browser then determines whether a confirmation is specified in the information (i.e., a markup language document) at block **452**. If a confirmation is not specified in the information at block **452**, the process proceeds to block **442** as described above. If a confirmation is specified at block **452**, the voice browser determines whether the response was recognized from the user a block **454** and then determines whether the response is affirmative at block **456**.

If the voice browser receives an affirmative response at block **456**, the process proceeds to block **442** as described above. If the voice browser does not receive an affirmative response from the user at block **456**, the process proceeds to block **448**.

The following text describes an exemplary markup language processed by the voice browser of the communication node **212**. The markup language preferably includes text, recorded sound samples, navigational controls, and input controls for voice applications as further described below. The markup language enables system designers or developers of service or content providers to create application programs for instructing the voice browser to provide a desired user interactive voice service. The markup language also enables designers to dynamically customize their content. For example, designers can provide up-to-date news, weather, traffic, etc.

The markup language can be designed to express flow of control, state management, and the content of information flow between the communication node **212** and the user. The structure of the language can be designed specifically for voice applications and the markup language is preferably designed and delivered in units of dialog.

The markup language can include elements that describe the structure of a document or page, provide pronunciation of words and phrases, and place markers in the text to control interactive voice services. The markup language also provides elements that control phrasing, emphasis, pitch, speaking rate, and other characteristics. The markup language documents are preferably stored on databases of the information sources, such as the content providers **208** and **209** and the markup language servers **251** and **257**.

FIG. **6** illustrates an exemplary markup language document that the voice browser of the communication node can process. The markup language document has a hierarchical structure, in which every element (except the dialog element) is contained by another element. Elements between another elements are defined to be children or a lower element of the tree. FIG. **7** illustrates a tree stucture of the markup language document of FIG. **6**.

As shown in FIG. **6**, the markup language document includes tags, denoted by <> symbols, with the actual element between the brackets. The markup language includes start tags ("<>") and end tags ("</>"). A start tag begins a markup element and the end tags ends the corresponding markup element. For example, in the markup language document as shown in FIG. **6**, the DIALOG element (<dialog>) on line **2** begins a markup language document or page, and the dialog element (<dialog>) on line **26** indicates the markup language document has ended. The elements often have attributes which are assigned values as further described below.

The DIALOG element and STEP elements of a markup language document provide the basic structure of the document. The DIALOG element defines the scope of the markup language document, and all other elements are contained by the DIALOG element. The STEP elements define states within a DIALOG element (i.e., the STEP element defines an application state). For example, an application state can include initial prompts, help messages, error messages, or cleanup and exit procedures.

The DIALOG element and the associated STEP elements of a markup language document define a state machine that represents an interactive dialogue between the voice browser and the user. When the voice browser interprets the markup language document, the voice browser will navigate through the DIALOG element to different STEP elements as a result of the user's responses.

The following example illustrates an exemplary markup language document that the voice browser of the communication node can process. The example has one DIALOG element and two STEP elements.

```
<?XML VERSION="1.0"?>
<DIALOG>
    <STEP NAME="init">
        <PROMPT> Please select a soft drink. </PROMPT>
        <HELP> Your choices are coke, pepsi, 7 up,
            or root beer. </HELP>
```

-continued

```
            <INPUT TYPE="optionlist" NAME="drink">
                <OPTION NEXT="#confirm"> coke </OPTION>
                <OPTION NEXT="#confirm"> pepsi </OPTION>
            <OPTION NEXT="#confirm"> 7 up </OPTION>
            <OPTION NEXT="#confirm"> root beer </OPTION>
        </INPUT>
    </STEP>
    <STEP NAME="confirm">
        <PROMPT> You ordered a <VALUE NAME="drink"/>.
            </PROMPT>
    </STEP>
</DIALOG>
```

When the above markup language document is interpreted by the voice browser, the voice browser initially executes the STEP element called "init". First, the user will hear the text contained by the prompt element (i.e., "Please select a soft drink."). If the user responds "help" before making a selection, the user would hear the text contained with the HELP element (i.e., "Your choices are coke, pepsi, 7up, or root beer."). After the user makes a selection, the voice browser will execute the STEP element named "confirm", which will read back the user's selection and then exit the application. It is noted that the STEP elements in a markup language document are executed based on the user's responses not on the order of the STEP elements within the source file.

Although the definition of the "init" STEP element appears before and the definition of the "confirm" STEP element, the order in which they are defined has no impact on the order in which the voice browser navigates through them.

The following text describes the markup language elements, their attributes, and their syntax. The DIALOG element of the markup language (i.e., <DIALOG [BARGEIN="value"]> markup language document </DIALOG>) is the fundamental element of the markup language. The DIALOG element includes a BARGEIN attribute. The value of the BARGEIN attribute can be "Y" and "N". The BARGEIN attribute allows the DIALOG element to be interrupted at any time based upon a predetermined response from the user (i.e., wake up).

The DIALOG element defines the basic unit of context within an application, and typically, there is one DIALOG element per address (i.e., URL). Each DIALOG element contains one STEP element named "init". The execution of the DIALOG element begins with the STEP named "init".

The following example of a markup language document or page contains the DIALOG element.

```
<DIALOG>
    <STEP NAME="init">
        <PROMPT> Welcome to VoxML ™ voice markup
        language. </PROMPT>
    </STEP>
</DIALOG>
```

In the example above, the DIALOG element contains a single STEP element named "init". The STEP element has a single PROMPT element that will be read to the user via the text-to-speech unit **252**. Since there is no INPUT element defined in the STEP element, the markup language application will terminate immediately after the PROMPT element is read.

The STEP element of the markup language (i.e., <STEP NAME="value"[PARENT="value"][BARGEIN="value"] [COST="value"]> text </STEP>) defines a state in a markup language document or page. The STEP element is contained by a DIALOG element. The STEP element includes a NAME attribute, a PARENT attribute, a BARGEIN attribute, and a COST attribute. The value of the NAME and PARENT attribute can be an identifier (i.e., a pointer or a variable name), the value of the BARGEIN attribute can be "Y" and "N", and the value of the COST attribute can be an integer.

The STEP element typically has an associated PROMPT element and INPUT element that define the application state. The following example illustrates the use of the STEP element in a markup language document.

```
<STEP NAME="askpython" PARENT="tvrating">
    <PROMPT> Please rate Monty Python's Flying
    Circus
        on a scale of 1 to 10. </PROMPT>
    <INPUT NAME="python" TYPE="number" NEXT="#drwho"
    />
</STEP>
```

The example shown above illustrates a STEP element that collects the user's opinion on one of several public television shows. The STEP element uses the PARENT attribute to share a common set of help and error elements with other TV-show-rating STEP elements. For example, the PARENT attribute can contain a HELP element explaining what a rating of 1, 5, and 10 would mean, and a common error message can remind the user that a numeric rating is expected.

The PROMPT element of the markup language (i.e., <PROMPT> text </PROMPT>) is used to define content (i.e., text or an audio file) that is to be presented to the user. Typically, the PROMPT element will contain text and several markup elements (i.e., the BREAK or EMP elements as described below) that are read to the user via the text-to-speech unit.

The PROMPT element can be contained within a STEP or a CLASS element. The following example illustrates the use of the PROMPT element in markup language document or page.

```
<STEP NAME="init">
    <PROMPT> How old are you? </PROMPT>
    <INPUT TYPE="number" NAME="age" NEXT=#weight"/>
</STEP>
```

In the example shown above, the text "How old are you?" will be played to the user via the text-to-speech unit, and then the voice browser will wait for the user to say his or her age.

The INPUT element of the markup language is used to define a valid user input within each STEP element. The INPUT element is contained within a STEP element. The INPUT element of the markup language includes an INPUT attribute. The value of the INPUT attribute can be a DATE input, a DIGIT input, a FORM input, a GRAMMAR input, a HIDDEN input, a MONEY input, a NONE element, a NUMBER input, an OPTIONLIST input, a PHONE input, a PROFILE input, a RECORD input, a TIME input, and a YORN element.

The DATE input of the INPUT attribute of the markup language (i.e., <INPUT TYPE="DATE" NAME="value" NEXT="value" [NEXTMETHOD="value"][TIsMEOUT= "value"]/>) is used to collect a calendar date from the user. The DATE input includes a NAME attribute, a NEXT attribute, a NEXTMETHOD attribute, and a TIMEOUT attribute. The value of the NAME attribute can be an identifier, and the value of the NEXT attribute can be the next STEP address (i.e., a URL). The value of the NEXT-METHOD attribute can be a get and a post (i.e., an input into a Java Script program or a markup language server), and the value of the TIMEOUT attribute can be a number represented in milliseconds.

The following example illustrates the use of the DATE input in a markup language document.

```
<STEP NAME="init">
<PROMPT> What is your date of birth? <PROMPT>
    <INPUT TYPE="date" NAME="dob" NEXT="#soc"/>
    </STEP>
```

In the example above, the DATE input is used to gather the user's birthday, store it in a variable "dob", and then go to the STEP element named "sock". The DATE input makes use of an input grammar to interpret the user's response and store that response in a standard format.

The DATE input grammar can interpret dates expressed in several different formats. A fully defined date, such as, "next Friday, Jul. 10th, 1998" is stored as "07101998|Jul.|10|1998|Friday|next". If the date cannot be determined by the user's response, the ambiguous parts of the response will be omitted from the data. The response "July 4th", is stored as "????????|july|4|||", "Tomorrow" becomes "????????|||||tomorrow", "The 15th" is stored as "????????||15|||", and "Monday" becomes "????????||||Monday|".

The DIGITS input of the INPUT attribute of the markup language (i.e., <INPUT TYPE="DIGITS" NAME="value" NEXT="value" [NEXTMETHOD="value"] [TIMEOUT= "value"] [MIN="value"] [MAX="value"]/>) is used to collect a series of digits from the user. The DIGITS input includes a NAME attribute, a NEXT attribute, a NEXT-METHOD attribute, a TIMEOUT attribute, a MIN attribute, and a MAX attribute. The value of the NAME attribute can be an identifier, the value of the NEXT attribute can be a next step address (i.e., a URL), the value of the NEXT-METHOD attribute can be a get and a post, and the value of the TIMEOUT attribute can be a number represented in milliseconds. The value of the MIN and MAX attributes can be minimum and maximum integer values, respectively.

The following example illustrates the use the DIGITS input in a markup language document or page.

```
<STEP NAME="init">
    <PROMPT> Please say your pin now. </PROMPT>
    <INPUT TYPE="digits" NAME="pin" NEXT="#doit"/>
</STEP>
```

In the example above, the DIGITS input is used to collect digits from the user, store the number in the a variable named "pin", and then go to the STEP named "doit". If the user were to speak, "four five six", in response to the PROMPT element, the value "**456**" would be stored in the variable "pin". The DIGITS input can collect the digits 0 (zero) through 9 (nine), but not other numbers like **20** (twenty). To collect double-digit numbers (i.e., 20 (twenty) or 400 (four-hundred), the NUMBER input can be used as further described below.

The FORM input of INPUT attribute of the markup language (i.e., <INPUT TYPE="FORM" NAME="value" MEHOD="value" ACTION="value" TIMEOUT="value"/ >is used to collect input from the user, convert the input to text using the speech to text unit, and send the text to the markup language server. The FORM input includes a NAME attribute, a NEXT attribute, a METHOD attribute, an ACTION attribute and a TIMEOUT attribute. The value of the NAME attribute can be an identifier, and the value of the NEXT attribute can be a next step address (i.e., a URL, pointer or mamory address). The value of the METHOD attribute can be a get or a post, and the value of the ACTION attribute is a pointer to a script that processes the input on the server. The value of the TIMEOUT attribute can be a number represented in milliseconds.

The FORM input makes use of the speech to text unit to convert user input to text. The user input is then sent to the markup language server in a standard HTML <FORM> text format to be processed by a script on the server. If the user said "John Smith" then the text string "john smith" would be sent to the server using the pointer and address indicated by the ACTION attribute using the method indicated by the METHOD attribute in a <FORM>format.

The following is an example, also shown in FIG. **10**, of the user of the FORM input in a markup language document.

```
<STEP NAME="order form">
    <PROMPT> What you like to order? </PROMPT>
    <INPUT TYPE="form" NAME="order" NEXT="#next
    order" METHOD="post"
    ACTION="http://www.test.com/cgi-bin/post-query"
    TIMEOUT="200" />
</STEP>
```

In the example shown above, the FORM input is used to collect an order input from the user, store the user input converted to text in the variable named "order", go to the next step named "next order", post the text to the address "htt:p://www.test.com/cai-bin/post-auery", and use a timeout value of 200 milliseconds.

The GRAMMAR input of the of the INPUT attribute of the markup language (i.e., <INPUT TYPE="GRAMMAR" SRC="value" NEXT="value" [NEXTMETHOD="value"] [TIMEOUT="value"]/>, <INPUT TYPE="GRAMMAR" SRC="value" NEXT="value" [NEXTMETHOD="value"] [TIMEOUT="value"]>RENAME elements </INPUT>, or <INPUT TYPE="GRAMMAR" SRC="value" [TIMEOUT="value"] [NEXT="value" [NEXTMETHOD= "value"]]>RESPONSE elements</INPUT>) is used to specify an input grammar when interpreting the user's responses. The GRAMMAR input includes a SCR attribute, a NEXT attribute, a NEXTMETHOD attribute, and a TIMEOUT attribute. The value of the SCR attribute can be a grammar address (i.e., a URL), and the value of the NEXT attribute can be a next step address (i.e., a URL). The value of the NEXTMETHOD attribute can be a get and a post, and the value of the TIMEOUT attribute can be a number represented in milliseconds.

The following example illustrates the use of the GRAMMAR input in a markup language document.

```
<STEP NAME="init">
    <PROMPT> Say the month and year in which the
        credit card expires. </PROMPT>
    <INPUT TYPE="GRAMMAR"
        SRC="gram://.SomeGrammar/month/year"
        NEXT="#stepNineteen"/>
</STEP>
```

The above example illustrates the use of the GRAMMAR input to generate a predetermined grammar corresponding to a month and year from the user, store the interpreted values in variables named "month" and "year", and then go to the step named "stepNineteen".

The HIDDEN input of the INPUT attribute of the markup language (i.e., <INPUT TYPE="HIDDEN" NAME="value" VALUE="value"/>) is used to store a value in a variable. The HIDDEN input includes a NAME attribute and a VALUE attribute. The value of the NAME attribute can be an identifier, and the value of the VALUE attribute can be a literal value.

The following example illustrates the use of the HIDDEN input in a markup language document.

```
<STEP NAME="init">
    <PROMPT> Login sequence complete.
        Are you ready to place your order?
        </PROMPT>
    <INPUT TYPE="hidden" NAME="firstname"
        VALUE="Bill"/>
    <INPUT TYPE="hidden" NAME="lastname"
        VALUE="Clinton"/>
    <INPUT TYPE="hidden" NAME="favorite"
        VALUE="fries"/>
    <INPUT TYPE="optionlist">
        <OPTION NEXT="#order"> yes </OPTION>
        <OPTION NEXT="#wait"> not yet </OPTION>
    </INPUT>
</STEP>
```

In the example shown above, the HIDDEN input is used to create variables and assign values to those variables.

In this example, the user has completed the login sequence and certain information is stored in variables as soon as the user's identity has been established. This information could then be used later in the application without requiring another access into the database.

The MONEY input of the INPUT attribute of the markup language (i.e., <INPUT TYPE="MONEY" NAME="value" NEXT="value" [NEXTMETHOD="value"] [TIMEOUT= "value"]/>) is used to collect monetary amounts from the user. The MONEY input includes a NAME attribute, a NEXT attribute, a NEXTMETHOD attribute, and a TIMEOUT attribute. The value of the NAME attribute can be an identifier, and the value of the NEXT attribute can be a next step address (i.e., a URL). The value of the NEXTMEHOD attribute can be a get and a post, and the value of the TIMEOUT attribute can be a number represented in milliseconds.

The MONEY input makes use of an input grammar to interpret the user's response and store that response in a standard format. The input grammar is able to interpret various ways to express monetary amounts. The data is preferably stored in integer format, in terms of cents. "Five cents" is stored as "5", "five dollars" is stored as "500", and "a thousand" is stored as "100000". In the case where the units are ambiguous, the grammar assumes dollars, in which "a thousand" is stored as if the user had said "a thousand dollars".

The following example illustrates the use of the MONEY input in a markup language document.

```
<STEP NAME="init">
    <PROMPT> How much would you like to deposit?
        </PROMPT>
    <INPUT TYPE="money" NAME="dep"
    NEXT="#deposit"/>
</STEP>
```

The example shown above, the MONEY input is used to collect the amount of money that the user would like to deposit in his account, store that amount in a variable named "dep", and then go to the STEP named "deposit".

The NONE input of the INPUT attribute of the markup language (i.e., <INPUT TYPE="NONE" NEXT="value" [NEXTMETHOD="value"]/>) is used to specify the next location for the voice browser to go to continue execution when no response is collected from the user. The NONE input includes a NEXT attribute and a NEXTMETHOD attribute. The value of the NEXT attribute can be a next step address (i.e., a URL), and the value of the NEXTMETHOD attribute can be a get and a post.

The following example illustrates the use of the NONE input in a markup language.

```
<STEP NAME="init">
    <PROMPT> Welcome to the system. </PROMPT>
    <INPUT TYPE="none" NEXT="#mainmenu"/>
</STEP>
```

In the example shown above, the NONE input is used to jump to another STEP element in this dialog without waiting for any user response. In this example, the user would hear the phrase "Welcome to the system" followed immediately by the prompt of the main menu.

The NUMBER input of INPUT attribute of the markup language (i.e., <INPUT TYPE="NUMBER" NAME= "value" NEXT="value" [NEXTMETHOD="value"] [TIMEOUT="value"]/>) is used to collect numbers from the user. The NUMBER input includes a NAME attribute, a NEXT attribute, a NEXTMETHOD attribute, and a TIMEOUT attribute. The value of the NAME attribute can be an identifier, and the value of the NEXT attribute can be a next step address (i.e., a URL). The value of the NEXTMETHOD attribute can be a get and a post, and the value of the TIMEOUT attribute can be a number represented in milliseconds.

The following example illustrates the use of the NUMBER input in a markup language document or page.

```
<STEP NAME="init">
    <PROMPT> Please say your age now. </PROMPT>
    <INPUT TYPE="number" NAME="age" NEXT="#doit"/>
</STEP>
```

In the example shown above, the NUMBER input is used to collect numbers from the user, store the number in a variable named "age", and then go to the STEP element named "doit". If the user were to say, "eighteen", in response to the PROMPT element, the value "18" would be stored in the variable "age". The NUMBER input will collect numbers like 20 (i.e. twenty), but only one number per input. To collect a series of digits like "four five six" (i.e. "456"), the DIGITS input can be used as described above.

The OPTIONLIST input of INPUT attribute of the markup language (i.e., <INPUT TYPE="OPTIONLIST" [NAME="value"] [TIMEOUT="value"] [NEXT="value" [NEXTMETHOD="value"]]>OPTION elements </INPUT>) is used to specify a list of options from which the user can select. The OPTIONLIST input includes a NAME attribute, a NEXT attribute, a NEXTMETHOD attribute, and a TIMEOUT attribute. The value of the NAME attribute can be an identifier, and the value of the NEXT attribute can be a next step URL. The value of the NEXTMETHOD attribute can be a get and a post, and the value of the TIMEOUT attribute can be a number represented in milliseconds.

The OPTIONLIST input is used in conjunction with the OPTION element, which defines the specific user responses and the behavior associated with each OPTION element. The following example illustrates the use of the OPTIONLIST element in a markup language document.

```
<STEP NAME="init">
    <PROMPT> What would you like to drink? </PROMPT>
    <INPUT TYPE="optionlist">
        <OPTION NEXT="#coke"> coke </OPTION>
        <OPTION NEXT="#coke"> coca-cola </OPTION>
        <OPTION NEXT="#pepsi"> pepsi </OPTION>
        <OPTION NEXT="#rc"> r c </OPTION
    </INPUT>
</STEP>
```

In the example shown above, the voice browser will go to a different STEP element or state depending on which cola the user selects. If the user said "cokes" or "coca-cola", the voice browser would go to the STEP element named "coke".

The PHONE input of INPUT attribute of the markup language (i.e., <INPUT TYPE="PHONE" NAME="value" NEXT="value" [NEXTMETHOD="value"] [TIMEOUT= "value"]/>) is used to collect telephone numbers from the user. The PHONE input includes a NAME attribute, a NEXT attribute, a NEXTMETHOD attribute, and a TIMEOUT attribute. The value of the NAME attribute can be an identifier, and the value of the NEXT attribute can be a next step address (i.e., a URL). The value of the NEXTMETHOD attribute can be a get and a post, and the value of the TIMEOUT attribute can be a number represented in milliseconds.

The PHONE input makes use of an input grammar to interpret the user's response and store that response in a standard format. The phone number is interpreted as a string of digits and stored in a variable. If a user said "One, eight zero zero, seven five nine, eight eight eight eight", the response would be stored as "18007598888".

The following is an example of the use of the PHONE input in a markup language document.

```
<STEP NAME="phone">
    <PROMPT> What is your phone number? </PROMPT>
    <INPUT TYPE="phone" NAME="ph" NEXT="#fax"/>
</STEP>
```

In this example shown above, the PHONE input is used to collect a telephone number from the user, store the number in the variable named "ph", and go to the STEP named "fax".

The PROFILE input of INPUT attribute of the markup language (i.e., <INPUT TYPE="PROFILE" NAME= "value" PROFNAME="value" [SUBTYPE="value"]/>) is used to collect the user's profile information (i.e, first name, last name, mailing address, email address, and notification address). The user profile information is stored in the database **244** of the system.

The PROFILE input includes a NAME attribute, a PROFNAME attribute, and a SUBTYPE attribute. The value of the NAME attribute can be an identifier, the value of the PROFNAME attribute can be a profile element name (string), and the value of the SUBTYPE attribute can be profile element subtype (string).

The following example illustrates the use of the PROFILE input in a markup language document.

```
<STEP NAME="getinfo">
    <INPUT TYPE="profile" NAME="firstname"
        PROFNAME="N" SUBTYPE="first"/>
    <PROMPT> Hello, <VALUE NAME="firstname"/>
        Please say your pin. </PROMPT>
    <INPUT TYPE="digits" NAME="pin"
    NEXT="#verify"/>
</STEP>
```

In the example above, the PROFILE input is used to retrieve the user's first name and store the string in a variable named "firstname". The string containing the name is then inserted into the PROMPT element using a VALUE element as further described below. When using the PROFILE input, more than one INPUT element can be included in the same STEP element because the PROFILE input is not an interactive INPUT element. Each STEP element contains only one INPUT element that accepts a response from the user.

The following table lists the valid combinations of profile names and their associated subtypes

Profile Name	Subtype	Description
ADR	POSTAL	postal address
	PARCEL	parcel address
	HOME	home address
	WORK	work address
	DOM	domestic address
	INTL	international address
BDAY	none	birthday
EMAIL	none	primary email address
	NOTIFICATION	notification email address
FN	none	formatted name
GEO	none	geographic location (longitude:lattitude)
KEY	none	public encryption key
LABEL	none	mailing label
MAILER	none	email program used
N	FIRST	first name
	LAST	last name
	MIDDLE	middle name
	PREFIX	prefix (e.g. Mr., Mrs., Dr.)
	SUFFIX	suffix (e.g. Jr., D.D.S, M.D.)
ORG	none	organization
ROLE	none	job role or position
TEL	HOME	home telephone number
	WORK	work telephone number
	MSG	voicemail telephone number

-continued

Profile Name	Subtype	Description
	VOICE	voice call telephone number
	FAX	fax call telephone number
	CELL	cellular telephone number
	PREF	preferred telephone number
TITLE	none	job title
TZ	none	time zone
UID	none	globally unique id
URL	none	URL of home page
VERSION	none	version of Vcard

The notification address shown above can be used to send a user urgent or timely information (i.e., sending information to a pager). The format of the notification address is preferably of an email address provided by the user when his or her subscription is activated. The user's notification address would be stored a variable named "n_{13} addr". The application could then use this email address to send a message to the user. To retrieve the notification address from the voice browser, the PROFILE input can be used in a markup language document in the following manner:

```
<INPUT TYPE="profile" NAME="n_addr"
    PROFNAME="email" SUBTYPE="notification"/>
```

The RECORD input of the INPUT attribute of the markup language (i.e., <INPUT TYPE="RECORD" TIMEOUT= "value" STORAGE="value" [FORMAT="value"] [NAME= "value"] NEXT="value" [NEXTMETHOD="value"]/>) is used to record an audio sample and to store that audio sample in a specified location. The RECORD input includes a TIMEOUT attribute, a FORMAT attribute, a NAME attribute, a STORAGE attribute, a NEXT attribute, and a NEXTMETHOD attribute. The value of the TIMEOUT attribute can be the maximum record time represented in milliseconds, the value of the FORMAT attribute can be a recorded audio format (audio/wav), the value of the NAME attribute can be an identifier, the value of the STORAGE attribute can be a file and a request, the value of the NEXT attribute can be a next step address (i.e., a URL), and the value of the NEXTMETHOD attribute can be a get, post and put.

The following two examples illustrate the RECORD input in a markup language document.

```
<STEP NAME="init">
    <PROMPT> Please say your first and last name.
        </PROMPT>
    <INPUT TYPE="record" TIMEOUT="7000"
        NAME="theName" STORAGE="REQUEST"
        NEXT="http://wavhost/acceptwav.asp"
        NEXTMETHOD="POST"/>
</STEP>
```

In the example shown above, the RECORD input is used to record a seven second audio sample, and then "POSTS" that sample to the remote machine named "wavhost". The response to the "POST" has to be a dialog which continues the execution of the application.

```
<STEP NAME="init">
    <PROMPT> Please say your first and last name.
        </PROMPT>
    <INPUT TYPE="record" TIMEOUT="7000"
        NAME="theName" STORAGE="FILE"
        NEXT="#reccomplete" NEXTMETHOD="GET"/>
</STEP>
```

In the example shown above, the RECORD input is used to record another seven second audio sample. However, the sample is stored in a file, instead of sent in the HTTP request as it was in the previous example. The name of the file is chosen by the voice browser automatically and is stored in a variable named "theName". After storing the audio sample in the file, the voice browser will continue execution at the URL specified by the NEXT attribute. In contrast to the previous example, the value of the variable "theName" will be the name of the audio file. In the earlier example (where the audio sample was transmitted via the HTTP request), the value of the variable "theName" would be null.

The TIME input type of the INPUT attriute of the markup language (i.e., <INPUT TYPE="TIME" NAME="value" NEXT="value" [NEXTMETHOD="value"] [TIMEOUT= "value"]/>) is used to collect a time of day from the user. The TIME input includes a NAME attribute, a NEXT attribute, a NEXTMETHOD attribute, and a TIMEOUT attribute. The value of the NAME attribute can be an identifier, and the value of the NEXT attribute can be a next step address (i.e., a URL). The value of the NEXTMETHOD attribute can be a get and a post, and the value of the TIMEOUT attribute can be a number represented in milliseconds.

The TIME input makes use of an input grammar to interpret the user's response and to store that response in a standard format. This grammar will interpret responses of various forms, including both 12-hour and 24-hour conventions. "Four oh three PM" becomes "403P". Note that "P" is appended to the time. Likewise, "Ten fifteen in the morning" becomes "1015A". "Noon" is stored as "1200P"f, and "Midnight" is stored as "1200A". Military time, such as, "Thirteen hundred hours" becomes "100P". If the user does not specify the morning or evening, no indication is stored in the variable (i.e.,"Four o'clock" is stored as "1400").

The following example illustrates the TIME input in a markup language document.

```
<STEP NAME="init">
    <PROMPT> What time would you like your wakeup
        call? </PROMPT>
    <INPUT TYPE="time" NAME="wakeup"
NEXT="#record"/>
    </STEP>
```

In the example shown above, the TIME input is used to collect a time of day from the user, store that data in the variable named "wakeup", and then go to the STEP element named "record".

The YORN input of the INPUT attribute of the markup language (i.e, <INPUT TYPE="YORN" NAME="value" [TIMEOUT="value"] NEXT="value" [NEXTMETHOD= "value"]/>, or <INPUT TYPE="YORN" [NAME="value"] [TIMEOUT="value"] [NEXT="value" (NEXTMETHOD= "value"]]>CASE elements </INPUT>) is used to collect "yes" or "no" responses from the user. The YORN input includes a NAME attribute, a NEXT attribute, a NEXTMETHOD attribute, and a TIMEOUT attribute. The value of

the NAME attribute can be an identifier, and the value of the NEXT attribute can be a next step address (i.e., a URL). The value of the NEXTMETHOD attribute can be a get and a post, and the value of the TIMEOUT attribute can be a number represented in milliseconds.

The YORN input maps a variety of affirmative and negative responses to the values "Y" and "N". The YORN input stores the value "Y" for affirmative responses and the value "N" for negative responses. Affirmative and negative responses are determined using an input grammar that maps various user responses to the appropriate result.

The following example illustrates the user of the YORN input in a markup language document.

```
<STEP NAME="ask">
    <PROMPT> Fire the missles now? </PROMPT>
    <INPUT TYPE="YORN" NAME="fire"
    NEXT="#confirm"/>
</STEP>
```

In the example shown above, the YORN input is used to collect a "yes" or "no" response from the user, store that response into a variable named "fire", and then go to the STEP named "confirms".

The OPTION element of the markup language (i.e. <OPTION [NEXT="value" [NEXTMETHOD="value"]] [VALUE="value"]> text </OPTION>) is used to define the type of response expected from the user in a STEP element or state. The OPTION input includes a VALUE attribute, a NEXT attribute, and a NEXTMETHOD attribute. The value of the VALUE attribute can be a literal value, the value of the NEXT attribute can be a next step address (i.e., a URL), and the value of the NEXTMETHOD attribute can be a get and a post. The OPTION element can exist within the INPUT element, and then only when using the OPTION-LIST input.

The following two examples illustrate the use of the OPTION element in a markup language document.

```
<INPUT NAME="choice" TYPE="optionlist">
    <OPTION NEXT="#doit" VALUE="1"> one </OPTION>
    <OPTION NEXT="#doit" VALUE="2"> two </OPTION>
</INPUT>
```

The example shown above illustrates the use of the OPTION element within the INPUT element. In this example, the first OPTION element would be executed when the user responded with "one", and the second OPTION would be executed when the user responded with "two". If the user said "one", the value of the variable named "choice" would be "1", because of the use of the VALUE attribute. Because the NEXT attributes for both of the OPTION element in this OPTIONLIST element are the same, the voice browser would proceed to the STEP element named "doit" when either "one"**0** or "two" was recognized.

```
<INPUT TUPE="optionlist">
    <OPTION
    NEXT="http://localhost/vml/weather.asp">
        weather </OPTION>
    <OPTION NEXT="http://localhost/vml/news.asp">
        news </OPTION>
    <OPTION
```

-continued

```
    NEXT="http://localhost/vml/traffic.asp">
        traffic </OPTION>
</INPUT>
```

The example shown above illustrates the use of the OPTION element to select one of three applications. Note that the URLs used in the NEXT attributes are full HTTP URLS, and that unlike the previous example, each OPTION element has a unique NEXT attribute.

The OPTIONS element of the markup language (i.e., <OPTIONS/>) describes the type of input expected within a given STEP element. The OPTIONS element can be used in HELP elements to present the user with a complete list of valid responses. The OPTIONS element can be used anywhere that text is read to the user. The OPTIONS element can be contained by a PROMPT, EMP, PROS, HELP, ERROR, or ACK element.

The following example illustrates the use of the OPTIONS element in a markup language document.

```
<CLASS NAME="helpful">
    <HELP> Your choices are: <OPTIONS/> </HELP>
</CLASS>
```

The example shown above illustrates how the OPTIONS element can be used to construct a "helpful" class. Any STEP elements that directly or indirectly name "helpful" as a PARENT element respond to a helpful request (i.e., "help") by speaking the message, in which the OPTIONS element expands to a description of what can be said by the user at this point in the dialog.

The ACK element of the markup language (i.e., <ACK [CONFIRM="value"] [BACKGROUND="value"] [REPROMPT="value"] > text </ACK>) is used to acknowledge the transition between Step elements, usually as a result of a user response. The ACK element includes a CONFIRM attribute, a BACKGROUND attribute, and a REPROMPT attribute. The value of the BACKGROUND and REPROMPT attributes can be a "Y" and "N", and the CONFIRM attribute can be a YORN element as described above. The ACK element can be contained within a STEP element or a CLASS element as further described below.

The following is an example of a markup language document containing the Ack element.

```
<STEP NAME="card_type">
    <PROMPT>
        What type of credit card do you have?
    </PROMPT>
    <INPUT NAME="type" TYPE="optionlist">
        <OPTION NEXT="#exp">visa </OPTION>
        <OPTION NEXT="#exp">mastercard </OPTION>
        <OPTION NEXT="#exp">discover </OPTION>
    </INPUT>
    <ACK CONFIRM="YORN" REPROMPT="Y">
        I thought you said <VALUE NAME="type"/>
        <BREAK/> Is that correct?
    </ACK>
</STEP>
```

In the example above, the ACK element is used to confirm the user's choice of credit card. When this element is interpreted by the voice browser, the PROMPT element is read to the user using text-to-speech unit **252**. The system

waits until the user responds with "visa", "mastercard", or "discover" and then asks the user to confirm that the type of card was recognized correctly. If the user answers "yes" to the ACK element, the voice browser will proceed to the STEP element named "exp". If the user answers "no" to the ACK element, the text of the PROMPT element will be read again, and the user will be allowed to make his or her choice again. The voice browser then re-enters or executes the STEP element again.

The AUDIO element of the markup language (i.e., <AUDIO SRC="value"/>) specifies an audio file that should be played. The AUDIO element includes a SRC attribute. The value of the SRC attribute can be an audio file URL. The AUDIO element can be contained within a PROMPT, EMP, PROS, HELP, ERROR, CANCEL, or ACK element.

The following markup language contains the AUDIO element.

```
<PROMPT>
    At the tone, the time will be 11:59 p m
    <AUDIO SRC="http://localhost/sounds/beep.wav"/>
</PROMPT>
```

In the example above, the AUDIO element is included in a PROMPT element. When interpreted by the voice browser, a prompt (i.e., "At the tone, the time will be 11:59 pm.") will be played to the user, and the WAV file "beep.wav" will be played to the user as specified by the AUDIO element.

The BREAK element of the markup language (i.e., <BREAK [MSECS="value"|SIZE="value"]/>) is used to insert a pause into content or information to be played to the user. The BREAK element includes a MSEC attribute and a SIZE attribute. The value of the MSEC attribute can include a number represented in milliseconds, and the value of the SIZE attribute can be none, small, medium, and large.

The BREAK element can be used when text or audio sample is to be played to the user. The BREAK element can be contained within a PROMPT, EMP, PROS, HELP, ERROR, CANCEL, or ACK element. The following markup language contains the BREAK element.

```
<PROMPT>
    Welcome to Earth. <BREAK MSECS="250"/>
    How may I help you?
</PROMPT>
```

In the example above, the BREAK element is used with a MSECS attribute, inside a PROMPT element. When interpreted by the voice browser, a prompt (i.e, "Welcome to Earth.") is read to the user. The system will then pause for 250 milliseconds, and play "How may I help you?".

Alternatively, the SIZE attribute (i.e., "small", "medium", and "large") of the BREAK element can be used to control the duration of the pause instead of specifying the number of milliseconds as shown below.

```
<PROMPT>
    Welcome to Earth. <BREAK SIZE="medium"/>
    How may I help you?
</PROMPT>
```

The OR element of the markup language (i.e., <OR/>) is used to define alternate recognition results in an OPTION element. The OR element is interpreted as a logical OR, and is used to associate multiple recognition results with a single NEXT attribute.

The following example illustrates the use of the OR element in a markup language document.

```
<INPUT TYPE="optionlist">
<OPTION NEXT="#coke_chosen">
        coke <OR/> coca-cola
    </OPTION>
    <OPTION NEXT="#pepsi_chosen"> pepsi </OPTION>
</INPUT>
```

The example shown above illustrates the use of the OR element within an OPTION element. As shown above, the user may respond with either "coke" or "coca-cola", and the voice browser will proceed to the STEP named "coke_chosen".

The CANCEL element of the markup language (i.e., <CANCEL NEXT="value" [NEXTMETHOD="value"]/>or <CANCEL NEXT="value" [NEXTMETHOD="value"]> text </CANCEL>) is used to define the behavior of the application in response to a user's request to cancel the current PROMPT element. The CANCEL element includes a NEXT attribute and a NEXTMETHOD attribute. The value the NEXT attribute can be a next step address (i.e., a URL), and the value of the NEXTMETHOD attribute can be a get and a post.

The CANCEL element can be invoked through a variety of phrases. For example, the user may say only the word "cancel", or the user may say "I would like to cancel, please."The CANCEL element can be contained within a STEP element or a CLASS element. When the voice browser detects "cancel" from the user, the voice browser responds based upon the use of the CANCEL element in markup language document. If no CANCEL element is associated with a given STEP element, the current prompt will be interrupted (if it is playing) and will stay in the same application state and then process any interactive inputs.

The following example illustrates a markup language containing the CANCEL element.

```
<STEP NAME="report">
    <CANCEL NEXT="#traffic_menu"/>
    <PROMPT> Traffic conditions for Chicago,
    Illinois,
    Monday, May 18.  Heavy
    congestion on ... </PROMPT>
    INPUT TYPE="optionlist">
        <OPTION NEXT="#report"> repeat </OPTION>
        <OPTION NEXT="#choose"> new city </OPTION>
    </INPUT>
</STEP>
```

The example above illustrates the use of the CANCEL element to specify that when the user says "cancel", the voice browser proceeds to the STEP element named "traffic_menu", instead of the default behavior, which would be to stop the PROMPT element from playing and wait for a user response. The user can also interrupt the PROMPT element by speaking a valid OPTION element. In this example, the user could interrupt the PROMPT element and get the traffic conditions for a different city by saying "new city".

The CASE element of the markup language (i.e., <CASE VALUE="value" NEXT="value" [NEXTMETHOD=

"value"]/>) is used to define the flow of control of the application, based on the values of internal markup language variables. The CASE input includes a VALUE attribute, a NEXT attribute, and a NEXTMETHOD attribute. The value of the VALUE attribute can be a literal value, the value of the NEXT attribute can be a next step address (i.e. a URL), and the value of the NEXTMETHOD attribute can be a get and a post. The CASE element can be contained by a SWITCH element or an INPUT element, when using an input type of the INPUT element that collects a single value (i.e., DATE, DIGITS, MONEY, PHONE, TIME, YORN).

The following example illustrates a markup language containing a CASE element.

```
<SWITCH FILED="pizza">
<CASE VALUE="pepperoni" NEXT="#p_pizza"/>
<CASE VALUE="sausage" NEXT="#s_pizza"/>
<CASE VALUE="veggie" NEXT="#v_pizza"/>
</SWITCH>
```

In the example above, the markup language shows the use of the CASE element within the SWITCH element. In this example, the CASE elements are used to direct the voice browser to different URLs based on the value of the markup language variable "pizza".

The CLASS element of the markup language (i.e., <CLASS NAME="value" [PARENT="value"] [BARGEIN="value"][COST="value"]> text </CLASS>) is used to define a set of elements that are to be reused within the content of a dialog. For example, application developers can define a set of elements once, and then use them several times. The CLASS input includes a NAME attribute, a PARENT attribute, a BARGEIN attribute, and a COST attribute. The value of the NAME and the PARENT attribute can be an identifier. The value of the BARGEIN attribute can be "Y" and "N", and the value of the COST attribute can be an integer number.

The CLASS element can be used to define the default behavior of an ERROR element, a HELP element, and a CANCEL element, within a given DIALOG element. The CLASS element can be contained by a DIALOG element. The following example shows a markup language document containing the CLASS element.

```
<CLASS NAME="simple">
  <HELP> Your choices are <OPTIONS/> </HELP>
  <ERROR> I did not understand what you said.
    Valid responses are <OPTIONS/> </ERROR>
</CLASS>
<STEP NAME="beverage" PARENT="simple">
  <PROMPT> Please choose a drink. </PROMPT>
  <INPUT NAME="drink" TYPE="optionlist">
    <OPTION NEXT="#food"> coke </OPTION>
    <OPTION NEXT="#food"> pepsi </OPTION>
  </INPUT>
</STEP>
<STEP NAME="food" PARENT="simple">
  <PROMPT> Please choose a meal. </PROMPT>
  <INPUT NAME="meal" TYPE="optionlist">
<OPTION NEXT="#deliver"> pizza </OPTION>
<OPTION NEXT="#deliver"> tacos </OPTION>
</INPUT>
</STEP>
```

In the example above, the markup language document illustrates the use of the CLASS element to define a HELP element and an ERROR element that will be used in-several steps within this DIALOG element. The markup language also illustrates the use of the PARENT attribute in the STEP element to refer to the CLASS element, and therefore inherit the behaviors defined within it. When interpreted by the voice browser, the STEP element will behave as if the HELP and ERROR elements that are defined in the CLASS element were defined explicitly in the steps themselves

The EMP element of the markup language (i.e., <EMP [LEVEL="value"]> text </EMP>) is used to identify content within text that will be read to the user where emphasis is to be applied. The EMP element includes a LEVEL attribute. The value of the LEVEL element can be none, reduced, moderate, and strong. The EMP element can be contained within a PROMPT, EMP, PROS, HELP, ERROR, CANCEL, or ACK element. The following example of a markup language document contains the EMP element.

```
<PROMPT>
    This example is
    <EMP LEVEL="strong"> really </EMP>
    simple.
</PROMPT>
```

In the above example, the EMP element is used to apply "strong" emphasis to the word "really" in the PROMPT element. The actual effect on the speech output is determined by the text-to-speech (TTS) software of the system. To achieve a specific emphatic effect, the PROS element, as further described below, can be used instead of the EMP element.

The ERROR element of the markup language (i.e., <ERROR [TYPE="value"] [ORDINAL="value"] [REPROMPT="value"] [NEXT="value" [NEXTMETHOD="value"]]> text </ERROR>) is used to define the behavior of the application in response to an error. The ERROR element includes a TYPE attribute, an ORDINAL attribute, a REPROMPT attribute, a NEXT attribute, and a NEXTMETHOD attribute. The value of the TYPE attribute can be all, nomatch, nospeech, toolittle, toomuch, noauth, and badnext. The value of the ORDINAL attribute can be an integer number, the value of the REPROMPT attribute can be GYP or "N", the value of the NEXT attribute can be a next step address (i.e., a URL), and the value of the NEXTMETHOD attribute can be a get and a post.

If the application developer does not define the behavior of an ERROR element for a given STEP element, the default behavior will be used. The default behavior for the ERROR element is to play the phrase "An error has occurred.", remain in the current STEP element, replay the PROMPT element, and wait for the user to respond. The ERROR element can be contained within a STEP or a CLASS element.

The following example illustrates the use of the ERROR element in a markup language document.

```
1  <STEP NAME="errors">
2    <ERROR TYPE="nomatch"> First error message.
3      I did not understand what you said. </HELP>
4    <ERROR TYPE="nomatch" ORDINAL="2">
5      Second error message.
6      I did not understand what you said </HELP>
7    <PROMPT> This step tests error messages.
8      Say 'oops' twice. Then say 'done' to
9      choose another test. </PROMPT>
```

-continued

```
10      <INPUT TYPE="OPTIONLIST">
11          <OPTION NEXT="#end"> done </OPTION>
12      </INPUT>
13  </STEP>
```

In the example above, the ERROR element is used to define the application's behavior in response to an error. On line **2**, the error message is defined to be used the first time an error of type "nomatch" occurs in this STEP element. On line **4**, the error message is to be used the second and all subsequent times an error of type "nomatch" occurs in this STEP.

The ORDINAL attribute of the ERROR element of the markup language determines which message will be used in the case of repeated errors within the same STEP element. The voice browser can choose an error message based on the following algorithm. If the error has occurred three times, the voice browser will look for an ERROR element with an ORDINAL attribute of "3". If no such ERROR element has been defined, the voice browser will -look for an ERROR element with an ORDINAL attribute of "2", and then "1", and then an ERROR element with no ORDINAL attribute defined. Thus, if the ERROR element is defined with the ORDINAL attribute of "6" in the STEP element shown above, and the same error occurred six times in a row, the user would hear the first error message one time, then the second error message four times, and finally the error message with ORDINAL attribute of "6".

The HELP element of the markup language (i.e.,<HELP [ORDINAL="value"] [REPROMPT="value"] [NEXT= "value" [NEXTMETHOD="value"]]> text </HELP>) is used to define the behavior of the application when the user asks for help. The HELP element includes an ORDINAL attribute, a REPROMPT attribute, a NEXT attribute, and a NEXTMETHOD attribute. The value of the ORDINAL attribute can be an integer number, and the value of the REPROMPT attribute can be a "Y" and "N". The value of the NEXT attribute can be a next step address (i.e., a URL), and the value of the NEXTMETHOD attribute can be a get and a post.

The HELP element, like CANCEL the element, can be detected through a variety of phrases. The user may say only the word "help", or the user may say "I would like help, please." In either case, the HELP element will be interpreted. The HELP element can be contained within a STEP element or a CLASS element.

When the voice browser detects "help" from the user, the voice browser responds based upon the use of the HELP element in markup language document. If no HELP element is associated with a given STEP, the current prompt will be interrupted (if it is playing), the user will hear "No help is available.", and will stay in the same application state and process any interactive inputs.

The following example illustrates the use of the HELP element in a markup language document.

```
1   <STEP NAME="helps">
2     <HELP REPROMPT="Y"> First help message.
3       You should hear the prompt again. </HELP>
4     <HELP ORDINAL="2"> Second help message.
5       You should not hear the prompt now. </HELP>
6     <PROMPT> This step tests help prompts.
7       Say 'help' twice. Then say 'done' to
8       choose another test. </PROMPT>
9     <INPUT TYPE="OPTIONLIST">
10      <OPTION NEXT="#end"> done </OPTION>
11    </INPUT>
12  </STEP>
```

In the example above, the HELP element is used to define the application's behavior in response to the user input "help". On line **2**, the help message is defined to be used the first time the user says "help". On line **4**, the help message is defined to be used the second and all subsequent times the user says "help". It should also be noted that through the use of the REPROMPT attribute, the prompt will be repeated after the first help message, but it will not be repeated after the second help message.

The ORDINAL attribute of the HELP element of the markup language determines which message will be used in the case of repeated utterances of "help" within the same STEP element. The voice browser will choose a help message based on the following algorithm. If the user has said "help" three times, the voice browser will look for a HELP element with an ORDINAL attribute of "3". If no such HELP element has been defined, the voice browser will look for a HELP element with an ORDINAL attribute of "2", and then "1", and then a HELP element with no ORDINAL attribute defined. Thus, if a HELP element is defined with ORDINAL attribute of "6" in the STEP element shown above, and the user said "help" six times in a row, the user would hear the first help message one time, then the second help message four times, and finally the help message with ORDINAL attribute of "6".

The PROS element of the markup language (i.e., <PROS [RATE="value"] [VOL="value"] [PITCH="value"] [RANGE="value"]> text </PROS>) is used to control the prosody of the content presented to the user via PROMPT, HELP, ERROR, CANCEL, and ACK elements. Prosody affects certain qualities of the text-to-speech presentation, including rate of speech, pitch, range, and volume. The PROS element includes a RATE attribute, a VOL attribute, a PITCH attribute, and a RANGE attribute. The value of the RATE attribute can be an integer number representing words per minute, and the value of the VOL attribute can be an integer number representing volume of speech. The value of the PITCH attribute can be an integer number representing pitch in hertz, and the value of the RANGE attribute can be an integer number representing range in hertz. The PROS element can be contained within a PROMPT, EMP, PROS, HELP, ERROR, CANCEL, or ACK element.

The following example illustrates the use of the pros element.

```
<PROMPT> Let me tell you a secret:
<PROS VOL="0.5"> I ate the apple. </PROS>
</PROMPT>
```

In the example shown above, the phrase "I ate the apple" is spoken with one half of the normal volume.

The RENAME element of the markup language (i.e., <RENAME RECNAME="value" VARNAME="value"/>) is used to rename recognition slots in grammars, such that the resulting variable name can be different from the name of the recognition slot defined in the grammar. The rename element includes a VARNAME attribute and a RECNAME attribute. The value of the VARNAME and the RECNAME attributes can be identifiers. The RENAME element can exist only within the INPUT element, and then only when using the GRAMMAR input type.

The following example illustrates the use of the RENAME element in a markup language document.

```
<INPUT TYPE="GRAMMAR"
  SRC="http://www.foo.com/mygram.grm"
  NEXT="http://www.fancyquotes.com/vmlstocks.asp">
  <RENAME VARNAME="sym" RECNAME="symbol">
  <RENAME VARNAME="detail" RECNAME="quotetype">
</INPUT>
```

In the example shown above, the RENAME element is used to account for differences in the variable names collected from a grammar and those expected by another script. In particular, a grammar from foo.com is used to provide input to an application hosted by fancyquotes.com. Because, in this example, the grammar and script have been developed independently, the RENAME element is used to help connect the grammar and the stock-quoting application.

The RESPONSE element of the markup language (i.e., <RESPONSE FIELDS="value" [NEXT="value" [NEXTMETHOD="value"]]/> or <RESPONSE FIELDS= "value" [NEXT="value" [NEXTMETHOD="value"]]> SWITCH elements </RESPONSE>) is used to define the behavior of an application in response to different combinations of recognition slots. The response element includes a FIELDS attribute, a NEXT attribute, and a NEXTMETHOD attribute. The value of the FIELDS attribute can be a list of identifiers, the value of the NEXT attribute can be a next step address (i.e., a URL), and the value of the NEXTMETHOD attribute can be a get and a post.

The RESPONSE element enables application developers to define a different NEXT attribute depending on which of the grammar's slots were filled. The RESPONSE element can exist within an INPUT element, and then only when using an input type of grammar.

The following example illustrates the RESPONSE element in a markup language document.

```
<INPUT TYPE="GRAMMAR"
  SRC="gram://.Banking/action/amt/fromacct/toacct"
  NEXT="#notenoughfields">
  <RESPONSE FIELDS="action,amt,fromacct,toacct"
    NEXT=#doit"/>
  <RESPONSE FIELDS="action,amt,fromacct"
    NEXT="#asktoacct"/>
  <RESPONSE FIELDS="action,amt,toacct"
    NEXT="#askfromacct"/>
  <RESPONSE FIELDS="action,amt" NEXT="#askaccts"/>
  <RESPONSE FIELDS="action" NEXT="#askamtaccts"/>
</INPUT>
```

The example shown above illustrates the use of the RESPONSE element where the user specifies less than all the possible variables available in the grammar. Using the RESPONSE element, the application can arrange to collect the information not already filled in by prior steps. In particular, this example transfers to the "askaccts" STEP element if neither the source nor destination account is specified (i.e., the user said "transfer 500 dollars"), but it transfers to the "askfromacct" STEP element if the user said what account to transfer to, but did not specify a source account (i.e., if the user had said "transfer 100 dollars to savings"). The next URL of the INPUT element is used when the user's response does not match any of the defined responses.

The SWITCH element of the markup language (i.e., <SWITCH FIELD="value">vml </SWITCH>) is used to define the application behavior dependant on the value of a specified recognition slot. The switch element includes a FIELD attribute. The value of the FIELD attribute can be an identifier. The SWITCH element is used in conjunction with the CASE element. The SWITCH element can exist within the INPUT element, and then only when using the grammar input type.

The following example illustrates the use of the SWITCH element in a markup language document.

```
  <INPUT TYPE="GRAMMAR"
SRC="gram://.Banking/action/amount/fromacct/toacct">
<SWITCH FIELD="action">
<CASE VALUE="transfer" NEXT="#transfer" />
<CASE VALUE="balance" NEXT="#balance" />
<CASE VALUE="activity">
    <SWITCH FIELD="fromacct">
      <CASE VALUE="checking" NEXT="#chxact" />
      <CASE VALUE="savings" NEXT="#savact" />
    </SWITCH>
</CASE>
</SWITCH>
</INPUT>
```

In the example shown above, the SWITCH element is used to determine the next STEP element to execute in response to a banking request. In this example, the grammar may fill in some or all of the variables (i.e., "action", "amount", "fromacct", and "toacct"). If the user asks for a transfer or balance action, the next STEP element to execute is the transfer or balance step. If the user asks for a report of account activity, a second SWITCH element determines the next STEP element based on the account type for which a report is being requested (assumed to be available in the "ffromacct" variable).

The VALUE element of the markup language (i.e., <VALUE NAME="value"/>) is used to present the value of a variable to the user via the text-to-speech unit. The VALUE element includes a FIELD attribute. The value of the FIELD attribute can be an identifier. The VALUE element can be used anywhere that text is read to the user. The VALUE element can be contained by a PROMPT, EMP, PROS, HELP, ERROR, CANCEL, or ACK element.

The following example illustrates the use of the value element in a markup language document.

```
<STEP NAME="thanks">
  <PROMPT> Thanks for your responses. I'll record
    that <VALUE NAME="first"/> is your favorite
    and that <VALUE NAME="second"/> is your
    second choice.
    </PROMPT>
    <INPUT TYPE="NONE" NEXT="/recordresults.asp" />
</STEP>
```

The example shown above illustrates the use of the VALUE element to read the user's selections back to the user. As shown above, the value of the variable named "first" would be inserted into the PROMPT element, and the value of the variable named "second" would be inserted into the PROMPT element.

The COST attribute of the STEP element of the markup language enables is used to charge a user for various services. The COST attribute can be used in the definition of one of more STEP or CLASS elements. The value of the COST attribute is the integer number of credits the user is to be charged for viewing the content. For example, to charge

10 credits for listening to a particular step element a provider might write the following markup language:

<STEP NAME="premiumContent" COST="10">

. . . premium content goes here . . .

</STEP>

If a content provider wishes to maintain a record of subscriber charges, the content provider need only request identifying data for the user using the PROFILE input type as in:

<INPUT TYPE="PROFILE" PROFNAME="UID" NAME="subID"/>

Using the resulting value and examining the SUB_CHARGE query-string parameter at each page request, the content provider can maintain records on a per-subscriber basis.

The following text describes a weather application **500** that can be executed by the system **200** of FIG. **3**. FIG. **8** shows an exemplary state diagram of the weather application containing states that prompt the user for input in order to access the weather database. After speaking the current or forecast weather information, the application expects the user to say a city name or the word "exit" to return to the main welcome prompt. The user can select to hear the forecast after the current weather conditions prompt. It will be recognized that the application could be designed to address errors, help and cancel requests properly.

The markup language set forth below is a static version of the weather application. The initial state or welcome prompt is within the first step, init (lines **11–20**). The user can respond with a choice of "weather", "market", "news" or "exit". Once the application detects the user's response of "weather", the next step, weather (lines **21–29**), begins. The prompt queries the user for a city name. Valid choices are "London", "New York", and "Chicago".

The steps called london_current, london_forecast, newyork_current, newyork_forecast, chicago_current, and chicago_forecast provide weather information prompts for each city. It is noted that Market and news steps are just placeholders in the example (lines **111** and **115**).

```
<?XML VERSION="1.0"?>
<!- -
------------------------------------------------------------------------------
_- ->
<!- - (c) 1998 Motorola Inc.
- ->
<!- - weather.vml
- ->
<!- -
------------------------------------------------------------------------------
_- ->
<DIALOG>
<CLASS NAME="help_top">
    <HELP>You are at the top level menu. For weather
information,
say weather. </HELP>
</CLASS>
<STEP NAME="init" PARENT="help_top">
    <PROMPT>Welcome to Genie.<BREAK SIZE="large"/>
    How may I help you? </PROMPT>
    <INPUT TYPE="OPTIONLIST">
        <OPTION NEXT="#weather">weather</OPTION>
        <OPTION NEXT="#market">market</OPTION>
        <OPTION NEXT="#news">news</OPTION>
        <OPTION NEXT="#bye">exit</OPTION>
    </INPUT>
</STEP>
<STEP NAME="weather" PARENT="help_top">
    <PROMPT>What city? </PROMPT>
    <INPUT TYPE="OPTIONLIST">
        <OPTION
NEXT="#london_current">london</OPTION>
        <OPTION NEXT="#newyork_current">new
york</OPTION>
        <OPTION
NEXT="#chicago_current ">chicago</OPTION>
        <OPTION NEXT="#init">exit</OPTION>
    </INPUT>
</STEP>
<CLASS NAME="help_generic">
    <HELP>Your choices are <OPTIONS/>.</HELP>
</CLASS>
<STEP NAME="london_current" PARENT="help_generic">
    <PROMPT>It is currently 46 degrees in London,
with rain.
    <BREAK SIZE="large"/>
    To hear the 3 day forecast for London, say
forecast, or say
    another city name, such as Chicago or New
York.</PROMPT>
    <INPUT TYPE="OPTIONLIST">
        <OPTION
NEXT="#london_forecast">forecast</OPTION>
        <OPTION
NEXT="·london_current">london</OPTION>
        <OPTION NEXT="#newyork_current">new
york</OPTION>
        <OPTION
NEXT="#chicago_current">chicago</OPTION>
        <OPTION NEXT="#init">exit</OPTION>
    </INPUT>
</STEP>
<STEP NAME="london_forecast" PARENT="help_generic">
    <PROMPT>London forecast for
    Tuesday. Showers. High of 50. Low of 44.
    Wednesday. Partly cloudy. High of 39. Low of 35.
    <BREAK SIZE="large"/>
    Choose a city, or say exit to return to the main
menu. </PROMPT>
    <INPUT TYPE="OPTIONLIST">
        <OPTION
NEXT="#london_current">london</OPTION>
        <OPTION NEXT="newyork_current">new
york</OPTION>
        <OPTION
NEXT="#chicago_current">chicago</OPTION>
        <OPTION NEXT="#init">exit</OPTION>
    </INPUT>
</STEP>
<STEP NAME="chicago_current" PARENT="help_generic">
    <PROMPT>It is currently 31 degrees in Chicago,
with snow.
    <BREAK SIZE="large"/>
    To hear the 3 day forecast for Chicago, say
forecast, or say
    another city name, such as London or New
York. </PROMPT>
    <INPUT TYPE="OPTIONLIST">
        <OPTION
NEXT="#chicago_forecast ">forecast</OPTION>
        <OPTION
NEXT="#london_current">london</OPTION>
        <OPTION NEXT="#newyork_current">new
york</OPTION>
        <OPTION
NEXT="#chicago_current">chicago</OPTION>
        <OPTION NEXT="#init">exit</OPTION>
    </INPUT>
</STEP>
<STEP NAME="chicago_forecast" PARENT="help_generic">
    <PROMPT>Chicago forecast for
    Tuesday. Flurries. High of 27. Low of 22.
    Wednesday. Snow showers. High of 27. Low of 12.
    <BREAK SIZE="large"/>
    Choose a city, or say exit to return to the main
menu. </PROMPT>
    <INPUT TYPE="OPTIONLIST">
        <OPTION
```

-continued

-continued

```
NEXT="#london_current">london</OPTION>
            <OPTION NEXT="#newyork_current">new
york</OPTION>
            <OPTION
NEXT="#chicago_current">chicago</OPTION>
            <OPTION NEXT="#init">exit</OPTION>
        </INPUT>
    </STEP>
    <STEP NAME="newyork_current" PARENT="help_generic">
        <PROMPT>It is currently 39 degrees in New York
City, with
        cloudy skies.<BREAK SIZE="large"/>
        To hear the 3 day forecast for New York, say
forecast, or say
    another city name, such as London or New
York.</PROMPT>
        <INPUT TYPE="OPTIONLIST">
            <OPTION
NEXT="#newyork_forecast">forecast</OPTION>
            <OPTION NEXT="#london_">london</OPTION>
            <OPTION NEXT="#newyork">new york</OPTION>
            <OPTION NEXT="#chicago">chicago</OPTION>
            <OPTION NEXT="#init">exit</OPTION>
        </INPUT>
    </STEP>
    <STEP NAME="newyork_forecast" PARENT="help_generic">
        <PROMPT>New York City forecast for
        Tuesday. Windy. High of 48. Low of 43.
        Wednesday. Rain. High of 43. Low of 28.
    <BREAK SIZE="large"/>
    Choose a city, or say exit to return to the main
menu. </PROMPT>
        <INPUT TYPE="OPTIONLIST">
            <OPTION
NEXT="#london_current ">london</OPTION>
            <OPTION NEXT="#newyork_current">new
york</OPTION>
            <OPTION NEXT="#chicago.">chicago</OPTION>
            <OPTION NEXT="#init">exit</OPTION>
      </INPUT>
    </STEP>
    <STEP NAME="market">
        <PROMPT>Market update is currently not
supported. </PROMPT>
        <INPUT TYPE="NONE" NEXT="#init"/>
    </STEP>
    <STEP NAME="news">
        <PROMPT>News update is currently not supported.
</PROMPT>
        <INPUT TYPE="NONE" NEXT="#init"/>
    </STEP>
    <STEP NAME="bye" PARENT="help_top">
        <PROMPT>Thanks for using Genie. Goodbye.
</PROMPT>
        <INPUT TYPE="NONE" NEXT="#exit"/>
    </STEP>
    </DIALOG>
```

FIG. **9** illustrates the same state diagram for the weather application as shown in FIG. **8** with labels for each dialog boundary. The initial dialog and dialog**1** contains the user prompts for welcome and city name. The Dialog**1** also controls the prompts for transitioning to hear a city's current or forecast weather and returning to the main menu. Dialog**2** handles access of the weather database for the current conditions of the city specified by the user and the information is read to the user. The Dialog**2** then returns control to dialog**1** again to get the user's next request. Similarly, dialog**3** handles access of the weather database for the forecast of the city requested and speaks the information. It returns control to dailog**1** to get the next user input.

The markup language set forth below illustrates an example of the weather application corresponding to the dialog boundaries as presented in the state diagram of FIG. **9**. The implementation of the application is with Active Server Pages using VBscript. It consists of three files called dialog**1**.asp, dialog**2**.asp, and dialog**3**.asp, each corresponding to the appropriate dialog.

For dialog**1**, there are two help message types, help_top and help dialog**1** (lines **16** and **29**). The first step, init, is at line **19**. The weather step follows at line **32**. Valid city names are those from the citylist table (line **36**) of the weather database. Lines **7** and **8** accomplish the database connection via ADO. Line **38** is the start of a loop for creating an option list of all possible city responses. If the user chooses a city, control goes to the step getcurrentweather in dialog**2**, as shown at line **40**. In this case, the city name is also passed to dialog**2** via the variable CITY at line **34**. The last major step in dialog**1** is nextcommand and can be referenced by dialog**2** or dialog**3**. It prompts the user for a cityname or the word forecast. Similar to the weather step, nextcommand uses a loop to create the optionlist (line **53**). If the user responds with a city name, the step getcurrentweather in dialog**2** is called. If the user responds with the word forecast, step getforecastweather _is called instead.

Dialog**2** contains a single step getcurrentweather. The step first reads the city name into local variable strcity (line **95**). A database query tries to find a match in the weather database for the city (lines **97** and **98**). If there is no weather information found for the city, the application will speak a message (line **101**) and proceed to init step in dialog**1** (line **110**). Otherwise, the application will speak the current weather information for the city (line **105**) and switch to the nextcommand step in dialog**1** (line **112**).

Dialog**3** is similar to dialog**2**. It contains a single step getforecastweather. The database query is identical to the one in dialog**2**. If there is weather information available for the city, the application will speak the weather forecast (line **105**), otherwise a notification message is spoken (line **101**). Dialog**3** relinquishes control back to dialog**1** with either the init step (line **110**) or next command (line **112**).

```
    <%@LANGUAGE="VBSCRIPT" %>
    <%
        Option Explicit
        Private objConnection, rsCities
        Private strCity, SQLQuery
        ' Create and open a connection to the database.
        Set objConnection =
Server.CreateObject("ADODB.Connection")
        objConnection.Open "Weather Database"
    %>
    <?XML VERSION="1.0"?>
    <!- -
-------------------------------------------------------------------------
_- ->
    <!- - (c) 1998 Motorola Inc.
- ->
    <!- - dialog1.asp
- ->
    <!- -
-------------------------------------------------------------------------
_- ->
    <DIALOG>
    <CLASS NAME="help_top">
        <HELP>You are at the top level menu. For weather
information,
            say weather. </HELP>
    </CLASS>
    <STEP NAME="init" PARENT="help_top">
        <PROMPT>Welcome to Genie.<BREAK SIZE="large"/>
        How may I help you? </PROMPT>
        <INPUT TYPE="OPTIONLIST">
            <OPTION NEXT="#weather">weather</OPTION>
            <OPTION NEXT="#market">market</OPTION>
            <OPTION NEXT="#news">news</OPTION>
            <OPTION NEXT="#bye">exit</OPTION>
```

-continued

```
        </INPUT>
    </STEP>
    <CLASS NAME="help_dialog1">
        <HELP>Your choices are <OPTIONS/>.</HELP>
    </CLASS>
    <STEP NAME="weather" PARENT="help_dialog1">
        <PROMPT>What city? </PROMPT>
        <INPUT TYPE="optionlist" NAME="CITY">
            <% ' Get all city names. %>
            <% SQLQuery = "SELECT * FROM CityList" %>
            <% Set rsCities =
objConnection.Execute (SQLQuery) %>
            <% Do Until rsCities.EOF %>
                <% ' Create an OPTION element for each
city. %>
                <OPTION
NEXT="dialog2.asp#getcurrentweather"
                    VALUE="<%=rsCities("City") %>">
                    <%=rsCities("City") %></OPTION>
                    <% rsCities.MoveNext %>
            <% Loop %>
                <OPTION NEXT="#init">exit</OPTION>
        </INPUT>
    </STEP>
    <STEP NAME="nextcommand" PARENT="help_dialog1">
        <% strCity = Request.QueryString("CITY") %>
        <PROMPT> To hear the 3 day forecast for
<%=strCity%>,say
         forecast, or say another city name.</PROMPT>
        <INPUT TYPE="optionlist" NAME="CITY">
            <% ' Get all city names. %>
            <% SQLQuery = "SELECT * FROM CityList" %>
            <% Set rsCities =
objConnection.Execute (SQLQuery) %>
            <% Do Until rsCities.EOF %>
                <% ' Create an OPTION element for each
city. %>
                <OPTION
NEXT="dialog2.asp#getcurrentweather"
                    VALUE="<%= rsCities("City") %>">
                    <%= rsCities("City") %></OPTION>
                    <% rsCities.MoveNext %>
            <% Loop %>
                <OPTION
NEXT="dialog3.asp#getforecastweather"
                    VALUE="<%= strCity
%>">forecast</OPTION>
                <OPTION NEXT="#init">exit</OPTION>
        </INPUT>
    </STEP>
    <STEP NAME="market">
        <PROMPT>Market update is currentiy not
supported. </PROMPT>
        <INPUT TYPE="NONE" NEXT="#init"/>
    </STEP>
    <STEP NAME="news">
        <PROMPT>News update is currentiy not supported.
</PROMPT>
        <INPUT TYPE="NONE" NEXT="#init"/>
    </STEP>
    <STEP NAME="bye" PARENT="help_top">
        <PROMPT>Thanks for using Genie. Goodbye.
</PROMPT>
        <INPUT TYPE="NONE" NEXT="#exit"/>
    </STEP>
    </DIALOG>
    <!- - _________End of
Dialog1.asp_____________- - >
    <%@ LANGUAGE="VBSCRIPT" %>
    <%
        Option Explicit
        Private objConnection, rsWeather, SQLQuery
        Private strCity, Valid
        ' Create and open a connection to the database.
        Set objConnection =
Server.CreateObject("ADODB.Connection")
        objConnection.Open "Weather Database"
    %>
    <?XML VERSION="1.0"?>
```

-continued

```
    <!- -
-------------------------------------------------------------------------
_- ->
    <!- - (c) 1998 Motorola Inc.
- ->
    <!- - dialog2.asp
- ->
    <!- -
-------------------------------------------------------------------------
_- ->
    <DIALOG>
    <CLASS NAME="help_dialog2">
        <HELP>Your choices are <OPTIONS/>.</HELP>
    </CLASS>
    <STEP NAME="getcurrentweather">
        <% strCity = Request.QueryString("CITY") %>
        <% Valid = "TRUE" %>
        <% SQLQUery = "SELECT * FROM WDB WHERE( City='"
& strCity & "'"
            )" %>
        <% Set rsWeather =
objConnection.Execute (SQLQuery) %>
        <% If rsWeather.EOF Then %>
            <% Valid = "FALSE" %>
            <PROMPT>Sorry, <BREAK/>There are no current
weather
            conditions available for
<%=strCity%>.<BREAK/></PROMPT>
        <% Else %>
            <% ' Speak current weather information %>
            <PROMPT><%=rsWeather("Current")%></PROMPT>
        <%End If %>
        <INPUT TYPE = "Hidden" NAME="CITY"
VALUE="<%=strCity%>" >
        </INPUT>
        <% If ( Valid = "FALSE") Then %>
            <INPUT TYPE="none"
NEXT="dialog1.asp#init"</INPUT>
        <% Else %>
            <INPUT TYPE="none"
NEXT="dialog1.asp#nextcommand "></INPUT>
        <% End If %>
    </STEP>
    </DIALOG>
    <!- - _________End of
Dialog2.asp_____________- - >
    <%@ LANGUAGE="VBSCRIPT" %>
    <%
    Option Explicit
    Private objConnection, rsWeather, SQLQuery
    Private strCity, Valid
    ' Create and open a connection to the database.
    Set objConnection =
Server.CreateObject("ADODB.Connection")
        objConnection.Open "Weather Database"
    %>
    <? XML VERSION="1.0"? >
    <!- -
-------------------------------------------------------------------------
_- ->
    <!- - (c) 1998 Motorola Inc.
- ->
    <!- - dialog3.asp
- ->
    <!- -
-------------------------------------------------------------------------
_- ->
    <DIALOG>
    <CLASS NAME="help_dialog3">
        <HELP>Your choices are <OPTIONS/>.</HELP>
    </CLASS>
    <STEP NAME="getforecastweather">
    <% strCity = Request.QueryString("CITY") %>
    <% Valid = "TRUE" %>
    <% SQLQuery = "SELECT * FROM WDB WHERE( City='" &
strCity & "' )"%>
    <% Set rsWeather = objConnection.Execute(SQLQuery) %>
    <% If rsWeather.EOF Then%>
        <% Valid = "FALSE" %>
```

-continued

```
        <PROMPT> Sorry, <BREAK/> There is no forecast
weather
        available for <%=strCity%>.<BREAK/></PROMPT>
    <% Else %>
        <% ' Speak forecast weather information %>
        <PROMPT> <%=rsWeather("Forecast")%> </PROMPT>
    <% End If %>
    <INPUT TYPE = "Hidden" NAME="CITY"
VALUE="<%=strCity%>" > </INPUT>
    <% If ( Valid = "FALSE") Then%>
    <INPUT TYPE="none" NEXT="dialog1.asp#init"</INPUT>
    <% Else %>
    <INPUT TYPE="none"
NEXT="dialog1.asp#nextcommand"></INPUT>
    <% End If %>
    </STEP>
    </DIALOG>
    <!- - _____End of
Dialog3.asp_____- - >
```

Accordingly, there has been described herein methods and systems to allow users to access information from any location in the world via any suitable network access device. The user can access up-to-date information, such as, news updates, designated city weather, traffic conditions, stock quotes, and stock market indicators. The system also allows the user to perform various transactions (i.e., order flowers, place orders from restaurants, place buy or sell orders for stocks, obtain bank account balances, obtain telephone numbers, receive directions to destinations, etc.)

It will be apparent to those skilled in the art that the disclosed embodiment may be modified in numerous ways and may assume many embodiments other than the preferred form specifically set out and described above. Accordingly, it is intended by the appended claims to cover all modifications of the invention which fall within the true spirit and scope of the invention.

What is claimed is:

1. A markup language document stored on a computer-readable medium to provide interactive services comprising:

a dialog element including a plurality of markup language elements, each of the plurality of markup language elements being identifiable by at least one markup tag;

a step element contained within the dialog element to define a state within the dialog element, the step element including a prompt element and an input element;

the prompt element including an announcement to be read to the user; and

the input element including an attribute to allow an audible user input to be converted to a text string.

2. The markup language document of claim **1**, wherein the attribute includes one of a name attribute, a next attribute, a method attribute, and action attribute and a timeout attribute.

3. The markup language document of claim **2**, wherein the name attribute comprises an identifier.

4. The markup language document of claim **2**, wherein the next attribute comprises a next step address.

5. The markup language document of claim **1**, wherein the method attribute comprises one of a get and post.

6. The markup language document of claim **1** wherein the action attribute comprises a pointer to a script that processes the input on a server.

7. The markup language document of claim **1**, wherein the timeout attribute comprises a period of time for the user to enter audible input.

8. The markup language document of claim **1**, wherein the attribute comprises a form input.

9. A The markup language document of claim **1**, wherein announcement comprises one of voice over internet protocol data and textual data.

10. The markup language file of claim **1**, wherein the markup language document contains a begin tag and an end tag for the dialog element.

11. The markup language document of claim **1**, wherein the step element further contains one of a name attribute, a bargein attribute, a parent attribute, and a cost attribute.

12. The markup language document of claim **1**, wherein a bargein attribute interrupts the step element in response to a selected user input.

13. The markup language document of claim **1**, wherein the input element includes an input attribute.

14. The markup language document of claim **13**, wherein the input attribute includes one of a date input, a digits input, a grammar input, a hidden input, a money input, a none input, a number input, an optionlist input, a phone input, a profile input, a record input, a time input, and a yorn input.

15. The markup language document of claim **14**, wherein the grammar input includes a grammar corresponding to a user input.

16. The markup language document of claim **14**, wherein the grammar input includes an electronic address to provide an indication of the location of a grammar.

17. The markup language document of claim **14**, wherein the electronic address includes one of a URL and an address of a second step element.

18. The markup language document of claim **14**, wherein the grammar input includes a nextmethod attribute having a memory address of a grammar.

19. The markup language document of claim **14**, wherein the grammar input includes a selected period of time for the user to enter audio input.

20. The markup language document of claim **14**, wherein the input element further contains a hidden input including a variable to store information.

21. The markup language document of claim **14**, wherein each of the date input, the digit input, the money input, the number input, the phone input, and the time input includes a predetermined grammar.

22. The markup language document of claim **14**, wherein the yorn element defines one of an affirmative response and a negative response.

23. The markup language document of claim **14**, wherein the optionlist input includes at least one option corresponding to user input.

24. The markup language document of claim **14**, wherein the form input is used to send information to an information source.

25. The markup language document of claim **14**, wherein the record input is used to record an audio input from a user for a selected length of time.

26. The markup language document of claim **14**, wherein the profile input is used to store user data.

27. The markup language document of claim **1**, wherein the dialog element further contains ad option element including at least one option corresponding to a numerical value.

28. The markup language document of claim **1**, wherein the dialog element further contains an ack element including a user acknowledgment.

29. The markup language document of claim **1**, wherein the dialog element further contains an audio element including audio data to be played to the user.

30. The markup language document of claim **29**, wherein the audio data is contained in a voice over internet protocol.

31. The markup language document of claim 1, wherein the dialog element further contains a break element to provide a silent period.

32. The markup language document of claim 1, wherein the dialog element further contains a case element including data to correspond to at least one user input.

33. The markup language document of claim 1, wherein the dialog element further contains a class element including a plurality of elements to be reused with a dialog element.

34. The markup language document of claim 1, wherein the dialog element further contains an emp element to allow at least a portion of a word to be emphasized during playback.

35. The markup language document of claim 1, wherein the dialog element further contains an options element including at least one option for a user to select.

36. The markup language document of claim 35, wherein the options element includes at least one element to be reused in the dialog step.

37. The markup language document of claim 1, wherein the dialog further contains an OR element including alternative options corresponding to the phonetic representation of the user input.

38. The markup language document of claim 1, wherein the dialog element further contains a pros element to define one of a rate of speech, the pitch of speech the range of speech, and the volume of speech to be read to the user.

39. The markup language document of claim 1, wherein the dialog element further contains a response element including a next step element to execute.

40. The markup language document of claim 1, wherein the dialog element further contains a response element including a next step element to execute.

41. The markup language document of claim 1, wherein the dialog element further contains a value element including a variable to store information to be played back to the user.

42. The markup language document of claim 1, wherein the step element includes a cost attribute used to track content usage for billing purposes.

43. A method of creating a voice application program for providing interactive voice services, the method comprising the steps of:

creating a markup language document having a plurality of elements;

selecting a prompt element;

defining a voice communication in the prompt element to be read to the user;

selecting an input element; and

selecting an attribute to allow an audible user input to be converted to a text string.

44. A program stored on a computer-readable medium to provide interactive services comprising: a prompt element including a voice communication to be read to a user; and an input element including an attribute to allow an audible user input to be converted to a text string, —has been inserted wherein the attribute includes one of a name attribute, a next attribute, a method attribute, and action attribute and a timeout attribute.

45. The program of claim 44, wherein the name attribute comprises an identifier.

46. The program claim 44, wherein the next attribute comprises a next step address.

47. The program of claim 44, wherein the method attribute comprises one of a get and post.

48. The program of claim 44, wherein the action attribute comprises a pointer to a script that processes the input on a server.

49. The markup language document of claim 44, wherein the timeout attribute comprises a period of time for the user to enter audible input.

50. A markup language document stored on a computer-readable medium to provide interactive voice services comprising:

a dialog element being identified by at least one markup tag; and

an input element including an attribute to allow an audible user input to be converted to a text string.

51. A method comprising the steps of:

providing an audible output to a user;

receiving an audible input from the user;

converting the audible input to text; and

posting the text to a destination associated with an electronic address.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 6,493,673 B1
DATED : December 10, 2002
INVENTOR(S) : Ladd, David et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 46,
Line 11, delete "-has been inserted"

Signed and Sealed this

Tenth Day of June, 2003

JAMES E. ROGAN
Director of the United States Patent and Trademark Office

EXHIBIT 5

US006983370B2

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

(10) **Patent No.: US 6,983,370 B2**
(45) **Date of Patent: Jan. 3, 2006**

(54) **SYSTEM FOR PROVIDING CONTINUITY BETWEEN MESSAGING CLIENTS AND METHOD THEREFOR**

(75) Inventors: **Eric Thomas Eaton**, Lake Worth, FL (US); **David Jeffery Hayes**, Lake Worth, FL (US); **Von Alan Mock**, Boynton Beach, FL (US)

(73) Assignee: **Motorola, Inc.**, Schaumberg, IL (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 875 days.

(21) Appl. No.: **09/995,338**

(22) Filed: **Nov. 27, 2001**

(65) **Prior Publication Data**

US 2003/0101343 A1 May 29, 2003

(51) **Int. Cl.**
G06F 1/26 (2006.01)

(52) **U.S. Cl.** **713/182**; 713/153; 713/161; 713/189; 713/193; 713/200; 713/201

(58) **Field of Classification Search** 713/182, 713/153, 161, 189, 193, 200, 201
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,811,377	A	3/1989	Krolopp et al.	
5,535,436	A	7/1996	Yoshida et al.	
6,038,296	A	3/2000	Brunson et al.	
6,041,229	A	3/2000	Turner	
6,101,531	A *	8/2000	Eggleston et al.	709/206
6,189,098	B1 *	2/2001	Kaliski, Jr.	713/168

* cited by examiner

Primary Examiner—Thomas R. Peeso
(74) *Attorney, Agent, or Firm*—Randi L. Karpinia; Sylvia Chen

(57) **ABSTRACT**

A messaging communication system (**10**) includes a plurality of messaging clients (**12**). A first messaging client (**14**) establishes a first communication connection (**16**) operating using a plurality of client data (**25**). The first messaging client (**14**) transfers the plurality of client data (**25**) to a second messaging client (**20**). The second messaging client (**20**) establishes a second communication connection (**22**) operating using the plurality of client data (**25**).

67 Claims, 24 Drawing Sheets

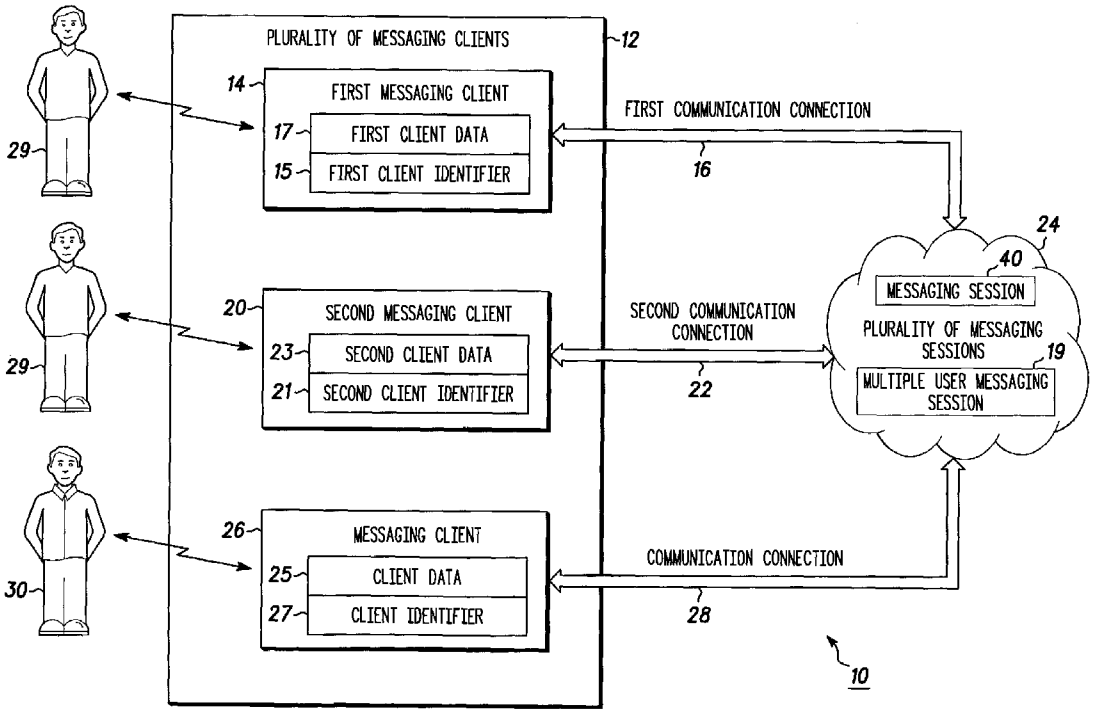
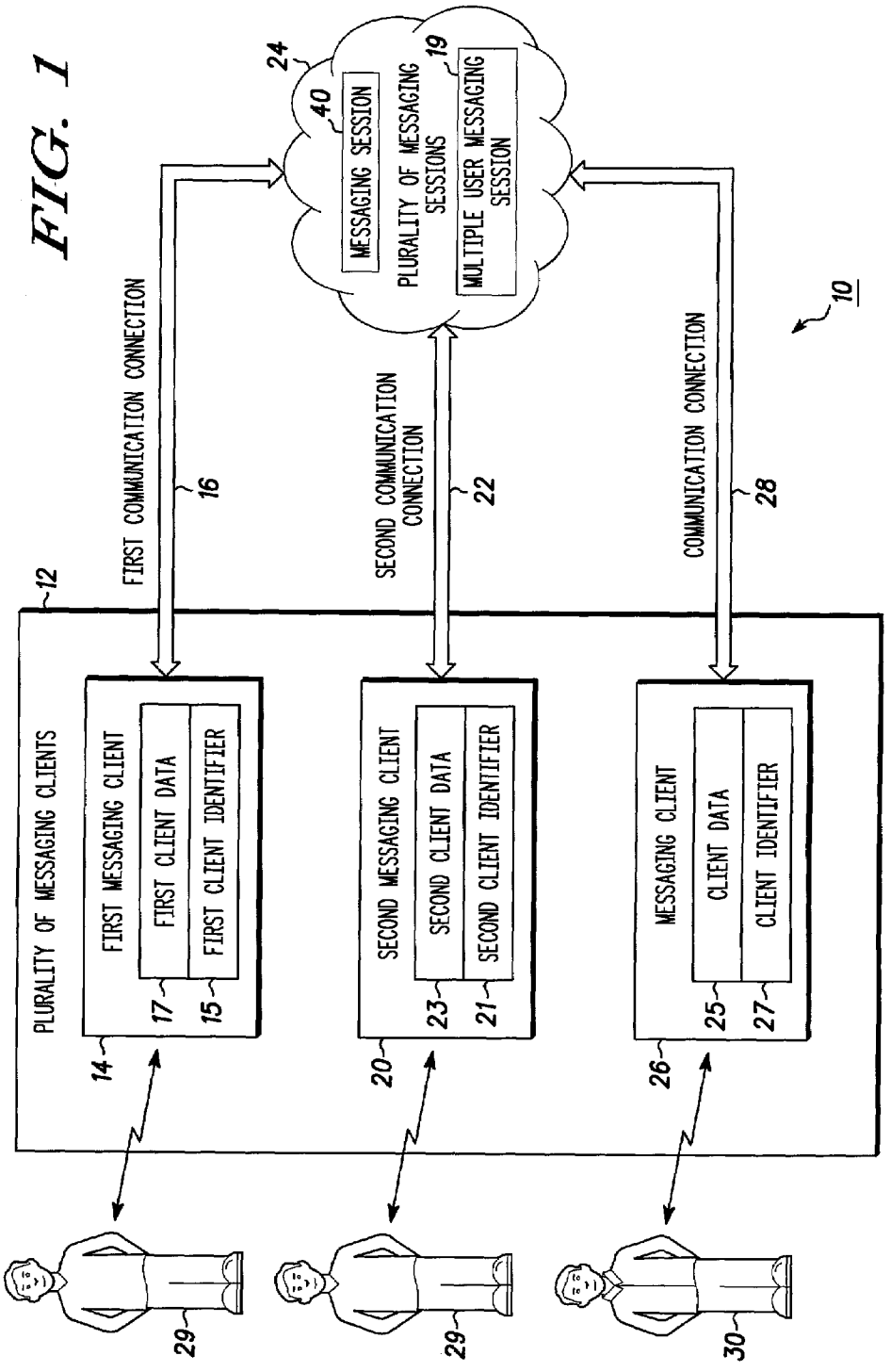


FIG. 1


24
40
MESSAGING SESSION
19
PLURALITY OF MESSAGING SESSIONS
MULTIPLE USER MESSAGING SESSION
10
FIRST COMMUNICATION CONNECTION
16
SECOND COMMUNICATION CONNECTION
22
COMMUNICATION CONNECTION
28
12
PLURALITY OF MESSAGING CLIENTS
FIRST MESSAGING CLIENT
FIRST CLIENT DATA
FIRST CLIENT IDENTIFIER
17
15
14
SECOND MESSAGING CLIENT
SECOND CLIENT DATA
SECOND CLIENT IDENTIFIER
23
21
20
MESSAGING CLIENT
CLIENT DATA
CLIENT IDENTIFIER
25
27
26
29
29
30

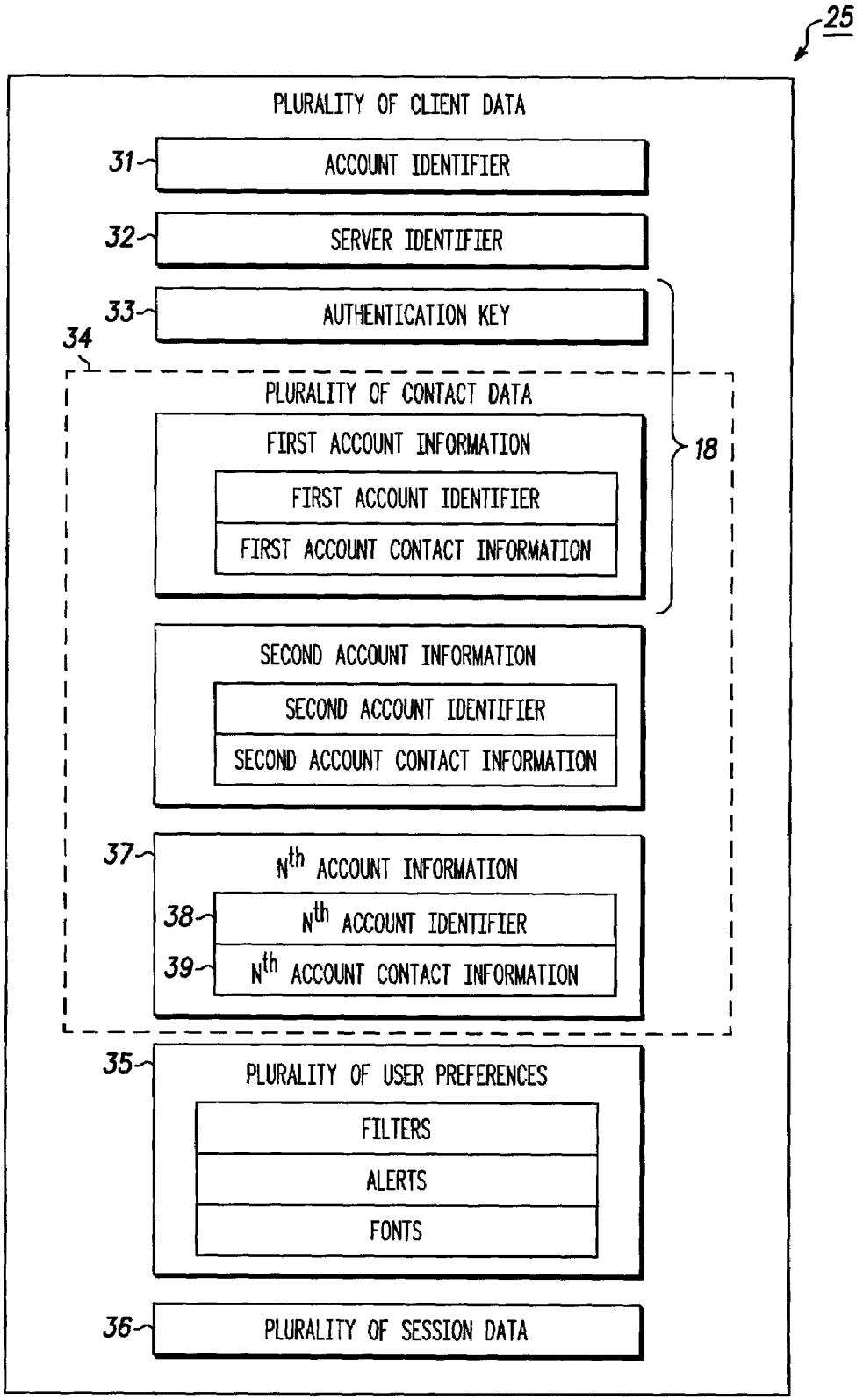

25
PLURALITY OF CLIENT DATA
31
ACCOUNT IDENTIFIER
32
SERVER IDENTIFIER
33
AUTHENTICATION KEY
34
PLURALITY OF CONTACT DATA
FIRST ACCOUNT INFORMATION
FIRST ACCOUNT IDENTIFIER
FIRST ACCOUNT CONTACT INFORMATION
18
SECOND ACCOUNT INFORMATION
SECOND ACCOUNT IDENTIFIER
SECOND ACCOUNT CONTACT INFORMATION
37
Nth ACCOUNT INFORMATION
38
Nth ACCOUNT IDENTIFIER
39
Nth ACCOUNT CONTACT INFORMATION
35
PLURALITY OF USER PREFERENCES
FILTERS
ALERTS
FONTS
36
PLURALITY OF SESSION DATA

FIG.2

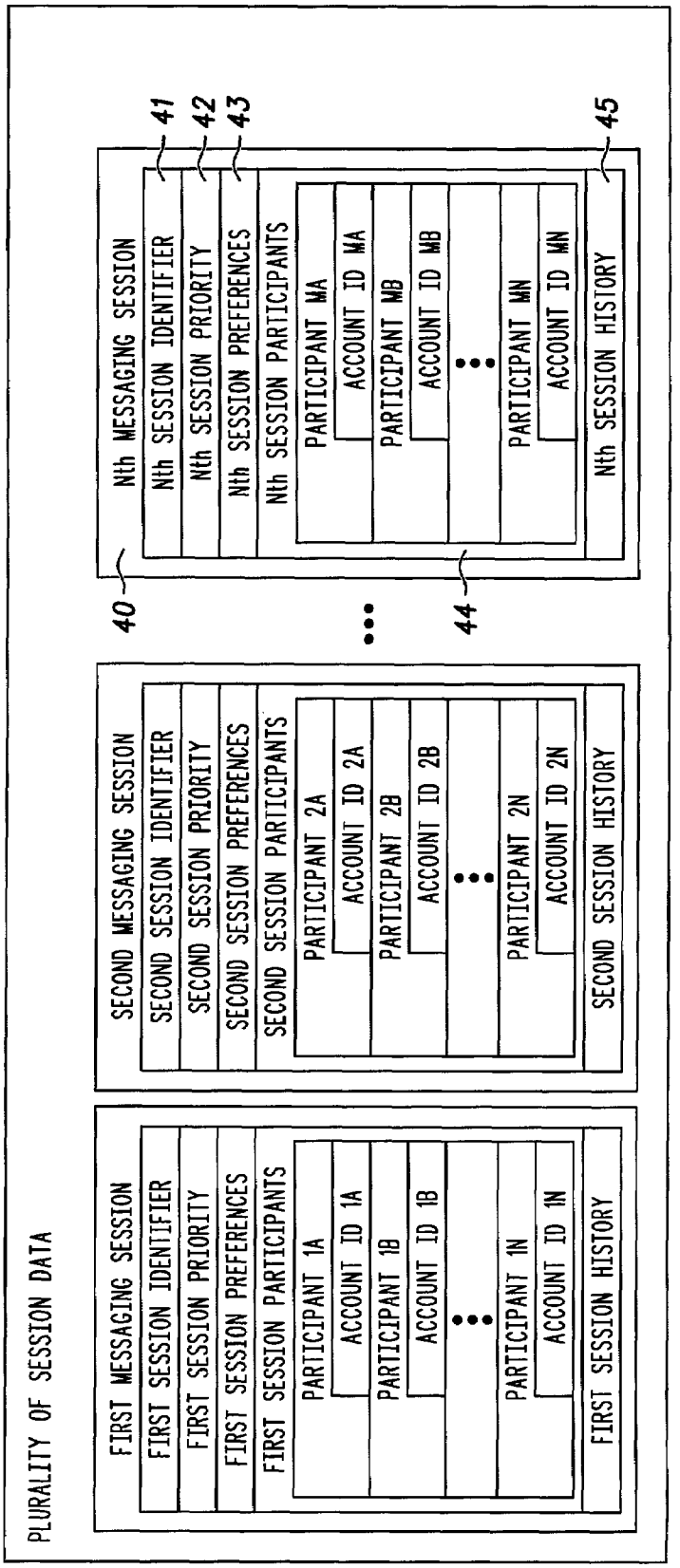

PLURALITY OF SESSION DATA
FIRST MESSAGING SESSION
FIRST SESSION IDENTIFIER
FIRST SESSION PRIORITY
FIRST SESSION PREFERENCES
FIRST SESSION PARTICIPANTS
PARTICIPANT 1A
ACCOUNT ID 1A
PARTICIPANT 1B
ACCOUNT ID 1B
PARTICIPANT 1N
ACCOUNT ID 1N
FIRST SESSION HISTORY
SECOND MESSAGING SESSION
SECOND SESSION IDENTIFIER
SECOND SESSION PRIORITY
SECOND SESSION PREFERENCES
SECOND SESSION PARTICIPANTS
PARTICIPANT 2A
ACCOUNT ID 2A
PARTICIPANT 2B
ACCOUNT ID 2B
PARTICIPANT 2N
ACCOUNT ID 2N
SECOND SESSION HISTORY
Nth MESSAGING SESSION
Nth SESSION IDENTIFIER
Nth SESSION PRIORITY
Nth SESSION PREFERENCES
Nth SESSION PARTICIPANTS
PARTICIPANT MA
ACCOUNT ID MA
PARTICIPANT MB
ACCOUNT ID MB
PARTICIPANT MN
ACCOUNT ID MN
Nth SESSION HISTORY
40
41
42
43
44
45
36
1

FIG. 3

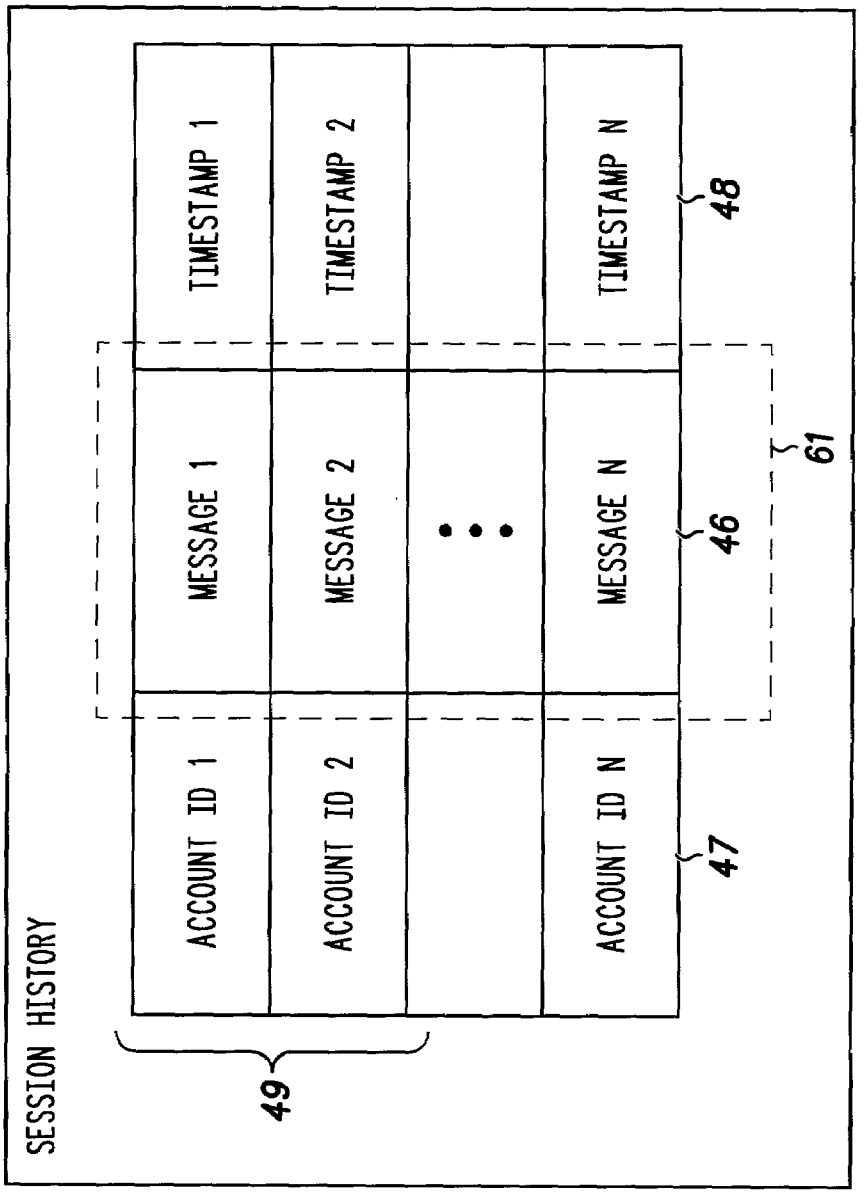

SESSION HISTORY
ACCOUNT ID 1
ACCOUNT ID 2
ACCOUNT ID N
MESSAGE 1
MESSAGE 2
MESSAGE N
TIMESTAMP 1
TIMESTAMP 2
TIMESTAMP N
49
47
46
48
61
45

FIG. 4

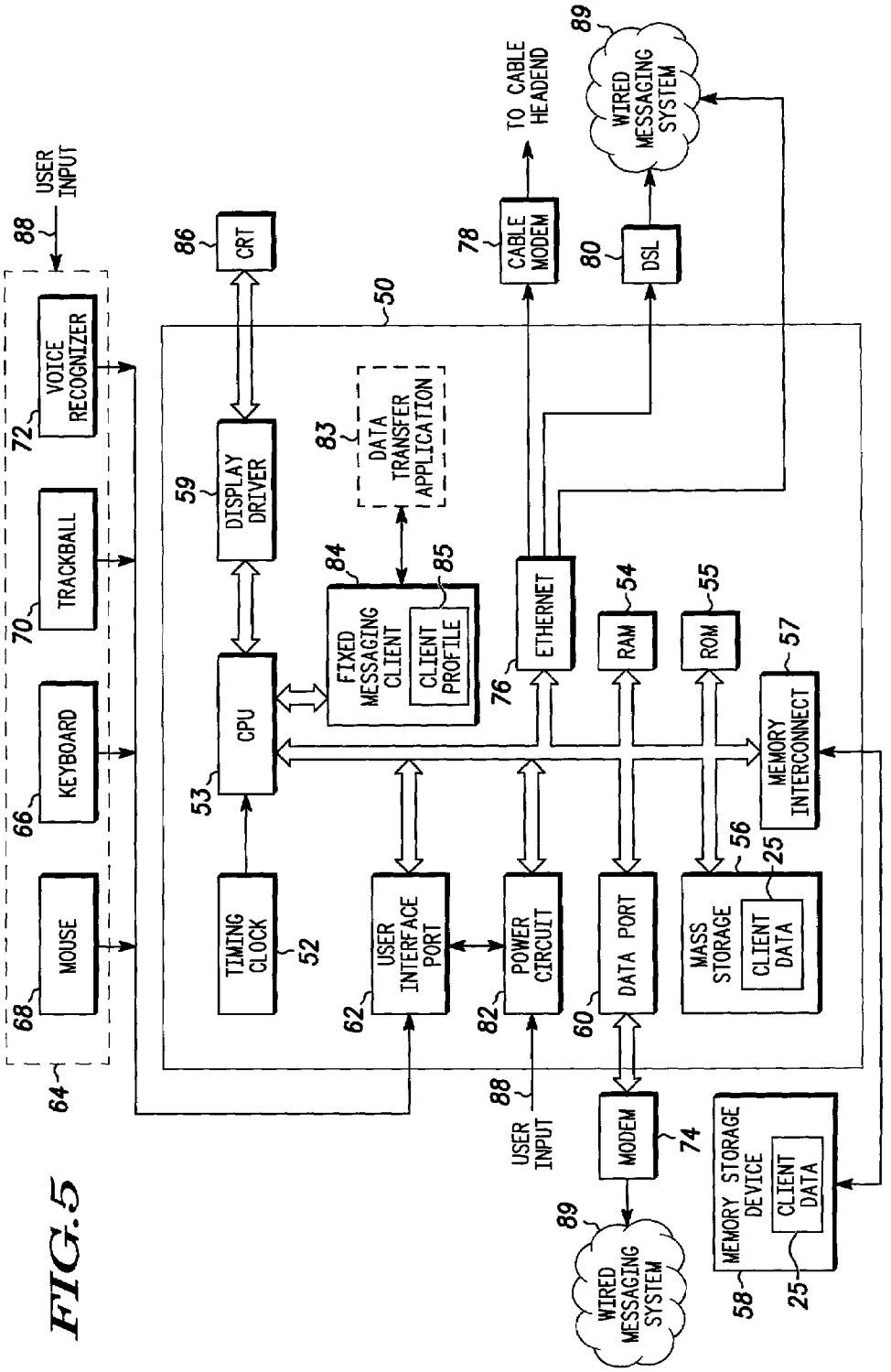

USER INPUT
88
VOICE RECOGNIZER
72
TRACKBALL
70
KEYBOARD
66
MOUSE
68
64
CRT
86
DISPLAY DRIVER
59
CPU
53
TIMING CLOCK
52
50
DATA TRANSFER APPLICATION
83
FIXED MESSAGING CLIENT
84
CLIENT PROFILE
85
USER INTERFACE PORT
62
POWER CIRCUIT
82
USER INPUT
88
TO CABLE HEADEND
CABLE MODEM
78
DSL
80
WIRED MESSAGING SYSTEM
89
ETHERNET
76
RAM
54
ROM
55
MEMORY INTERCONNECT
57
DATA PORT
60
MASS STORAGE
56
CLIENT DATA
25
MODEM
74
WIRED MESSAGING SYSTEM
89
MEMORY STORAGE DEVICE
58
CLIENT DATA
25

FIG. 5

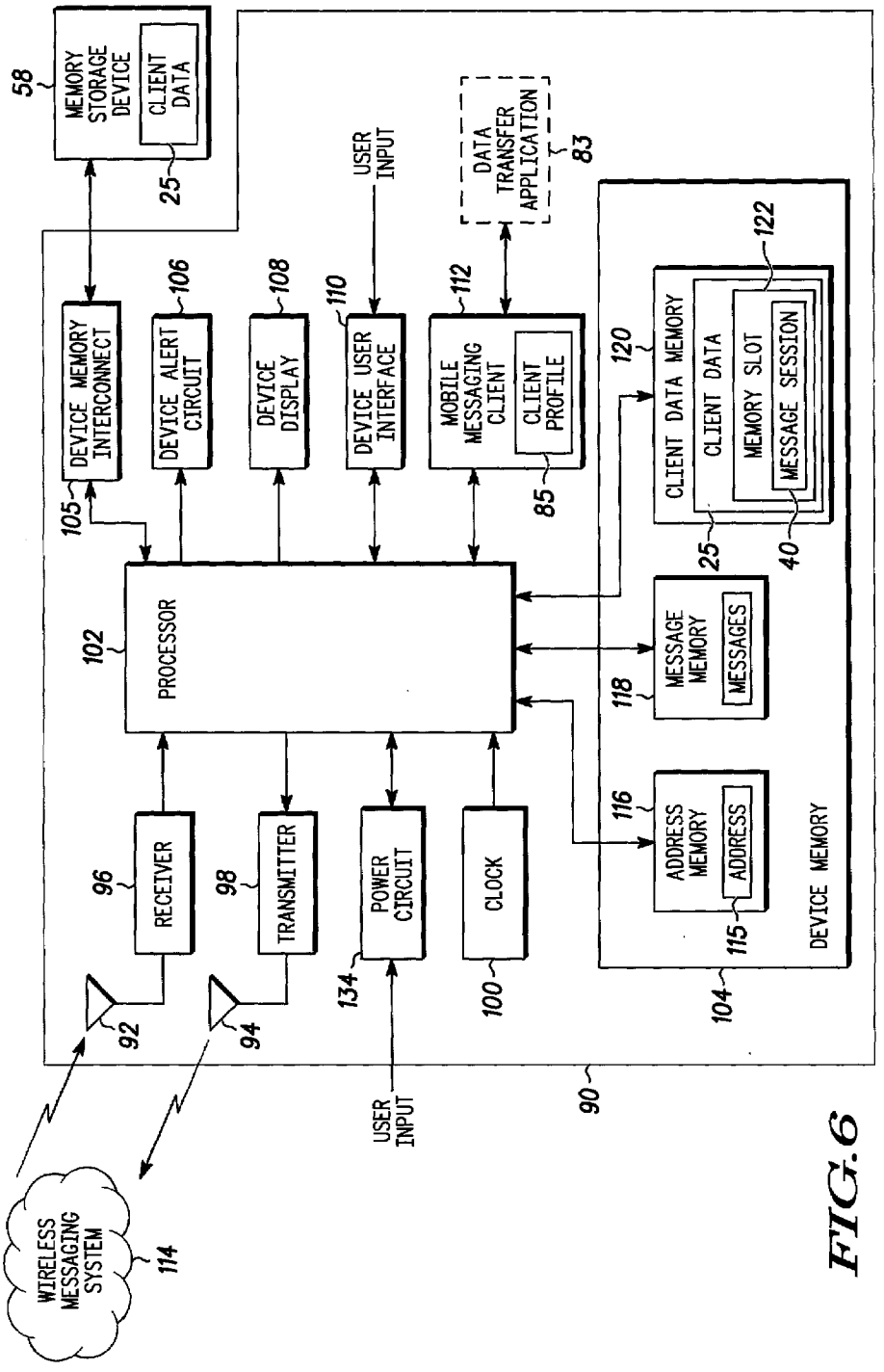

58
MEMORY STORAGE DEVICE
CLIENT DATA
25
DEVICE MEMORY INTERCONNECT
105
DEVICE ALERT CIRCUIT
106
DEVICE DISPLAY
108
DEVICE USER INTERFACE
110
USER INPUT
MOBILE MESSAGING CLIENT
CLIENT PROFILE
85
112
DATA TRANSFER APPLICATION
83
CLIENT DATA MEMORY
120
CLIENT DATA
MEMORY SLOT
MESSAGE SESSION
122
40
PROCESSOR
102
MESSAGE MEMORY
MESSAGES
118
ADDRESS MEMORY
ADDRESS
116
115
DEVICE MEMORY
104
RECEIVER
96
TRANSMITTER
98
POWER CIRCUIT
134
CLOCK
100
92
94
USER INPUT
90
WIRELESS MESSAGING SYSTEM
114

FIG. 6

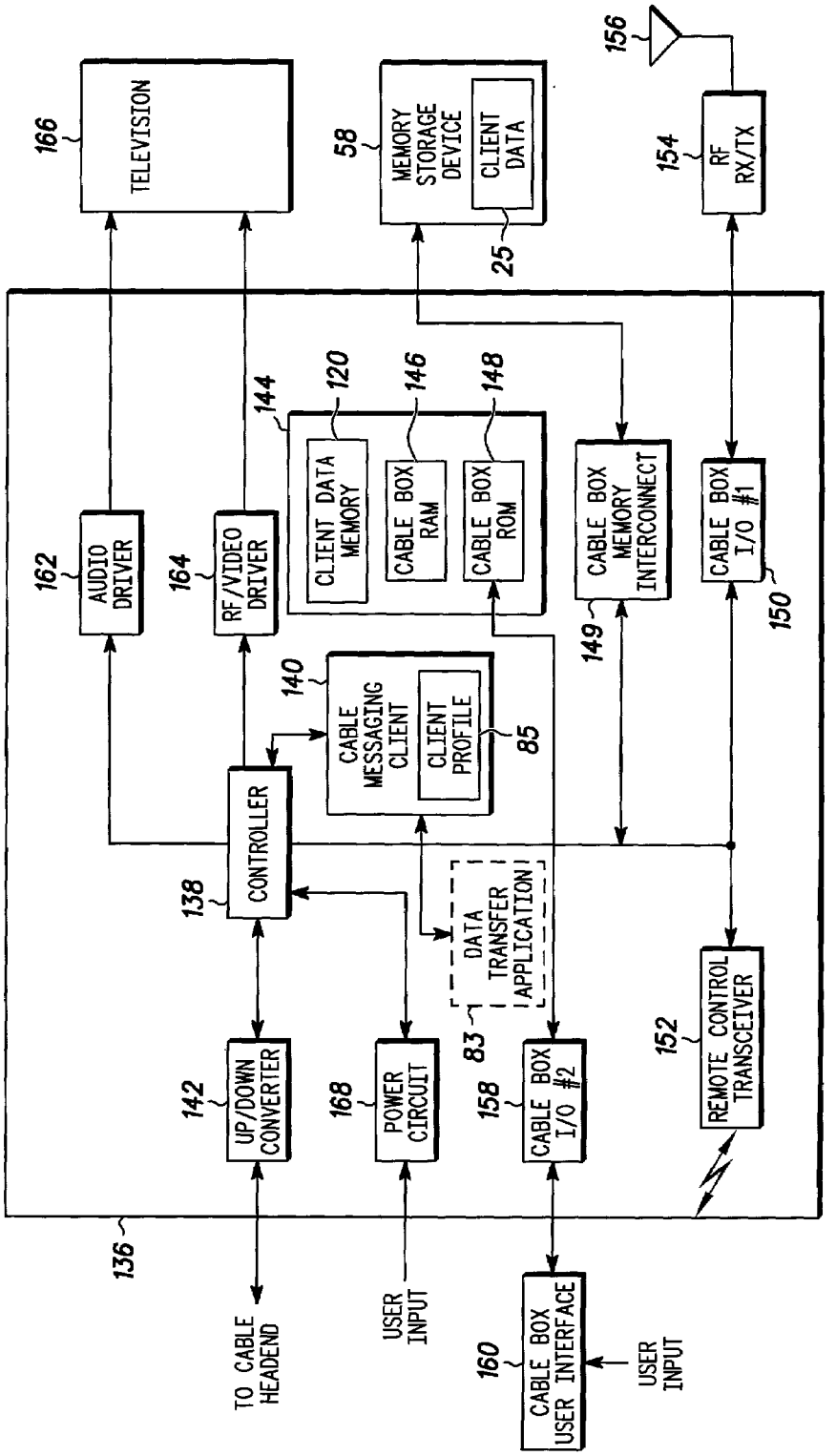

166
TELEVISION
58
MEMORY STORAGE DEVICE
CLIENT DATA
25
156
154
RF RX/TX
144
120
146
148
CLIENT DATA MEMORY
CABLE BOX RAM
CABLE BOX ROM
162
AUDIO DRIVER
164
RF/VIDEO DRIVER
CABLE BOX MEMORY INTERCONNECT
149
CABLE BOX I/O #1
150
140
CABLE MESSAGING CLIENT
CLIENT PROFILE
85
CONTROLLER
138
DATA TRANSFER APPLICATION
83
142
UP/DOWN CONVERTER
168
POWER CIRCUIT
158
CABLE BOX I/O #2
152
REMOTE CONTROL TRANSCEIVER
136
TO CABLE HEADEND
USER INPUT
160
CABLE BOX USER INTERFACE
USER INPUT

FIG. 7

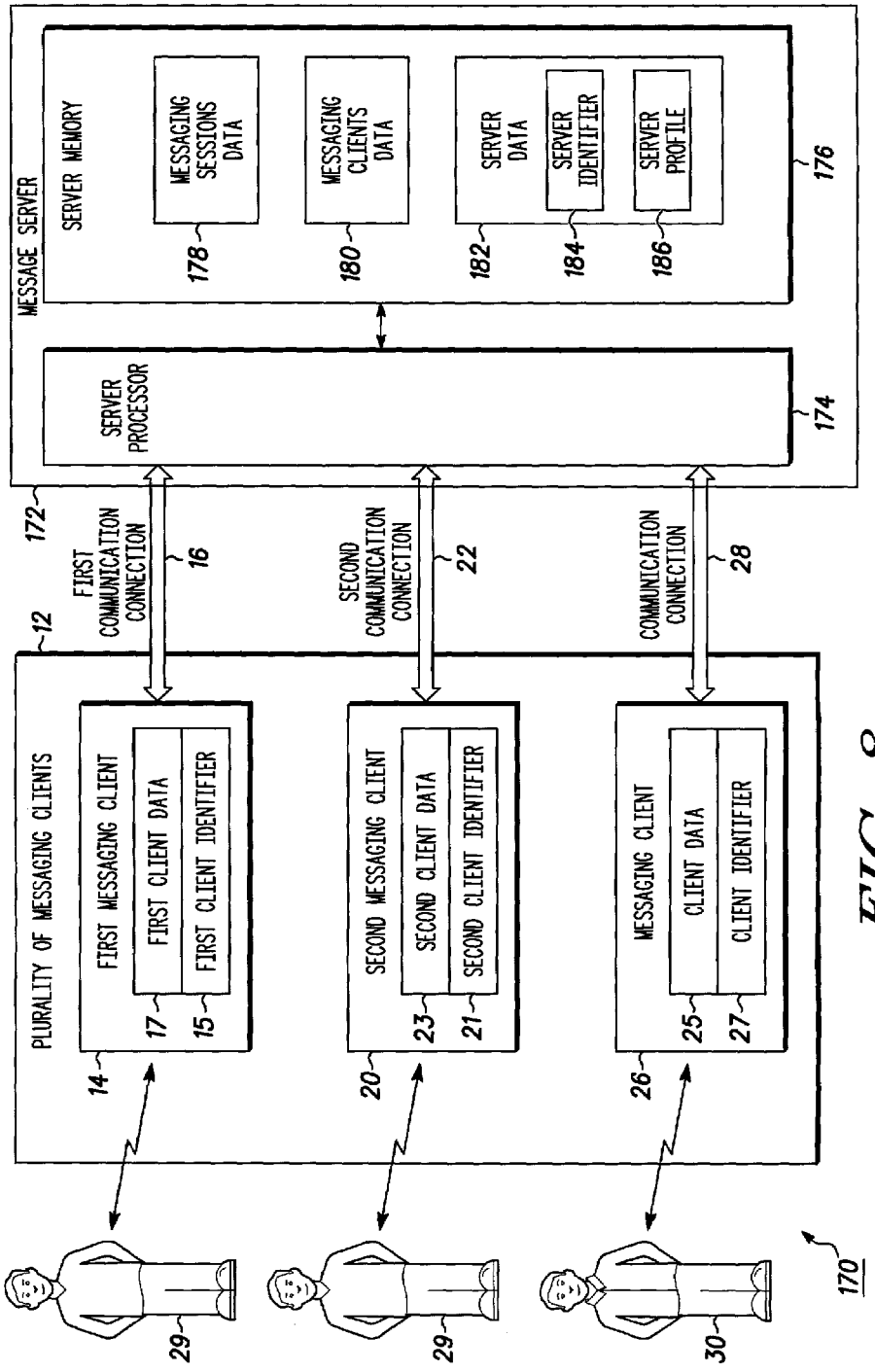

MESSAGE SERVER
SERVER MEMORY
MESSAGING SESSIONS DATA
MESSAGING CLIENTS DATA
SERVER DATA
SERVER IDENTIFIER
SERVER PROFILE
176
178
180
182
184
186
SERVER PROCESSOR
174
172
FIRST COMMUNICATION CONNECTION
16
SECOND COMMUNICATION CONNECTION
22
COMMUNICATION CONNECTION
28
12
PLURALITY OF MESSAGING CLIENTS
FIRST MESSAGING CLIENT
FIRST CLIENT DATA
FIRST CLIENT IDENTIFIER
17
15
14
SECOND MESSAGING CLIENT
SECOND CLIENT DATA
SECOND CLIENT IDENTIFIER
23
21
20
MESSAGING CLIENT
CLIENT DATA
CLIENT IDENTIFIER
25
27
26
29
29
30
170

FIG. 8

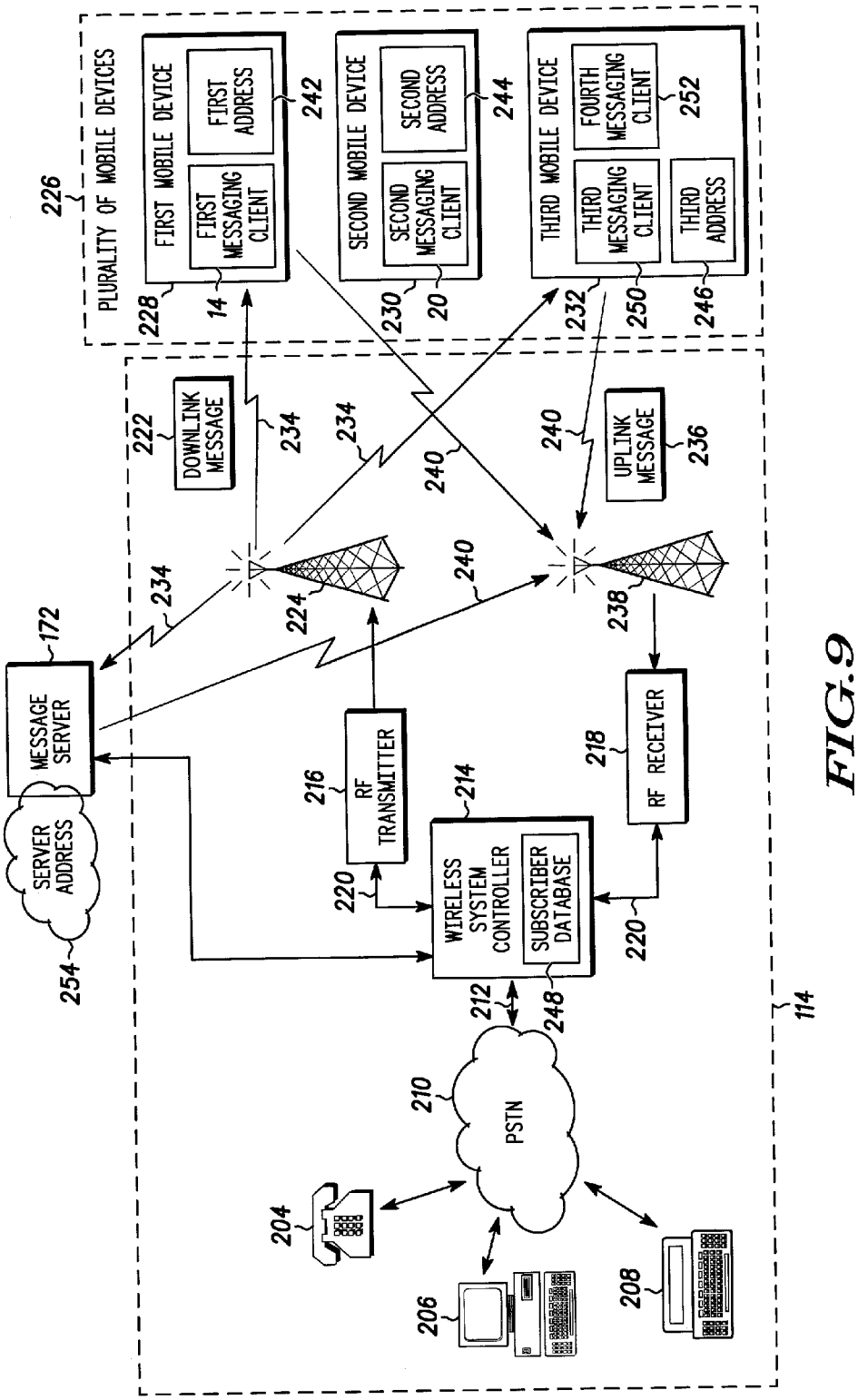

PLURALITY OF MOBILE DEVICES
226
FIRST MOBILE DEVICE
FIRST MESSAGING CLIENT
FIRST ADDRESS
242
SECOND MOBILE DEVICE
SECOND MESSAGING CLIENT
SECOND ADDRESS
244
THIRD MOBILE DEVICE
THIRD MESSAGING CLIENT
THIRD ADDRESS
FOURTH MESSAGING CLIENT
252
228
14
230
20
232
250
246
DOWNLINK MESSAGE
222
UPLINK MESSAGE
236
234
240
224
238
MESSAGE SERVER
172
SERVER ADDRESS
254
RF TRANSMITTER
216
RF RECEIVER
218
WIRELESS SYSTEM CONTROLLER
214
SUBSCRIBER DATABASE
220
212
248
PSTN
210
204
206
208
114

FIG. 9

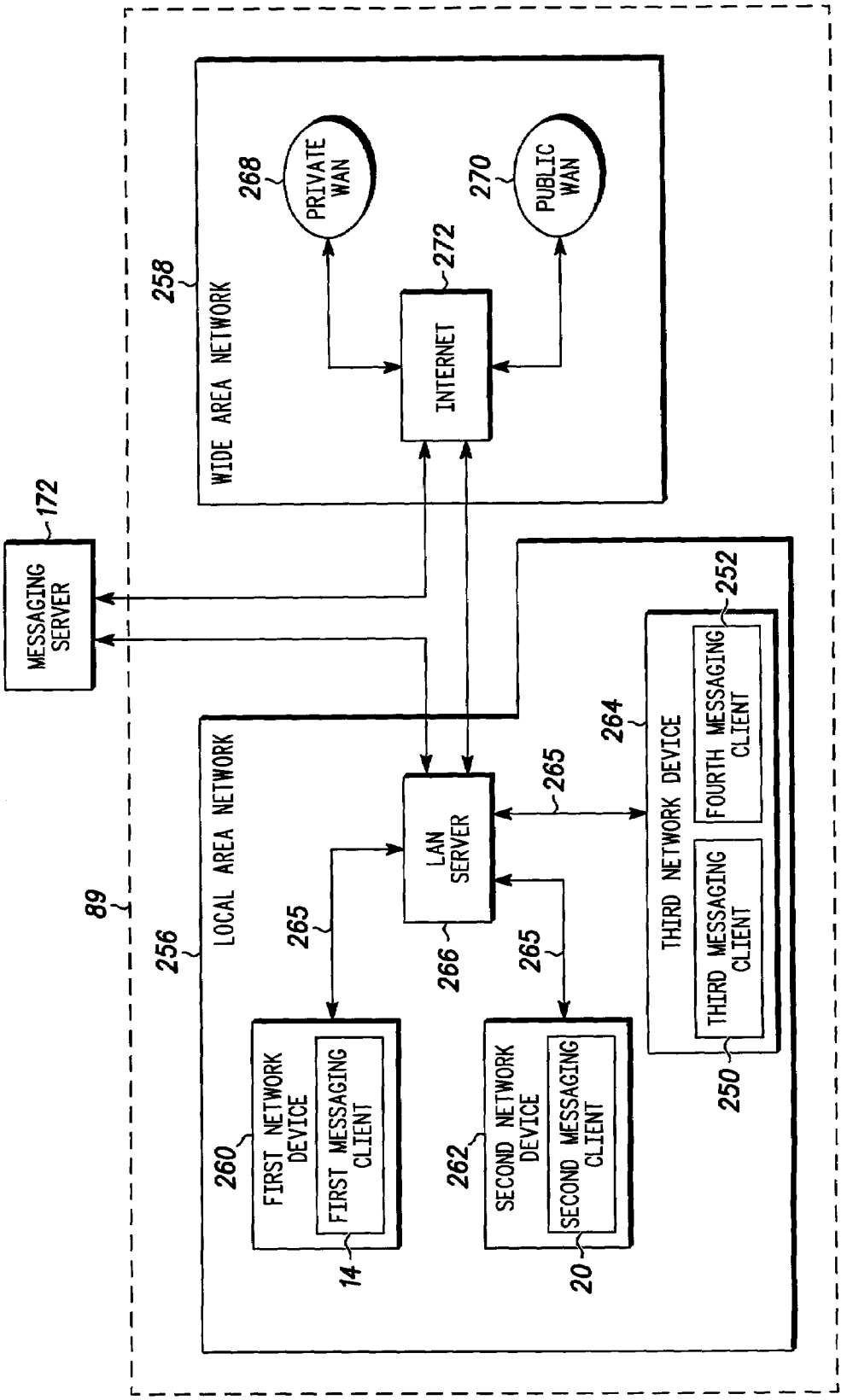

89
MESSAGING SERVER
172
WIDE AREA NETWORK
258
PRIVATE WAN
268
PUBLIC WAN
270
INTERNET
272
LOCAL AREA NETWORK
256
LAN SERVER
266
265
265
265
264
THIRD NETWORK DEVICE
252
FOURTH MESSAGING CLIENT
THIRD MESSAGING CLIENT
250
FIRST NETWORK DEVICE
260
FIRST MESSAGING CLIENT
14
SECOND NETWORK DEVICE
262
SECOND MESSAGING CLIENT
20

FIG. 10

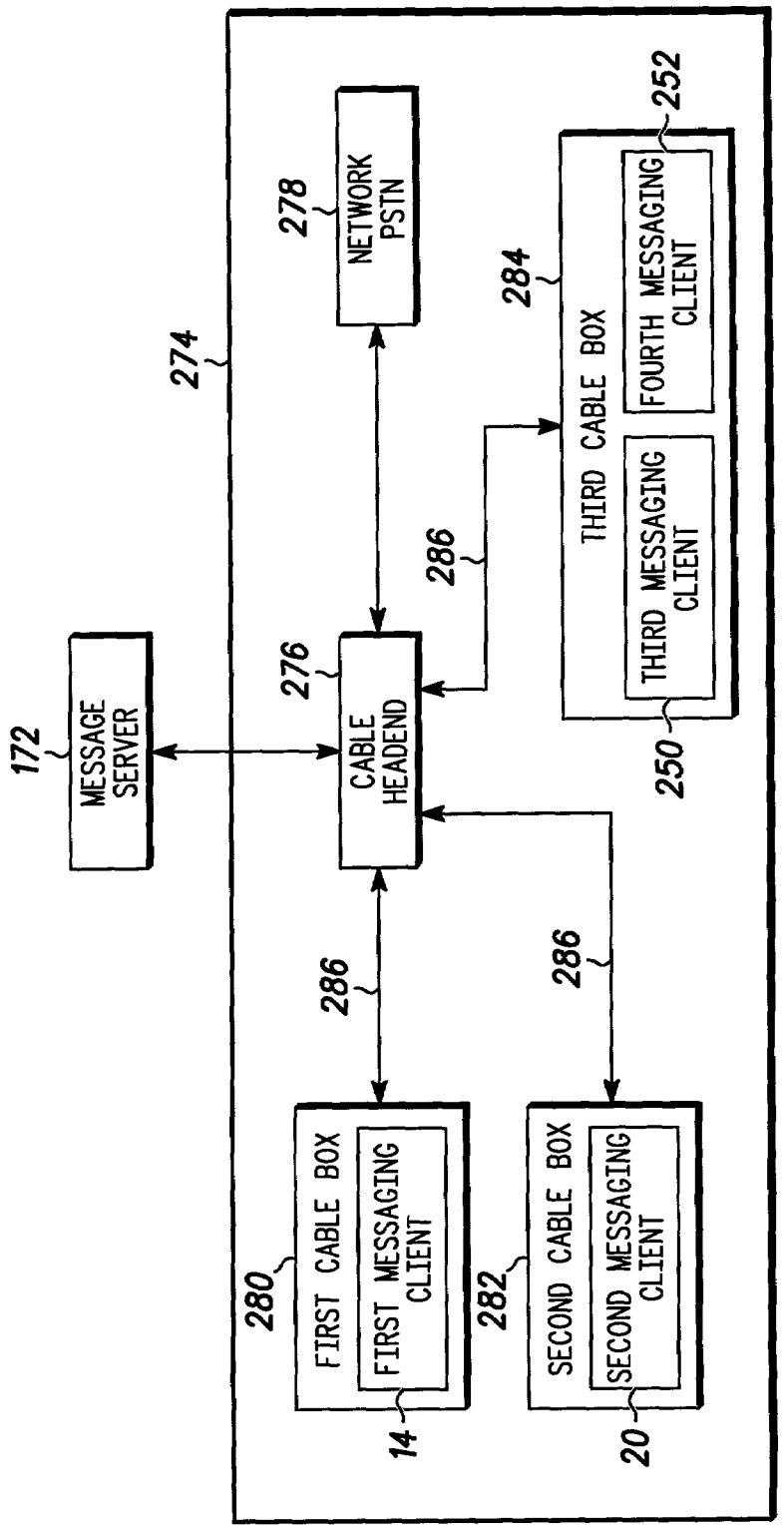

274
278
NETWORK PSTN
284
252
THIRD CABLE BOX
FOURTH MESSAGING CLIENT
THIRD MESSAGING CLIENT
286
172
MESSAGE SERVER
276
CABLE HEADEND
250
286
286
280
FIRST CABLE BOX
FIRST MESSAGING CLIENT
282
SECOND CABLE BOX
SECOND MESSAGING CLIENT
14
20

FIG. 11

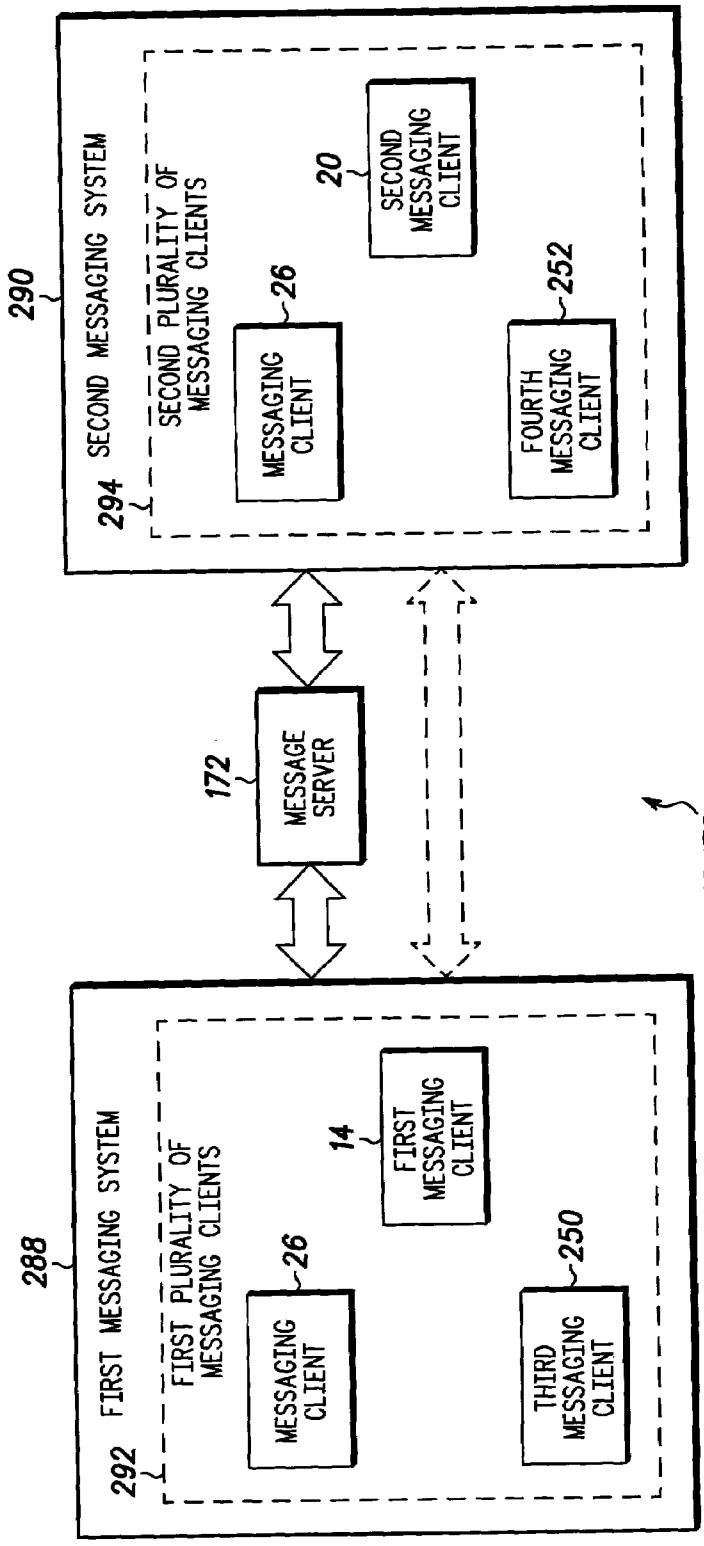

290
SECOND MESSAGING SYSTEM
294
SECOND PLURALITY OF MESSAGING CLIENTS
20
SECOND MESSAGING CLIENT
26
MESSAGING CLIENT
252
FOURTH MESSAGING CLIENT
172
MESSAGE SERVER
10,170
288
FIRST MESSAGING SYSTEM
292
FIRST PLURALITY OF MESSAGING CLIENTS
14
FIRST MESSAGING CLIENT
26
MESSAGING CLIENT
250
THIRD MESSAGING CLIENT

FIG. 12

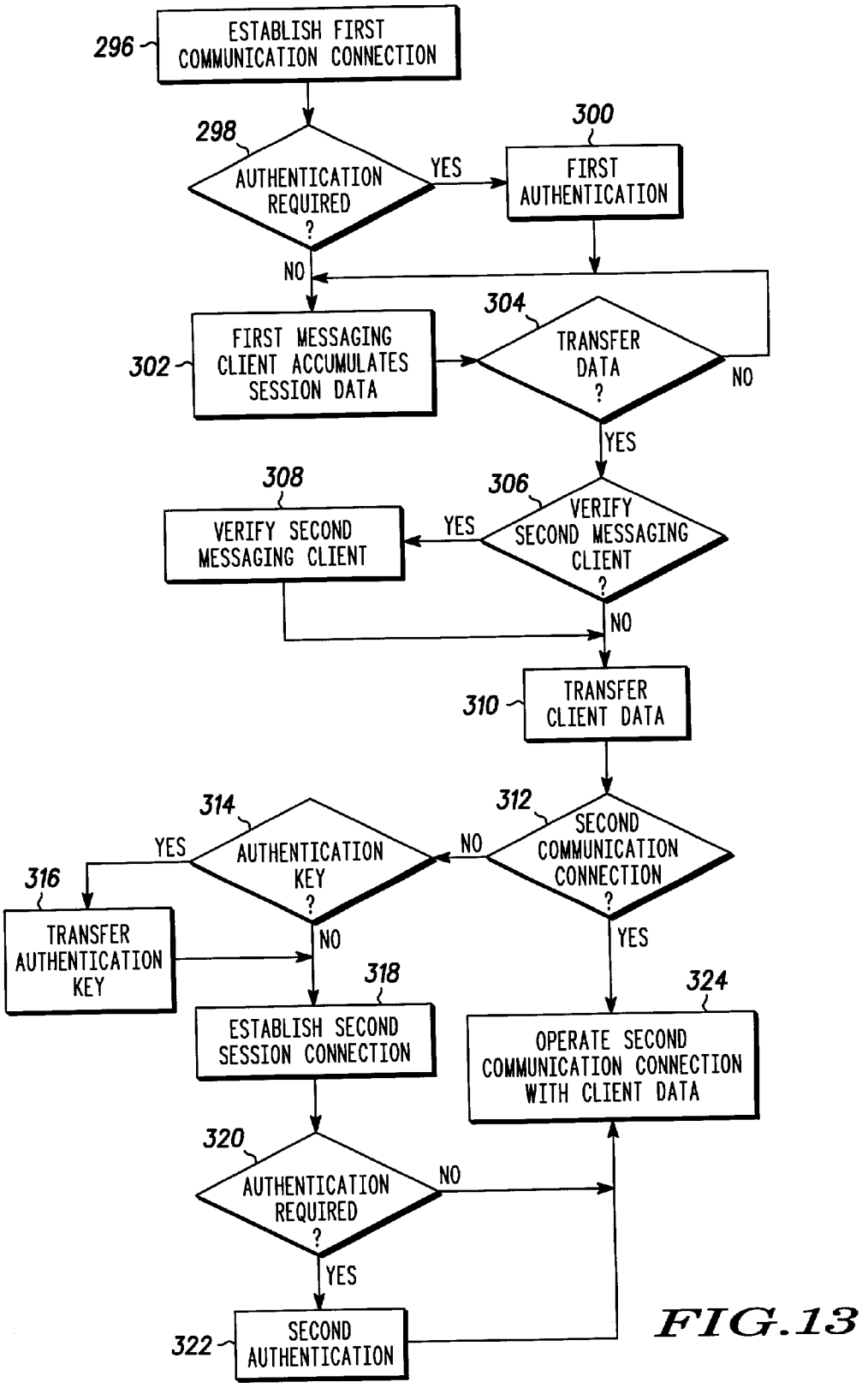

296
ESTABLISH FIRST COMMUNICATION CONNECTION
298
AUTHENTICATION REQUIRED ?
YES
300
FIRST AUTHENTICATION
NO
302
FIRST MESSAGING CLIENT ACCUMULATES SESSION DATA
304
TRANSFER DATA ?
NO
YES
306
VERIFY SECOND MESSAGING CLIENT ?
YES
308
VERIFY SECOND MESSAGING CLIENT
NO
310
TRANSFER CLIENT DATA
312
SECOND COMMUNICATION CONNECTION ?
NO
314
AUTHENTICATION KEY ?
YES
316
TRANSFER AUTHENTICATION KEY
NO
318
ESTABLISH SECOND SESSION CONNECTION
YES
324
OPERATE SECOND COMMUNICATION CONNECTION WITH CLIENT DATA
320
AUTHENTICATION REQUIRED ?
NO
YES
322
SECOND AUTHENTICATION

FIG.13

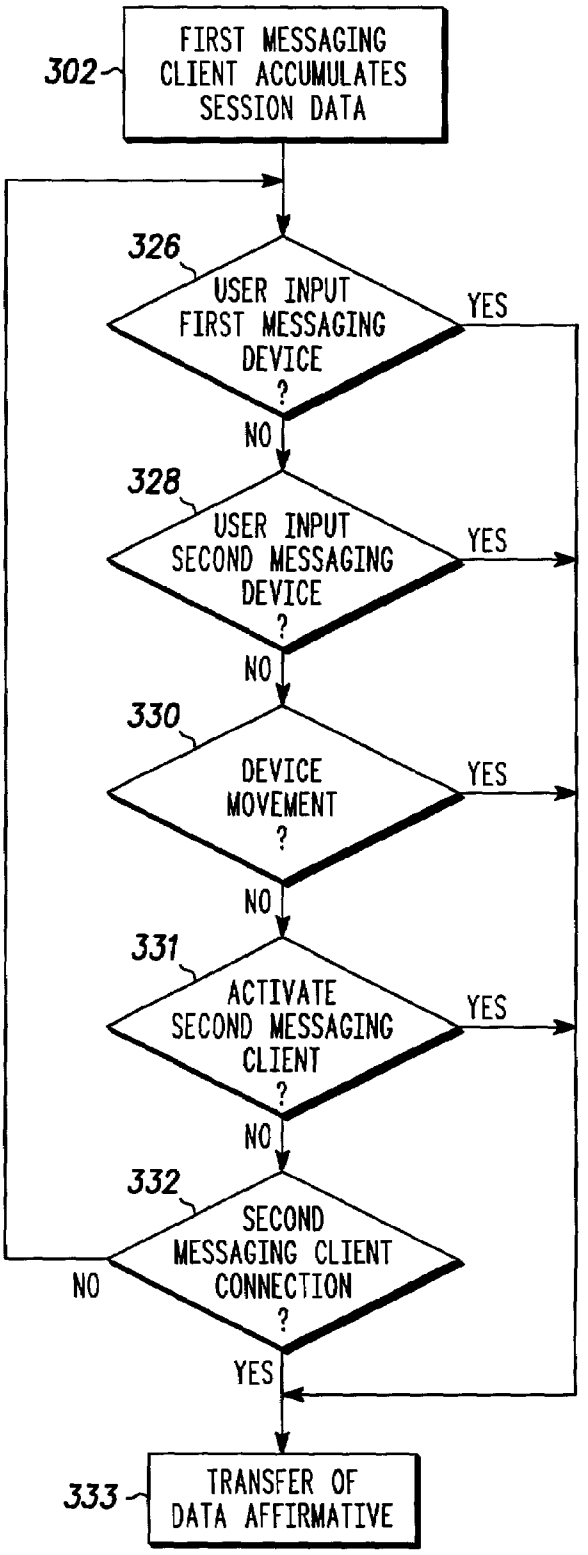

302
FIRST MESSAGING CLIENT ACCUMULATES SESSION DATA
326
USER INPUT FIRST MESSAGING DEVICE ?
YES
NO
328
USER INPUT SECOND MESSAGING DEVICE ?
YES
NO
330
DEVICE MOVEMENT ?
YES
NO
331
ACTIVATE SECOND MESSAGING CLIENT ?
YES
NO
332
SECOND MESSAGING CLIENT CONNECTION ?
NO
YES
333
TRANSFER OF DATA AFFIRMATIVE

FIG. 14

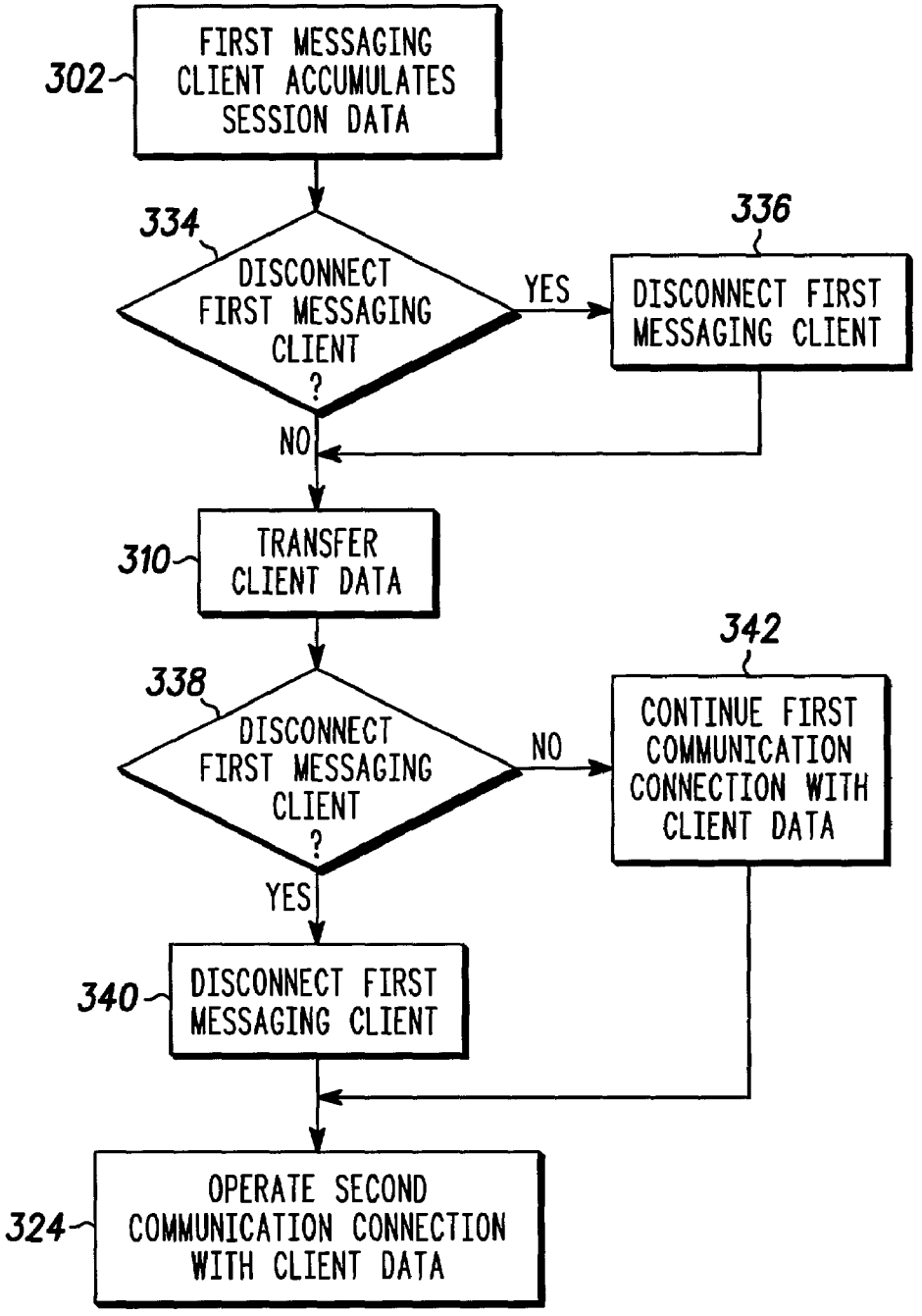

302
FIRST MESSAGING CLIENT ACCUMULATES SESSION DATA
334
DISCONNECT FIRST MESSAGING CLIENT ?
YES
336
DISCONNECT FIRST MESSAGING CLIENT
NO
310
TRANSFER CLIENT DATA
338
DISCONNECT FIRST MESSAGING CLIENT ?
NO
342
CONTINUE FIRST COMMUNICATION CONNECTION WITH CLIENT DATA
YES
340
DISCONNECT FIRST MESSAGING CLIENT
324
OPERATE SECOND COMMUNICATION CONNECTION WITH CLIENT DATA

FIG.15

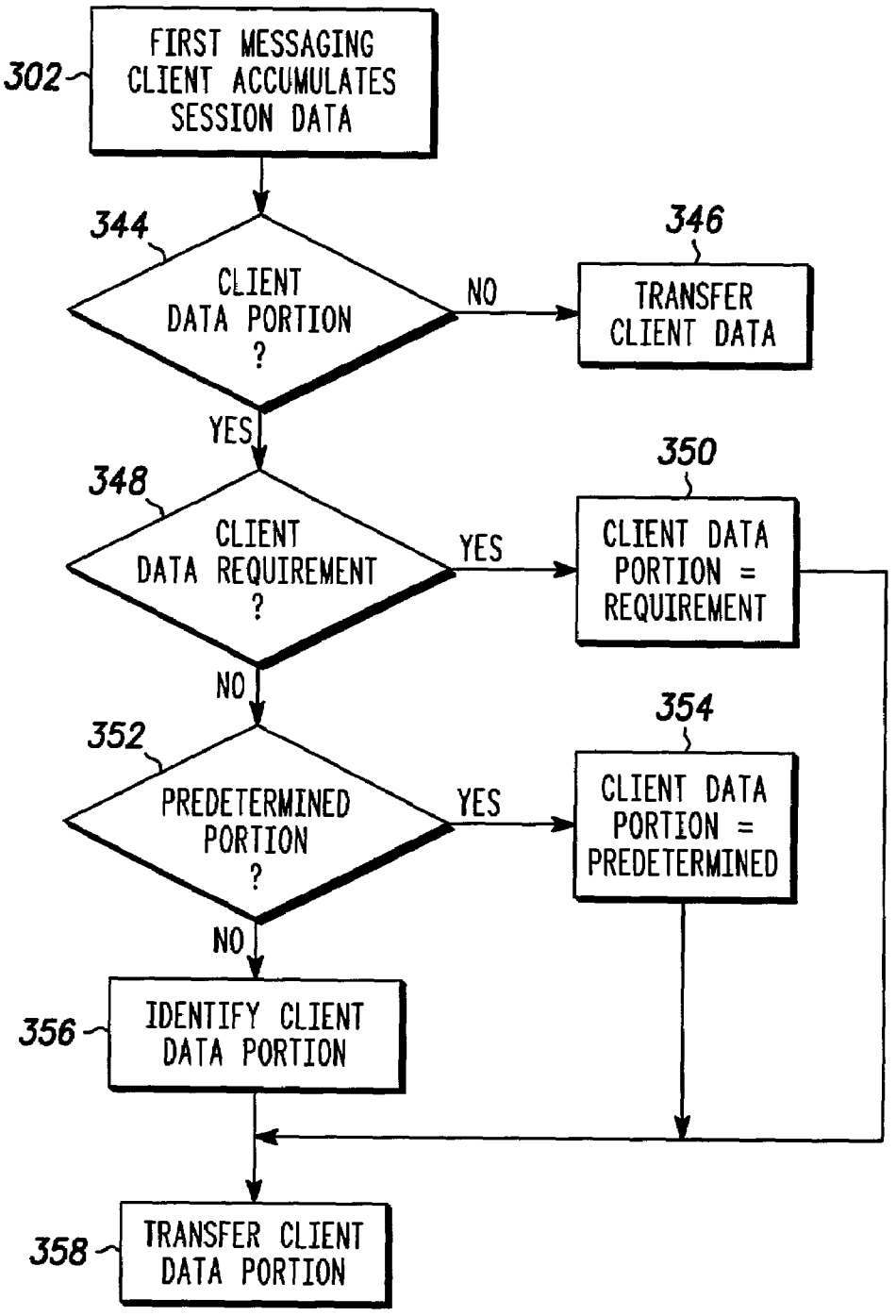

302
FIRST MESSAGING CLIENT ACCUMULATES SESSION DATA
344
CLIENT DATA PORTION ?
NO
346
TRANSFER CLIENT DATA
YES
348
CLIENT DATA REQUIREMENT ?
YES
350
CLIENT DATA PORTION = REQUIREMENT
NO
352
PREDETERMINED PORTION ?
YES
354
CLIENT DATA PORTION = PREDETERMINED
NO
356
IDENTIFY CLIENT DATA PORTION
358
TRANSFER CLIENT DATA PORTION

FIG.16

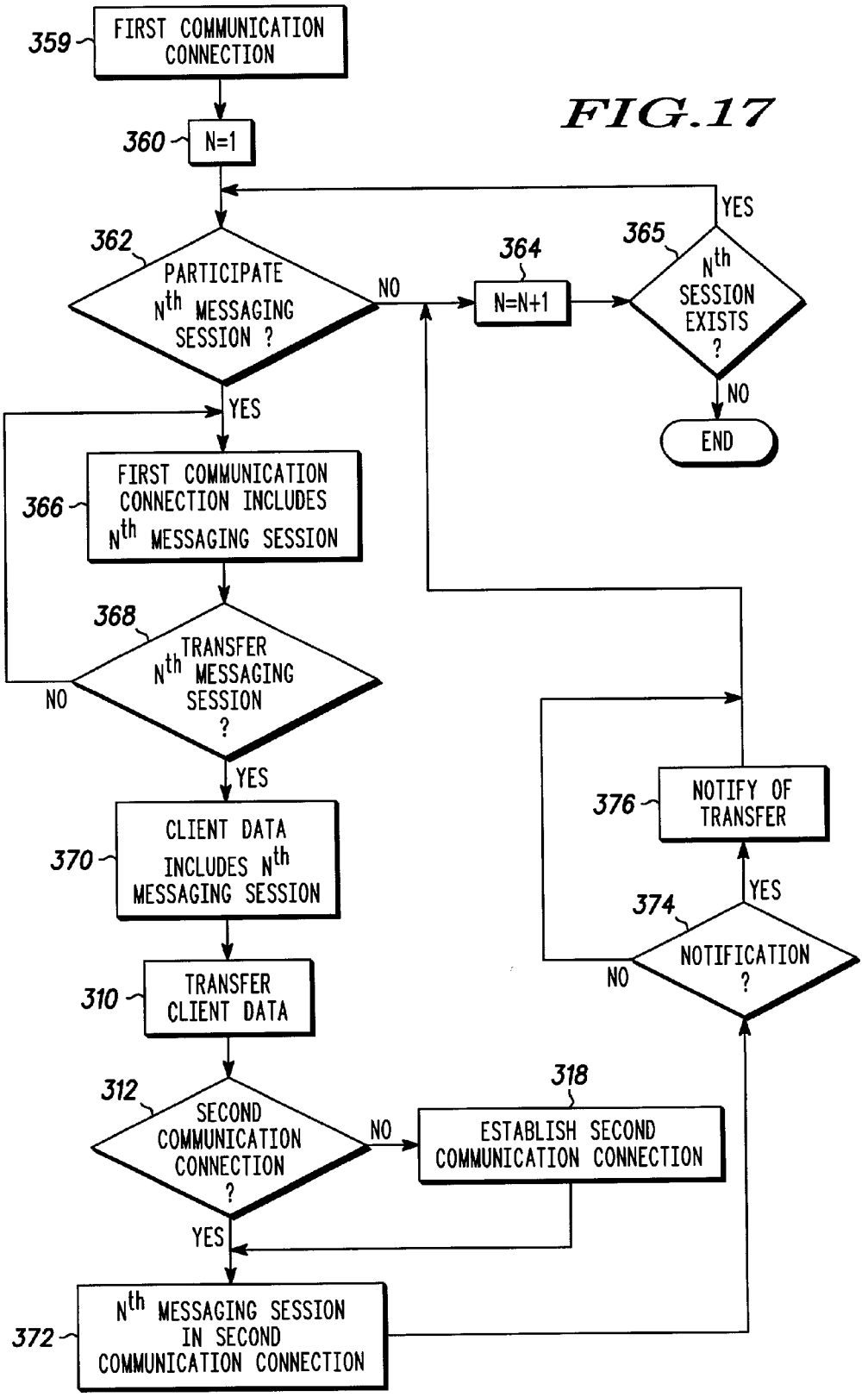

359
FIRST COMMUNICATION CONNECTION
360
N=1
362
PARTICIPATE Nth MESSAGING SESSION ?
NO
364
N=N+1
365
Nth SESSION EXISTS ?
YES
NO
END
YES
366
FIRST COMMUNICATION CONNECTION INCLUDES Nth MESSAGING SESSION
368
TRANSFER Nth MESSAGING SESSION ?
NO
YES
370
CLIENT DATA INCLUDES Nth MESSAGING SESSION
310
TRANSFER CLIENT DATA
312
SECOND COMMUNICATION CONNECTION ?
NO
318
ESTABLISH SECOND COMMUNICATION CONNECTION
YES
372
Nth MESSAGING SESSION IN SECOND COMMUNICATION CONNECTION
376
NOTIFY OF TRANSFER
374
NOTIFICATION ?
YES
NO

FIG.17

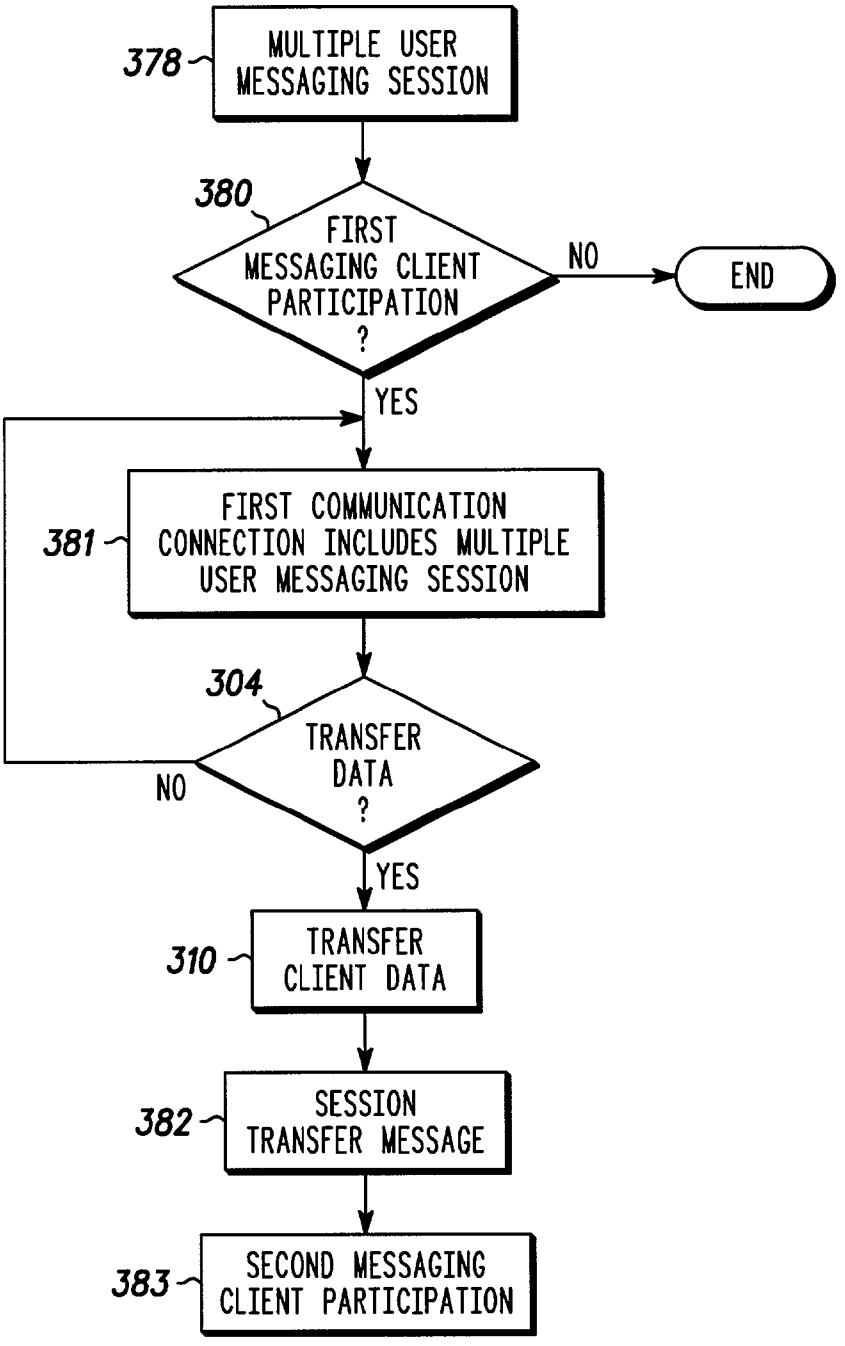

378
MULTIPLE USER MESSAGING SESSION
380
FIRST MESSAGING CLIENT PARTICIPATION ?
NO
END
YES
381
FIRST COMMUNICATION CONNECTION INCLUDES MULTIPLE USER MESSAGING SESSION
304
TRANSFER DATA ?
NO
YES
310
TRANSFER CLIENT DATA
382
SESSION TRANSFER MESSAGE
383
SECOND MESSAGING CLIENT PARTICIPATION

FIG.18

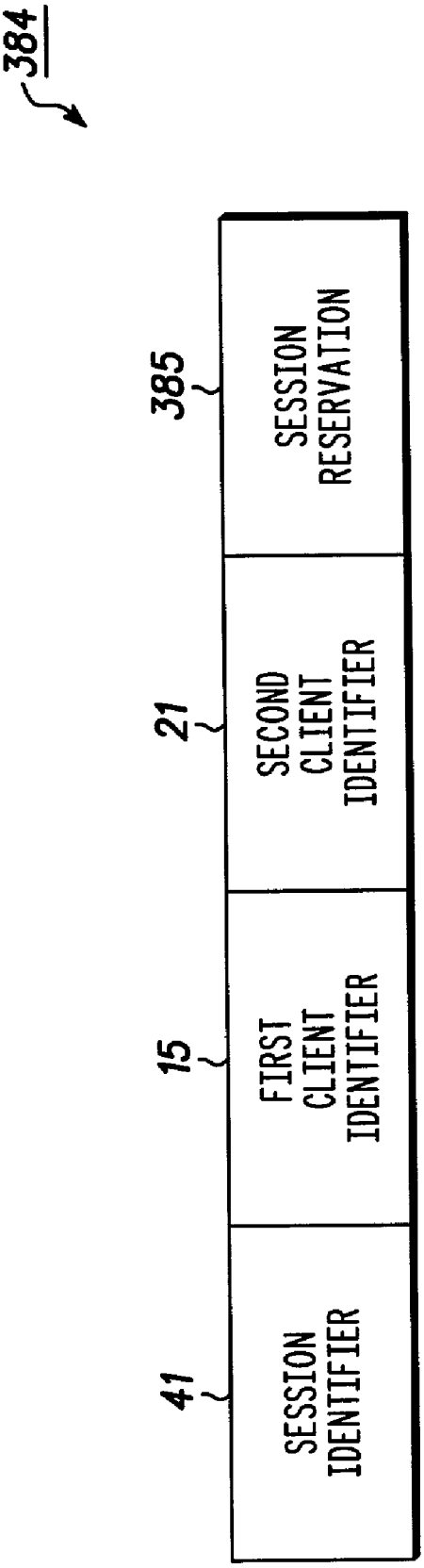

384
385
SESSION RESERVATION
21
SECOND CLIENT IDENTIFIER
15
FIRST CLIENT IDENTIFIER
41
SESSION IDENTIFIER

FIG. 19

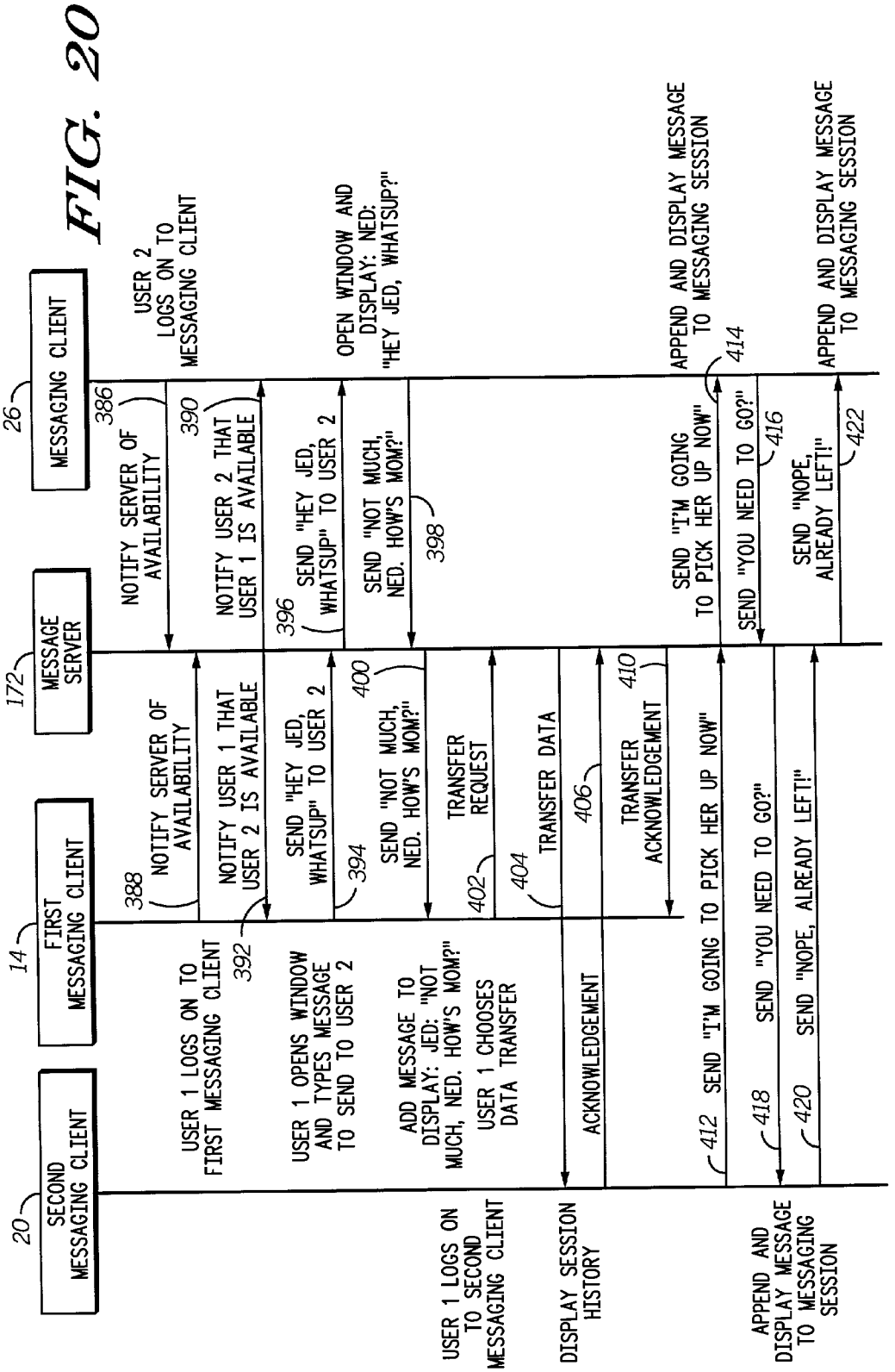

20
SECOND MESSAGING CLIENT
14
FIRST MESSAGING CLIENT
172
MESSAGE SERVER
26
MESSAGING CLIENT
USER 1 LOGS ON TO FIRST MESSAGING CLIENT
392
388
NOTIFY SERVER OF AVAILABILITY
NOTIFY USER 1 THAT USER 2 IS AVAILABLE
386
NOTIFY SERVER OF AVAILABILITY
390
NOTIFY USER 2 THAT USER 1 IS AVAILABLE
USER 2 LOGS ON TO MESSAGING CLIENT
USER 1 OPENS WINDOW AND TYPES MESSAGE TO SEND TO USER 2
394
SEND "HEY JED, WHATSUP" TO USER 2
396
SEND "HEY JED, WHATSUP" TO USER 2
OPEN WINDOW AND DISPLAY: NED: "HEY JED, WHATSUP?"
ADD MESSAGE TO DISPLAY: JED: "NOT MUCH, NED. HOW'S MOM?"
400
SEND "NOT MUCH, NED. HOW'S MOM?"
SEND "NOT MUCH, NED. HOW'S MOM?"
398
USER 1 LOGS ON TO SECOND MESSAGING CLIENT
USER 1 CHOOSES DATA TRANSFER
402
TRANSFER REQUEST
404
TRANSFER DATA
406
ACKNOWLEDGEMENT
DISPLAY SESSION HISTORY
TRANSFER ACKNOWLEDGEMENT
410
SEND "I'M GOING TO PICK HER UP NOW"
APPEND AND DISPLAY MESSAGE TO MESSAGING SESSION
414
412
SEND "I'M GOING TO PICK HER UP NOW"
SEND "YOU NEED TO GO?"
416
418
SEND "YOU NEED TO GO?"
SEND "NOPE, ALREADY LEFT!"
APPEND AND DISPLAY MESSAGE TO MESSAGING SESSION
422
420
SEND "NOPE, ALREADY LEFT!"

FIG. 20

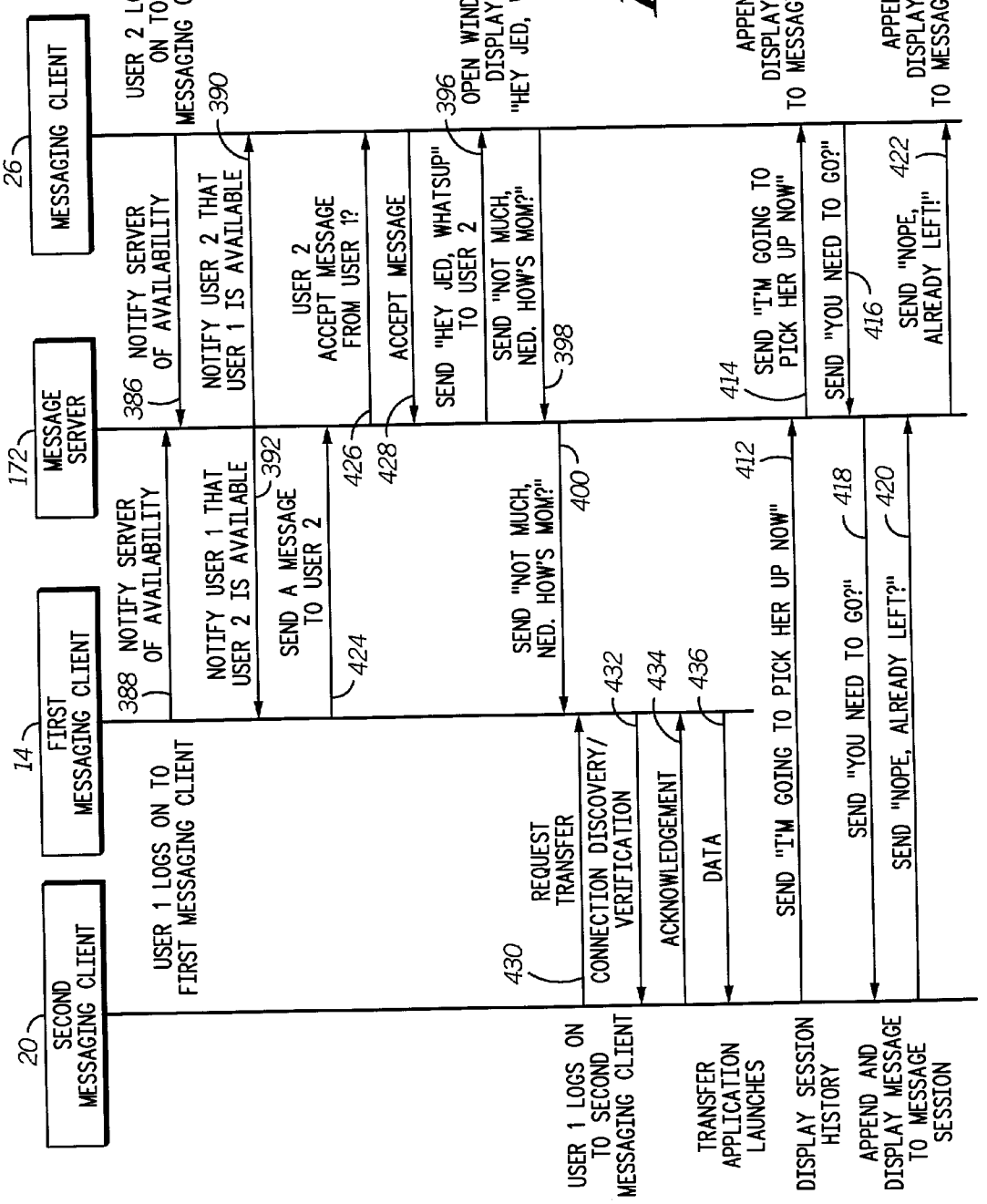

20
SECOND MESSAGING CLIENT
14
FIRST MESSAGING CLIENT
172
MESSAGE SERVER
26
MESSAGING CLIENT
USER 1 LOGS ON TO FIRST MESSAGING CLIENT
388 NOTIFY SERVER OF AVAILABILITY
NOTIFY USER 1 THAT USER 2 IS AVAILABLE
386 NOTIFY SERVER OF AVAILABILITY
NOTIFY USER 2 THAT USER 1 IS AVAILABLE
USER 2 LOGS ON TO MESSAGING CLIENT
390
392
SEND A MESSAGE TO USER 2
424
USER 2 ACCEPT MESSAGE FROM USER 1?
426
ACCEPT MESSAGE
428
SEND "HEY JED, WHATSUP" TO USER 2
396
OPEN WINDOW AND DISPLAY: NED: "HEY JED, WHATSUP"
SEND "NOT MUCH, NED. HOW'S MOM?"
398
SEND "NOT MUCH, NED. HOW'S MOM?"
400
APPEND AND DISPLAY MESSAGE TO MESSAGING SESSION
414
SEND "I'M GOING TO PICK HER UP NOW"
412
SEND "YOU NEED TO GO?"
416
SEND "YOU NEED TO GO?"
418
APPEND AND DISPLAY MESSAGE TO MESSAGING SESSION
422
SEND "NOPE, ALREADY LEFT!"
SEND "NOPE, ALREADY LEFT?"
420
430
REQUEST TRANSFER
CONNECTION DISCOVERY/ VERIFICATION
432
ACKNOWLEDGEMENT
434
DATA
436
SEND "I'M GOING TO PICK HER UP NOW"
USER 1 LOGS ON TO SECOND MESSAGING CLIENT
TRANSFER APPLICATION LAUNCHES
DISPLAY SESSION HISTORY
APPEND AND DISPLAY MESSAGE TO MESSAGE SESSION

FIG. 21

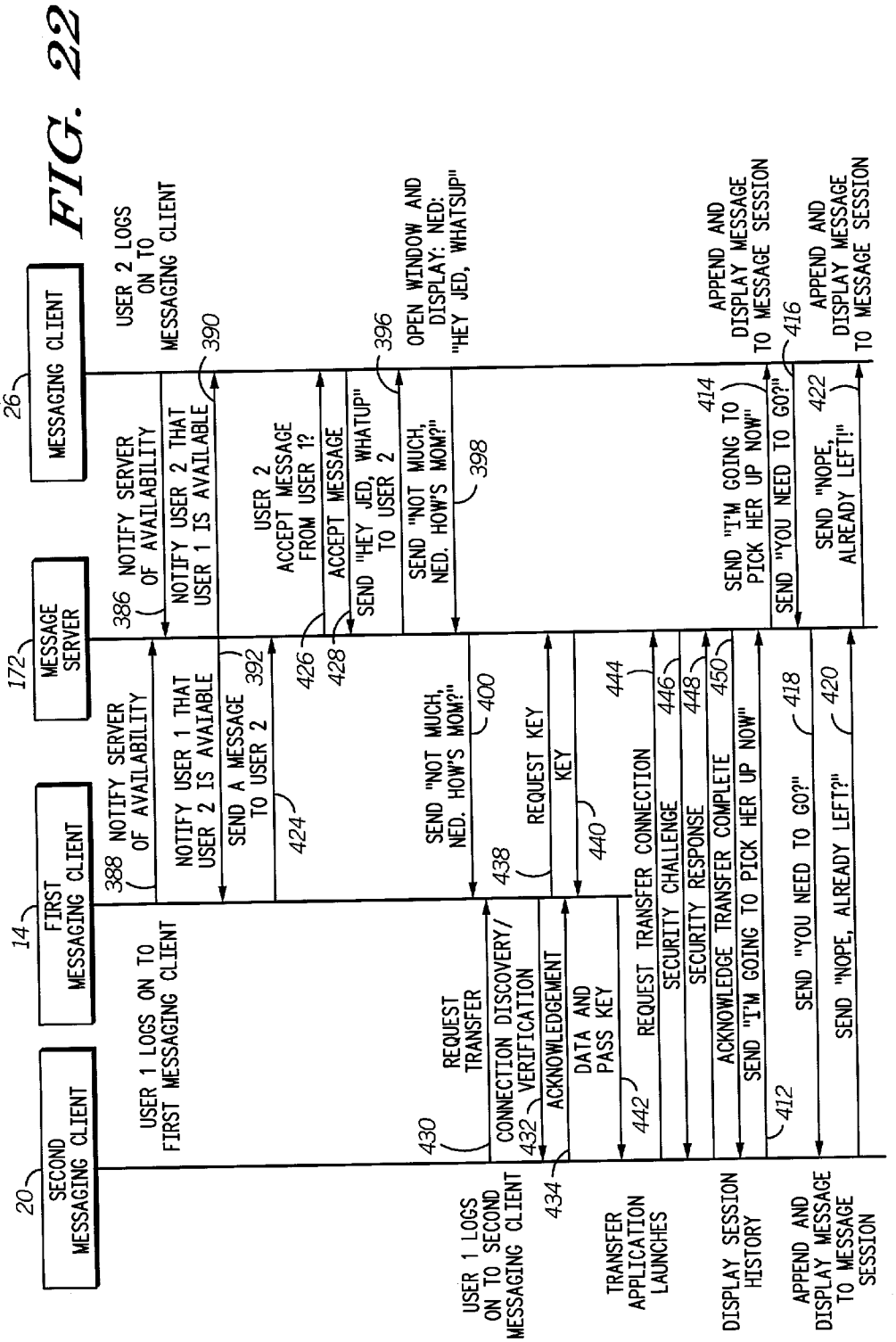
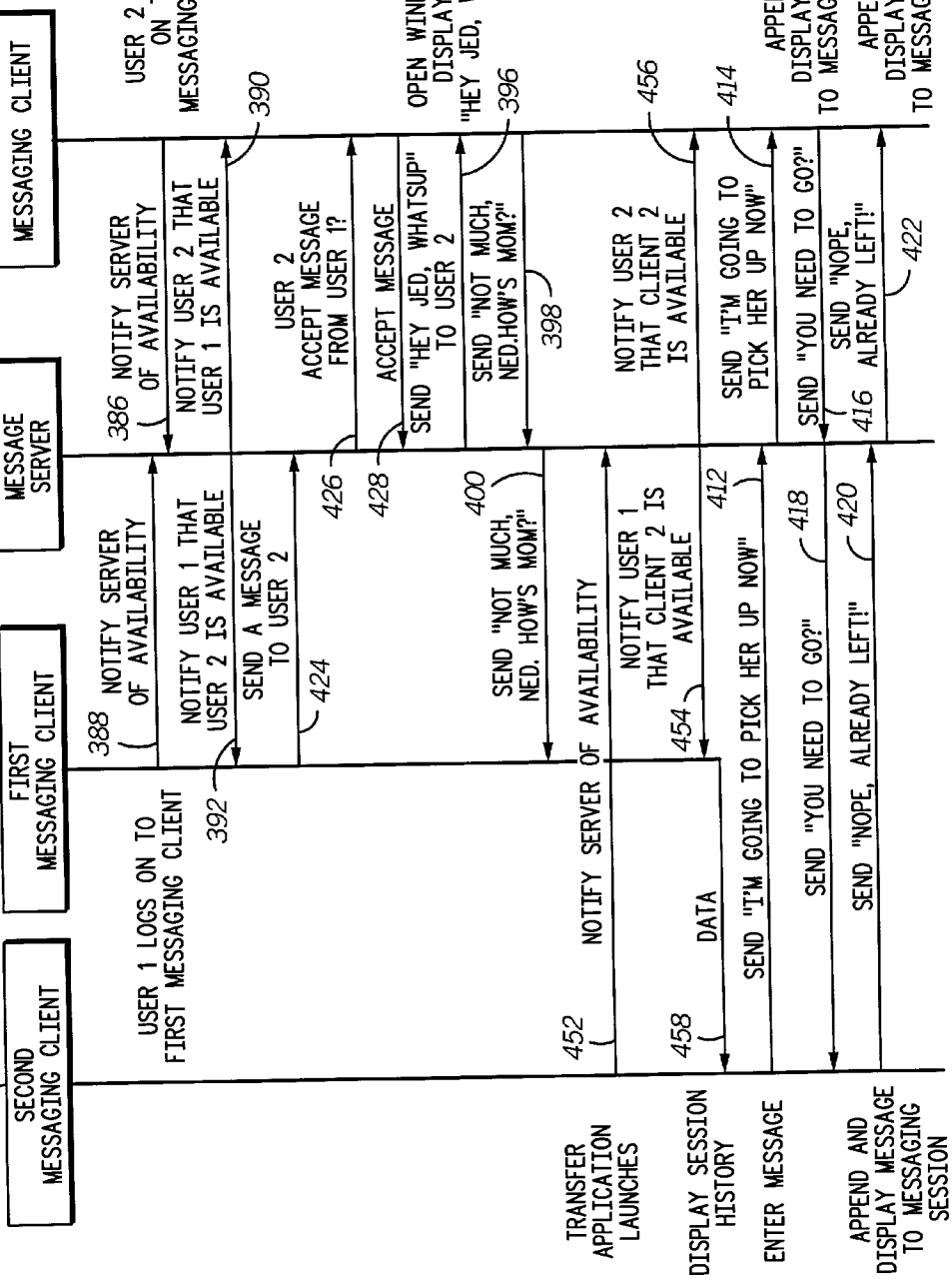

20
SECOND MESSAGING CLIENT
14
FIRST MESSAGING CLIENT
172
MESSAGE SERVER
26
MESSAGING CLIENT
USER 1 LOGS ON TO FIRST MESSAGING CLIENT
388
NOTIFY SERVER OF AVAILABILITY
NOTIFY USER 1 THAT USER 2 IS AVAIABLE
386
NOTIFY SERVER OF AVAILABILITY
NOTIFY USER 2 THAT USER 1 IS AVAILABLE
390
USER 2 LOGS ON TO MESSAGING CLIENT
SEND A MESSAGE TO USER 2
392
424
USER 2 ACCEPT MESSAGE FROM USER 1?
426
ACCEPT MESSAGE
428
SEND "HEY JED, WHATUP" TO USER 2
396
OPEN WINDOW AND DISPLAY: NED: "HEY JED, WHATSUP"
SEND "NOT MUCH, NED. HOW'S MOM?"
400
398
SEND "NOT MUCH, NED. HOW'S MOM?"
438
REQUEST KEY
KEY
440
444
REQUEST TRANSFER
430
CONNECTION DISCOVERY/ VERIFICATION
432
ACKNOWLEDGEMENT
434
DATA AND PASS KEY
442
REQUEST TRANSFER CONNECTION
446
SECURITY CHALLENGE
448
SECURITY RESPONSE
450
ACKNOWLEDGE TRANSFER COMPLETE
SEND "I'M GOING TO PICK HER UP NOW"
412
SEND "YOU NEED TO GO?"
418
SEND "NOPE, ALREADY LEFT?"
420
414
SEND "I'M GOING TO PICK HER UP NOW"
SEND "YOU NEED TO GO?"
416
APPEND AND DISPLAY MESSAGE TO MESSAGE SESSION
422
SEND "NOPE, ALREADY LEFT!"
APPEND AND DISPLAY MESSAGE TO MESSAGE SESSION
USER 1 LOGS ON TO SECOND MESSAGING CLIENT
TRANSFER APPLICATION LAUNCHES
DISPLAY SESSION HISTORY
APPEND AND DISPLAY MESSAGE TO MESSAGE SESSION

FIG. 22

FIG. 23


20
SECOND MESSAGING CLIENT
14
FIRST MESSAGING CLIENT
172
MESSAGE SERVER
26
MESSAGING CLIENT
USER 1 LOGS ON TO FIRST MESSAGING CLIENT
392
388 NOTIFY SERVER OF AVAILABILITY
NOTIFY USER 1 THAT USER 2 IS AVAILABLE
SEND A MESSAGE TO USER 2
424
386 NOTIFY SERVER OF AVAILABILITY
NOTIFY USER 2 THAT USER 1 IS AVAILABLE
USER 2 LOGS ON TO MESSAGING CLIENT
390
426
USER 2 ACCEPT MESSAGE FROM USER 1?
428
ACCEPT MESSAGE
SEND "HEY JED, WHATSUP" TO USER 2
OPEN WINDOW AND DISPLAY: NED: "HEY JED, WHATSUP?"
396
400
SEND "NOT MUCH, NED. HOW'S MOM?"
SEND "NOT MUCH, NED.HOW'S MOM?"
398
452
NOTIFY SERVER OF AVAILABILITY
NOTIFY USER 2 THAT CLIENT 2 IS AVAILABLE
456
TRANSFER APPLICATION LAUNCHES
458
DATA
454
NOTIFY USER 1 THAT CLIENT 2 IS AVAILABLE
412
SEND "I'M GOING TO PICK HER UP NOW"
SEND "I'M GOING TO PICK HER UP NOW"
414
DISPLAY SESSION HISTORY
ENTER MESSAGE
SEND "YOU NEED TO GO?"
418
SEND "YOU NEED TO GO?"
APPEND AND DISPLAY MESSAGE TO MESSAGING SESSION
APPEND AND DISPLAY MESSAGE TO MESSAGING SESSION
SEND "NOPE, ALREADY LEFT!"
420
416
SEND "NOPE, ALREADY LEFT!"
422

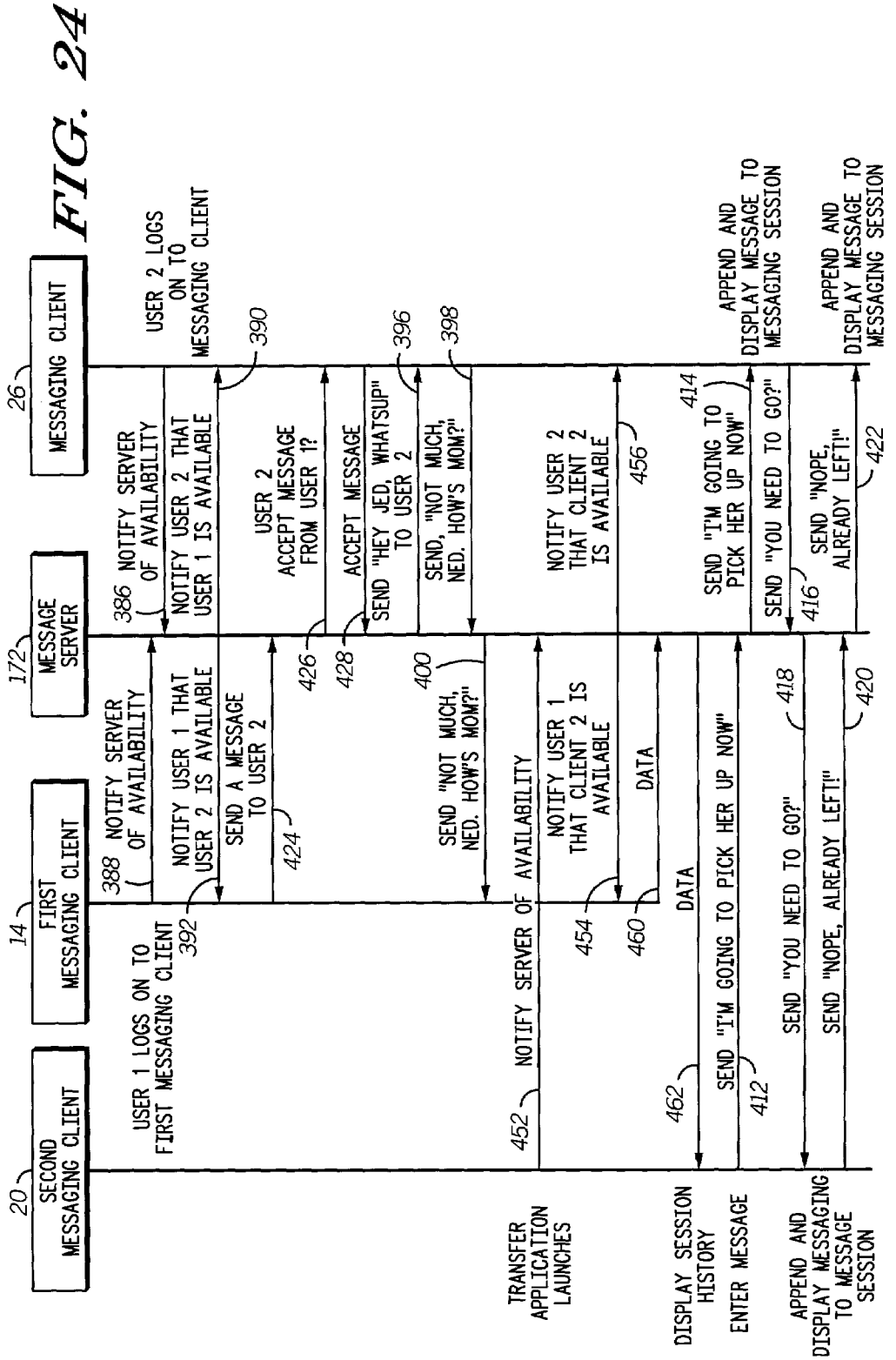

20
SECOND MESSAGING CLIENT
14
FIRST MESSAGING CLIENT
172
MESSAGE SERVER
26
MESSAGING CLIENT
USER 1 LOGS ON TO FIRST MESSAGING CLIENT
392
388 NOTIFY SERVER OF AVAILABILITY
NOTIFY USER 1 THAT USER 2 IS AVAILABLE
SEND A MESSAGE TO USER 2
424
426
428
400
SEND "NOT MUCH, NED. HOW'S MOM?"
386 NOTIFY SERVER OF AVAILABILITY
NOTIFY USER 2 THAT USER 1 IS AVAILABLE
USER 2 LOGS ON TO MESSAGING CLIENT
390
USER 2 ACCEPT MESSAGE FROM USER 1?
ACCEPT MESSAGE
SEND "HEY JED, WHATSUP" TO USER 2
396
SEND, "NOT MUCH, NED. HOW'S MOM?"
398
APPEND AND DISPLAY MESSAGE TO MESSAGING SESSION
TRANSFER APPLICATION LAUNCHES
452 NOTIFY SERVER OF AVAILABILITY
NOTIFY USER 1 THAT CLIENT 2 IS AVAILABLE
454
DATA
460
NOTIFY USER 2 THAT CLIENT 2 IS AVAILABLE
456
DISPLAY SESSION HISTORY
462
DATA
ENTER MESSAGE
SEND "I'M GOING TO PICK HER UP NOW"
412
414
SEND "I'M GOING TO PICK HER UP NOW"
APPEND AND DISPLAY MESSAGING TO MESSAGE SESSION
SEND "YOU NEED TO GO?"
418
416
SEND "YOU NEED TO GO?"
SEND "NOPE, ALREADY LEFT!"
420
SEND "NOPE, ALREADY LEFT!"
422
APPEND AND DISPLAY MESSAGE TO MESSAGING SESSION

FIG. 24

SYSTEM FOR PROVIDING CONTINUITY BETWEEN MESSAGING CLIENTS AND METHOD THEREFOR

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates in general to communication systems and in particular to communication systems incorporating capabilities to provide continuity between messaging clients.

2. Description of the Related Art

With the proliferation of the Internet, the way people communicate is changing. Electronic communication such as electronic mail (email), and real time electronic messaging (e.g. instant messaging and chat messaging) is quickly replacing traditional telephonic communication and handwritten letters. Real time electronic messaging allows the simultaneous access to a message or a plurality of messages by multiple account users, with each account user capable of inputting a message or a plurality of messages to a messaging session. Each inputted message is relayed to messaging clients operating on messaging devices (such as computers) of the other account users who have chosen to participate in that messaging session. The other session participants can respond with their own messages, which are likewise relayed to all the participating devices. These messages are typically text messages that are delivered to the intended recipient(s) of the message in a real time manner. However, these messages may be in a variety of different media formats or combinations thereof such as audio, animation, video, images, etc. A session history of the messages received and transmitted by all participants involved in the messaging session is typically maintained on the individual participants' devices and typically presented on the screen of the respective device in the form of a scrolling dialog. This text history constitutes one of the attributes of the look and feel of the real time electronic messaging experience.

There are currently several standard methods available for account users to participate in messaging communications. For example, personal instant messaging typically occurs between two individuals. An individual can establish a private chat room so that multiple users can participate in a common on-line conversation. Participants gain access to the private chat room by accepting an invitation from the creator of the private chat room. Non-restricted public chat rooms are available to anyone interested in the topic being discussed by simply selecting the desired chat room descriptor on the account user's device. In addition to the non-restricted public chat rooms there are limited access public chat rooms. An example of one such restriction is the limiting of the number of account users allowed to participate in the chat room. Electronic gaming is available to participants who register and login to join at least one other participant in playing one or more games. Communication of participants' "moves" are made through electronic message communications in messaging sessions.

Each messaging session can have various types of session attributes such as session type, session connection info, participants, filter settings, colors, relative font sizes, etc. The account user can also have various types of preferences such as colors, relative font sizes, buddy lists, nicknames, and parental control settings. These are only a few examples of the attributes and preferences that can effect a messaging session.

Several different devices, such as personal computers, interactive broadcast receivers and mobile communication devices, can be utilized to participate in messaging communications. Although these different devices can be utilized to participate in a messaging communication they all do not have the same set of features and capabilities. For example, display size, support for different media types, and cost of sending messages can all be different for different devices. Some messaging service providers maintain the same user preference settings such as screen names, buddy list groups, electronic mailboxes, and parental control settings regardless of which device is used to access the service. By having this capability the service providers are providing continuity of user preferences from device to device, which simplifies the use of multiple devices in the messaging system.

In order to utilize real time electronic messaging, a device is pre-configured with, or the user installs, application client software distributed by a particular service provider. The client software connects to a message server for communication. To access the message server, via a logon procedure, the client sends the routing information (e.g. IP address and number of the port assigned to the client, mobile phone number) of the device being used, the account user's username, the account user's password, and the account user's current availability setting to the message server. The message server temporarily stores the routing and availability information for the account user's device. In response to the account user's login information (username and password), the message server provides the messaging client with the account user's contact list. The message server then determines the presence and availability of the account users in that contact list. If the message server finds any of the contacts logged in (i.e. presence setting is online), it sends a message back to the messaging client on the account user's device with the presence and availability information for that account user. The message server also sends the account user's presence and availability information to the people that have the account user in their contact list. The account user can click on the name of a person in his/her contact list who is online, and a window is created in which the account user can enter a message. The account user enters a message and clicks "send" to communicate with that person. The other person gets the message and can respond. Messages between account users may be addressed directly to the account user's device or may be addressed to the username and sent via the message server.

The window that the session participants see on their respective messaging devices typically includes a scrolling dialog of the session history. Each participant's messages appear in this window on all participating devices. Messages can have different attributes such as message formatting, sender identification, timestamps and others. For example, messages related to a particular electronic game could contain graphics enhancing the "look and feel" of the electronic game for the participants.

When the messaging session is complete, the account users close the message window for that messaging session. When the account user signs off, his/her messaging client sends a message to the message server to terminate the account user's participation in the plurality of messaging sessions. The message server then sends an update of the account user's presence and availability information to the people that have the account user in their contact list to indicate the account user has signed off. Finally, the message server discards the routing and availability information for the account user's device.

Some messaging services support access of a single account from multiple devices. Further, some messaging services also support simultaneous login of devices on the

same account. Still further, some messaging services utilize a resource extension to describe the device that is being utilized to communicate. For example an account user logging in with a mobile device can choose to use "mobile device" as their resource extension while logging into the same account from the home personal computer may utilize a resource extension of "home computer".

When using messaging services that allow access from multiple devices, an account user can log on with a first messaging device and engage in conversations with other account users and later log on with a second messaging device. For example, users of mobile devices would typically benefit if a messaging session in progress on a fixed network device could be continued on a mobile device. This would allow the account user to continue the messaging session when the account user is no longer in proximity to the fixed network device. In addition the account user would benefit if a messaging session that was in progress on a mobile device could be continued on a fixed network device that may have a superior user interface.

In order to switch to a different device with existing technology, the account user may have to cause the currently connected device to disconnect from the message server. The account user would then have to cause the second device to connect to the message server and login. Finally, the account user would have to re-initiate each messaging session (one-to-one, public chat, private chat, electronic game) that was in progress on the first device. The disadvantage of this method is the numerous manual operations required of the account user to change devices. A further disadvantage is the lack of messaging session continuity. For example, the second device will not have the session history that was available on the first device, and the second device may not be able to re-connect to chat rooms that restrict the number of active account users since another account user may have connected to the chat room after the account user's first device disconnected.

What is needed is a system and method for maintaining continuity between messaging clients.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention will be described by way of exemplary embodiments, but not limitations, illustrated in the accompanying drawings in which like references denote similar elements, and in which:

FIG. **1** is an electronic block diagram of a messaging communication system, in accordance with the preferred embodiment of the present invention;

FIG. **2** illustrates client data for use within the messaging communication system of FIG. **1**, in accordance with the preferred embodiment of the present invention;

FIGS. **3** and **4** illustrate more detail of the client data of FIG. **2**, in accordance with the preferred embodiment of the present invention;

FIGS. **5**, **6**, and **7** are electronic block diagrams of various embodiments of a messaging device in which a messaging client of FIG. **1** operates;

FIG. **8** is an electronic block diagram of an alternate embodiment of a messaging communication system, in accordance with the preferred embodiment of the present invention;

FIGS. **9** to **12** are electronic block diagrams of various embodiments of the messaging communication system of FIGS. **1** and **8**, in accordance with the preferred embodiment of the present invention;

FIGS. **13** to **18** are flowcharts illustrating the operation of the messaging communication system of FIGS. **1** and **8**, in accordance with the preferred embodiment of the present invention;

FIG. **19** illustrates a message for use within the messaging communication system of FIGS. **1** and **8**, in accordance with the preferred embodiment of the present invention; and

FIGS. **20** to **24** are signaling flow diagrams illustrating the interaction between the elements of the messaging communication system of FIGS. **1** and **8**, in accordance with the preferred embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT(S)

As required, detailed embodiments of the present invention are disclosed herein; however, it is to be understood that the disclosed embodiments are merely exemplary of the invention, which may be embodied in various forms. Therefore, specific structural and functional details disclosed herein are not to be interpreted as limiting, but merely as a basis for the claims and as a representative basis for teaching one skilled in the art to variously employ the present invention in virtually any appropriately detailed structure. Further, the terms and phrases used herein are not intended to be limiting; but rather, to provide an understandable description of the invention.

Referring to FIG. **1**, an electronic block diagram of a messaging communication system **10** in accordance with the present invention is shown. As illustrated, the messaging communication system **10** preferably includes a plurality of messaging clients **12** for participation within a plurality of messaging sessions **24**. As illustrated, the plurality of messaging sessions **24** can include a messaging session **40** and/or a multiple user messaging session **19**. The plurality of messaging sessions **24**, for example, can include the communication of a plurality of electronic messages such as chat sessions, instant message sessions, and electronic mail, facilitating substantially real time communication among the plurality of messaging clients **12**. Similarly, the plurality of messaging sessions **24** can include communication of gaming messages for one or more gaming sessions (e.g. battleship, checkers, chess, tic tac toe and doom). It will be appreciated by one of ordinary skill in the art that the plurality of messaging sessions **24** can include any of the messaging sessions mentioned herein or an equivalent. Each of the plurality of messaging clients **12** such as a first messaging client **14** and a second messaging client **20** includes client software to interface within the messaging communication system **10**. The client software, for example, can include a software application for communication through an Internet service provider. Further, the client software can include a software application for participation in one or more electronic games offered by a gaming software provider. It will be appreciated by one of ordinary skill in the art that the client software can be any of those mentioned herein or an equivalent. Further, it will be appreciated by one of ordinary skill in the art that in accordance with the present invention, the interface capabilities of the client software can also be designed into client hardware of a messaging client. Each messaging client **26** of the plurality of messaging clients **12** further includes a client identifier **27**. For example, the first messaging client **14** includes a first client identifier **15** and the second messaging client **20** includes a second client identifier **21**. The client identifier **27** of the messaging client **26** is a unique identification within the messaging communication system **10** for directing mes-

sages to a particular messaging client. For example, the client identifier **27** can be an address of a mobile device or an IP address and number of the port of a fixed network device. To communicate within at least one of the plurality of messaging sessions **24** the messaging client **26** establishes a communication connection **28**. For example, the first messaging client **14** establishes a first communication connection **16** for communication within at least one of the plurality of messaging sessions **24**. Similarly, the second messaging client **20** establishes a second communication connection **22** for communication within at least one of the plurality of messaging sessions **24**. It will be appreciated by one of ordinary skill in the art that the communication connection **28**, the first communication connection **16**, and the second communication connection **22** can be a physical connection, or alternatively can be a logical connection where the act of connecting and disconnecting is a logical one. Each messaging client **26** of the plurality of messaging clients **12** is operated by at least one account user **30**. The account user **30** is an individual who uses one or more messaging clients to communicate with other account users within the plurality of messaging sessions **24**. It will be appreciated by one of ordinary skill in the art that the account user **30** can communicate using one or more messaging clients. For example, a first account user **29** can establish communication within the plurality of messaging sessions **24** using the first messaging client **14**, and, in accordance with the present invention, also using the second messaging client **20**.

Each messaging client **26** preferably includes a plurality of client data **25**. The plurality of client data **25** includes data associated with the messaging client **26** and data associated with each messaging session for which the messaging client **26** is currently participating, has participated in, or plans to participate in. The plurality of client data **25** can be divided up into one or more client data portions **18** as illustrated in FIG. **2**. The first messaging client **14** includes a first client data **17** and the second messaging client **20** includes a second client data **23**. FIG. **2** illustrates the plurality of client data **25** included within the messaging client **26** of FIG. **1**. It will be appreciated by one of ordinary skill in the art that the plurality of client data **25** as illustrated in FIG. **2** can be the first client data **17** or the second client data **23**. As illustrated, the plurality of client data **25** preferably includes a client version identifier (not shown), an account identifier **31**, a server identifier **32**, an authentication key **33**, a plurality of contact data **34**, a plurality of user preferences **35**, and a plurality of session data **36**. It will be appreciated by one of ordinary skill in the art that the plurality of client data **25** can include any of the client data mentioned herein or an equivalent.

The client version identifier is preferably the name and version or other similar indication of the messaging client being used. The account identifier **31** is preferably a user name or other identification of the account user **30** currently using the messaging client **26**. In an embodiment of the present invention in which a message server is utilized to manage the plurality of messaging sessions **24**, the server identifier **32** identifies the message server. For example, the server identifier **32** can be a wireless address, an IP (internet protocol) address, or an IP address accompanied by a number of the port assigned to the message server. The authentication key **33** preferably includes a code that is used to authenticate the account user **30** to the messaging communication system **10**. For example, the authentication key **33** could be derived from a password known only to the account user **30** and the messaging communication system **10**. The plurality of contact data **34** is a listing of information relating to the various account users in which the account user **30** currently using the messaging client **26** communicates, plans to communicate, or has communicated with in the past. For example, the plurality of contact data **34** can be a "buddy list" for the account user **30**. The plurality of contact data **34** preferably includes a plurality of account information **37** for each of a plurality of accounts. For example, the plurality of account information **37** for an Nth account can include an Nth account identifier **38** and further can include an Nth account contact information **39** associated with the Nth account identifier **38**. The Nth account contact information **39**, for example, can include Nth account user presence, Nth account user availability, Nth account phone number, Nth account mailing address, or Nth account user preferred communication means. It will be appreciated by one of ordinary skill in the art that the Nth account contact information can be any of the contact information mentioned herein or an equivalent. The plurality of account information **37** can further include, for example, billing information, favorite topics, associates, group lists, age, obscenity rating, and optional services. It will be appreciated by one of ordinary skill in the art that the plurality of account information **37** can include any of the information mentioned herein or an equivalent. In one embodiment of the present invention, each of the plurality of messaging clients **12** of FIG. **1** operates using at least one account. Further, each account user can have one or more accounts. For example, the account user **30** can have a business account and a personal account both operated using the messaging client **26**.

The plurality of user preferences **35** defines certain attributes settable by the account user **30** for communicating within the plurality of messaging sessions **24** using the messaging client **26**. The plurality of user preferences **35**, for example, can include text font attributes, filter settings, blocking settings, screen names per account identifier, alert settings per screen name, buddy list groups, electronic mailboxes, electronic voice mail, and parental control settings. It will be appreciated by one of ordinary skill in the art that the plurality of user preferences **35**, in accordance with the present invention, can include any of those preferences mentioned herein or an equivalent.

The plurality of session data **36** included within the plurality of client data **25** contains information relating to each of the plurality of messaging sessions **24** for which the account user **30** is currently participating, has previously participated, or plans to participate in, using the messaging client **26**. FIG. **3** illustrates a preferred embodiment of the plurality of session data **36** in accordance with the present invention. As illustrated in FIG. **3**, for each messaging session **40**, the plurality of session data **36** includes a session identifier **41**, a session priority **42**, a plurality of session preferences **43**, a plurality of session participants **44**, and a session history **45**. It will be appreciated by one of ordinary skill in the art that the plurality of session data **36**, in accordance with the present invention, can include any of the session data mentioned herein or an equivalent.

Preferably, the session identifier **41** identifies the messaging session **40** of the plurality of messaging sessions **24**. In one embodiment, the messaging session **40** is assigned the session priority **42**. The session priority **42** determines or identifies the priority of the messaging session **40** within the plurality of messaging sessions **24** for which the account user **30** is currently participating. The session priority **42** can be set manually by the account user **30** or through a predetermined algorithm in the messaging client **26** taking

into account the various characteristics of the messaging session **40** and the messaging client **26**. The session priority **42** can for example, specify a stacking order (e.g.: order of display window layering for viewing) of the messaging windows within the messaging client **26**. Alternatively, when the messaging client **26** operates within a messaging device capable of only displaying one session at a time, the session priority **42** can identify the session to display at any given point in time.

The plurality of session preferences **43** defines certain attributes settable by the account user **30** for communicating within the messaging session **40** using the messaging client **26**. The plurality of session preferences **43**, for example, can include text font attributes, filter settings, blocking settings, alert settings, screen names, buddy list groups, electronic mailboxes, parental control settings, an alert option such as alert on receipt of a new real time message or no alert on receipt of a new real time message, guaranteed or non-guaranteed delivery, timeout setting for participation in the messaging session **40**, and number of real time messages to retain in the session history **45** and to display. It will be appreciated by one of ordinary skill in the art that the plurality of session preferences **43**, in accordance with the present invention, can include any of the session preferences mentioned herein or an equivalent. In one embodiment, the plurality of session preferences **43** includes a session timer. (not shown) The session timer is a preset time period upon which the messaging client **26** is active within the messaging session **40**. The plurality of session preferences **43** in one embodiment is transferred to the messaging client **26** when the messaging session **40** is activated. Alternatively, the account user **30** manually can set the plurality of session preferences **43**. Alternatively, a default set of session preferences can be preprogrammed in the messaging client **26** to enhance the efficiency of managing the participation in the plurality of messaging sessions **24**. The plurality of session participants **44** includes each account user participating in the messaging session **40** along with the account identifier for each participating account user.

FIG. **4** illustrates a preferred embodiment of the session history **45** of FIG. **3** in accordance with the present invention. As illustrated in FIG. **4**, the session history **45** preferably includes a plurality of session messages **61** in which each session message **46** is associated with a plurality of message information including an account identifier **47** for an associated message originator such as the account user **30**. The associated originator for example is one of the plurality of session participants **44**. Each session message **46** further can be associated with a message timestamp **48** identifying the time that the session message **46** was entered into the messaging session **40** by the message originator. The session history **45** is further composed of at least one session portion **49**. Each session portion **49** comprises at least one session message **46** and associated information. It will be appreciated by one of ordinary skill in the art that although the session portion **49** is illustrated as a portion of the session history **45**, alternatively, in accordance with the present invention, the session portion **49** can be any portion of the plurality of session data **36**.

FIG. **5** is an electronic block diagram of one embodiment of a messaging device in which a messaging client of FIG. **1** operates. Specifically, FIG. **5** illustrates a fixed network device **50**. The fixed network device **50** can operate for example on a local area network (LAN) or a wide area network (WAN) or a combination of both. The fixed network device **50** can be one of a plurality of spatially co-located computers which are typically located within a room, building or campus of buildings and are sharing common resources and communicating with each other on a computer network in a manner well known to one of ordinary skill in the art. Typical resources shared are files on a file server, printers on a print server, and electronic message (email) services on an email server. The fixed network device **50** can operate on a network that uses a physical network such as ARCNET, Ethernet, Token-ring, Local Talk or other network media to connect the computers, which represent wired network nodes into the network. The fixed network device **50** can operate on a LAN that employs any one of a number of networking protocols, such as TCP/IP (Transmission Control Protocol/Internet Protocol), AppleTalk™, IPX/SPX (Inter-Packet Exchange/Sequential Packet Exchange), Net BIOS (Network Basic Input Output System) or any other packet structures to enable the communication among the devices and/or between the devices and the shared resources. Further the fixed network device **50** can operate on a WAN that uses a different physical network media such as X.25, Frame Relay, ISDN, Modem dial-up or other media to connect other computers or other local area networks. In the following description, the term "fixed network device" includes any of the messaging devices operating as described above or an equivalent.

As illustrated, the fixed network device **50** comprises a timing clock **52**, a central processing unit **53**, an electronic memory preferably in the form of a random access memory (RAM) **54** and/or a read only memory (ROM) **55**, and a mass storage element (e.g., a disk drive or the like) **56**. In one embodiment, the fixed network device **50** includes a memory interconnect **57** for operatively connecting a memory storage device **58** to the fixed network device **50**. The memory interconnect **57** can, for example, comprise a structure for physically engaging external contacts on the memory storage device **58** so that the memory storage device **58** is directly connected to the fixed network device **50**. It will be appreciated by one of ordinary skill in the art that the memory interconnect **57** can also be a wireless connection such as an infrared, Bluetooth or radio frequency interface. When the memory interconnect **57** is connected to the memory storage device **58**, the fixed network device **50** can access a plurality of memory information such as the plurality of client data **25** from the memory storage device **58**.

The fixed network device **50** further preferably comprises a display driver **59**, a general I/O interface or data port **60**, and a user interface port **62** that accommodates a user interface **64** including any number of input means for general information entry. In the preferred embodiment, the user interface **64**, e.g., a keyboard **66**, a "mouse," **68**, a pen or puck activated tablet (not shown), a trackball **70**, an audio activated command recognition processor **72**, or the like, allows a device user to enter and manipulate information using a user input **88**. After information is entered, it may be communicated to a wired messaging system **89** via a conventional modem **74** or the like. Preferably, the fixed network device **50** also includes an Ethernet connection **76** for communicating to the wired messaging system **89** or for communicating through either a conventional cable modem **78** to a cable headend, or a (Digital Subscriber Line) DSL connection **80** to the wired messaging system **89**. The fixed network device **50** can be changed from an active to an inactive state or from an inactive state to an active state through the user input **88** to the power circuit **82**. The power circuit **82** can be operated manually via the user input **88**

directly to the power circuit **82**, the user input **88** to the user interface **64**, or alternatively automatically via the programming of the CPU **53**.

In a preferred embodiment of the present invention, the fixed network device **50** of FIG. **5** includes a fixed messaging client **84**. It will be appreciated by one of ordinary skill in the art that the fixed messaging client **84** can be the first messaging client **14**, the second messaging client **20**, or any other of the plurality of messaging clients **12** of FIG. **1**. The fixed network device **50** performs messaging functions within the fixed messaging client **84** using a plurality of messages stored in the electronic memory of the fixed network device **50**. The fixed messaging client **84** may be hard coded or programmed into the fixed network device **50** during manufacturing, may be programmed over-the-air upon customer subscription, or may be a downloadable application. It will be appreciated that other programming methods can be utilized for programming the fixed messaging client **84** into the fixed network device **50**. It will be further appreciated by one of ordinary skill in the art that the fixed messaging client **84** can be hardware circuitry within the fixed network device **50**.

Preferably the fixed messaging client **84** automatically updates a CRT **86** when a new message has been sent or received by sending a command to the display driver **59**. This allows the message to be updated while the device user is reading it without disturbing the CRT **86**. The fixed messaging client **84** uses the plurality of client data **25** stored in the electronic memory or stored in the memory storage device **58** to perform functions relating to various received and/or sent messages. It will be appreciated by one of ordinary skill in the art that fixed networked devices having software-programming capabilities may include client data that is specialized and personalized such as the plurality of user preferences **35** including display options and screens for each account user, or similarly may include the plurality of session preferences **43** for each messaging session **40**. Alternatively, fixed networked devices that do not include software-programming capabilities may include the plurality of client data **25** including the plurality of user preferences **35** that are standard, pre-defined display options and screens for the plurality of messaging sessions **24**.

The plurality of user preferences **35** of the plurality of client data **25** used by the fixed messaging client **84** further includes various alert options. In one embodiment, the fixed messaging client **84** notifies the CPU **53** to send a command to an alert circuit (not shown) when a new message is received. In another embodiment, the fixed messaging client **84** notifies the CPU **53** to send a command to the alert circuit when an unread message is to be deleted from the memory. Alternatively, no alert may be sent when a new message is received and stored in the memory. It will be appreciated by one of ordinary skill in the art that other alerting schemes are within the scope of the present invention. Further, the CPU **53**, in response to the user input **88** to the user interface **64** through to the user interface port **62**, such as a device user depressing a button or series of buttons, or in response to receipt of a message initiates an input signal to the fixed messaging client **84**. The fixed messaging client **84**, in response to the input signal, accesses a plurality of messages stored in the electronic memory for use in operation of the fixed messaging client **84**.

Preferably, the fixed messaging client **84** includes a client profile **85**. The client profile **85** includes information regarding the capabilities and limitations of the fixed messaging client **84** and also of the fixed network device **50**. For example, the client profile **85** can include indication of the media supported by the fixed messaging client **84** (e.g. audio, video), indication of which features are supported by the fixed messaging client **84**, device type, device display, device battery life, device battery capacity, device processing power, and access to alternate networks. It will be appreciated by one of ordinary skill in the art that the client profile **85** can include any of those mentioned above in any combination or an equivalent.

In accordance with the present invention, the fixed messaging client **84** includes software capability for transferring all or a portion of the plurality of client data **25** to one or more other messaging clients for use by the other messaging client to participate within one or more of the plurality of messaging sessions **24**. The fixed messaging client **84**, in accordance with the present invention, further includes software capability for receiving all or a portion of the plurality of client data **25** from at least one other messaging client to participate within one or more of the plurality of messaging sessions **24**. As illustrated in FIG. **5**, the software capability for transferring and/or the capability for receiving the plurality of client data **25** can be incorporated into the fixed messaging client **84**, or alternatively can be contained within a separate data transfer application **83**. The data transfer application **83**, for example can be a third party software add-on that is compatible with existing messaging software applications (e.g. the fixed messaging client **84**) already programmed into the fixed network device **50**. Maintaining the data transfer software on a separate data transfer application **83** minimizes incorporation timeframes and also the cost of upgrading the fixed network device **50** to include this feature.

FIG. **6** is an electronic block diagram of one embodiment of a messaging device in which a messaging client of FIG. **1** operates. Specifically, FIG. **6** illustrates a mobile device **90**. It will be appreciated by one of ordinary skill in the art that the mobile device **90** in accordance with the present invention, can be a mobile cellular telephone, a mobile radio data terminal, a mobile cellular telephone having an attached data terminal, or a two way pager, such as the "Pagewriter 2000X" manufactured by Motorola Inc. of Schaumburg, Ill. In the following description, the term "mobile device" refers to any of the messaging devices mentioned above or an equivalent.

As illustrated in FIG. **6**, the mobile device **90** includes a first antenna **92**, a second antenna **94**, a receiver **96**, a transmitter **98**, a clock **100**, a processor **102**, a device memory **104**, a device memory interconnect **105**, a device alert circuit **106**, a device display **108**, a device user interface **110** and a mobile messaging client **112**.

The first antenna **92** intercepts transmitted signals from a wireless messaging system **114**. It will be appreciated by one of ordinary skill in the art that the wireless messaging system **114**, in accordance with the present invention, can function utilizing any wireless RF channel, for example, a one or two-way pager channel, a mobile cellular telephone channel, or a mobile radio channel. Similarly, it will be appreciated by one of ordinary skill in the art that the wireless messaging system **114** can function utilizing other types of communication channels such as infrared channels. In the following description, the term "wireless messaging system" refers to any of the wireless messaging systems mentioned above or an equivalent.

The first antenna **92** is coupled to the receiver **96**, which employs conventional demodulation techniques for receiving the communication signals transmitted by the wireless messaging system **114**. Coupled to the receiver **96**, is the processor **102** utilizing conventional signal-processing tech-

niques for processing received messages. Preferably, the processor **102** is similar to the MC68328 micro-controller manufactured by Motorola, Inc. of Schaumburg, Ill. It will be appreciated by one of ordinary skill in the art that other similar processors can be utilized for the processor **102**, and that additional processors of the same or alternative type can be utilized as required to handle the processing requirements of the processor **102**. The processor **102** decodes an address in the demodulated data of the received message, compares the decoded address with one or more addresses **115** stored in an address memory **116** of the device memory **104**; and when a match is detected, proceeds to process the remaining portion of the received message.

To perform the necessary functions of the mobile device **90**, the processor **102** is coupled to the device memory **104**, which preferably includes a random access memory (RAM), a read-only memory (ROM), and an electrically erasable programmable read-only memory (EEPROM)(not shown). The device memory **104** includes the address memory **116**, a message memory **118**, and a client data memory **120**.

Once the processor **102** has processed a received message, it stores the decoded message in the message memory **118** of the device memory **104**. It will be appreciated by one of ordinary skill in the art that the message memory **118**, in accordance with the present invention, can be a voicemail box or a group of memory locations in a data storage device. In the following description, the term "message memory" refers to any of the memory means mentioned above or an equivalent. Preferably, when the received message is a message for participation in one of the plurality of messaging sessions **24**, for example the session message **46** of the messaging session **40**, the processor **102** stores the decoded message in the client data memory **120**.

In one embodiment, the mobile device **90** includes the device memory interconnect **105** for operatively connecting the memory storage device **58** to the mobile device **90**. The device memory interconnect **105** can, for example, comprise a structure for physically engaging external contacts on the memory storage device **58** so that the memory storage device **58** is directly connected to the mobile device **90**. It will be appreciated by one of ordinary skill in the art that the device memory interconnect **105** can also be a wireless connection such as an infrared, Bluetooth or radio frequency interface. When the device memory interconnect **105** is connected to the memory storage device **58**, the mobile device **90** can access a plurality of memory information such as the plurality of client data **25** from the memory storage device **58**.

The client data memory **120** includes the plurality of client data **25** as described previously in FIGS. **2** to **4**. The client data memory **120** includes a memory slot **122** for each messaging session **40** in which the mobile device **90** has subscribed. The memory slot **122**, in accordance with the present invention, includes the plurality of session data **36** as illustrated in FIG. **2**. The plurality of session messages **61** associated with the messaging session **40** is stored together in chronological order in the memory slot **122** similar to the session history **45** of FIG. **4**. The memory slot **122** is allocated a fixed amount of memory for storing associated plurality of session messages **61**. The memory slot **122** holds multiple session messages in a single message memory slot. Any session message **46** received for the messaging session **40** along with its associated session message information is appended at the end of the plurality of session messages **61** already in the memory slot **122**. If the amount of allocated memory for the memory slot **122** is exceeded, the older session messages are deleted. It will be appreciated by one of ordinary skill in the art that, in accordance with the present invention, the client data memory **120** and associated operation herein described, in accordance with the present invention, can be included in the fixed network device **50** of FIG. **5**, a cable box **136** of FIG. **7** or any other messaging device in which the messaging client **26** operates.

Upon receipt and processing of a message, the processor **102** preferably generates a command signal to the device alert circuit **106** as a notification that the message has been received and stored. The device alert circuit **106** can include a speaker (not shown) with associated speaker drive circuitry capable of playing melodies and other audible alerts, a vibrator (not shown) with associated vibrator drive circuitry capable of producing a physical vibration, or one or more LEDs (not shown) with associated LED drive circuitry capable of producing a visual alert. It will be appreciated by one of ordinary skill in the art that other similar alerting means as well as any combination of the audible, vibratory, and visual alert outputs described can be used for the device alert circuit **106**.

Upon receipt and processing of a message, the processor **102** preferably also generates a command signal to the device display **108** to generate a visual notification of the receipt and storage of the message. When the device display **108** receives the command signal from the processor **102** that the message has been received and stored in the device memory **104**, a message indication is displayed. The message indication, for example can be the activation of one of a plurality of message icons on the device display **108**. The device display **108** can be, for example, a liquid crystal display utilized to display text. It will be appreciated by one of ordinary skill in the art that other similar displays such as cathode ray tube displays can be utilized for the device display **108**.

The mobile device **90** preferably further includes the clock **100**. The clock **100** provides timing for the processor **102**. The clock **100** can include the current time for use in the operation of the mobile device **90**. The clock **100** also provides a source for timing of feature enhancements such as active and inactive periods of operation or periods of alerting.

In a preferred embodiment, the mobile device **90** includes the mobile messaging client **112**. It will be appreciated by one of ordinary skill in the art that the mobile messaging client **112** can be the first messaging client **14**, the second messaging client **20**, or any other of the plurality of messaging clients **12** of FIG. **1**. The mobile messaging client **112** performs messaging functions within the mobile device **90** using the plurality of client data **25** stored in the client data memory **120**. The mobile messaging client **112** may be hard coded or programmed into the mobile device **90** during manufacturing, may be programmed over-the-air upon customer subscription, or may be a downloadable application. It will be appreciated that other programming methods can be utilized for programming the mobile messaging client **112** into the mobile device **90**. It will be further appreciated by one of ordinary skill in the art that the mobile messaging client **112** can be hardware circuitry within the mobile device **90**. Preferably the mobile messaging client **112** automatically updates the device display **108** when a new session message has been sent or received. This allows the session history **45** to be updated while the account user **30** is reading it without disturbing the device display **108**. The mobile messaging client **112** uses the plurality of client data **25** stored in the electronic memory or stored in the memory storage device **58** to perform functions relating to various received and/or sent session messages. It will be appreciated

by one of ordinary skill in the art that mobile devices having software-programming capabilities may include specialized and personalized display options and screens for each messaging session **40**. Alternatively, mobile devices that do not include software-programming capabilities may include standard, pre-defined display options and screens for the plurality of messaging sessions **24**. In accordance with the present invention, the display options for the plurality of messaging sessions **24** in which the messaging client **26** within the mobile device **90** is participating can be included in the plurality of session preferences **43** for each messaging session **40** or alternately, the display options can be stored independently within the plurality of user preferences **35** of the plurality of client data **25**

The mobile messaging client **112** further operates using various alert options. In one embodiment, the mobile messaging client **112** notifies the processor **102** to send a command to the device alert circuit **106** when a new session message is added to the memory slot **122** of the client data memory **120** for the messaging session **40**. In another embodiment, the mobile messaging client **112** notifies the processor **102** to send a command to the device alert circuit **106** when an unread session message is to be deleted from the memory slot **122**. Alternatively, no alert may be sent when a new session message is received and stored in the client data memory **120**. It will be appreciated by one of ordinary skill in the art that other alerting schemes are within the scope of the present invention. In accordance with the present invention, the alert options for the plurality of messaging sessions **24** in which the messaging client **26** within the mobile device **90** is participating can be included in the plurality of session preferences **43** for each messaging session **40** or alternately, the alert options can be stored independently within the plurality of user preferences **35** of the plurality of client data **25**.

In accordance with the present invention, the mobile messaging client **112** includes software capability for transferring all or a portion of the plurality of client data **25** to at least one other messaging client for use by the other messaging client to participate within one or more of the plurality of messaging sessions **24**. The mobile messaging client **112**, in accordance with the present invention, further includes software capability for receiving all or a portion of the plurality of client data **25** from another messaging client to participate within one or more of the plurality of messaging sessions **24**. As illustrated in FIG. **6**, the software capability for transferring and receiving client data can be incorporated into the mobile messaging client **112** or alternatively contained within a separate data transfer application **83**. The data transfer application **83**, for example can be a third party software add-on that is compatible with existing messaging software applications (e.g. the mobile messaging client **112**) already programmed into the mobile device **90**. Maintaining the data transfer software on a separate data transfer application **83** minimizes incorporation timeframes and also the cost of upgrading a messaging device to include this feature.

Preferably, the device user interface **110** is coupled to the processor **102**. The device user interface **110** can be one or more buttons used to generate a button press, a series of button presses, a voice response from the device user, or some other similar method of manual response initiated by the device user (such as the account user **30**) of the mobile device **90**. The processor **102**, in response to the device user interface **110**, such as a device user depressing a button or series of buttons, or in response to receipt of a session message, initiates an input signal to the mobile messaging client **112**. The mobile messaging client **112**, in response to the user input signal, accesses the plurality of session messages **61** stored in the client data memory **120** for use in operation of the mobile messaging client **112**.

Preferably, the mobile messaging client **112** includes the client profile **85**. The client profile **85** includes information regarding the capabilities and limitations of the mobile messaging client **112** and also of the mobile device **90**. For example, the client profile **85** can include indication of the media supported by the mobile messaging client **112** (e.g. audio, video), indication of which features are supported by the mobile messaging client **112**, device type, device protocol usage, device display, device battery life, device battery capacity, device processing power, and access to alternate networks. It will be appreciated by one of ordinary skill in the art that the client profile **85** can include any of those mentioned above in any combination or an equivalent.

The transmitter **98** is coupled to the processor **102** and is responsive to commands from the processor **102**. When the transmitter **98** receives a command from the processor **102**, the transmitter **98** sends a signal via the second antenna **94** to the wireless messaging system **114**.

In an alternative embodiment (not shown), the mobile device **90** includes one antenna performing the functionality of the first antenna **92** and the second antenna **94**. Further, the mobile device **90** alternatively includes a transceiver circuit performing the functionality of the receiver **96** and the transmitter **98**. It will be appreciated by one of ordinary skill in the art that other similar electronic block diagrams of the same or alternate type can be utilized for the mobile device **90** to handle the requirements of the mobile device **90**.

The mobile device **90** can be changed from an active state to an inactive state or from an inactive state to an active state through a user input to the power circuit **134**. The power circuit **134** can be operated manually via the user input to the power circuit **134**, the user input to the user interface **110**, or alternatively automatically via the programming of the processor **102**.

FIG. **7** is an electronic block diagram of one embodiment of a messaging device in which a messaging client of FIG. **1** operates. Specifically, FIG. **7** illustrates an interactive broadcast receiver such as the cable box **136**. The cable box **136** preferably allows network operators to deploy a wide range of interactive television broadcast services and applications on their networks. Further the cable box **136** preferably offers cable operators a combined, all-in-one, hardware and software solution for deploying interactive television services on their networks, thereby creating the ability for real time electronic message communication using television sets and networks.

As illustrated in FIG. **7**, the cable box **136** preferably includes a controller **138** for controlling the operation of the cable box **136**. Preferably, the controller **138** is similar to the MC68328 micro-controller manufactured by Motorola, Inc. of Schaumburg, Ill. It will be appreciated by one of ordinary skill in the art that other similar processors can be utilized for the controller **138**, and that additional processors of the same or alternative type can be utilized as required to handle the processing requirements of the controller **138**. Preferably, the controller **138** is programmed to function with the cable messaging client **140**. The cable messaging client **140**, in accordance with the present invention, operates similarly to the fixed messaging client **84** of FIG. **5** and the mobile messaging client **112** of FIG. **6** as described above. It will be appreciated by one of ordinary skill in the art that the cable messaging client **140** illustrated in FIG. **7** can be the first

messaging client **14**, the second messaging client **20**, or any other of the plurality of messaging clients **12** of FIG. **1**.

In accordance with the present invention, the cable messaging client **140** includes software capability for transferring all or a portion of the plurality of client data **25** to at least one other messaging client for use by the other messaging client to participate within one or more of the plurality of messaging sessions **24**. The cable messaging client **140** further includes software capability for receiving all or a portion of the plurality of client data **25** from another messaging client to participate within one or more of the plurality of messaging sessions **24**. As illustrated in FIG. **7**, the software capability for transferring and receiving client data can be incorporated into the cable messaging client **140** or alternatively contained within a separate data transfer application **83**. The data transfer application **83**, for example can be a third party software add-on that is compatible with existing messaging software applications (e.g. the cable messaging client **140**) already programmed into the cable box **136**. Maintaining the data transfer software on a separate data transfer application **83** minimizes incorporation timeframes and also the cost of upgrading a device to include this feature.

Preferably, the cable messaging client **140** includes the client profile **85**. The client profile **85** includes information regarding the capabilities and limitations of the cable messaging client **140** and of the cable box **136**. For example, the client profile **85** can include indication of the media supported by the cable messaging client **140** (e.g. audio, video), indication of which features are supported by the cable messaging client **140**, device type, device protocol usage, device display, device battery life, device battery capacity, device processing power, and access to alternate networks. It will be appreciated by one of ordinary skill in the art that the client profile **85** can include any of those mentioned above in any combination or an equivalent.

The cable box **136** further includes an up/down converter **142** coupled to the controller **138** for communicating with a cable headend. To perform the necessary functions of the cable box **136**, the controller **138** is further coupled to a cable box memory **144**, which preferably includes a cable box random access memory (RAM) **146**, a cable box read-only memory (ROM) **148**, and an electrically erasable programmable read-only memory (EEPROM) (not shown). The cable box memory **144** of the cable box **136** preferably includes the client data memory **120** as previously described and illustrated in FIG. **6**.

In one embodiment, the cable box **136** includes a cable box memory interconnect **149** for operatively connecting the memory storage device **58** to the cable box **136**. The cable box memory interconnect **149** can, for example, comprise a structure for physically engaging external contacts on the memory storage device **58** so that the memory storage device **58** is directly connected to the cable box **136**. It will be appreciated by one of ordinary skill in the art that the cable box memory interconnect **149** can also be a wireless connection such as an infrared, Bluetooth or radio frequency interface. When cable box memory interconnect **149** is connected to the memory storage device **58**, the cable box **136** can access a plurality of memory information such as the plurality of client data **25** from the memory storage device **58**.

Further coupled to the controller **138** is a first cable box I/O **150** for driving a remote control transceiver **152** and further for driving a radio frequency transceiver **154** connected to a cable box antenna **156**. A second cable box I/O **158** for inputs from a user input via a cable box user interface **160** is further coupled to the controller **138**. Also coupled to the controller **138** are an audio driver **162** and a radio frequency/video driver **164** for communicating with a television **166**.

The cable box **136** can be changed from an active state to an inactive state or from an inactive state to an active state through a user input to the cable box power circuit **168**. The cable box power circuit **168** can be operated manually via the user input to the cable box power circuit **168**, the user input to the cable box user interface **160** or alternatively automatically via the programming of the controller **138**.

FIG. **8** is an electronic block diagram of an alternate embodiment of a messaging communication system **170** in accordance with the present invention. The messaging communication system **170** includes the plurality of messaging clients **12** and a message server **172**.

The message server **172** manages the communication of a plurality of electronic messages among the plurality of messaging clients **12**, facilitating substantially real time communication among the plurality of messaging clients **12** within the messaging communication system **170**. The message server **172** provides numerous services to manage the plurality of messaging sessions **24**. The message server **172** also offers various options to the plurality of session participants **44** to reduce cost or enhance the features of the plurality of messaging sessions **24**.

Each messaging client **26** of the plurality of messaging clients **12** such as the first messaging client **14** and the second messaging client **20** includes client software to interface within the messaging communication system **10**. It will be appreciated by one of ordinary skill in the art that in accordance with the present invention, the interface capabilities of the client software can also be designed into client hardware of a messaging client. Each mess aging client **26** of the plurality of messaging clients **12** further includes the client identifier **27**. For example, the first messaging client **14** includes the first client identifier **15** and the second messaging client **20** includes the second client identifier **21**. The client identifier **27** of the messaging client **26** is a unique identification within the messaging communication system **170** for providing individualized messages to be directed to a particular messaging client. For example, the client identifier **27** can be an address of the mobile device **90** or an IP address and number of the port of the fixed network device **50**. To communicate within the messaging communication system **170**, the messaging client **26** establishes the communication connection **28** via the message server **172**. For example, the first messaging client **14** establishes the first communication connection **16** via the message server **172** for communication within at least one of the plurality of messaging sessions **24**. Similarly, the second messaging client **20** establishes the second communication connection **22** via the message server **172** for communication within at least one of the plurality of messaging sessions **24**. It will be appreciated by one of ordinary skill in the art that the communication connection **28**, the first communication connection **16**, and the second communication connection **22** can be a physical connection, or alternatively can be a logical connection where the act of connecting and disconnecting is a logical one. Each of the plurality of messaging clients **14** belongs to the account user **30**. The account user **30** is an individual who uses one or more of the plurality of messaging clients **12** to communicate with other account users within the plurality of messaging sessions **24**. It will be appreciated by one of ordinary skill in the art that the account user **30** can communicate using one or more of the plurality of messaging clients **12**. For example, the first

account user **29** can establish communication within the plurality of messaging sessions **24** using the first messaging client **14**, and, in accordance with the present invention, also using the second messaging client **20**.

Each messaging client **26** preferably includes the plurality of client data **25**. The plurality of client data **25** includes data associated with the messaging client **26** and data associated with each messaging session for which the messaging client **26** is currently participating, has participated in, or plans to participate in. The first messaging client **14** includes the first client data **17** and the second messaging client **20** includes the second client data **23**.

The message server **172** includes a server processor **174** and a server memory **176**. The server processor **174** utilizes conventional signal processing techniques for processing received electronic messages. Preferably, the server processor **174** is similar to the MC68328 micro-controller manufactured by Motorola, Inc. of Schaumburg, Ill. It will be appreciated that other similar processors can be utilized for the server processor **174**, and that additional processors of the same or alternative type can be added as required to handle the processing requirements of the server processor **174**.

To perform the necessary functions of the message server **172**, the server processor **174** is coupled to the server memory **176**, which preferably includes a random access memory (RAM), a read-only memory (ROM), an electrically erasable programmable read-only memory (EEPROM), and/or a magnetic storage memory (for example a hard drive). The server memory **174** preferably includes a messaging sessions data memory **178**, a messaging clients data memory **180**, and a server data memory **182**. The messaging sessions data memory **178** stores the plurality of session data for all messaging sessions for which the message server **172** is managing. The plurality of session data stored for each messaging session for which the message server **172** is managing is similar to the plurality of session data **36** as illustrated in FIGS. **3** and **4** and described previously. The messaging clients data memory **178** stores the plurality of client data **25** for each of the messaging clients **26** that have established the communication connection **28** with the message server **172**. For example, the plurality of client data **25** can include the type of device being utilized by each messaging client **26**, the account user **30** utilizing each messaging client **26**, the plurality of user preferences **35** for each messaging client **26**, and the messaging sessions for which each messaging client **26** is participating. It will be appreciated by one of ordinary skill in the art that the messaging client data memory **178** can store any of the plurality of client data **25** mentioned herein or an equivalent. The server memory **176** further includes the server data memory **182**. The server data memory **182** preferably includes a server identifier **184** for the message server **172**. The server identifier **184** can be, for example, a unique selective call address in the wireless messaging system **114**. Alternatively, the server identifier **184** can be an IP address, or an IP address and associated number of the port assigned to the message server **172** of the wired messaging system **89**. It will be appreciated by one of ordinary skill in the art that the server identifier **184** can be one mentioned herein or an equivalent. The server identifier **184** enables the communication between the plurality of messaging clients **12** and the message server **172** using the communication connections such as the communication connection **28**, the first communication connection **16**, and the second communication connection **22**. The server data memory **182** also preferably includes a server profile **186**. The server profile **186** includes information regarding the capabilities of the message server **176**. For example, the server profile **186** can include server processing power, server client capability, server messaging session capability, and server access to secondary networks. It will be appreciated by one of ordinary skill in the art that the server profile **186** can include any of those mentioned above in any combination or an equivalent.

FIG. **9** is an electronic block diagram of one embodiment of the messaging communication system **10, 170** of FIGS. **1** and **8** respectively. Specifically, FIG. **9** illustrates an embodiment of the present invention in which the messaging communication system **10, 170** is the wireless messaging system **114** of FIG. **6**.

The wireless messaging system **114**, as illustrated in FIG. **9** includes a message input device for initiating messages into the wireless messaging system **114**. The message input device can be, for example, a telephone **204**, a computer **206**, a desktop messaging unit **208**, or the message server **172** connected through a conventional public switched telephone network (PSTN) **210** through a plurality of telephone links **212** to a wireless system controller **214**. The telephone links **212**, for example, can be a plurality of twisted wire pairs, a fiber optic cable, or a multiplexed trunk line.

The wireless system controller **214** is coupled to and oversees the operation of at least one radio frequency (RF) transmitter **216** and at least one radio frequency (RF) receiver **218** through one or more communication links **220**. The communication links **220** typically are twisted pair telephone wires, and additionally can include radio frequency (RF), microwave, or other communication links. The RF transmitter **216** and the RF receiver **218** typically are used with message store and forward stations that encode and decode inbound and outbound messages into formats that are compatible with landline message switched computers and personal radio addressing requirements, such as cellular messages, short messaging service, or paging protocols. The wireless system controller **214** can also function to encode and decode wireless messages that are transmitted to or received by the RF transmitter **216** or the RF receiver **218**. Telephony signals are typically transmitted to and received from the wireless system controller **214** by telephone sets such as the telephone **204** or a mobile device. The wireless system controller **214** encodes and schedules outbound messages such as a downlink message **222**. The wireless system controller **214** then transmits the encoded outbound messages through the RF transmitter **216** via a transmit antenna **224** to a plurality of mobile devices **226** such as the mobile device **90** of FIG. **6** on at least one outbound radio frequency (RF) channel **234**. The plurality of mobile devices **226**, for example, includes a first mobile device **228**, a second mobile device **230**, and a third mobile device **232** each communicating through a wireless connection such as the outbound RF channel **234** and an inbound RF channel **240**. The downlink message **222** can be, for example, a data message or a voice call such as the session message **46**. Similarly, the wireless system controller **214** receives and decodes inbound messages such as an uplink message **236** received by the RF receiver **218** via a receive antenna **238** on at least one inbound radio frequency (RF) channel **240** from one of the plurality of mobile devices **226**. The uplink message **236** can be, for example, a data message, a reply to a data message, a response message based on at least one data message, a voice call, or a reply to a voice call, such as the session message **46**.

Each of the plurality of mobile devices **226** assigned for use in the wireless messaging system **114** has an address or

identity assigned thereto which is a unique selective call address in the wireless messaging system **114**. For example, the first mobile device **228** has a first address **242**, the second mobile device **230** has a second address **244**, and the third mobile device **232** has a third address **246**. It will be appreciated by one of ordinary skill in the art that other mobile devices assigned for use in the wireless messaging system **114** have an address assigned thereto which is a unique selective call address in the wireless messaging system **114**. The address enables the transmission of the downlink message **222** from the wireless system controller **214** only to the mobile device having the address, and identifies the messages and responses received at the wireless system controller **214** from the mobile device with the address. In one embodiment, each of the plurality of mobile devices **226** also has a pin number assigned thereto, the pin number being associated with a telephone number within the PSTN **210**. A list of the assigned addresses and correlated telephone numbers for each of the plurality of mobile devices **226** is stored in the wireless system controller **214** in the form of a subscriber database **248**.

Preferably, at least one messaging client operates within a mobile device. For example, as illustrated in FIG. **9**, the first messaging client **14** operates within the first mobile device **228** and the second messaging client **20** operates within the second mobile device **230**. Similarly, a plurality of messaging clients can operate within the same mobile device. For example, a third messaging client **250** and a fourth messaging client **252** operate within the third mobile device **232**. It will be appreciated by one of ordinary skill in the art that, in accordance with the present invention, a mobile device can include no messaging client, one messaging client, or a plurality of messaging clients.

In one embodiment of the present invention, the message server **172** is coupled to the wireless system controller **214** of the wireless messaging system **114**. The message server **172** provides a means for real time electronic message communication with the plurality of mobile devices **226**. The message server **172**, for example, receives a request and can in response to such receipt, sends a response, both via the wireless system controller **214**. The wireless system controller **214** then routes the response to the requesting device which may be a message input device, such as the telephone **204**, the computer **206**, or the desktop messaging unit **208**, or alternatively may be an individual or one of the plurality of mobile devices **226**. In the following description, the term requester refers to any of the requesting devices mentioned above or an equivalent.

Preferably, the message server **172** includes a server address **254**, which is a unique selective call address in the wireless messaging system **114**. The server address **254** enables the transmission, via the inbound RF channel **240**, to the message server **172** of various real time electronic communication messages such as conversation service requests, subscription requests, conversation messages, availability settings, and other information. The message server **172** similarly sends real time electronic communication messages such as sending an availability setting or the forwarding of a session message to the plurality of mobile devices **226** via the outbound RF channel **234**. Furthermore, the message server **172** can also have a pin number assigned thereto, the pin number being associated with a telephone number within the PSTN **210**. The server address **254** and correlated telephone number is stored in the in the subscriber database **248** of the wireless system controller **214**.

The coupling of the message server **172** to the wireless messaging system **114** enhances the operation of the wireless messaging system **114** by adding intelligence for multiple mobile devices to communicate in substantially real time. The message server **114** interactively manages the messaging traffic associated with the plurality of messaging sessions **24** in an efficient manner.

FIG. **10** is an electronic block diagram of one embodiment of the messaging communication system **10, 170** of FIGS. **1** and **8** respectively. Specifically, FIG. **10** illustrates an embodiment of the present invention in which the messaging communication system **10, 170** is the wired messaging system **89** of FIG. **5**. The wired messaging system **89**, for example, can include a LAN **256** (local area network), a WAN **258** (wide area network), or a combination of LAN **256** and WAN **258** networks. It will be appreciated that while only a single LAN **256** and a single WAN **258** are shown, multiple LAN **256** networks and/or WAN **258** networks can be interconnected in a manner well known to one of ordinary skill in the art for the transfer of electronic communication such as electronic mail (email), and real time electronic messaging (i.e.: instant messaging and chat messaging) including the plurality of session messages **61**.

The general function and operation of the LAN **256** is one of allowing spatially co-located computers which are typically located within a room, building or campus of buildings to communicate with each other and/or share common resources on a computer network in a manner well known to one of ordinary skill in the art. The spatially co-located computers are represented pictorially in FIG. **10** as a plurality of messaging devices, such as the fixed network device **50** of FIG. **5**, three of which are shown by example. (a first network device **260**, a second network device **262**, and a third network device **264**) Each of the plurality of messaging devices communicates using a network connection **265**. Preferably, at least one messaging client operates within a network device. For example, as illustrated in FIG. **10**, the first messaging client **14** operates within the first network device **260** and the second messaging client **20** operates within the second network device **262**. Similarly, a plurality of messaging clients can operate within the same network device. For example, the third messaging client **250** and the fourth messaging client **252** operate within the third network device **264**. It will be appreciated by one of ordinary skill in the art that, in accordance with the present invention, a network device can include no messaging client, one messaging client, or a plurality of messaging clients.

Typical resources shared on the LAN **256** through a LAN server **266** are files on a file server, printers on a print server, and electronic message (email) services on an email server. The LAN **256** uses a physical network such as ARCNET, Ethernet, Token-ring, Local Talk or other network media to connect the computers, which represent wired network nodes into the network. The LAN **256** can employ any one of a number of networking protocols, such as TCP/IP (Transmission Control Protocol/Internet Protocol), AppleTalk™, IPX/SPX (Inter-Packet Exchange/Sequential Packet Exchange), Net BIOS (Network Basic Input Output System) or any other packet structures to enable the communication between E-mail clients and the E-mail server. In the following description, the term "local area network" refers to a network utilizing any of the networking protocols mentioned above or an equivalent. The LAN **256** can also use routers (not shown) to subnet the LAN **256** organizationally or physically. In this context, the definition of the LAN **256** as described herein refers to a geographic locality of computers and the type of wired media used to interconnect the computers for communication.

The general function and operation of the WAN **258** is also one of allowing computers to share common resources. However, in this context the definition used herein is one where the computers are not spatially co-located. The typical resources shared are similar to, if not the same, as found in the LAN **256**. However, the WAN **258** uses a different physical network media such as X.25, Frame Relay, ISDN, Modem dial-up or other media to connect other computers or other local area networks to the WAN **258** network. The WAN **258**, for example, can include a number of well-known private wide area networks, one **(268)** of which is shown by example; and public wide area networks, one **(270)** of which is show by example, such as CompuServe™, America Online™ (AOL), the MIT computer network, the Motorola™ computer network and Prodigy™. In the following description, the term "wide area network" refers to any of the networks mentioned above or an equivalent. The WAN **258** described above can operate independently, or can be interconnected through the well-known worldwide Internet computer network **272**. Likewise, the LAN **256** can also be interconnected to the WAN **258** through the worldwide Internet computer network **272**, as shown, in a manner well known to one of ordinary skill in the art.

In a one embodiment of the present invention, the message server **172** is coupled to the LAN **256** and to the WAN **258** of the wired messaging system **89**. The message server **172** provides a means for real time electronic message communication with all messaging devices communicating within the wired messaging system **89** such as the first network device **260**, the second network device **262**, and the third network device **264**. The message server **172**, for example, receives a request and preferably in response to such receipt, sends a response, via the LAN server **266**, via the worldwide Internet computer network **272**, or an equivalent. The LAN server **266**, the worldwide Internet computer network **272**, or the equivalent then routes the response to the requesting device, which can be an individual or one of the networked devices. In the following description, the term requester refers to any of the requesting devices mentioned above or an equivalent.

FIG. **11** is an electronic block diagram of one embodiment of the messaging communication system **10, 170** of FIGS. **1** and **8** respectively. Specifically, FIG. **11** illustrates an alternate embodiment of the present invention in which the messaging communication system **10, 170** is the wired messaging system **89** of FIG. **5**. The wired messaging system **89** illustrated in FIG. **11** is, for example, a broadcast messaging system **274**.

The broadcast messaging system **274** preferably includes a cable headend **276**, a network PSTN **278**, and a plurality of cable boxes, such as the cable box **136** of FIG. **7**, three of which are shown by way of example. (a first cable box **280**, a second cable box **282**, and a third cable box **284**. Each of the plurality of cable boxes communicates within the broadcast messaging system **274** via a wired connection **286**. Preferably, at least one messaging client operates within a cable box. For example, as illustrated in FIG. **11**, the first messaging client **14** operates within the first cable box **280** and the second messaging client **20** operates within the second cable box **282**. Similarly, a plurality of messaging clients can operate within the same cable box. For example, the third messaging client **250** and the fourth messaging client **252** operate within the third cable box **284**. It will be appreciated by one of ordinary skill in the art that, in accordance with the present invention, a cable box can include no messaging client, one messaging client, or a plurality of messaging clients.

The cable headend **276** is coupled to the first cable box **280**, the second cable box **282**, the third cable box **284**, the network PSTN **278**, and, in one embodiment, the message server **172**. The cable headend **276** enables operators to deliver services such as conventional video and audio broadcasting, NVOD, VOD, Pay TV, advertising, information, interactive shopping and more. The cable headend **276** preferably offer functions such as MPEG-2/DVB encoding of local and non-compressed programs, insertion of local advertising and events data insertion, conditional access (CA) scrambling, interactive services, and monitoring and control of the entire network. At the multiplexing stage, broadcasters can create program bouquets and add PSI/SI information before the outgoing transport stream is delivered to a conditional access (CA) system for scrambling. Following processing, transport streams are modulated and then transmitted to the cable headend **276** via telecom networks, terrestrial or satellite systems.

In one embodiment of the present invention, the message server **172** is coupled to the cable headend **276** of the broadcast messaging system **274**. The message server **172** provides a means for real time electronic message communication with all cable boxes communicating within the broadcast messaging system **274**. The message server **172**, for example, receives a request and preferably in response to such receipt, sends a response via the cable headend **276**. The cable headend **276** then routes the response to the requesting device, which can be an individual, or can be a cable box. In the following description, the term requester refers to any of the requesting devices mentioned above or an equivalent.

FIG. **12** is an electronic block diagram illustrating an alternative embodiment of the messaging communication system **10,170** in accordance with the present invention. As illustrated, the messaging communication system **10,170** preferably includes a first messaging system **288** having a first plurality of messaging clients **292**, and a second messaging system **290** having a second plurality of messaging clients **294**. In one embodiment, the messaging communication system **10,170** also includes the message server **172**. It will be appreciated by one of ordinary skill in the art that while only two messaging systems are shown by way of example, multiple messaging systems can be interconnected in a manner well known to one of ordinary skill in the art for the transfer of electronic communication such as electronic mail (email), and real time electronic messaging (i.e.: instant messaging and chat messaging) either directly between the messaging systems and/or by using the messaging server **172**.

It will be appreciated by one of ordinary skill in the art that, in accordance with the present invention, the first messaging system **288** and the second messaging system **290** can be the wireless messaging system **114** of FIG. **9**, the wired messaging system **89** of FIG. **10**, the broadcast messaging system **274** of FIG. **11** or any other equivalent messaging system. Further, in accordance with the present invention, the messaging communication system **10,170** can include a plurality of wireless messaging systems, a plurality of wired messaging systems, or any combination thereof. Similarly, each messaging client of the first plurality of messaging clients **292** and the second plurality of messaging clients **294** can operate within the mobile device **90** of FIG. **6**, the fixed network device **50** of FIG. **5**, or the cable box **136** of FIG. **7**. The first plurality of messaging clients **292** and the second plurality of messaging clients **294**, in accordance with the present invention, can include a plurality of

wireless messaging devices, a plurality of wired messaging devices, a plurality of networked devices, or any combination thereof.

FIG. **13** is a flowchart illustrating the operation of the messaging communication system **10,170** in accordance with the present invention. Beginning with Step **296**, the first messaging client **14** establishes the first communication connection **16** for communication within at least one of the plurality of messaging sessions **24** within the messaging communication system **10,170**. For example, when the first messaging client **14** operates within the fixed network device **50**, the first messaging client **14** accesses the appropriate network and notifies the messaging communication system **10,170** of its connection information (i.e.: IP address and number of the port assigned to the first messaging client **14**). Next, in Step **298**, the process determines whether or not an authentication is required. It will be appreciated by one of ordinary skill in the art that an authentication can be required of the first messaging client **14**, of the first account user **29** utilizing the first messaging client **14**, or of the messaging device in which the first messaging client **14** operates, or an equivalent. In Step **300**, when an authentication is required in Step **298**, a first authentication is performed. The first authentication of Step **300** checks that the first account user **29** or alternatively the first messaging client **14** is authorized to establish the first communication connection **16** and/or authorized to participate within one or more of the plurality of messaging sessions **24**. Next, in Step **302**, when the first authentication of Step **300** is completed, and also when the authentication is not required in Step **298**, the first messaging client **14** operates using the first communication connection **16** and accumulates the plurality of session data **36** for each messaging session **40** for which the first messaging client **14** is participating. In accordance with the present invention, the plurality of session data **36** can include the session identifier **41**, the session priority **42**, the session preferences **43**, the session participants **44**, or the session history **45**. It will be appreciated by one of ordinary skill in the art that the plurality of session data **36** can include any of the items mentioned herein or an equivalent. Next, in Step **304**, the process determines whether a data transfer is required or requested. A data transfer, in accordance with the present invention, is the capability for a first account user **29** to change communication means within the messaging communication system **10, 170** from the first messaging client **14** to the second messaging client **20**. For example, when the first account user **29** establishes the first communication connection **16** using the fixed network device **50** and thereafter needs to become mobile, the first account user **29** can transfer the first client data **17** including the plurality of session data **36** accumulated for the first communication connection **16** to the second messaging client **20** which for example can operate on the mobile device **90**. When no data transfer is required or requested in Step **304**, the first communication connection **16** is maintained in Step **302**, whereby the first messaging client **14** continues to accumulate the plurality of session data **36** for each messaging session **40** for which the first messaging client **14** participates. In Step **306**, when a data transfer is required or requested in Step **304**, the process determines if it is necessary to verify the second messaging client **20** prior to transferring the first client data **17** including the plurality of session data **36** to the second messaging client **20**. When verification of the second messaging client **20** is required, the second messaging client **20** is verified in step **308**. For example, the first messaging client **14** and the second messaging client **20** can both be pre-configured with a private value and the first messaging client **14** can exchange messages with the second messaging client **20** that verify that the second messaging client **20** has the correct private value. Next, in Step **310**, after the second messaging client **20** is verified in Step **308** or when no verification is required in Step **306**, the first client data **17** including the plurality of session data **36** is transferred from the first messaging client **14** to the second messaging client **20**. It will be appreciated by one of ordinary skill in the art that a portion of the first client data **17** can alternatively be transferred in Step **310**. It will further be appreciated by one of ordinary skill in the art that the transfer of the first client data **17** can be accomplished using a direct connection between the first messaging client **14** and the second messaging client **20** or a connection through the message server **172**, both either via a network connection, a wireless connection such as through the wireless communication system **114**, a Bluetooth connection, or IRDA connection, a wired connection such as through the wired communication system **89**, a network connection separate from the wireless communication system, an RS-232 connection or the broadcast messaging system **274**, or an equivalent.

Next, in Step **312**, the process determines whether or not the second communication connection **22** has already been established. For example, the second messaging client **20** can establish the second communication connection **22** independently from the establishment of the first communication connection **16** by the first messaging client **14**. The second messaging client **20** can establish the second communication connection **22** but not yet be participating in a messaging session. Alternatively, the second messaging client **20** can independently be participating in one or more messaging session of the plurality of messaging session **24** which can be the same or different messaging sessions from the ones that the first messaging client **14** is participating. In Step **314**, when no second communication connection **22** has been established for the second messaging client **20**, the process determines whether or not the authentication key **33** is required. In Step **316**, when the authentication key **33** is required in Step **314**, the first messaging client **14** transfers the authentication key **33** to the second messaging client **20**. It will be appreciated by one of ordinary skill in the art that the second messaging client **20** can include a plurality of authentication keys and that in step **316** the first messaging client **14** can send an indicator of which of the plurality of authentication keys should be used. It will be appreciated by one of ordinary skill in the art that an authentication key can be required of the second messaging client **20**, of the first account user **29**, or any other account user **30** utilizing the second messaging client **20**, or of the particular messaging device in which the second messaging client **20** operates, or an equivalent. It will further be appreciated by one of ordinary skill in the art that the transfer of the authentication key **33** can be accomplished using a direct connection between the first messaging client **14** and the second messaging client **20** or a connection through the message server **172**, both either via a network connection, a wireless connection such as through the wireless communication system **114**, a Bluetooth connection, or IRDA connection, a wired connection such as through the wired communication system **89**, a network connection separate from the wireless communication system, an RS-232 connection or the broadcast messaging system **274**, or an equivalent. Next, in Step **318**, when no authentication key is required in Step **314** or after the transfer of the authentication key in Step **316**, the second communication connection **22** is established. The second messaging client **20** establishes the second commu-

nication connection **22** for communication within at least one of the plurality of messaging sessions **24** within the messaging communication system **10,170**. For example, when the second messaging client **20** is the mobile device **90**, the second messaging client **20** accesses the appropriate network through the wireless communication system **114** and notifies the messaging communication system **10,170** of its connection information (i.e.: the second address **244** of the second mobile device **230** when the second messaging client **20** operates within the second mobile device **230**). Next, in Step **320**, the process determines whether or not an authentication is required. It will be appreciated by one of ordinary skill in the art that an authentication can be required of the second messaging client **20**, of the first account user **29**, or any other account user **30** utilizing the second messaging client **20**, or of the particular messaging device in which the second messaging client **20** operates, or an equivalent. In Step **322**, when an authentication is required in Step **320**, a second authentication is performed. Next, in Step **324**, when the second communication connection **22** is already established in Step **312**, or after the second session connection **22** is established in Step **318** and authentication is not required in Step **320**, or after the second authentication in Step **322**, the second communication connection **22** is operated using the first client data **17** including the plurality of session data **36** transferred from the first messaging client **14** to the second messaging client **20** in Step **310**.

The method illustrated by the flowchart of FIG. **13** allows messaging sessions to be easily transferred between messaging clients while maintaining session continuity and assuring session security. The account user can switch to a different messaging client on a different messaging system without being required to re-initiate each messaging session that was in progress on the first messaging client. Session continuity is maintained within the two messaging clients, and optionally the transfer does not affect other messaging session participants.

Similarly, the method illustrated by the flowchart of FIG. **13** allows messaging sessions to be easily transferred between different account users. For example, if the first account user **29** is a customer service representative and the first account user **29** is a participant in the plurality of messaging sessions **24** with customers. The first account user **29** may want to transfer a portion of the plurality of messaging sessions **24** to another account user **30** such as a second customer service representative. The second customer service representative would benefit from having access to the session history **45** of the transferred messaging sessions. For example, the second customer representative can avoid asking the customer for information already provided to the first account user **29**. FIG. **14** is a flowchart illustrating more detail of the operation of the messaging communication system **10,170**. Specifically, FIG. **14** illustrates various methods in which the data transfer query (Step **304** of FIG. **13**) can be answered in the affirmative. The operation begins with Step **302**, in which the first messaging client **14** operates using the first communication connection **16** and accumulates the plurality of session data **36** for each messaging session **40** for which the first messaging client **14** is participating. Next, in Step **326**, the process determines whether or not the first messaging device in which the first messaging client **14** operates has received a user input requesting the transfer of at least a portion of the first client data **17** including the plurality of session data **36**. For example, when the messaging device in which the first messaging client **14** operates is the fixed network device **50** of FIG. **5**, the first account user **29** can enter and manipulate information (including requesting the transfer of the first client data **17**) by the user input **88** to the user interface **64**, e.g., the keyboard **66**, the "mouse," **68**, the pen or puck activated tablet (not shown), the trackball **70**, the audio activated command recognition processor **72**, or the like. Similarly, when the first messaging device in which the first messaging client **14** operates is the mobile device **90** of FIG. **6**, the first account user **29** can enter a user input such as a button press, a series of button presses, a voice response, or some other similar method of manual response initiated by the first account user **29** to the device user interface **110** of the mobile device **90**. Similarly, when the first messaging device in which the first messaging client **14** operates is the cable box **136** of FIG. **7**, the user input is made via the cable box user interface **160**. It will be appreciated by one of ordinary skill in the art that the user input can be any of the inputs mentioned herein or an equivalent. When a user input requesting the transfer of at least a portion of the first client data **17** including the plurality of session data **36** is not received by the first messaging device in which the first messaging client **14** operates, the process next, in Step **328** determines whether a user input requesting the transfer of at least a portion of the first client data **17** including the plurality of session data **36** has been received by a second messaging device in which the second messaging client **20** operates. For example, when the second messaging device in which the second messaging client **20** operates is the fixed network device **50** of FIG. **5**, the first account user **29** can enter and manipulate information (including requesting the transfer of the first client data **17**) by the user input **88** to the user interface **64**, e.g., the keyboard **66**, the "mouse," **68**, the pen or puck activated tablet (not shown), the trackball **70**, the audio activated command recognition processor **72**, or the like. Similarly, when the second messaging device in which the second messaging client **20** operates is the mobile device **90** of FIG. **6**, the first account user **29** of the mobile device **90** can enter a user input such as a button press, a series of button presses, a voice response, or some other similar method of manual response initiated by the first account user **29** to the device user interface **110** of the mobile device **90**. Similarly, when the second messaging device in which the second messaging client **20** operates is the cable box **136** of FIG. **7**, the user input is made via the cable box user interface **160**. It will be appreciated by one of ordinary skill in the art that the user input can be any of the inputs mentioned herein or an equivalent. When a user input requesting the transfer of at least a portion of the first client data **17** including the plurality of session data **36** is not received by the second messaging device in which the second messaging client **20** operates, the process next, in Step **330** determines whether the second messaging client **20** is the mobile device **90**, and if so, whether the transfer of at least a portion of the first client data **17** including the plurality of session data **36** is initiated in response to detection of a movement of the mobile device **90**. For example, the server processor **174** of the message server **172** can be programmed to track the location of each of the plurality of messaging clients **12**, and transfer the plurality of session data **36** to the second messaging client **20** in response to the detection of a change of location of the mobile device **90** in which the second messaging client **20** operates. Alternatively, the mobile device **90** can include location-sensing capabilities such as a Global Positioning Satellite receiver, and in response to the detection of a change of location, send a request to transfer the plurality of session data **36**. Alternatively, the second messaging device in which the second messaging client **20** operates can detect

its removal from a charging base. Alternatively, the second messaging device in which the second messaging client **20** operates may have a motion-sensing device such as a tilt sensor whose electrical properties change when under motion. When no device movement is detected or alternatively a device movement program is not included in either the mobile device **90** or the message server **172**, in Step **330**, the process continues to Step **331** in which it is determined whether or not the transfer of at least a portion of the first client data **17** including the plurality of session data **36** is required due to the activation of the second messaging client **20**. The activation of the second messaging client **20** can be, for example, in response to a user input to a power circuit that powers the second messaging client **20**. Alternatively, the activation of the second messaging client **20** can be in response to an instruction command to activate sent from CPU **53** to the fixed messaging client **84** of the fixed network device **50**, from the processor **102** to the mobile messaging client **112** of the mobile device **90**, or from the controller **138** to the cable messaging client **40** of the cable box **136**. In one embodiment of the present invention, the message server **172** is programmed to detect the activation of the second messaging client **20**. In an alternate embodiment, the second messaging client **20** can request the transfer of at least a portion of the first client data **17** including the plurality of session data **36** upon being activated. When the second messaging client **20** is not activated in Step **331**, the process continues to Step **332** in which it is determined whether the second messaging client **20** has connected to the message server **172**. When no connection of the second messaging client **20** is detected, the process returns to Step **326** and continues checking for the various methods in which the data transfer query (Step **304** of FIG. **13**) can be answered in the affirmative. In Step **333**, when there is an affirmative answer to any of the previous Steps **326** to **332**, the query of Step **304** of FIG. **13** of whether or not to transfer data is answered in the affirmative.

FIG. **15** is a flowchart illustrating more detail of the operation of the messaging communication system **10,170**. Specifically, FIG. **15** illustrates various ways in which the first messaging client **14** can operate in relation to the transfer of the plurality of the first client data **17**. The operation begins with Step **302**, in which the first messaging client **14** operates using the first communication connection **16** and accumulates the plurality of session data **36** for each messaging session **40** for which the first messaging client **14** is participating. Next, in Step **334**, the process determines whether it is required or requested to disconnect the first messaging client **14** from the first communication connection **16**. When it is required or requested to disconnect the first messaging client **14** from the first communication connection **16**, in Step **336** the first messaging client **14** is disconnected from the first communication connection **16**. Next, in Step **310**, when the first messaging client **14** is disconnected from the first communication connection **16** in Step **336** and when it is not desired to disconnect the first messaging client **14** from the first communication connection **16** in Step **334**, at least a portion of the first client data **17** including the plurality of session data **36** is transferred from the first messaging client **14** to the second messaging client **20**. Next, in Step **338**, the process once again determines whether it is required or requested to disconnect the first messaging client **14** from the first communication connection **16**. In Step **340**, when it is required or requested to disconnect the first messaging client **14** from the first communication connection **16**, the first messaging client **14** is disconnected from the first communication connection **16**. When no disconnection of the first messaging client **14** is required or requested in Step **338**, the process moves to Step **342** in which the first messaging client **14** continues the first communication connection **16** using the first client data **17** and accumulating the plurality of session data **36**. Next, in Step **324**, and also after disconnecting the first messaging client **14** in Step **340**, the second messaging client **20** operates the second communication connection **22** using the transferred portion of the first client data **17** including the plurality of session data **36**.

The flowchart of FIG. **15** as described herein provides an efficient and flexible method for disconnecting the first messaging client **14** from the first communication connection **16** prior to the transfer of the first client data **17** including the plurality of session data **36** or after the transfer of the first client data **17** including the plurality of session data **36**. Further, it provides a method for the continued operation of the first messaging client **14** on the first communication connection **16** and the second messaging client **20** on the second communication connection **22** using the same plurality of session data **36** included in at least a portion of the first client data **17**.

FIG. **16** is a flowchart illustrating more detail of the operation of the messaging communication system **10,170**. Specifically, FIG. **16** illustrates more detail of the transfer of the first client data **17** from the first messaging client **14** to the second messaging client **20** or alternatively a portion of the first client data **17** such as the client data portion **18** or alternatively the session portion **49**. The operation begins with Step **302**, in which the first messaging client **14** operates using the first communication connection **16** and accumulates the plurality of session data **36** for each messaging session **40** for which the first messaging client **14** is participating. While operating within the first communication connection **16**, the first client data **17** of the first messaging client **14** includes both the client data such as the first client identifier **15** as well as an accumulation of the plurality of session data **36**. Next, in Step **344**, the process determines whether only a portion of the first client data **17** such as the client data portion **18** or the session portion **49** is being transferred. In Step **346**, when the entire first client data **17** is being transferred in Step **344**, the first client data **17** is transferred from the first messaging client **14** to the second messaging client **20**. Thereafter, the second client data **23** of the second messaging client **20** includes the first client data **17** along with any other client data already included within the second client data **23**. It will be appreciated by one of ordinary skill in the art that the transfer of the first client data **17** can be accomplished using a direct connection between the first messaging client **14** and the second messaging client **20** or a connection through the message server **172** both either via a network connection, a wireless connection such as through the wireless communication system **114**, a wired connection such as through the wired communication system **89** or the broadcast messaging system **274**, or an equivalent.

Next, in Step **348**, when a portion of the first client data **17** is being transferred, the process determines whether a client data requirement has been sent from the second messaging client **20** to the first messaging client **14**. In Step **350**, when a client data requirement has been sent from the second messaging client **20** to the first messaging client **14**, the client data portion **18** is determined using the client data requirement. For example, due to memory limitations of the device in which the second messaging client **20** operates, the client data portion **18** can be a defined, limited portion of the session history **45**. As another example, the client data

requirement can be the plurality of user preferences **35** for the first messaging client **14** set by the first account user **29**. It will be appreciated by one of ordinary skill in the art that the client data requirement can be a requirement for all or any portion of the first client data **17** sent from the second messaging client **20**. When no client data requirement has been received by the first messaging client **14** from the second messaging client **20**, the process moves to Step **352** in which the process determines whether a predetermined client data portion **18** has been programmed either into the first messaging client **14** or alternatively into the message server **172**. In Step **354**, when the predetermined client data portion **18** has been programmed, the client data portion **18** is determined using the predetermined client data portion **18**. In Step **356**, when no predetermined portion has been defined in Step **352**, some other method is used to identify the client data portion **18**. It will be appreciated by one of ordinary skill in the art that any other method can be used to identify the client data portion **18** in accordance with the present invention. Next, in Step **358**, when the client data portion **18** has been identified in Step **350**, **354**, or **356**, the client data portion **18** is transferred from the first messaging client **14** to the second messaging client **20**. It will be appreciated by one of ordinary skill in the art that the transfer of the client data portion **18** can be accomplished using a direct connection between the first messaging client **14** and the second messaging client **20** or a connection through the message server **172**, both either via a network connection, a wireless connection such as through the wireless communication system **114**, a wired connection such as through the wired communication system **89** or the broadcast messaging system **274**, or an equivalent.

The flowchart of FIG. **16** provides a method for limiting the amount of client data transferred from the first messaging client **14** to the second messaging client **20**, optimizing the efficiency of the transfer of client data to maintain overall optimal system utilization.

FIG. **17** is a flowchart illustrating further operation of the messaging communication system **10**,**170** in which the first messaging client **14** participates in a plurality of messaging sessions **24**. The operation begins with Step **359** in which the first messaging client establishes the first communication connection **16**. Next, in Step **360**, a counter is set to N=1. Next, in Step **362**, the process determines whether the first messaging client **14** is participating in an Nth messaging session. In Step **364**, when the first messaging client **14** is not participating in the Nth messaging session, the counter is incremented by one (1). Next, in Step **365**, the process determines whether or not the Nth messaging session exists. When the Nth session does not exist, the process ends. When the Nth messaging session does exist, the process returns to Step **362** in which it is determined whether the first messaging client **14** is participating in the Nth messaging session. In Step **366**, when the first messaging client **14** is participating in the Nth messaging session in Step **362**, the Nth messaging session including its associated session data is included in the first communication connection **16**. Next, in Step **368**, it is determined whether the Nth messaging session is requested or required to be transferred from the first messaging client **14** to the second messaging client **20**. When the Nth messaging session is not requested or required to be transferred in Step **368**, the process returns to Step **366** in which the first communication connection **16** continues to include the Nth messaging session. In Step **370**, when, in Step **368**, the Nth messaging session is being transferred, session data for the Nth messaging session is included in the plurality of session data **36** of the first client data **17**. The session data included for the Nth messaging session can be, for example, the Nth session identifier **41**, the Nth session priority **42**, the Nth session preferences **43**, the Nth session participants **44** and/or the Nth session history **45**. Next, in Step **310**, at least a portion of the first client data **17** including the session data for the Nth messaging session is transferred from the first messaging client **14** to the second messaging client **20**. It will be appreciated by one of ordinary skill in the art that, in accordance with the present invention, the plurality of session data for each messaging session to be transferred can be transferred separately. Alternatively, in accordance with the present invention, the plurality of session data for all messaging sessions being transferred can be transferred at one time in one or more communications. Next, in Step **312**, the process determines whether or not the second communication connection **22** has been established. In Step **318**, when the second communication connection **22** has not been established, the second communication connection **22** is established. In Step **372**, when the second communication connection **22** is already established in Step **312**, or after the second communication connection **22** is established in Step **318**, the Nth messaging session, along with any portion of the first client data **17** transferred, is included within the second communication connection **22** for the second messaging client **20**. Next, in Step **374**, the process determines whether a notification of data transfer is required or requested. In step **376**, when a notification is required or requested, the notification is sent. It will be appreciated by one of ordinary skill in the art that the notification of data transfer can be sent for each messaging session separately. Alternatively, in accordance with the present invention, the notification of data transfer can be sent for all messaging sessions being transferred in one notification message. In accordance with the present invention, the notification of data transfer can be sent to at least one of the plurality of messaging clients **12** participating in the Nth messaging session. Alternatively or additionally, the notification of data transfer can be sent to the message server **172**. The notification of data transfer can be sent from the first messaging client **14**, from the second messaging client **20**, from the message server **172**, or an equivalent. Preferably, the notification includes the client profile **85** stored in the device in which the second messaging client **20** operates. The message communication system **10**, **170**, including one or more of the plurality of messaging clients **12** and/or the message server **172**, can modify the content sent to the second messaging client **20** based on the client profile **85**. Next, the process returns to Step **364** in which the counter is incremented.

FIG. **18** is a flowchart illustrating one embodiment of the operation of the messaging communication system **170** in which the plurality of messaging sessions **24** includes the multiple user messaging session **19**. The operation begins with Step **378** in which the multiple user messaging session **19** is established within the messaging communication system **170**. The multiple user messaging session **19** includes the plurality of session messages **61** among the plurality of messaging clients **12**. Next, in Step **380**, the process determines whether the first messaging client **12** is participating in the multiple user messaging session **19**. When the first messaging client **12** is not participating in the multiple user messaging session **19**, the process ends. In Step **381**, when the first messaging client **12** is participating in the multiple user messaging session **19**, the first communication connection **16** includes the multiple user messaging session **19**. Further, the data for the multiple user messaging session **19** is part of the plurality of session data **36**. Data for the

multiple user messaging session **19** can be, for example, the session identifier **41**, the session priority **42**, the session preferences **43**, the session participants **44** and/or the session history **45** of the multiple user messaging session **19**. Next, in Step **304**, the process determines whether a data transfer is required or requested. When no data transfer is required or requested in Step **304**, the first communication connection **16** including the multiple user messaging session **19** is maintained in Step **381**. It will be appreciated by one of ordinary skill in the art that the plurality of session data **36** for the multiple user messaging session **19** is updated periodically as the multiple user messaging session **19** continues. (not shown) Next, in Step **310**, when a data transfer is requested or required in Step **304** at least a portion of the first client data **17** including the plurality of session data **36** is transferred from the first messaging client **14** to the second messaging client **20**.

Next, in Step **382**, a data transfer message **384** is sent to the message server **170**. Preferably, the data transfer message **384** is as illustrated in FIG. **19**. The data transfer message **384** preferably includes a session reservation **385**. For example, the session reservation **385** could save a connection within the multiple user messaging session **19** for any messaging client that is being used by the same account identifier used in the first messaging client **14**. As shown in FIG. **19**, the data transfer message **384** alternatively includes the session identifier **41** of the multiple user messaging session **19**, the first client identifier **15** of the first messaging client **14**, the second client identifier **21** of the second messaging client **20**, and the session reservation **385**. The session reservation **385** saves a connection within the multiple user messaging session **19** for the second messaging client **20** having the second client identifier **21**. It will be appreciated by one of ordinary skill in the art that the data transfer message **384** can be sent using a network connection, a wireless connection such as through the wireless communication system **114**, a wired connection such as through the wired communication system **89** or the broadcast messaging system **274**, or an equivalent.

Referring back to FIG. **18**, next, in Step **383**, the second messaging client **20** establishes the second communication connection **22** for participating within the multiple user messaging session **19**. In one embodiment of the present invention, the message server **172** can require that Step **383** be performed within a specific time period after it received the data transfer message **384**. (not shown) If this time is exceeded, the message server **172** can release the reserved seat to be used by any of the plurality of messaging clients **12**. The operation of the message communication system **170** as illustrated in FIG. **18** provides a means for the first account user **29** to ensure that there is an opening within the multiple user messaging session **19** when the first account user **29** transfers at least a portion of the first client data **17** including the plurality of session data **36** (and accordingly the communication means) from the first messaging client **14** to the second messaging client **20**. This operation is especially beneficial in situations in which there are a limited number of available openings within the multiple user messaging session **19** and the first account user **29** could lose his/her space during the transfer of data from one messaging client to another messaging client.

FIG. **20** is a signaling flow diagram illustrating an example of the interaction between the elements of the messaging communication system **10**, **170**, according to the present invention. Specifically, FIG. **20** illustrates the interaction between the first messaging client **14**, the second messaging client **20**, the messaging client **26**, and the message server **172**. In accordance with the present invention, as illustrated in FIG. **20**, a second account user, such as the account user **30**, logs onto the messaging client **26** and sends a notification signal **388** to the message server **172**. Preferably, the notification signal **388** further includes the second account identifier of the second account user. The notification signal **388** for example, includes the connection information (i.e.: IP address and number of the port assigned to the messaging client) of the messaging client **26**. Similarly, the first account user **29** logs onto the first messaging client **14** and sends a notification signal **386** to the message server **172**. The notification signal **386** for example, includes the connection information (i.e.: IP address and number of the port assigned to the messaging client) of the first messaging client **14**. Preferably, the notification signal **386** also includes the first account identifier of the first account user **29**. It will be appreciated by one of ordinary skill in the art that alternatively, the notification signals **386** and **388** can be sent directly to one or more of the plurality of messaging clients **12**. In response to receiving the notification signal **386** from the messaging client **26**, and receiving the notification signal **388** from the first messaging client **14**, the message server **172** sends a client availability signal **390** to the messaging client **26**. The client availability signal **390** informs the second account user via the messaging client **26** that the first account user **29** is available for real time electronic communications such as for participation in one or more of the plurality of messaging sessions **24**. Similarly, in response to receiving the notification signal **386** from the messaging client **26**, and receiving the notification signal **388** from the first messaging client **14**, the message server **172** sends a client availability signal **392** to the first messaging client **14**. The client availability signal **392** informs the first account user **29** via the first messaging client **14** that the second account user is available for real time electronic communications such as for participation in one or more of the plurality of messaging sessions **24**. Next, the first account user **29** initiates the messaging session **40** with the second account user by sending a session message **394** to the message server **172**. The message server **172**, acting as a store and forward device, sends a session message signal **396** containing substantially the same message information as the session message **394** to the second account user via the messaging client **26**. In response to receiving the session message signal **396**, a window is created on the display of the messaging device in which the second messaging client **26** operates and the session message **46**, preferably along with the first account identifier of the first account user **29**, is displayed in the created window. Next, the second account user via the messaging client **26** sends a response message **398** to the message server **172**. The message server **172**, acting as a store and forward device, sends a response message signal **400** to the first account user **29** via the first messaging client **14** containing substantially the same message information as the response message **398**. In response to receiving the response message signal **400**, the created messaging session window is updated on the display of the messaging device in which the first messaging client **14** operates and the session message contained within the response message **398**, preferably along with the second account identifier of the second account user, is displayed. Although one session message **394** and one response message **398** is illustrated by way of example in FIG. **20**, it will be appreciated by one of ordinary skill in the art that the messaging session **40** between the first account user's first messaging client **14** and the second account user's messaging client **26** can include a plurality of session messages and

a plurality of response messages. Further, although the interaction of two account users and two messaging clients is illustrated by way of example in FIG. **20**, it will be appreciated by one of ordinary skill in the art that the messaging session **40** can include a plurality of messaging clients and an associated plurality of account users. Further, it will be appreciated by one of ordinary skill in the art that the session message **394** can be sent directly from the first messaging client **14** to the messaging client **26**; and similarly the response message **398** can be sent directly from the messaging client **26** to the first messaging client **14**, without the interface of the message server **170**, in accordance with the present invention.

According to the present invention, an account user can choose to initiate a data transfer. As illustrated in FIG. **20**, the first account user **29** via the first messaging client **14** sends a transfer request signal **402** including at least a portion of the first client data **17** such as the plurality of session data **36** currently contained on the first account user's first messaging client **14** to the message server **172**. The transfer request signal **402** preferably also includes a request to the message server **172** to transfer at least a portion of the first client data **17** including the plurality of session data **36** to a new messaging client such as the second messaging client **20**. For example, the first messaging client **14** can be a fixed personal computer such as the fixed network device **50** in the office of the first account user **29**. The first account user **29** has the need to become mobile. The second messaging client **20** can be a cellular telephone such as the mobile device **90**. The first account user **29**, according to the present invention, can pass the current messaging session from his/her personal computer to his/her cellular telephone with no loss of communication or of session data. Similarly, the transfer request signal **402** can include a request for the message server **172** to pass the plurality of session data **36** for more than one messaging session. In response to receiving the transfer request signal **402**, the message server **172** determines whether the second messaging client **20** is currently connected onto the message server **172**. (not shown) For example, the message server **172** determines whether the second messaging client **20** has established the second communication connection **22**. When the second messaging client **20** is not currently connected with the message server **172**, the message server **172** stores the plurality of session data **36** and/or the first client data **17** if so requested until the second messaging client **20** is connected. (not shown) When the second messaging client **20** is connected to the message server **172**, the message server **172** sends the data signal **404** including the plurality of session data **36** and/or any portion of the first client data **17** received from the first messaging client **14** within the transfer request signal **404** to the second messaging client **20**. The second messaging client **20** stores the plurality of session data **36** and/or the portion of the first client data **17** in memory and displays the session history **45** for access and use by the first account user **29** on the display of the messaging device in which the second messaging client **20** operates. It will be appreciated by one of ordinary skill in the art that the data signal **404** can include the plurality of session data **36** for one messaging session or for a plurality of messaging sessions, or can include the first client data **17** or the client data portion **18** of the first client data **17** for the first messaging client **14**. Similarly the messaging device in which the second messaging client **20** operates can store one messaging session or a plurality of messaging sessions, the first client data **17** or the client data portion **18** of the first client data **17** in its memory in response to receiving the data signal **404**. Preferably, in response to receiving the data signal **404**, the second messaging client **20** sends an acknowledgement signal **406** to the message server **172**. The message server **172** also preferably sends a transfer acknowledgement signal **410** to the first account user's first messaging client **14**. The messaging session **40** seamlessly continues between the first account user **29** and the second account user through the second messaging client **20** and the messaging client **26** as illustrated by the plurality of session messages **412** to **422**. The messaging session **40** continues seamlessly without the second account user being necessarily aware of the transfer of the first client data from the first account user's first messaging client **14** to his/her second messaging client **20**. It will be appreciated by one of ordinary skill in the art that although only a second account user is shown in FIG. **20** by way of example, the messaging session **40** can continue seamlessly between a plurality of account users and associated plurality of messaging clients.

Preferably, the first messaging client **14** is disconnected from the messaging session **40** upon completion of the data transfer. (not shown) It will be appreciated by one of ordinary skill in the art that the first messaging client **14** can be automatically disconnected from the messaging session **40** or alternatively the first messaging client **14** can be disconnected manually by the first account user **29**. Similarly, it will be appreciated by one of ordinary skill in the art the first messaging client **14** can continue to be active in the messaging session **40** along with the second messaging client **20**. (not shown)

FIG. **21** is a signaling flow diagram illustrating the interaction between the elements of the messaging communication system **10**,**170**, according to the present invention. Specifically, FIG. **21** illustrates the interaction between the first messaging client **14**, the second messaging client **20**, the messaging client **26**, and the message server **172**, according to the present invention. In accordance with the present invention, as illustrated in FIG. **21**, a second account user, such as the account user **30**, logs onto the messaging client **26** and sends a notification signal **388** to the message server **172**. For example, the messaging client **26** establishes the communication connection **28**. The notification signal **388** for example, includes the connection information (i.e.: IP address and number of the port assigned to the messaging client) of the messaging client **26**. Preferably, the notification signal **388** further includes the second account identifier of the second account user. Similarly, the first account user **29** logs onto the first messaging client **14** and sends a notification signal **386** to the message server **172**. For example, the first messaging client **14** establishes the first communication connection **16**. The notification signal **386** for example, includes the connection information (i.e.: IP address and number of the port assigned to the messaging client) of the first messaging client **14**. Preferably, the notification signal **386** also includes the first account identifier of the first account user **29**. It will be appreciated by one of ordinary skill in the art that alternatively, the notification signals **386** and **388** can be sent directly to one or more of the plurality of messaging clients **12**. In response to receiving the notification signal **386** from the messaging client **26**, and receiving the notification signal **388** from the first messaging client **14**, the message server **172** sends a client availability signal **390** to the messaging client **26**. The client availability signal **390** informs the second account user via the messaging client **26** that the first account user **29** is available for real time electronic communications such as participation in one or more messaging sessions. Similarly, in response to receiving the notification signal **386** from the

messaging client **26**, and receiving the notification signal **388** from the first messaging client **14**, the message server **172** sends a client availability signal **392** to the first messaging client **14**. The client availability signal **392** informs the first account user **29** via the first messaging client **14** that the second account user is available for real time electronic communications such as participation in one or more messaging sessions. Next, the first account user **29** initiates the messaging session **40** with the second account user by sending a session message **424** to the message server **172**. The message server **172**, in response to receiving the session message **424** sends a messaging session participation request **426** to the second account user via the messaging client **26**. The messaging client **26** asks the second account user if he/she wants to participate in the messaging session **40** with the first account user **29**. When the second account user does not accept the messaging session participation request **426**, the process stops. (not shown) When the second account user does accept the messaging session participation request **426**, the messaging client **26** sends a messaging session participation acceptance signal **428** to the message server **172**. The message server **172**, in response to receiving the messaging session participation acceptance signal **428**, sends a session message signal **396** containing substantially the same message information as the session message **424** to the second account user via the messaging client **26**. In response to receiving the session message signal **396**, a window is created on the display of the messaging device in which the messaging client **26** operates and the session message **46**, preferably along with the first account identifier of the first account user **29**, is displayed on the created window. Next, the second account user via the messaging client **26** sends a response message **398** to the message server **172**. The message server **172**, acting as a store and forward device, sends a response message signal **400** to the first account user **29** via the first messaging client **14** containing substantially the same message information as the response message **398**. In response to receiving the response message signal **400**, the open display window is updated on the display of the messaging device in which the first messaging client **14** operates and the session message, preferably along with the second account identifier of the second account user, is displayed. Although one session message **396** and one response message **398** is illustrated by way of example in FIG. **21**, it will be appreciated by one of ordinary skill in the art that the messaging session **40** between the first account user's first messaging client **14** and the second account user's messaging client **26** can include a plurality of session messages and a plurality of response messages. Further, although the interaction of two account users and two messaging clients is illustrated by way of example in FIG. **21**, it will be appreciated by one of ordinary skill in the art that the messaging session can include a plurality of messaging clients and an associated plurality of account users.

According to the present invention, an account user can choose to initiate a data transfer. As illustrated in FIG. **21**, the first account user **29** via the second messaging client **20** sends a transfer request signal **430** to the first account user's first messaging client **14**. In response, the first messaging client **14** sends a connection discovery and verification signal **432** to the second messaging client **20**. The second messaging client **20** then sends an acknowledgement signal **434** to the first messaging client **14**. The acknowledgement signal **434** preferably includes verification data in which the first messaging client **14** can verify the validity of the second messaging client **20**. The first messaging client **14** then sends a data signal **436** to the second messaging client **20**. In a preferred embodiment, the second messaging client **20** includes session transfer capabilities. Alternatively, the messaging device in which the second messaging client **20** operates includes the data transfer application **83**. The first account user **29** launches the data transfer application **83** or alternatively the data transfer capabilities of the second messaging client **20**, and utilizes the data transfer application **83** and/or the second messaging client **20** to achieve the transfer of at least a portion of the first client data **17** including the plurality of session data **36** from the first messaging client **14**. The data transfer application **83** stores first client data received including the plurality of session data **36** for the messaging session **40** in the memory of the messaging device and launches the second messaging client **20** if it is not already active. The messaging device in which the second messaging client **20** operates displays the session history **45** for access and use by the first account user **29** on the messaging device in which the second messaging client **20** operates. It will be appreciated by one of ordinary skill in the art that the session data signal **436** can include the plurality of session data **36** for one messaging session or for a plurality of messaging sessions, or can include the first client data **17** or the client data portion **18** of the first client data **17** for the first messaging client **14**. Similarly the messaging device in which the second messaging client **20** operates can store one messaging session or a plurality of messaging sessions, the first client data **17** or the client data portion **18** of the first client data **17** in its memory in response to receiving the session data signal **436**.

The messaging session **40** continues between the first account user **29** and the second account user through the second messaging client **20** and the messaging client **26** as illustrated by the plurality of session messages **412** to **422**. The messaging session **40** continues seamlessly without the second account user being necessarily aware of the transfer of the portion of the first client data **17** including the plurality of session data **36** from the first messaging client **14** to the second messaging client **20**. It will be appreciated by one of ordinary skill in the art that although only a second account user is shown in FIG. **21** by way of example, the messaging session **40** can continue seamlessly between a plurality of account users and associated plurality of messaging clients.

Preferably, the first messaging client **14** is disconnected from the messaging session **40** upon completion of the data transfer. (not shown) It will be appreciated by one of ordinary skill in the art that the first messaging client **14** can be automatically disconnected from the messaging session **40** or alternatively the first messaging client **14** can be disconnected manually by the first account user **29**. Similarly, it will be appreciated by one of ordinary skill in the art the first messaging client **14** can continue to be active in the messaging session **40** along with the second messaging client **20**. (not shown)

FIG. **22** is a signaling flow diagram illustrating the interaction between the elements of the messaging communication system **10**, **170**, according to the present invention. Specifically, FIG. **22** illustrates the interaction between the first messaging client **14**, the second messaging client **20**, the messaging client **26**, and the message server **172**. In accordance with the present invention, as illustrated in FIG. **22**, a second account user, such as the account user **30**, logs onto the messaging client **26** and sends a notification signal **388** to the message server **172**. For example, the messaging client **26** establishes the communication connection **28**. The notification signal **388** for example, includes the connection

information (i.e.: IP address and number of the port assigned to the messaging client) of the messaging client **26**. Preferably, the notification signal **388** further includes the second account identifier of the second account user. Similarly, the first account user **29** logs onto the first messaging client **14** and sends a notification signal **386** to the message server **172**. For example, the first messaging client **14** establishes the first communication connection **16**. The notification signal **386** for example, includes the connection information (i.e.: IP address and number of the port assigned to the messaging client) of the first messaging client **14**. Preferably, the notification signal **386** also includes the first account identifier of the first account user **29**. It will be appreciated by one of ordinary skill in the art that alternatively, the notification signals **386** and **388** can be sent directly to one or more of the plurality of messaging clients **12**. In response to receiving the notification signal **386** from the messaging client **26**, and receiving the notification signal **388** from the first messaging client **14**, the message server **172** sends a client availability signal **390** to the messaging client **26**. The client availability signal **390** informs the second account user via the messaging client **26** that the first account user **29** is available for real time electronic communications such as participation in one or more of the plurality of messaging sessions **24**. Similarly, in response to receiving the notification signal **386** from the messaging client **26**, and receiving the notification signal **388** from the first messaging client **14**, the message server **172** sends a client availability signal **392** to the first messaging client **14**. The client availability signal **392** informs the first account user **29** via the first messaging client **14** that the second account user is available for real time electronic communications such as participation in one or more of the plurality of messaging sessions **24**. Next, the first account user **29** initiates the messaging session **40** with the second account user by sending a session message **424** to the message server **172**. The message server **172**, in response to receiving the real time electronic message **424** sends a messaging session participation request **426** to the second account user via the messaging client **26**. The messaging client **26** asks the second account user if he/she wants to participate in the messaging session **40** with the first account user **29**. When the second account user does not accept the messaging session participation request **426**, the process stops. (not shown) When the second account user does accept the messaging session participation request **426**, the messaging client **26** sends a messaging session participation acceptance signal **428** to the message server **172**. The message server **172**, in response to receiving the messaging session participation acceptance signal **428**, sends a session message signal **396** containing substantially the same message information as the session message **424** to the second account user via the messaging client **26**. In response to receiving the session message signal **396**, a window is created on the display of the messaging device in which the messaging client **26** operates and the session message **46**, preferably along with the first account identifier of the first account user **29**, is displayed. Next, the second account user via the messaging client **26** sends a response message **398** to the message server **172**. The message server **172**, acting as a store and forward device, sends a response message signal **400** to the first account user **29** via the first messaging client **14** containing substantially the same message information as the response message **398**. In response to receiving the response message signal **400**, the open display window is updated on the display of the messaging device in which the first messaging client **14** operates and the session message, preferably along with the second account identifier of the second account user, is displayed. Although one session message **396** and one response message **398** is illustrated by way of example in FIG. **22**, it will be appreciated by one of ordinary skill in the art that the messaging session **40** between the first account user's first messaging client **14** and the second account user's messaging client **26** can include a plurality of session messages and a plurality of response messages. Further, although the interaction of two account users and two messaging clients is illustrated by way of example in FIG. **22**, it will be appreciated by one of ordinary skill in the art that the messaging session **40** can include a plurality of messaging clients and an associated plurality of account users.

According to the present invention, an account user can choose to initiate a data transfer. As illustrated in FIG. **22**, the first account user **29** via the second messaging client **20** sends a transfer request signal **430** to the first account user's first messaging client **14**. In response, the first messaging client **14** sends a connection discovery and verification signal **432** to the second messaging client **20**. The second messaging client **20** then sends an acknowledgement signal **434** to the first messaging client **14**. The acknowledgement signal **434** preferably includes verification data in which the first messaging client **14** can verify that the validity of the second messaging client **20**. In response to receiving the acknowledgement signal **434**, the first messaging client **14** sends a request for a key **438** to the message server **172**. Next, the message server **172** sends a key signal **440** to the first messaging client **14**. The first messaging client **14** then sends the data and key signal **442** to the second messaging client **20**. The key preferably includes a code by which the second messaging client **20** can access the messaging session **40**. The second messaging client **20** stores the transferred portion of the first client data **17** including the plurality of session data **36** and the key for the messaging session **40** in the memory of the messaging device in which the second messaging client **20** operates, and displays the session history **45** for access and use by the first account user **29** on the display of the messaging client in which the second messaging client **20** operates. It will be appreciated by one of ordinary skill in the art that the data and key signal **442** can include session data for one messaging session or for a plurality of messaging sessions, the first client data **17** or a portion of the first client data **17**; and similarly that the second messaging client **20** can store one messaging session or a plurality of messaging sessions, the first client data **17** or a portion of the first client data **17** in memory in response to receiving the data and key signal **442**. The second messaging client **20** then launches the data transfer application **83** or alternatively runs the data transfer software contained within the second messaging client **20**. Further, the second messaging client **20** can cause the messaging device in which the second messaging client **20** operates to display the session history **45** received from the first messaging client **14** within the data and key signal **442** for viewing by the first account user **29**. Next, the second messaging client **20** sends a request for connection signal **444** to the message server **172**. In response, the message server **172** sends a security challenge signal **446** to the second messaging client **20**. The second messaging client **20** responds to the security challenge signal **446** with a security response signal **448** which may be calculated from the security challenge signal and the key to the message server **172**. Then the message server **172** sends an acknowledgement of transfer complete signal **450** to the second messaging client **20**. The messaging session **40** has now been

transferred from the first messaging client **14** to the second messaging client **20**. Preferably, the second messaging client **20** also sends an acknowledgement of transfer to the message server **172**. (not shown)

The messaging session **40** continues between the first account user **29** and the second account user through the second messaging client **20** and the messaging client **26** as illustrated by the plurality of session messages **412** to **422**. The messaging session **40** continues seamlessly without the second account user being necessarily aware of the transfer of the portion of the first client data **17** including the plurality of session data **36** from the first messaging client **14** to the second messaging client **20**. It will be appreciated by one of ordinary skill in the art that although only a second account user is shown in FIG. **22** by way of example, the messaging session **40** can continue seamlessly between a plurality of account users and associated plurality of messaging clients.

Preferably, the first messaging client **14** is disconnected from the messaging session **40** upon completion of the data transfer. (not shown) It will be appreciated by one of ordinary skill in the art that the first messaging client **14** can be automatically disconnected from the messaging session **40** or alternatively the first messaging client **14** can be disconnected manually by the first account user **29**. Similarly, it will be appreciated by one of ordinary skill in the art that the first messaging client **14** can continue to be active in the messaging session **40** along with the second messaging client **20**. (not shown)

FIG. **23** is a signaling flow diagram illustrating the interaction between the elements of the messaging communication system **10**,**170**, according to the present invention. Specifically, FIG. **23** illustrates the interaction between the first messaging client **14**, the second messaging client **20**, the messaging client **26**, and the message server **172**. In accordance with the present invention, as illustrated in FIG. **23**, a second account user, such as the account user **30**, logs onto the messaging client **26** and sends a notification signal **388** to the message server **172**. For example, the messaging client **26** establishes the communication connection **28**. The notification signal **388** for example, includes the connection information (i.e.: IP address and number of the port assigned to the messaging client) of the messaging client **26**. Preferably, the notification signal **388** further includes the second account identifier of the second account user. Similarly, a first account user **29** logs onto the first messaging client **14** and sends a notification signal **386** to the message server **172**. For example, the first messaging client **14** establishes the first communication connection **16**. The notification signal **386** for example, includes the connection information (i.e.: IP address and number of the port assigned to the messaging client) of the first messaging client **14**. Preferably, the notification signal **386** also includes the first account identifier of the first account user **29**. It will be appreciated by one of ordinary skill in the art that alternatively, the notification signals **386** and **388** can be sent directly to one or more of the plurality of messaging clients **12**. In response to receiving the notification signal **386** from the messaging client **26**, and receiving the notification signal **388** from the first messaging client **14**, the message server **172** sends a client availability signal **390** to the messaging client **26**. The client availability signal **390** informs the second account user via the messaging client **26** that the first account user **29** is available for real time electronic communications such as for participation in one or more of the plurality of messaging sessions **24**. Similarly, in response to receiving the notification signal **386** from the messaging client **26**, and receiving the notification signal **388** from the first messaging client **14**, the message server **172** sends a client availability signal **392** to the first messaging client **14**. The client availability signal **392** informs the first account user **29** via the first messaging client **14** that the second account user is available for real time electronic communications such as participation in one or more of the plurality of messaging sessions **24**. Next, the first account user **29** initiates the messaging session **40** with the second account user by sending a session message **424** to the message server **172**. The message server **172**, in response to receiving the session message **424** sends a messaging session participation request **426** to the second account user via the messaging client **26**. The messaging client **26** asks the second account user if he/she wants to participate in the messaging session **40** with the first account user **29**. When the second account user does not accept the messaging session participation request **426**, the process stops. (not shown) When the second account user does accept the messaging session participation request **426**, the messaging client **26** sends a messaging session participation acceptance signal **428** to the message server **172**. The message server **172**, in response to receiving the messaging session participation acceptance signal **428**, sends a session message signal **396** containing substantially the same message information as the session message **424** to the second account user via the messaging client **26**. In response to receiving the session message signal **396**, a window is created on the display of the messaging device in which the messaging client **26** operates and the session message, preferably along with the first account identifier of the first account user **29**, is displayed. Next, the second account user via the messaging client **26** sends a response message **398** to the message server **172**. The message server **172**, acting as a store and forward device, sends a response message signal **400** to the first account user **29** via the first messaging client **14** containing substantially the same message information as the response message **398**. In response to receiving the response message signal **400**, the open window is updated on the display of the messaging device in which the first messaging client **14** operates and the session message, preferably along with the second account identifier of the second account user, is displayed. Although one session message **396** and one response message **398** is illustrated by way of example in FIG. **23**, it will be appreciated by one of ordinary skill in the art that the messaging session **40** between the first account user's first messaging client **14** and the second account user's messaging client **26** can include a plurality of session messages and a plurality of response messages. Further, although the interaction of two account users and two messaging clients is illustrated by way of example in FIG. **23**, it will be appreciated by one of ordinary skill in the art that the messaging session **40** can include a plurality of messaging clients and an associated plurality of account users.

According to the present invention, an account user can choose to launch data transfer software. As illustrated in FIG. **23**, the first account user **29** launches the data transfer software within the second messaging client **20** or alternatively the data transfer application **83**. Upon launching the data transfer application **83** or alternatively the data transfer software within the second messaging client **20**, the second messaging client **20** sends a notification of availability signal **452** to the message server **172**. In response, the message server **172** sends an availability signal **454** to the first messaging client **14** and an availability signal **456** to the messaging client **26**. Preferably the signals **452** and **454** include identification information for the second messaging

client **20** such as the second client identifier **21**. As illustrated, after receiving the availability notification signal **454**, the first messaging client **14** sends a data signal **458** to the second messaging client **20**. The second messaging client **20** stores the received portion of the first client data **17** including the plurality of session data **36** for the messaging session **40** in the memory of its associated messaging device and causes the session history **45** to be displayed on the display of the messaging device in which the second messaging client **20** operates for access and use by the first account user **29**. It will be appreciated by one of ordinary skill in the art that the data signal **458** can include session data for one messaging session or for a plurality of messaging sessions, the first client data **17**, or a portion of the first client data **17**; and similarly that the second messaging client **20** can store one messaging session or a plurality of messaging sessions, the first client data **17**, or a portion of the first client data **17** in its memory in response to receiving the session data signal **458**.

The messaging session **40** continues between the first account user **29** and the second account user through the second messaging client **20** and the messaging client **26** as illustrated by the plurality of session messages **412** to **422**. The messaging session **40** continues seamlessly. It will be appreciated by one of ordinary skill in the art that although only a second account user is shown in FIG. **23** by way of example, the messaging session **40** can continue seamlessly between a plurality of account users and associated plurality of messaging clients.

Preferably, the first messaging client **14** is disconnected from the messaging session **40** upon completion of the data transfer. (not shown) It will be appreciated by one of ordinary skill in the art that the first messaging client **14** can be automatically disconnected from the messaging session **40** or alternatively the first messaging client **14** can be disconnected manually by the first account user **29**. Similarly, it will be appreciated by one of ordinary skill in the art the first messaging client **14** can continue to be active in the messaging session **40** along with the second messaging client **20**. (not shown)

FIG. **24** is a signaling flow diagram illustrating the interaction between the elements of the messaging communication system **10**, **170**, according to the present invention. Specifically, FIG. **24** illustrates the interaction between the first messaging client **14**, the second messaging client **20**, the messaging client **26**, and the message server **172**. In accordance with the present invention, as illustrated in FIG. **24**, a second account user, such as the account user **30**, logs onto the messaging client **26** and sends a notification signal **388** to the message server **172**. The notification signal **388** for example, includes the connection information (i.e.: IP address and number of the port assigned to the messaging client) of the messaging client **26**. Preferably, the notification signal **388** further includes the second account identifier of the second account user. Similarly, a first account user **29** logs onto the first messaging client **14** and sends a notification signal **386** to the message server **172**. The notification signal **386** for example, includes the connection information (i.e.: IP address and number of the port assigned to the messaging client) of the first messaging client **14**. Preferably, the notification signal **386** also includes the first account identifier of the first account user **29**. It will be appreciated by one of ordinary skill in the art that alternatively, the notification signals **386** and **388** can be sent directly to one or more of the plurality of messaging clients **12**. In response to receiving the notification signal **386** from the messaging client **26**, and receiving the notification signal **388** from the first messaging client **14**, the message server **172** sends a client availability signal **390** to the messaging client **26**. The client availability signal **390** informs the second account user via the messaging client **26** that the first account user **29** is available for real time electronic communications such as participation in one or more of the plurality of messaging sessions **24**. Similarly, in response to receiving the notification signal **386** from the messaging client **26**, and receiving the notification signal **388** from the first messaging client **14**, the message server **172** sends a client availability signal **392** to the first messaging client **14**. The client availability signal **392** informs the first account user **29** via the first messaging client **14** that the second account user is available for real time electronic communications such as participation in one or more of the plurality of messaging sessions **24**. Next, the first account user **29** initiates the messaging session **40** with the second account user by sending a session message **424** to the message server **172**. The message server **172**, in response to receiving the session message **424** sends a messaging session participation request **426** to the second account user via the messaging client **26**. The messaging client **26** asks the second account user if he/she wants to participate in the messaging session **40** with the first account user **29**. When the second account user does not accept the messaging session participation request **426**, the process stops. (not shown) When the second account user does accept the messaging session participation request **426**, the messaging client **26** sends a messaging session participation acceptance signal **428** to the message server **172**. The message server **172**, in response to receiving the messaging session participation acceptance signal **428**, sends a session message signal **396** containing substantially the same message information as the session message **424** to the second account user via the messaging client **26**. In response to receiving the real time electronic message signal **396**, a window is created on the display of the messaging device in which the messaging client **26** operates and the session message **46**, preferably along with the first account identifier of the first account user **29**, is displayed. Next, the second account user via the messaging client **26** sends a response message **398** to the message server **172**. The message server **172**, acting as a store and forward device, sends a response message signal **400** to the first account user **29** via the first messaging client **14** containing substantially the same message information as the response message **398**. In response to receiving the response message signal **400**, the open window is updated on the display of the messaging device in which the first messaging client **14** operates and the session message, preferably along with the second account identifier of the second account user, is displayed. Although one session message **396** and one response message **398** is illustrated by way of example in FIG. **23**, it will be appreciated by one of ordinary skill in the art that the messaging session **40** between the first account user's first messaging client **14** and the second account user's messaging client **26** can include a plurality of session messages and a plurality of response messages. Further, although the interaction of two account users and two messaging clients is illustrated by way of example in FIG. **23**, it will be appreciated by one of ordinary skill in the art that the messaging session **40** can include a plurality of messaging clients and an associated plurality of account users.

According to the present invention, an account user can choose to launch data transfer software. As illustrated in FIG. **24**, the first account user **29** launches the transfer software within the second messaging client **20** or alternatively the data transfer application **83**. Upon launching the

data transfer application **83** or alternatively the transfer software within the second messaging client **20**, the second messaging client **20** sends a notification of availability signal **452** to the message server **172**. In response, the message server **172** sends an availability signal **454** to the first messaging client **14** and an availability signal **456** to the messaging client **26**. Preferably the signals **452** and **454** include identification information for the second messaging client **20** such as the second messaging client identification **382**. As illustrated, after receiving the availability signal **454**, the first messaging client **14** sends a data signal **460** to the message server **172**. In response, the message server **172** sends a data signal **462** to the second messaging client **20**. The second messaging client **20** stores the received portion of the first client data **17** including the plurality of session data **36** for the messaging session **40** in memory and causes the session history **45** to be displayed on the display of the messaging device in which the second messaging client **20** operates for access and use by the first account user **29** on the second messaging client **20**. It will be appreciated by one of ordinary skill in the art that the data signals **460** and **462** can include session data for one messaging session or for a plurality of messaging sessions, the first client data **17**, or a portion of the first client data **17**; and similarly that the second messaging client **20** can store one messaging session or a plurality of messaging sessions, the first client data **17**, or a portion of the first client data **17** in memory in response to receiving the data signals **460** and **462**.

The messaging session **40** continues between the first account user **29** and the second account user through the second messaging client **20** and the messaging client **26** as illustrated by the plurality of session messages **412** to **422**. It will be appreciated by one of ordinary skill in the art that although only the first account user **29** and a second account user are shown in FIG. **23** by way of example, the messaging session **40** can continue seamlessly between a plurality of account users and associated plurality of messaging clients.

Preferably, the first messaging client **14** is disconnected from the messaging session **40** upon completion of the data transfer. (not shown) It will be appreciated by one of ordinary skill in the art that the first messaging client **14** can be automatically disconnected from the messaging session **40** or alternatively the first messaging client **14** can be disconnected manually by the first account user **29**. Similarly, it will be appreciated by one of ordinary skill in the art the first messaging client **14** can continue to be active in the messaging session **40** along with the second messaging client **20**. (not shown)

Although the invention has been described in terms of preferred embodiments, it will be obvious to those skilled in the art that various alterations and modifications may be made without departing from the invention. Accordingly, it is intended that all such alterations and modifications be considered as within the spirit and scope of the invention as defined by the appended claims.

What is claimed is:

1. Within a messaging communication system having a message server for managing the communication of a plurality of messages among a plurality of messaging clients, a method for providing continuity between the plurality of messaging clients comprising:

establishing a first communication connection including a plurality of client data between a first messaging client and the message server;

transferring the plurality of client data from the first messaging client to a second messaging client; and

establishing a second communication connection including the plurality of client data between the second messaging client and the message server.

2. A method for providing continuity between a plurality of messaging clients as recited in claim **1** wherein the first messaging client further includes at least one user preference, the method further comprising:

transferring the at least one user preference from the first messaging client to the second messaging client; and

operating within the second communication connection by the second messaging client using the at least one user preference.

3. A method for providing continuity between a plurality of messaging clients as recited in claim **1** wherein the first messaging client operates within a first messaging device, and further wherein the first messaging device includes a user interface, the method further comprising prior to the transferring step:

requesting the transfer of the plurality of client data by a user input to the user interface of the first messaging device.

4. A method for providing continuity between a plurality of messaging clients as recited in claim **1** wherein the second messaging client operates within a second messaging device, and further wherein the second messaging device includes a user interface, the method further comprising prior to the transferring step:

requesting the transfer of the plurality of client data by a user input to the user interface of the second messaging device.

5. A method for providing continuity between a plurality of messaging clients as recited in claim **1** wherein the second messaging client operates within a mobile device, wherein in the transferring step the transfer of the plurality of client data is in response to a movement of the mobile device.

6. A method for providing continuity between a plurality of messaging clients as recited in claim **1** wherein in the transferring step the transfer of the plurality of client data is in response to an activation of the second messaging client.

7. A method for providing continuity between a plurality of messaging clients as recited in claim **1** wherein the second messaging client operates within a second messaging device, wherein the second messaging device includes a data transfer application, and further wherein in the transferring step the transfer of the plurality of client data is in response to an activation of the data transfer application.

8. A method for providing continuity between a plurality of messaging clients as recited in claim **1** wherein the first messaging client operates within a first messaging device, wherein the first messaging device includes a data transfer application, and further wherein in the transferring step the transfer of the plurality of client data is in response to an activation of the data transfer application.

9. A method for providing continuity between a plurality of messaging clients as recited in claim **1** wherein in the transferring step the transfer of the plurality of client data is in response to the second messaging client establishing the second communication connection.

10. A method for providing continuity between a plurality of messaging clients as recited in claim **1** wherein the second messaging client operates within a second messaging device, and further wherein in the transferring step the transfer of the plurality of client data is in response to activating the second messaging device.

11. A method for providing continuity between a plurality of messaging clients as recited in claim 1 further comprising:

disconnecting the first messaging client from the first communication connection prior to the transferring step.

12. A method for providing continuity between a plurality of messaging clients as recited in claim 1 further comprising:

disconnecting the first messaging client from the first communication connection after the transferring step.

13. A method for providing continuity between a plurality of messaging clients as recited in claim 1, wherein the plurality of client data includes at least one client data portion, and further wherein the transferring step comprises transferring the at least one client data portion.

14. A method for providing continuity between a plurality of messaging clients as recited in claim 13 further comprising prior to the transferring step, sending from the second messaging client to the first messaging client a client data requirement, wherein the client data portion is determined using the client data requirement.

15. Within a messaging communication system having a plurality of messaging clients and a message server, a method for providing continuity between the plurality of messaging clients comprising:

establishing for a first messaging client a first communication connection with the message server including a plurality of client data;

establishing for a second messaging client a second communication connection with the message server; and

transferring the plurality of client data from the first messaging client to the second messaging client in response to the second communication connection.

16. Within a messaging communication system having a plurality of messaging clients and a message server, a method for providing continuity between the plurality of messaging clients comprising:

establishing for a first messaging client a first communication connection with the message server including a plurality of client data, wherein the first messaging client includes a first account identifier;

providing the first account identifier for the first messaging client to the messaging communication system;

transferring the plurality of client data from the first messaging client to a second messaging client, wherein the second messaging client includes a second account identifier;

providing the second account identifier from the second messaging client to the messaging communication system; and

establishing for the second messaging client a second communication connection with the message server including the plurality of client data using the second account identifier.

17. A method for providing continuity between a plurality of messaging clients as recited in claim 1 further comprising:

authenticating an account user by the first messaging client using an authentication key prior to the transferring step,

transferring the authentication key from the first messaging client to the second messaging client; and

authenticating the account user by the second messaging client using the authentication key.

18. A method for providing continuity between a plurality of messaging clients as recited in claim 1 wherein the plurality of client data includes a plurality of contact data, and further wherein the plurality of contact data comprises at least one account identifier.

19. A method for providing continuity between a plurality of messaging clients as recited in claim 18 wherein the plurality of contact data further comprises a contact information for the at least one account identifier.

20. A method for providing continuity between a plurality of messaging clients as recited in claim 1 wherein the plurality of client data includes at least one user preference.

21. A method for providing continuity between a plurality of messaging clients as recited in claim 1 wherein the message server includes a sewer identity, wherein the plurality of client data includes the server identity, and further wherein the second communication connection is established using the server identity received within the plurality of client data.

22. Within a messaging communication system having a plurality of messaging clients, a method for providing continuity between the plurality of messaging clients comprising:

establishing a first communication connection for a first messaging client;

establishing at least one messaging session having a session identifier between the first messaging client and at least one other messaging client of the plurality of messaging clients;

transferring a plurality of session data for the first session connection including the session identifier from the first messaging client to a second messaging client;

establishing a second communication connection including the plurality of session data for the second messaging client; and

participating in the at least one messaging session in the second communication connection using the session identifier.

23. A method for providing continuity between a plurality of messaging clients as recited in claim 22 further comprising:

sending a notification of session data transfer to at least one other messaging client participating in the at least one messaging session.

24. A method for providing continuity between a plurality of messaging clients as recited in claim 23 wherein the notification includes a client profile of the second messaging client.

25. A method for providing continuity between a plurality of messaging clients as recited in claim 23 wherein the notification is sent from the first messaging client.

26. A method for providing continuity between a plurality of messaging clients as recited in claim 23 wherein the notification is sent from the second messaging client.

27. A method for providing continuity between a plurality of messaging clients as recited in claim 23 wherein the messaging communication system further includes a messaging server, and further wherein the notification is sent from the messaging server.

28. A method for providing continuity between a plurality of messaging clients as recited in claim 23 further comprising:

informing an account user of the session data transfer by the at least one other messaging client in response to receiving the notification.

29. A method for providing continuity between a plurality of messaging clients as recited in claim 22 wherein the

messaging session includes a session history having at least one session portion, and further wherein the plurality of session data further includes the session portion.

30. A method for providing continuity between a plurality of messaging clients as recited in claim **29** further comprising prior to the transferring step, sending from the second messaging client to the first messaging client a session data requirement, wherein the session portion is determined using the session data requirement.

31. A method for providing continuity between a plurality of messaging clients as recited in claim **22** wherein the plurality of session data further includes a session priority indicator, wherein the session priority indicator determines a priority of the messaging session within the messaging communication system.

32. A method for providing continuity between a plurality of messaging clients as recited in claim **22** wherein the plurality of session data further includes a session priority indicator, wherein the session priority indicator determines a priority of the messaging session within the second messaging client.

33. A method for providing continuity between a plurality of messaging clients as recited in claim **22** wherein the plurality of session data includes at least one user preference.

34. A method for providing continuity between a plurality of messaging clients as recited in claim **22** further comprising:

sending a notification of session data transfer, wherein the notification includes a client profile for the second messaging client; and

sending a plurality of content to the second messaging client using the client profile.

35. A method for providing continuity between a plurality of messaging clients as recited in claim **34** wherein the notification is sent from the first messaging client and the plurality of content is sent from at least one other messaging client.

36. Within a messaging communication system having a plurality of messaging clients, a method for providing continuity between the plurality of messaging clients comprising:

establishing a first communication connection for a first messaging client;

establishing a plurality of messaging sessions each having a session identifier between the first messaging client and at least one of the plurality of messaging clients;

transferring a plurality of client data for the first communication connection including at least one session identifier for at least one messaging session from the first messaging client to a second messaging client;

establishing a second communication connection including the plurality of client data for the second messaging client; and

participating in the at least one messaging session in the second communication connection using the session identifier.

37. A method for providing continuity between a plurality of messaging clients as recited in claim **36** further comprising:

sending a notification of data transfer to at least one of the plurality of messaging clients participating in the at least one messaging session.

38. A method for providing continuity between a plurality of messaging clients as recited in claim **37** wherein the notification is sent from the first messaging client.

39. A method for providing continuity between a plurality of messaging clients as recited in claim **37** wherein the notification is sent from the second messaging client.

40. A method for providing continuity between a plurality of messaging clients as recited in claim **37** wherein the notification includes a client profile of the second messaging client.

41. A method for providing continuity between a plurality of messaging clients as recited in claim **36** further comprising:

sending a notification of data transfer, wherein the notification includes a client profile for the second messaging client; and

sending a plurality of content to the second messaging client using the client profile.

42. A method for providing continuity between a plurality of messaging clients as recited in claim **36** wherein the messaging session includes a session history having at least one session portion, and further wherein the plurality of client data further includes the session portion.

43. A method for providing continuity between a plurality of messaging clients as recited in claim **36** further comprising prior to the transferring step, sending from the second messaging client to the first messaging client a client data requirement, wherein the session portion is determined using the client data requirement.

44. A method for providing continuity between a plurality of messaging clients as recited in claim **36** wherein the plurality of client data further includes a session priority indicator, wherein the session priority indicator determines the priority of the messaging session within the messaging communication system.

45. A method for providing continuity between a plurality of messaging clients as recited in claim **36** wherein the plurality of client data includes at least one user preference.

46. Within a messaging communication system having a plurality of messaging clients, a method for providing continuity between the plurality of messaging clients comprising:

establishing a first communication connection for a first messaging client;

establishing at least one messaging session having a session identifier between the first messaging client and at least one other messaging client of the plurality of messaging clients;

transferring a plurality of client data for the first communication connection including the session identifier from the first messaging client to a second messaging client;

establishing a second communication connection including the plurality of client data for the second messaging client; and

adding the second messaging client to the at least one messaging session using the session identifier.

47. Within a messaging communication system having a message server for managing a plurality of multiple user messaging sessions, wherein the multiple user messaging sessions comprise communication of a plurality of session messages among a plurality of messaging clients, a method for providing continuity between the plurality of messaging clients comprising:

establishing a first communication connection for a first messaging client within a multiple user messaging session of the message server;

transferring a plurality of client data for the first communication connection from the first messaging client to a second messaging client;

sending a data transfer message to the message server wherein the data transfer message includes a session reservation for the second messaging client; and

establishing a second communication connection for the second messaging client within the multiple user messaging session of the message server using the plurality of client data.

48. A method for providing continuity between a plurality of messaging clients as recited in claim **47** wherein the first messaging client has a first client identifier, wherein the multiple user messaging session has a session identifier, wherein the second messaging client has a second client identifier, wherein the plurality of client data includes the session identifier, and further wherein the data transfer message includes the session identifier, the first client identifier, and the second client identifier.

49. A method for providing continuity between a plurality of messaging clients as recited in claim **48**, wherein the multiple user messaging session includes at least one other messaging client, the method further comprising:

sending a notification of data transfer to the at least one other messaging client.

50. A plurality of messaging clients within a messaging communication system for providing continuity between the plurality of messaging clients comprising:

a first messaging client, for establishing a first communication connection including a plurality of client data with a message server; and

a second messaging client for receiving the plurality of client data from the first messaging client and for establishing a second communication connection including the plurality of client data with the message server.

51. A plurality of messaging clients as recited in claim **60** wherein the first messaging client operates within a first messaging device and the second messaging client operates within a second messaging device.

52. A plurality of messaging clients as recited in claim **51** wherein the first messaging device includes:

a memory coupled to the first messaging client for storing the plurality of client data, wherein the first messaging client accesses the plurality of client data from the memory, and further wherein the first messaging client transfers the plurality of client data to the second messaging device.

53. A plurality of messaging clients as recited in claim **51** wherein the first messaging device includes:

a memory coupled to the first messaging client for storing the plurality of client data, wherein the first messaging client accesses the plurality of client data from the memory, and

a data transfer application coupled to the first messaging client for transferring the plurality of client data to the second messaging device.

54. A plurality of messaging clients as recited in claim **51** wherein the second messaging device includes:

a memory coupled to the second messaging client, wherein the second messaging client receives the plurality of client data and stores the plurality of client data in the memory.

55. A plurality of messaging clients as recited in claim **51** wherein the second messaging device includes:

data transfer application coupled to the second messaging client for receiving the plurality of client data, wherein the second messaging client processes the received plurality of client data, and

a memory coupled to the second messaging client for storing the plurality of client data.

56. A plurality of messaging clients as recited in claim **51** wherein the first messaging device is a fixed device and further wherein the second device is a mobile device.

57. A plurality of messaging clients as recited in claim **51** wherein the first messaging device includes a first memory interconnect for connecting the first messaging device to a memory storage device, wherein the second messaging device includes a second memory interconnect for connecting the second messaging device to the memory storage device, wherein the first messaging device stores the plurality of client data on the memory storage device, and further wherein the second messaging device receives the plurality of client data from the memory storage device connecting to the second memory interconnect.

58. A plurality of messaging clients as recited in claim **57** wherein the first messaging client and the second messaging client operate within a messaging device.

59. A messaging communication system for providing continuity between a plurality of messaging clients comprising:

the plurality of messaging clients including:

a first messaging client,

a second messaging client, and

at least one other messaging client;

a message server for managing the communication of a plurality of session messages among the plurality of messaging clients, wherein the message server is programmed to:

establish a first communication connection for the first messaging client

establish at least one messaging session having a session identifier between the first messaging client and the at least one other messaging client,

transfer a plurality of client data for the first communication connection including the session identifier from the first messaging client to the second messaging client,

establish a second communication connection including the plurality of client data for the second messaging client, and

transfer the at least one messaging session from the first messaging client to the second messaging client using the session identifier.

60. A messaging communication system for providing continuity between a plurality of messaging clients as recited in claim **59** wherein the message server includes a server memory, wherein the first messaging client stores the plurality of client data in the server memory, and further wherein the second messaging client retrieves the plurality of client data from the server memory for use in the operation of the second communication connection.

61. A messaging communication system as recited in claim **59** wherein the first messaging client operates within a first messaging device and the second messaging client operates within a second messaging device.

62. A messaging communication system as recited in claim **59** wherein the first messaging client and the second messaging client operate within a messaging device.

63. A messaging communication system as recited in claim **59** wherein the messaging communication system comprises a first messaging system and a second messaging system, wherein the first messaging client functions within the first messaging system, and further wherein the second messaging client functions within the second messaging system.

64. A messaging communication system as recited in claim 63 wherein the first messaging system comprises a wired messaging system and further wherein the second messaging system comprises a wireless messaging system.

65. A messaging communication system as recited in claim 63 wherein the first messaging system comprises a wireless messaging system and further wherein the second messaging system comprises a wired messaging system.

66. A messaging communication system for providing continuity between a plurality of messaging clients comprising:

the plurality of messaging clients including:

a first messaging client for establishing a first communication connection including a plurality of client data, and

a second messaging client for establishing a second communication connection including the plurality of client data; and

a server memory coupled to the plurality of messaging clients, wherein the first messaging client stores the plurality of client data in the server memory, and further wherein the second messaging client retrieves the plurality of client data from the server memory for use in the operation of the second communication connection.

67. A messaging communication system as recited in claim 66 wherein the server memory is contained within a message server of the messaging communication system.

* * * * *

EXHIBIT 6

US007007064B2

(12) **United States Patent**
Faris

(10) **Patent No.: US 7,007,064 B2**
(45) **Date of Patent: Feb. 28, 2006**

(54) **METHOD AND APPARATUS FOR OBTAINING AND MANAGING WIRELESSLY COMMUNICATED CONTENT**

(75) Inventor: **Randi W. Faris**, Boca Raton, FL (US)

(73) Assignee: **Motorola, Inc.**, Schaumberg, IL (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 513 days.

(21) Appl. No.: **10/211,614**

(22) Filed: **Aug. 2, 2002**

(65) **Prior Publication Data**

US 2004/0021555 A1 Feb. 5, 2004

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

(52) **U.S. Cl.** **709/203**; 709/202; 709/216; 709/27; 709/231; 455/420

(58) **Field of Classification Search** 709/231, 709/201–203, 216–219, 225–229; 455/412.1–412.2, 455/418–420, 456.5–456.6
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

5,504,476	A		4/1996	Marrs et al.
5,561,802	A	*	10/1996	Orimo et al. 709/202
5,705,995	A		1/1998	Laflin et al.
5,991,800	A		11/1999	Burke et al.
6,064,975	A	*	5/2000	Moon et al. 705/8
6,202,023	B1	*	3/2001	Hancock et al. 701/201
6,260,059	B1	*	7/2001	Ueno et al. 709/202
6,295,502	B1	*	9/2001	Hancock et al. 701/201
6,363,419	B1	*	3/2002	Martin, Jr. et al. 709/219
6,397,259	B1	*	5/2002	Lincke et al. 709/236
6,470,381	B1	*	10/2002	De Boor et al. 709/217
6,526,284	B1	*	2/2003	Sharp et al. 455/456.6
6,731,612	B1	*	5/2004	Koss 370/310
6,775,291	B1	*	8/2004	Ryu et al. 370/401
6,795,827	B1	*	9/2004	Ohira 709/201
6,823,373	B1	*	11/2004	Pancha et al. 709/219
6,826,614	B1	*	11/2004	Hanmann et al. 709/227

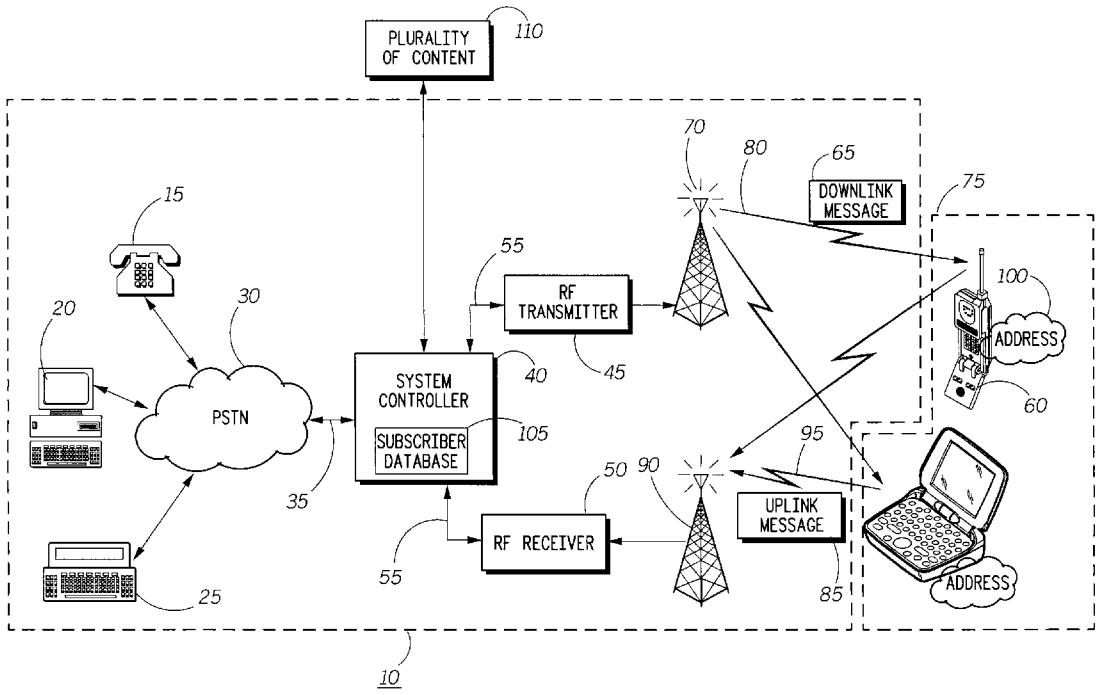
* cited by examiner

Primary Examiner—Bharat Barot
(74) *Attorney, Agent, or Firm*—Randi L. Karpinin; Daniel C. Crilly

(57) **ABSTRACT**

A wireless communication device **(60)** for receiving communicated content includes a receiver **(125)**, a processor **(140)**, a content memory **(180)**, and a content management application **(165)**. The receiver **(125)** receives a content message **(260)** from a wireless communication system **(10)**. The content message **(260)** includes a document type **(265)**, wherein the document type indicates that the content message **(260)** includes data relating to one or more communicated content portions **(280)**. The content management application **(165)** processes the data relating to the one or more communicated content portions **(280)** and stores the data relating to the one or more communicated content portions **(280)** in the content memory **(180)**.

26 Claims, 6 Drawing Sheets



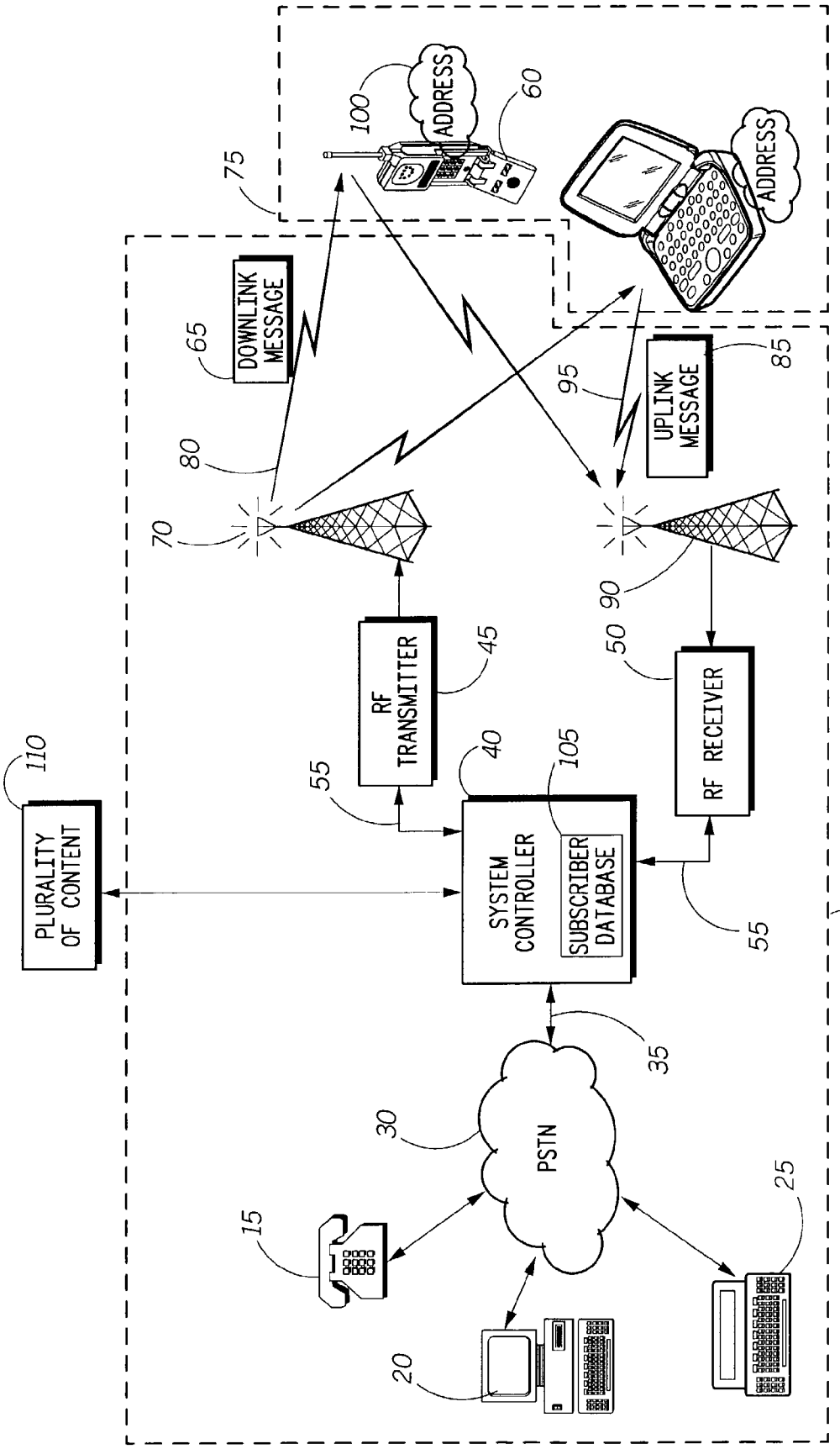


FIG. 1

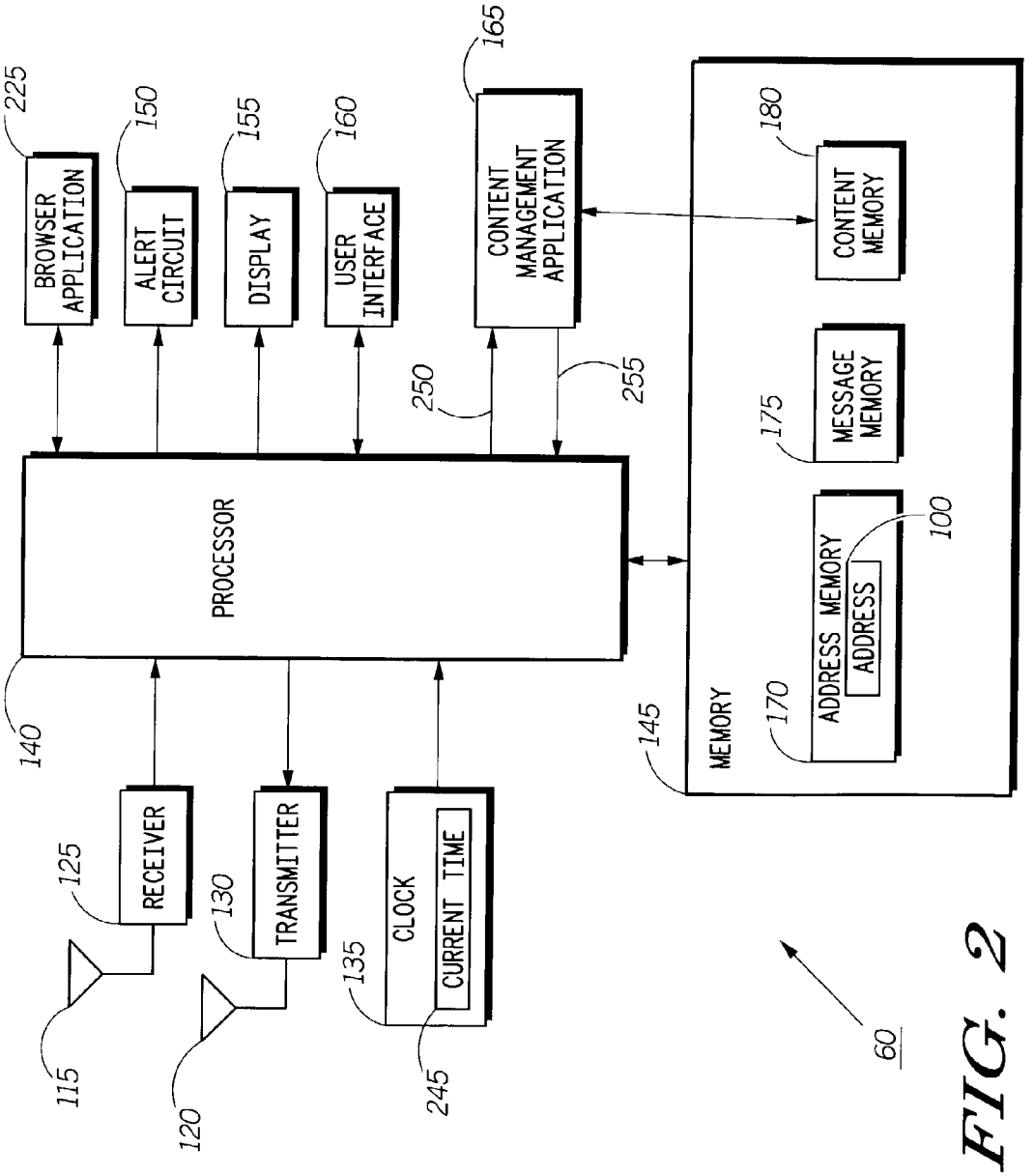


FIG. 2

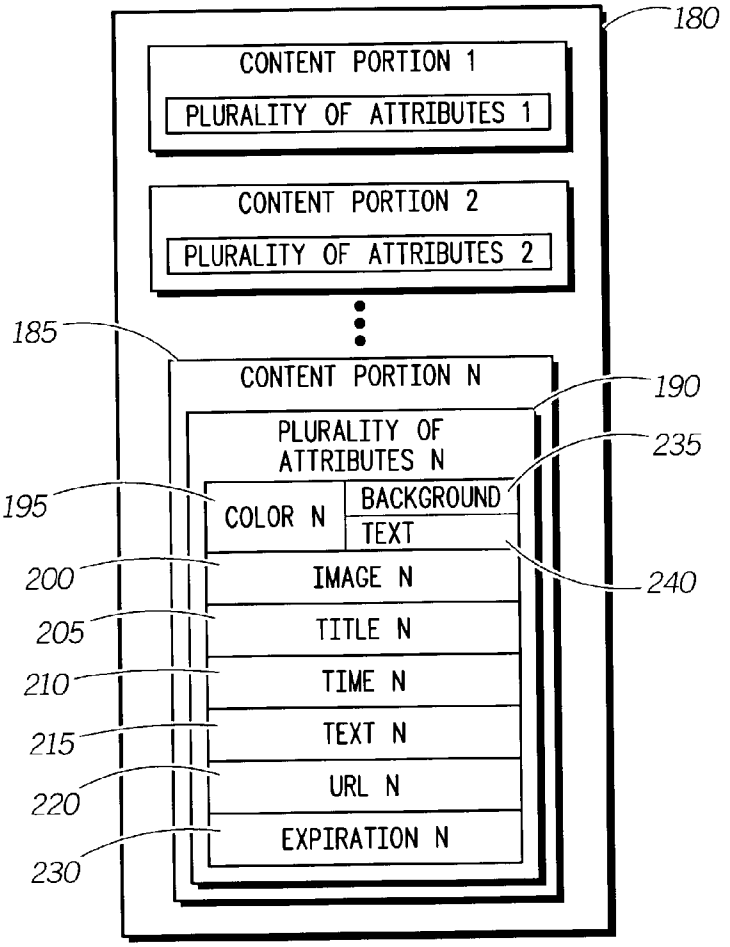


FIG. 3

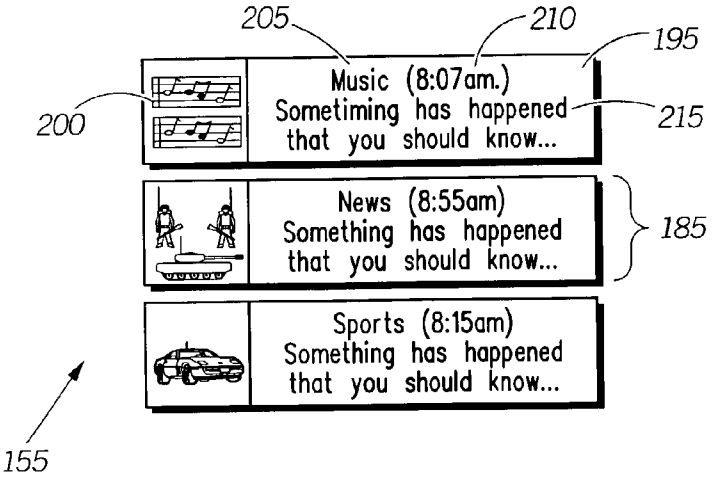


FIG. 4

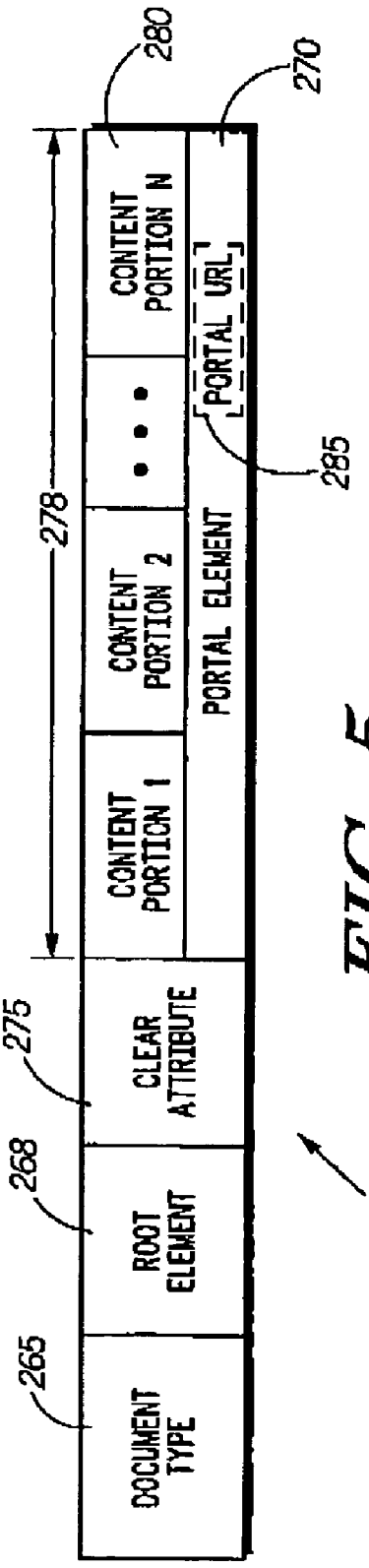


FIG. 5

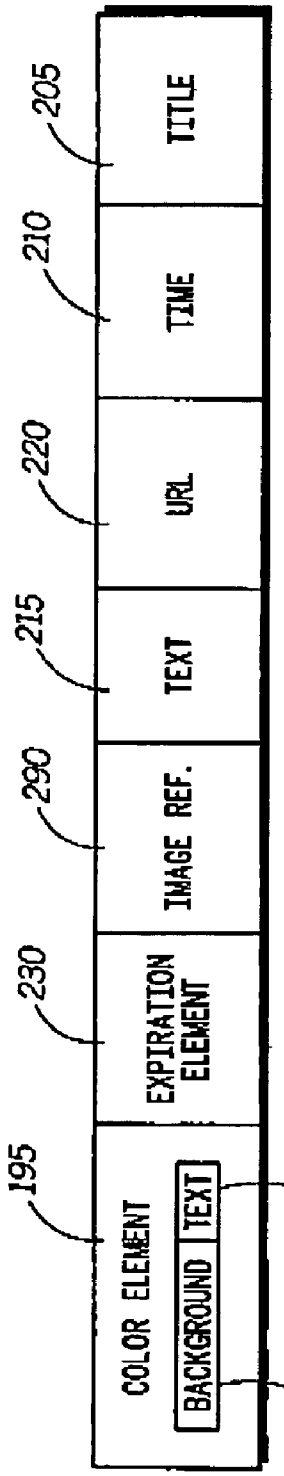


FIG. 6

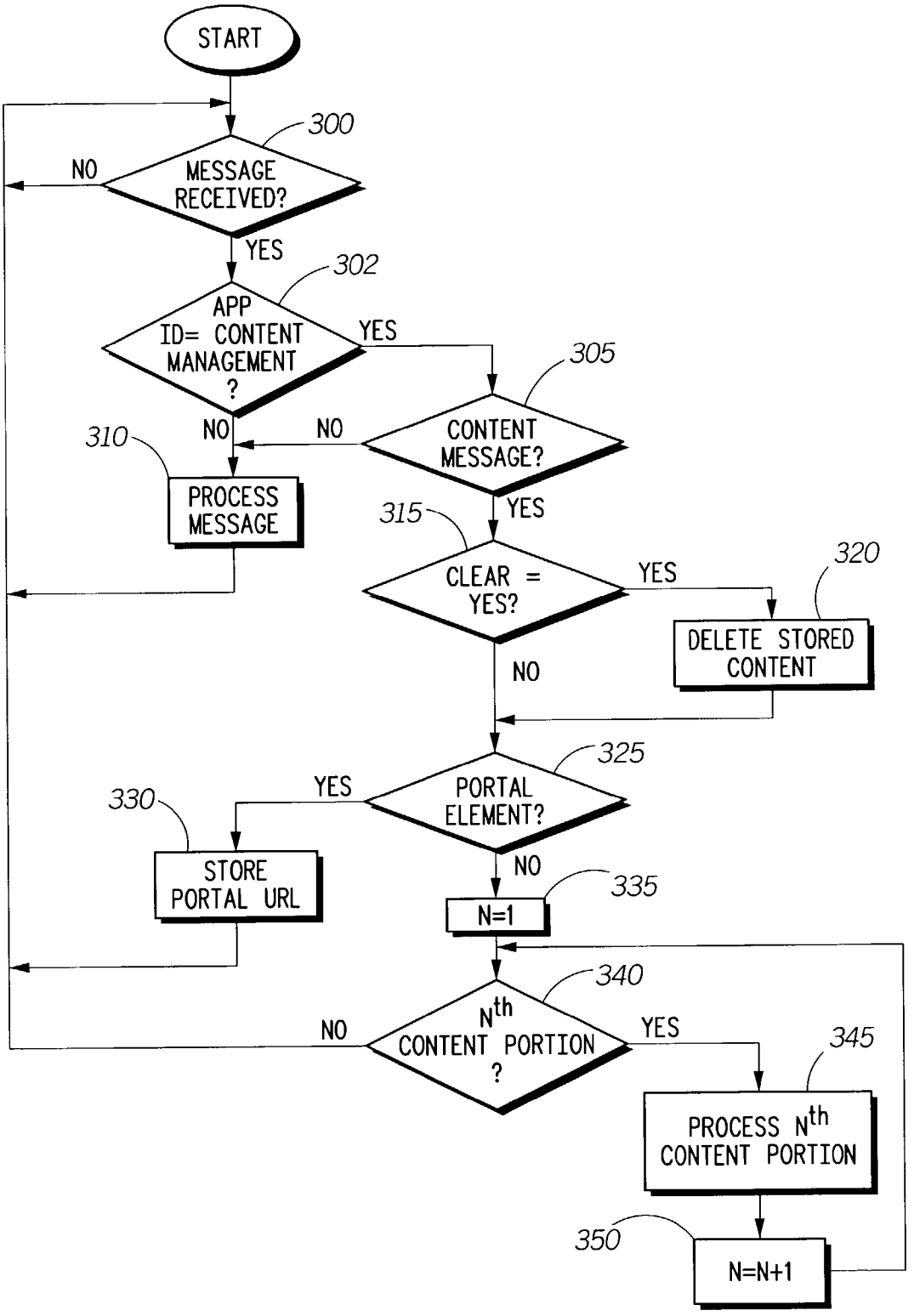


FIG. 7

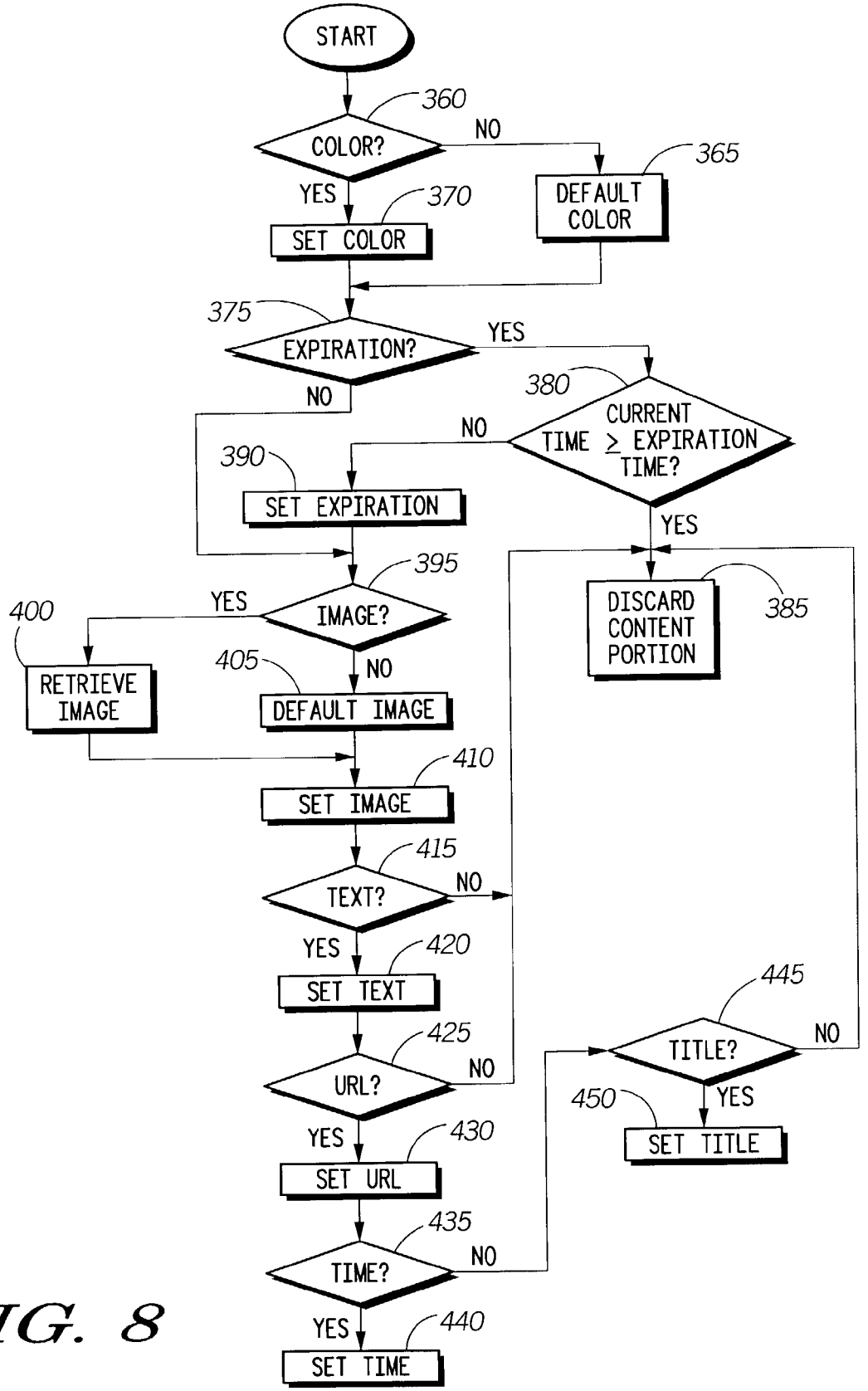


FIG. 8

METHOD AND APPARATUS FOR OBTAINING AND MANAGING WIRELESSLY COMMUNICATED CONTENT

BACKGROUND OF THE INVENTION

1. Field of the Invention

This invention relates in general to communication systems and in particular to wireless communication systems for providing content to wireless communication devices.

2. Description of the Related Art

Historically, wireless communication devices, such as cellular telephones and two way messaging devices, have had the ability to receive data and/or voice messages sent from a wireless communication system, and perform standard functions in response to message receipt such as storing the message, displaying the message, or alerting the user of receipt of the message. Data messages are typically a numeric message such as a phone number, or an alphanumeric message containing one unique piece of information such as "meeting in my office at 6:00 pm."

The user is alerted of receipt of a new message by an audible alert or a vibratory alert. The display of many wireless communication devices today includes a message indicator for each message it has received and stored in memory. This message indicator allows a quick view for the user of how many messages are in the wireless communication device and also allows the user to quickly pick a message to view. The display further can include a call receipt message indicating to the user a new call is being received and/or the source of the new call.

Today, wireless service providers are teaming up with content providers to provide content on wireless communication devices. The content providers benefit from such partnerships by being able to send content to a large number of potential consumers in a cost effective manner. The content, for example, can include weather, traffic, events, restaurants, shopping, services, sports, and the like. Adjunct content servers connected through the wireless communication system provide content information and other content services to device users that have their content configured according to personalized preferences, system generated information, and/or information generated directly from the wireless communication device.

Methods and systems have been developed in an attempt to satisfy the increasing demand for information, multimedia and other communication services on wireless communication devices. One of the most popular techniques of obtaining these services is through the Internet. The Internet is collection of over 25,000 computer networks connected through a communication backbone (NSFNET backbone) funded by the National Science Foundation (NSF) and is currently managed by Advanced Network System (ANS). A subscriber obtains an account with an organization's host computer (server) that is connected to the Internet through one or more networks. Traditionally, the subscriber is connected to the server through telephone lines using a personal computer (PC) and a modem. As use of the Internet becomes more popular, different methods of accessing the Internet have been developed. For example, users can access the Internet using a wireless communication device.

One drawback of sending content such as Internet content to wireless communication devices via various wireless communication systems today is the amount of channel space required to send the content information. Service providers want to offer the content services without impacting channel space for traditional message communication.

Another drawback is the amount of memory required to store such information on the wireless communication device. Users desire portable access to such content without the loss of memory storage for the traditional message communications, which is typically the dominant purpose of the wireless communication device.

BRIEF DESCRIPTION OF THE DRAWINGS

The present invention will be described by way of exemplary embodiments, but not limitations, illustrated in the accompanying drawings in which like references denote similar elements, and in which:

FIG. **1** is an electronic block diagram of a wireless communication system, in accordance with a preferred embodiment of the present invention.

FIG. **2** is an electronic block diagram of a wireless communication device for use within the wireless communication system of FIG. **1**, in accordance with a preferred embodiment of the present invention.

FIG. **3** is a block diagram of a content memory for use within the wireless communication device of FIG. **2**, in accordance with a preferred embodiment of the present invention.

FIG. **4** illustrates one embodiment of a display for use within the wireless communication device of FIG. **2** in accordance with the present invention.

FIG. **5** illustrates a content message for communicating within the wireless communication system of FIG. **1** in accordance with a preferred embodiment of the present invention.

FIG. **6** illustrates a content portion communicated within the content message of FIG. **5** in accordance with a preferred embodiment of the present invention.

FIGS. **7** and **8** are flowcharts illustrating the operation of the wireless communication device of FIG. **2** in accordance with a preferred embodiment of the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT(S)

As required, detailed embodiments of the present invention are disclosed herein; however, it is to be understood that the disclosed embodiments are merely exemplary of the invention, which can be embodied in various forms. Therefore, specific structural and functional details disclosed herein are not to be interpreted as limiting, but rather should be interpreted merely as a basis for the claims and as a representative basis for teaching one skilled in the art to variously employ the present invention in virtually any appropriately detailed structure. Further, the terms and phrases used herein are not intended to be limiting; but rather are intended to provide an understandable description of the invention.

The terms "a" or "an," as used herein, are defined as one or more than one. The term "plurality," as used herein, is defined as two or more. The term "another," as used herein, is defined as at least a second or more. The terms "including" and/or "having," as used herein, are defined as comprising (i.e., open language). The term "coupled," as used herein, is defined as connected, although not necessarily directly, and not necessarily mechanically. The term "program," as used herein, is defined as a sequence of instructions designed for execution on a computer system. A program, or computer program, may include a subroutine, a function, a procedure, an object method, an object implementation, an executable application, an applet, a servlet, a source code, an object

code, a shared library/dynamic load library and/or other sequence of instructions designed for execution on a computer system.

Referring to FIG. **1**, an electronic block diagram of a wireless communication system **10** is shown. The wireless communication system **10** includes a message input device for initiating messages into the wireless communication system **10**. The message input device can be, for example, a telephone **15**, a computer **20**, or a desktop messaging unit **25**, connected through a conventional public switched telephone network (PSTN) **30** through a plurality of telephone links **35** to a system controller **40**. The telephone links **35**, for example, can be a plurality of twisted wire pairs, a fiber optic cable, or a multiplexed trunk line.

The system controller **40** is coupled to and oversees the operation of at least one radio frequency (RF) transmitter **45** and at least one radio frequency (RF) receiver **50** through one or more communication links **55**. The communication links **55** typically are twisted pair telephone wires, and additionally can include radio frequency (RF), microwave, or other communication links. The radio frequency transmitter **45** and the radio frequency receiver **50** typically are used with message store and forward stations that encode and decode inbound and outbound messages into formats that are compatible with landline message switched computers and personal radio addressing requirements, such as cellular messages, short messaging service, or paging protocols. The system controller **40** can also function to encode and decode wireless messages that are transmitted to or received by the radio frequency transmitter **45** or the radio frequency receiver **50**. Telephony signals are typically transmitted to and received from the system controller **40** by telephone sets such as the telephone **15** or a wireless communication device **60**. The system controller **40** encodes and schedules outbound messages such as a downlink message **65**. The system controller **40** then transmits the encoded outbound messages through the radio frequency transmitter **45** via a transmit antenna **70** to one or more of a plurality of wireless communication devices **75** such as the wireless communication device **60** on at least one outbound radio frequency (RF) channel **80**. The downlink message **65** can be, for example, a data message or a voice call. Similarly, the system controller **40** receives and decodes inbound messages such as an uplink message **85** received by the radio frequency receiver **50** via a receive antenna **90** on at least one inbound radio frequency (RF) channel **95** from one of the plurality of wireless communication devices **75**. The uplink message **85** can be, for example, a data message, a reply to a data message, a voice call, or a reply to a voice call.

It will be appreciated by one of ordinary skill in the art that the wireless communication system **10**, in accordance with the present invention, can function utilizing any wireless RF channel, for example, a one or two-way pager channel, a mobile cellular telephone channel, or a mobile radio channel. Similarly, it will be appreciated by one of ordinary skill in the art that the wireless communication system **10** can function utilizing other types of communication channels such as infrared channels and/or Bluetooth channels. In the following description, the term "wireless communication system" refers to any of the wireless communication systems mentioned above or an equivalent.

Similarly, it will be appreciated by one of ordinary skill in the art that each of the plurality of wireless communication devices **75**, such as the wireless communication device **60**, in accordance with the present invention, can be a mobile cellular telephone, a mobile radio data terminal, a mobile cellular telephone having an attached or integrated data terminal, or a two way pager, such as the "Pagewriter 2000X" manufactured by Motorola Inc. of Schaumburg, Ill. In the following description, the term "wireless communication device" refers to any of the devices mentioned above or an equivalent.

Each of the plurality of wireless communication devices **75** assigned for use in the wireless communication system **10** has an address or identity assigned thereto which is a unique selective call address in the wireless communication system **10**. For example, the wireless communication device **60** assigned for use in the wireless communication system **10** has an address **100** assigned thereto which is a unique selective call address in the wireless communication system **10** for the wireless communication device **60**. The address **100** enables the transmission of the downlink message **65** from the system controller **40** only to the wireless communication device **60** having the address **100**, and identifies the messages and responses received at the system controller **40** from the wireless communication device **60** with the address **100**. In one embodiment, each of the plurality of wireless communication devices **75** also has a pin number assigned thereto, the pin number being associated with a telephone number within the PSTN **30**. A list of the assigned addresses and correlated telephone numbers for each of the plurality of wireless communication devices **75** is stored in the system controller **40** in the form of a subscriber database **105**.

Preferably, and in accordance with the present invention, the wireless communication system **10** is coupled to and receives a plurality of content **110**. As illustrated, the plurality of content **110** is preferably received and processed by the system controller **40**. It will be appreciated by those of ordinary skill in the art, that alternatively, the PSTN **30** can receive the plurality of content **110** and send it to the system controller **40** for processing (not shown). The plurality of content **110** can include, for example, content associated with physical events such as sporting events, content associated with television or radio broadcasts, wireless content, Internet content, or an equivalent. Each of the plurality of content **110** can be separately designated within the wireless communications system **10**, or alternatively can be grouped within one or more topics. The plurality of content **110**, for example, can include separately designated content such as weather, traffic, events, restaurants, shopping, services, sports, and the like. Further, for example, a general topic can be the National Football League (NFL) and all events relating to the NFL such as a live NFL game via the Internet or a pay per view broadcasted NFL game can be grouped within the general topic.

FIG. **2** is an electronic block diagram of a preferred embodiment of the wireless communication device **60** for use within the wireless communication system **10** of FIG. **1** in accordance with the present invention. It will be appreciated by one of ordinary skill in the art that the electronic block diagram of FIG. **2** is illustrative of each of the plurality of wireless communication devices **75** assigned for use in the wireless communication system **10**.

Referring to FIG. **2**, the wireless communication device **60** includes a first antenna **115**, a second antenna **120**, a receiver **125**, a transmitter **130**, a clock **135**, a processor **140**, a memory **145**, an alert circuit **150**, a display **155**, a user interface **160**, a browser application **225**, and a content management application **165**. It will be appreciated by one of ordinary skill in the art that the content management application **165** and the browser application **225** can be a software program or any other equivalent.

The first antenna **115** intercepts transmitted signals from the wireless communication system **10**. The first antenna **115** is coupled to the receiver **125**, which employs conventional demodulation techniques for receiving the communication signals transmitted by the wireless communication system **10** such as the downlink message **65** of FIG. **1**.

Coupled to the receiver **125**, is the processor **140** utilizing conventional signal-processing techniques for processing received messages. Preferably, the processor **140** is similar to the MC68328 micro-controller manufactured by Motorola, Inc. of Schaumburg, Ill. It will be appreciated by one of ordinary skill in the art that other similar processors can be utilized for the processor **140**, and that additional processors of the same or alternative type can be utilized as required to handle the processing requirements of the processor **140**.

The processor **140** decodes an address in the demodulated data of the received message, compares the decoded address with one or more addresses such as the address **100** stored in an address memory **170** of the memory **145**; and when a match is detected, proceeds to process the remaining portion of the received message.

To perform the necessary functions of the wireless communication device **60**, the processor **140** is coupled to the memory **145**, which preferably includes a random access memory (RAM), a read-only memory (ROM), and an electrically erasable programmable read-only memory (EEPROM)(not shown). The memory **145** is comprised of the address memory **170**, a message memory **175**, and a content memory **180**.

FIG. **3** illustrates further detail of the content memory **180** in accordance with a preferred embodiment of the present invention. As illustrated, the content memory **180** preferably stores one or more content portions **185** for one or more of the plurality of content **110** received from the wireless communication system **10**. Each content portion **185** includes one or more attributes **190**. The one or more attributes **190** can, for example, include a color element **195**, an image element **200**, a title attribute **205**, a time attribute **210**, and a text element **215**. The time attribute **210** specifies the time for the related content with which the content portion **185** is associated. The title attribute **205** specifies the title text, which shall be displayed on the display **155** for the content portion **185**. The title attribute **205** gives the user an idea of what category the subject is in, such as music, news, weather, and the like. When the user interface **160** permits, the title attribute **205** can have a different appearance on the display **155** than the text element **215** of the content portion **185**.

The image element **200** specifies the image file to be displayed along with the content portion **185** on the display **155**. The actual image file must be fetched once received from the wireless communication system **10** such as within a content message. The image retrieved for example can be in GIF (Graphic Interchange Format) or PNG (Portable Network Graphics) format or an equivalent. When no image element **200** is specified for the content portion **185**, a default image can be used. The text element **215** contains the text description of the content. It acts as a teaser to the more detailed article specified by a Uniform Resource Locator (URL) attribute **220** if present.

The color element **195** allows the content provider to specify the background color and text color of the content portion text display area on the display **155**. If the color element **195** doesn't exist, the background and text color of the text area preferably can be a default color. Preferably, if either of the attributes (text or background) is missing, the color element **195** is ignored by the content management application **165**. In one embodiment, color values are specified in hexadecimal format as #RRGGBB, where RR=red value in hex, GG=the green value in hex, and BB=the blue value in hex. Preferably, the color element **195** includes a background color attribute **235** and a text color attribute **240**. The background color attribute **235** preferably specifies the sRGB (a standard default color space for the Internet) color of the text display area background in the format #RRGGBB as previously described. The text attribute **240** specifies the sRGB color for the snack text in the format #RRGGBB as previously described.

In one embodiment, the one or more attributes **190** of the content portion **185** further includes the URL attribute **220**. The URL attribute **220** allows the user to view more detailed information related to the content portion **185** using the browser application **225** or equivalent on the wireless communication device **60**.

In one embodiment of the present invention, the one or more attributes **190** of the content portion **185** includes an expiration element **230**. The expiration element **230** specifies the expiration time and date for the content portion **185**. If no expiration element **230** is stored, the content portion **185** will not expire and will remain in the content memory **180** of the wireless communication device **60** until some other event causes it to be deleted (for example: the content memory becomes full). The clock **135** on the wireless communication device **60** can be the time base for the deletion. In an alternate embodiment, the time attribute **10**, if present, can be used as an indication of when the content portion **185** is to be deleted.

Referring back to FIG. **2**, once the processor **140** has processed a received message, it stores the decoded message in the message memory **175**. It will be appreciated by one of ordinary skill in the art that the message memory **175**, in accordance with the present invention, can be a group of memory locations in a data storage device or an equivalent.

Upon receipt and processing of a message, the processor **140** preferably generates a command signal to the alert circuit **150** as a notification that the message has been received and stored. The alert circuit **150** can include a speaker (not shown) with associated speaker drive circuitry capable of playing melodies and other audible alerts, a vibrator (not shown) with associated vibrator drive circuitry capable of producing a physical vibration, or one or more LEDs (not shown) with associated LED drive circuitry capable of producing a visual alert. It will be appreciated by one of ordinary skill in the art that other similar alerting means as well as any combination of the audible, vibratory, and visual alert outputs described can be used for the alert circuit **150**.

Upon receipt of a message, the processor **140** preferably also generates a command signal to the display **155** to generate a visual notification of the receipt and storage of the message. When the display **155** receives the command signal from the processor **140** that the message has been received and stored in the message memory **175**, a message indication is displayed. The message indication, for example can be the activation of one of a plurality of message icons on the display **155**. The display **155** can be, for example, a liquid crystal display, a dot matrix display, or an equivalent.

The wireless communication device **60** preferably further includes the clock **135**. The clock **135** provides timing for the processor **140**. The clock **135** preferably includes a current time **245** for use in the operation of the wireless communication device **60** such as for use by the content management application **165**. The clock **135** also provides a

source for timing of feature enhancements such as active and inactive periods of operation or periods of alerting.

In a preferred embodiment, the wireless communication device **60** includes the content management application **165**. The content management application **165** is programmed to process the one or more content portions **185** received by the receiver **125** and/or stored in the content memory **180**, and to identify personalized content to be displayed to the user on the display **155**. The wireless communication device **60** performs content management functions within the content management application **165** using a processor command **250** sent from the processor **140**. The content management application **165** sends an application response **255** in reply to the processor command **250**. The content management application **165** can be hard coded or programmed into the wireless communication device **60** during manufacturing, can be programmed over-the-air upon customer subscription, or can be a downloadable application. It will be appreciated by one of ordinary skill in the art that other programming methods can be utilized for programming the content management application **165** into the wireless communication device **60**.

The content management application **165**, in response to the processor command **250**, preferably accesses the content memory **180** of the memory **145**. The content management application **165** is programmed with a set of rules identifying the management of the one or more content portions **185** received by the receiver **125** and/or stored in the content memory **180**. For example, the content management application **165** can identify the one or more content portions **185** to be displayed on the display **155**. The content management application **165** sends the application response **255** to the processor **140** including the one or more content portions **185** and an associated plurality of attributes **190**. The processor **140**, in response to receipt of the application response **255** sends a command signal to the display **155** identifying the content portion **185** and the plurality of attributes **190** for displaying on the display **155**. In a preferred embodiment, the content information is received in a manner that is non-obtrusive to the device user. The one or more content portions **185** is received by the receiver **125** and forwarded by the processor **140** to the content memory **180** for storage and for later processing by the content management application **165**. The user preferably is not alerted upon the receipt of each message containing the one or more content portions **185**.

The content management application **165** is programmed to handle incoming content portions and expired content portions. The content management application **165** can be programmed, for example, to check expiration times in conjunction with the receipt of an incoming content portion, expiration of the scroll timer, or other such events. Similarly, the content management application **165** can be programmed to update the display **155** with a newly received and stored content portion when a scroll timer expires. Consequently, the content management application **165** can consider other pending events prior to updating the display **155** in order to present a more pleasing experience to the user.

FIG. **4** illustrates an example of the display **155** including the one or more content portions **185**. As illustrated, the title attribute **205** and the time attribute **210** are preferably displayed on the first line of each content portion area. If the total length of the title attribute **205** and the time attribute **210** is longer than the number of characters, which can be displayed on that line, it can be truncated and an indication that it has been truncated displayed at the end of the line. The text element **215** is displayed on the second and third lines of the content portion area. If the length of the text element **215** is longer than can be displayed on those two lines, it can be truncated and an indication that it has been truncated displayed at the end of the line. Each image element **200**, for example, can be a 16-bit (or less) color image that is 55 pixels wide by 38 pixels high. If the image element **200** is larger than the available display space, it can be truncated to fit into the space. The image element **200** can either be centered, or upper left justified, or an equivalent.

Preferably, the user interface **160** is coupled to the processor **140**, as shown in FIG. **2**. The user interface **160** can be one or more buttons used to generate a button press, a series of button presses, a voice response from the device user, or some other similar method of manual response initiated by the device user of the wireless communication device **60**. The processor **140**, in response to the user interface **160**, initiates the processor command **250** to the content management application **165**. The content management application **165**, in response to the processor command **250** can alter the displayed content portions and/or perform other actions on the one or more content portions. For example, pressing a button of the user interface **160** can cause all content portions to be updated and then the updated content portions displayed on the display **155**. Similarly, pressing a button on the user interface **160** can cause all expired content portions to be deleted and all newly received content portions to be stored in the content memory **180**. In one embodiment, the user interface **160** can be used to highlight one or more attributes of a content portion. Preferably, the user interface **160** is used to scroll between various displayed content portions, to select various displayed content portions, and/or to select the URL attribute **220** for a displayed content portion to access the content associated with the content portion.

The transmitter **130** is coupled to the processor **140** and is responsive to commands from the processor **140**. When the transmitter **130** receives a command from the processor **140**, the transmitter **130** sends a signal via the second antenna **120** to the wireless communication system **10**. The signal, for example, can be the uplink message **85** as illustrated in FIG. **1**.

In an alternative embodiment (not shown), the wireless communication device **60** includes one antenna performing the functionality of the first antenna **115** and the second antenna **120**. Further, the wireless communication device **60** alternatively includes a transceiver circuit performing the functionality of the receiver **125** and the transmitter **130**. It will be appreciated by one of ordinary skill in the art that other similar electronic block diagrams of the same or alternate type can be utilized for the wireless communication device **60** to handle the requirements of the wireless communication device **60**.

Content information is preferably sent from the wireless communication system **10** to one or more of the plurality of wireless communication devices **75** using the extensible markup language (XML). XML is a universal format for structured documents and data on the World Wide Web. XML is a set of rules (i.e.: guidelines or conventions) for designing text formats for structuring data. XML makes use of tags (words bracketed by '<' and '>') and attributes (of the form name="value"). XML uses the tags only to delimit pieces of data, and leaves the interpretation of the data completely to the application that reads it.

Preferably, the content delivery mechanism uses the wireless application protocol (WAP) push method. This method allows content to be sent or "pushed" to devices by server-

based applications via a push proxy. Push functionality is especially relevant to real-time applications that send notifications to their users, such as messaging, stock price and traffic update alerts. Without push functionality, these types of applications would require the devices to poll application servers for new information or status. In wireless environments such polling activities would constitute inefficient and wasteful use of the resources of wireless communication systems. WAP's push functionality provides control over the lifetime of pushed messages, store & forward capabilities at the push proxy and control over the bearer's choice for delivery. Each pushed message contains a URL to the content associated with each of the one or more content portions **185**. In a preferred embodiment, the wireless communication device **60** can fetch the content associated with each of the one or more content portions **185** when it receives a push message. One or more fetch actions may be required when the content portion **185** includes a URL to the content and a URL to the image associated with the content. Once the content portion has been fetched, it will be sent up to the content memory **180** for storage. Content portions that do not contain the required elements and attributes are preferably discarded.

FIG. **5** illustrates a content message **260** for communicating content information. The content message **260** can be, for example, the downlink message **65** of FIG. **1**. As illustrated in FIG. **5**, the content message **260** preferably includes a document type **265**, a root element **268**, a clear attribute **275**, and an information message **278**. In accordance with the present invention, the information message **278** can be either a portal element **270** or one or more communicated content portions **280**.

When the downlink message **65** sent from the wireless communication system **10** to one or more of the plurality of wireless communication devices **75** is the content message **260**, the document type **265** preferably is set to indicate that the data to follow relates to one or more communicated content portions **280**. For example, the document type **265** can be indicated by "text/x-motorola.sxl." Upon receipt of a message with the document type **265**=text/x-motorola.sxl, the processor **140** sends the processor command **250** to the content management application **165**.

The root element **268** is the top-level element in the XML document hierarchy. The root element contains all other elements. Each document can have only one root. All other elements are nested within it.

The portal element **270** indicates where the user can go to customize the one or more content portions **185** stored in the content memory **180**. For example, the portal element **270** indicates where the user can go to customize the one or more communicated content portions **280** sent within the content message **260**. The portal element **270** includes a portal URL **285** at which the user can access and update one or more preferences. The portal URL **285** is preferably stored in the memory **145** of the wireless communication device **60**. When the wireless communication device **60** receives a new portal element **270**, the new portal element **270** replaces the currently stored portal URL. It will be appreciated by those of ordinary skill in the art that the device user can access the portal URL **285** through the browser application **225** by initiating an input to the user interface **160**. For example, the portal URL **285** can be assigned to a button, which, when pressed, displays that web page using the browser application **225**. It will further be appreciated by those of ordinary skill in the art that the one or more preferences for content message receipt can be updated by a user at the portal URL **285** using the browser application **225** in response to an input from the user interface **160** of the wireless communication device **60**.

The clear attribute **275** is used to flush the existing queue of stored content portions **185**. When the clear attribute **275** is set to "yes" in the content message **260**, the wireless communication device **60** deletes all existing (received) content portions from the content memory **180** prior to storing any new one or more communicated content portions **280**. It will be appreciated by those of ordinary skill in the art that the new one or more communicated content portions **280** can be included in the content message **260** containing the clear attribute **275** set to "yes"; or alternatively can be included in a second content message communicated after the stored content portions have been deleted. When the clear attribute **275** is set to "no" or is not present in the content message **260**, the one or more communicated content portions **280** of the content message **260** will be considered an incremental update.

FIG. **6** illustrates the one or more communicated content portion **280** in further detail. As illustrated in FIG. **6**, the communicated content portion **280** preferably includes one or more attributes including the color element **195** (including the background color **235** and the text color **240**), the expiration element **230**, an image reference attribute **290**, the text element **215**, the URL attribute **220**, the time attribute **210**, and the title attribute **205**. The image reference attribute specifies the URL for the image data file. This data file can be retrieved once the content message **260** is received. The image URL is discarded once the image element **200** has been retrieved and stored in the content memory **180**. The attributes included in the communicated content portion **280** are substantially equivalent to those described previously herein for FIG. **3**.

One example of the data format of the content message **260** of FIGS. **5** and **6** is illustrated below:

```
<?xml version="1.0"? encoding="UTF-8" standalone="no"?>
<!DOCTYPE sxl SYSTEM "http://URI/xml/dtds/sxl.dtd">
<!-- a color using sRGB: #RRGGBB as Hex values -->
<!ENTITY % Color "CDATA">
<!ELEMENT SXL (PORTAL | SNACK+ )>
<!ATTLIST SXL
clear CDATA #IMPLIED
>
<!ELEMENT PORTAL EMPTY>
<!ATTLIST PORTAL
    href CDATA #REQUIRED
>
<!ELEMENT SNACK (COLOR?, EXP?, IMG, T)>
<!ATTLIST SNACK
    href CDATA #REQUIRED
    time CDATA #IMPLIED
    title CDATA #REQUIRED
    >
<!ELEMENT COLOR EMPTY>
<!ATTLIST COLOR
    bg %Color; #REQUIRED
    text %Color #REQUIRED
>
<!ELEMENT EXP EMPTY>
<!ATTLIST EXP
    time CDATA #REQUIRED
    >
<!ELEMENT IMG EMPTY>
<!ATTLIST IMG
    href CDATA #IMPLIED
>
<!ELEMENT T (#PCDATA)>
```

FIG. **7** is a flowchart illustrating the operation of the wireless communication device **60** of FIG. **2** in accordance with a preferred embodiment of the present invention. Specifically, FIG. **7** illustrates the processing of a received content message **260** by the wireless communication device **60**. The process begins with Step **300** wherein the wireless communication device **60** checks for a received message. The wireless communication device **60**, for example, can receive the downlink message **65** from the wireless communication system **10** via the first antenna **115** coupled to the receiver **125**. When no message has been received in Step **300**, the process cycles back and periodically checks for a received message. Next, in Step **302**, when a message has been received, the wireless communication device **60** queries whether the application identification (APP ID) in the WAP push is the APP ID designated for the content management application **165**. Next, in Step **305**, when the application identification is the APP ID designated for the content management application **165**, the received message is pushed to the content management application **165** for further processing. Specifically, the content management application **165** of the wireless communication device **60** determines whether the received message is a content message **260**. For example, the content management application **165** determines whether the received message includes the document type **265** set to indicate that the data to follow relates to one or more content communicated content portions **280**. In Step **310**, when the received message is not a content message **260** in Step **305** or when the application identification is not the APP ID designated for the content management application **165** in Step **302**, the message is processed as a standard received message. For example, the message is stored in the message memory **175** of the memory **145** and the processor sends a command signal to the alert circuit **150** and a command signal to the display **155** to notify the user of the received message. The process then cycles back to Step **300** and periodically checks for receipt of a message.

Next in Step **315**, when the received message is a content message **260**, the content management application **165** of the wireless communication device **60** determines whether the content message **260** includes a clear attribute **275** set to "yes". It will be appreciated by those of ordinary skill in the art that the clear attribute **275** being set as a "yes" is an example for illustration purposes only, and that the clear attribute **275** can be set to any affirmative indication that the one or more content portions **185** should be deleted from the content memory **180**. In Step **320**, when the content message **260** includes the clear attribute **275** set to "yes", the wireless communication device **60** deletes the one or more content portions **185** stored in the content memory **180** of the memory **145**. Next, in Step **325**, when the one or more content portions **185** stored in the content memory **180** have been deleted (i.e. the content memory **180** is empty), or when the clear attribute **275** is set to "no" (or any other indication not to delete the one or more content portions **185** from the content memory **180**) or not present in the content message **260**, the content management application **165** determines whether the information message **278** of the content message **260** includes a portal element **270**. In Step **330**, when the information message **278** includes a portal element **270**, the associated portal URL **285** is stored in the memory **145** for later access and use by the device user. The process then returns to Step **300** and periodically checks for received messages. In Step **335**, when the information message **278** does not include a portal element **325**, a counter is set to N=1. Next, in Step **340**, the content management application **165** determines whether the information message **278** of the content message **260** includes an Nth content portion. When the Nth content portion is not included in the content message **260**, the process cycles back to Step **300** and periodically checks for received messages. In Step **345**, when the content message **260** includes the Nth content portion, the content management application **165** processes the Nth content portion. Next, in Step **350**, the counter is incremented to N=N+1 and the process cycles back to Step **340** to check for the presence of the Nth content portion in the content message **260**.

FIG. **8** is a flowchart illustrating further detail of the operation of the wireless communication device **60** of FIG. **2** in accordance with a preferred embodiment of the present invention. Specifically, FIG. **8** illustrates the processing of a communicated content portion **280** as in Step **345** of FIG. **7** by the content management application **165** of the wireless communication device **60**. The process begins with Step **360** wherein the content management application **165** determines whether the content portion includes a color element **195**. In Step **365**, when no color element **165** is included, the background and text color of the text area to be displayed on the display **155** for the communicated content portion **280** is set to a default color. In Step **370**, when a color element **165** is included in the communicated content portion **280**, the background color and the text color of the text area to be displayed on the display **155** for the communicated content portion **280** is set to the background color and text color of the background color attribute **235** and the text color attribute **240** preferably sent with the color element **195**. Next, in Step **375**, the content management application **165** determines whether an expiration element **230** is included in the communicated content portion **280**. In Step **380**, when an expiration element **230** is included in the communicated content portion **280**, the content management application **165** compares the current time **245** with the expiration time of the expiration element **230**. In Step **385**, when the current time **245** is at least equal to (i.e., greater than or equal to) the expiration time, the communicated content portion **280** received within the content message **260** is discarded. In Step **390**, when the current time **245** is less than the expiration time, the expiration time is set and stored within the content memory location **180** for the communicated content portion **280**. In Step **395**, when no expiration element **230** is included within the communicated content portion **280** in Step **375**, or when the expiration time has been stored in Step **390**, the content management application **165** determines whether the communicated content portion **280** includes an image element **200**. In Step **400**, when the image element **200** is included, the image file is retrieved. For example, the content management application **165** informs the processor **140** to retrieve the image from the specified URL within the image element **200**. In Step **405**, when no image element **200** is included within the communicated content portion **280**, a default image is identified for the communicated content portion **280**. Next in Step **410**, the image for the communicated content portion **280** is stored in the content memory **180**. Next, in Step **415**, the content management application **165** determines whether the communicated content portion **280** includes a text element **215**. When no text element **214** is present, the process cycles to Step **385** and the communicated content portion **280** received within the content message **260** is discarded. In Step **420**, when the text element **215** is included, the text description of the content contained within the text element **215** is stored in the content memory **180**. Next, in Step **425**, the content management application **165** determines

whether the communicated content portion **280** includes a URL attribute **220**. When no URL attribute **220** is present, the process cycles to Step **385** and the communicated content portion **280** received within the content message **260** is discarded. Next, in Step **430**, when the URL attribute **220** is included, the URL of the content associated with the communicated content portion **180** is stored in the content memory **180**. The user can later retrieve the content associated with the communicated content portion **180** as desired using the stored URL. Next, in Step **435**, the content management application **165** determines whether the time attribute **210** is included within the communicated content portion **280**. In Step **440**, when the time attribute **210** is included, the time for the communicated content portion **280** is stored within the content memory **180**. Next, in Step **445**, when no time attribute is present in Step **435**, the content management application **165** determines whether a title attribute **205** is present in the communicated content portion **280**. When the title attribute is present, the title text is stored in the content memory **180** in Step **450**. When no title attribute **205** is present, the process cycles to Step **385** and the communicated content portion **280** received within the content message **260** is discarded.

The present invention as described herein provides a communication system and method for communication of content. Although the invention has been described in terms of preferred embodiments, it will be obvious to those skilled in the art that various alterations and modifications can be made without departing from the invention. Accordingly, it is intended that all such alterations and modifications be considered as within the spirit and scope of the invention as defined by the appended claims.

What is claimed is:

1. A wireless communication device for acquiring and managing content transmitted from a wireless communication system, the wireless communication device comprising:

a memory for storing content portions;

a receiver for receiving a content message transmitted from the wireless communication system, wherein the content message includes:

a first set of one or more content portions, and

a clear attribute, wherein the clear attribute indicates whether a second set of one or more content portions should be deleted from the memory, the second set of content portions having been stored in the memory prior to receipt of the content message;

a processor coupled to the receiver and the memory for processing the content message; and

a content management application executable by the processor, wherein the content management application, when executed, causes the processor to delete the second set of content portions from the memory in the event that the clear attribute indicates that the second set of content portions should be deleted.

2. The wireless communication device as recited in claim **1**, wherein the content management application, when executed, further causes the processor to store the first set of content portions in the memory after deleting the second set of content portions from the memory.

3. The wireless communication device as recited in claim **1**, wherein the content management application, when executed, further causes the processor to delete the first set of content portions responsive to receipt of a third set of one or more content portions.

4. The wireless communication device as recited in claim **1**, wherein the content message further includes a portal element including a portal URL, the wireless communication device further comprising:

a user interface, coupled to the processor, for receiving inputs from a user of the wireless communication device; and

a browser application executable by the processor, wherein the browser application, when executed, causes the processor to access the portal URL and, while thereat, updating one or more preferences for content receipt in response to one or more inputs from the user interface.

5. The wireless communication device as recited in claim **1**, wherein the content management application, when executed, further causes the processor to process at least one content portion of the first set of content portions.

6. The wireless communication device as recited in claim **5**, further comprising:

a display coupled to the processor for displaying the at least one content portion.

7. The wireless communication device as recited in claim **6**, wherein the at least one content portion includes a color element, and wherein the content management application, when executed, further causes the processor to set a background color and a text color of a text area to be used for displaying the at least one content portion on the display.

8. The wireless communication device as recited in claim **5**, wherein at least one content portion of the first set of content portions includes an expiration element having an expiration time, and wherein the content management application, when executed, further causes the processor to compare a current time to the expiration time, and to discard the at least one content portion when the current time is at least equal to the expiration time.

9. The wireless communication device as recited in claim **5**, wherein at least one content portion of the first set of content portions includes an image reference attribute specifying a predetermined URL, and further wherein the content management application, when executed, further causes the processor to retrieve an image from the predetermined URL and to store the image in the memory as part of the at least one content portion.

10. The wireless communication device as recited in claim **5**, wherein at least one content portion of the first set of content portions includes a text element comprising a text description of the content, and wherein the content management application, when executed, further causes the processor to store the text description of the content in the memory as part of the at least one content portion.

11. The wireless communication device as recited in claim **5**, wherein at least one content portion of the first set of content portions includes a URL attribute comprising a URL of content associated with the at least one content portion, and wherein the content management application, when executed, further causes the processor to store the URL in the memory as part of the at least one content portion, the wireless communication device further comprising:

a user interface, coupled to the processor, for receiving inputs from a user of the wireless communication device; and

a browser application executable by the processor, wherein the browser application, when executed, causes the processor to retrieve the content associated with the at least one content portion using the URL in response to one or more inputs from the user interface.

12. The wireless communication device as recited in claim **5**, wherein at least one content portion of the first set of content portions includes a time attribute comprising a time associated with the at least one content portion, and wherein the content management application, when

executed, further causes the processor to store the title attribute in the memory as part of the at least one content portion.

13. The wireless communication device as recited in claim **5**, wherein at least one content portion of the first set of content portions includes a title attribute comprising title text for the at least one content portion, and wherein the content management application, when executed, further causes the processor to store the title attribute in the memory as part of the at least one content portion.

14. A method for a wireless communication device to acquire and manage content transmitted from a wireless communication system, the method comprising:

receiving a content message transmitted from the wireless communication system wherein the content message includes:

a first set of one or more content portions, and

a clear attribute, wherein the clean attribute indicates whether a second set of one or more content portions should be deleted from a memory of the wireless communication device, the second set of content portions having been stored in the memory prior to receipt of the content message, and

deleting the second set of content portions from the memory in the event that the clear attribute indicates that the second set of content portions should be deleted.

15. The method of claim **14**, further comprising:

storing the first set of content portions in the memory after the deleting step.

16. The method of claim **14**, wherein the content message further includes a portal element including a portal URL, the method further comprising:

storing the portal URL in the memory;

accessing the portal URL though a browser application of the wireless communication device in response to an input from a user interface of the wireless communication device; and

updating one or more preferences for content receipt through the browser application in response to another input from the user interface while accessing the portal URL.

17. The method of claim **15**, further comprising:

processing the first set of content portions through use of a content management application.

18. The method of claim **17**, further comprising:

displaying at least one content portion of the first set of content portions on a display of the wireless communication device.

19. The method of claim **18**, wherein at least one content portion of the first set of content portions includes a color element, and further wherein the processing step comprises:

setting a background color and a text color of a text area to be used for displaying the at least one content portion on the display.

20. The method of claim **17**, wherein at least one content portion of the first set of content portions includes an expiration element having an expiration time, and further wherein the processing step comprises:

comparing a current time to the expiration time, and

discarding the at least one content portion when the current time is at least equal to the expiration time.

21. The method of claim **17**, wherein at least one content portion of the first set of content portions includes an image reference attribute specifying a predetermined URL, and wherein the processing step comprises:

retrieving an image from the predetermined URL; and

storing the image in the memory as part of the at least one content portion.

22. The method of claim **17**, wherein at least one content portion of the first set of content portions includes a text element comprising a text description of the content, wherein the processing step comprises:

storing the text description of the content in the memory as part of the at least one content portion.

23. The method of claim **17**, wherein at least one content portion of the first set of content portions includes a URL attribute comprising a URL of content associated with the at least one content portion, wherein the processing step comprises:

storing the URL in the memory as part of the at least one content portion, and

retrieving the content associated with the at least one content portion through a browser application using the URL in response to an input from a user interface.

24. The method of claim **17**, wherein at least one content portion of the first set of content portions includes a time attribute comprising a time associated with the at least one content portion, wherein the processing step comprises:

storing the time attribute in the memory as part of the at least one content portion.

25. The method of claim **17**, wherein at least one content portion of the first set of content portions includes a title attribute comprising title text for the at least one content portion, wherein the processing step comprises:

storing the title attribute in the memory as part of the at least one content portion.

26. A method for supplying content to a wireless communication device in a wireless communication system, the method comprising:

transmitting a content message to produce a transmitted content message, wherein the content message includes:

a first set of one or more content portions, and

a clear attribute, e clear attribute indicates whether a second set of one or more content portions should be deleted from a memory of the wireless communication device, the second set of content portions having been stored in the memory prior to transmission of the content message;

receiving the transmitted content message; and

deleting the second set of content portions from the memory in the event that the clear attribute indicates that the second set of content portions should be deleted.

* * * * *

EXHIBIT 7

US007383983B2

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

(10) **Patent No.: US 7,383,983 B2**
(45) **Date of Patent: *Jun. 10, 2008**

(54) **SYSTEM AND METHOD FOR MANAGING CONTENT BETWEEN DEVICES IN VARIOUS DOMAINS**

(75) Inventors: **Michael T. Gaumond**, Scottsdale, AZ (US); **Richard Mark Clayton**, Phoenix, AZ (US); **Parvathy Bhaskaran**, Mesa, AZ (US); **Lee Callaway**, Oak Park, IL (US)

(73) Assignee: **General Instrument Corporation**, Horsham, PA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 195 days.

This patent is subject to a terminal disclaimer.

(21) Appl. No.: **11/118,528**

(22) Filed: **Apr. 29, 2005**

(65) **Prior Publication Data**

US 2005/0252959 A1 Nov. 17, 2005

Related U.S. Application Data

(60) Provisional application No. 60/574,601, filed on May 7, 2004.

(51) **Int. Cl.**
G06F 17/00 (2006.01)
G06K 19/06 (2006.01)

(52) **U.S. Cl.** **235/375**; 235/492

(58) **Field of Classification Search** 235/375, 235/492; 340/425.5; 709/217
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,389,463	B2 *	5/2002	Bolas et al.	709/219
7,050,834	B2 *	5/2006	Harwood et al.	455/563
7,079,015	B2 *	7/2006	Gibeau	340/425.5
7,171,221	B1 *	1/2007	Amin et al.	455/462
2002/0183059	A1 *	12/2002	Noreen et al.	455/427
2005/0009561	A1 *	1/2005	Hollstrom et al.	455/556.1

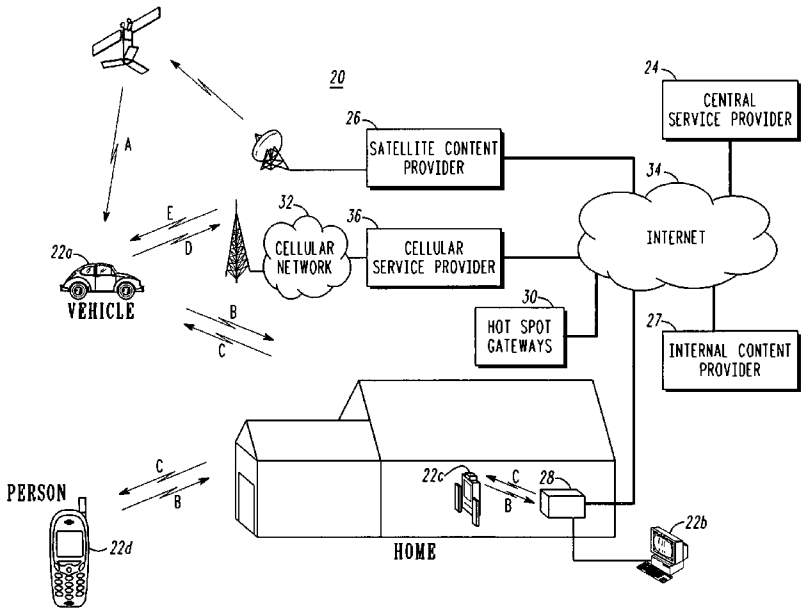
* cited by examiner

Primary Examiner—Steven S. Paik
Assistant Examiner—April A Taylor
(74) *Attorney, Agent, or Firm*—Stewart M. Wiener

(57) **ABSTRACT**

A system and method for managing content between different client devices in various domains (such as vehicle, home, and person). The system and method include receiving an input from a user on the first client device to pause the content. After receiving the input, the first client device determines whether the first client device is connected to a wireless communication system. If the first client device is connected to the wireless communication system, the first client device sends a data message to the second client device through a host system. In one embodiment, the second client device will then store the content in the second client device after receiving the data message to permit the user to resume playback of the content on the second client device. In another embodiment, the host system will store the content itself or access the content from a content provider and transfer the content directly to the second device with a data message that indicates the paused location of the content.

41 Claims, 9 Drawing Sheets



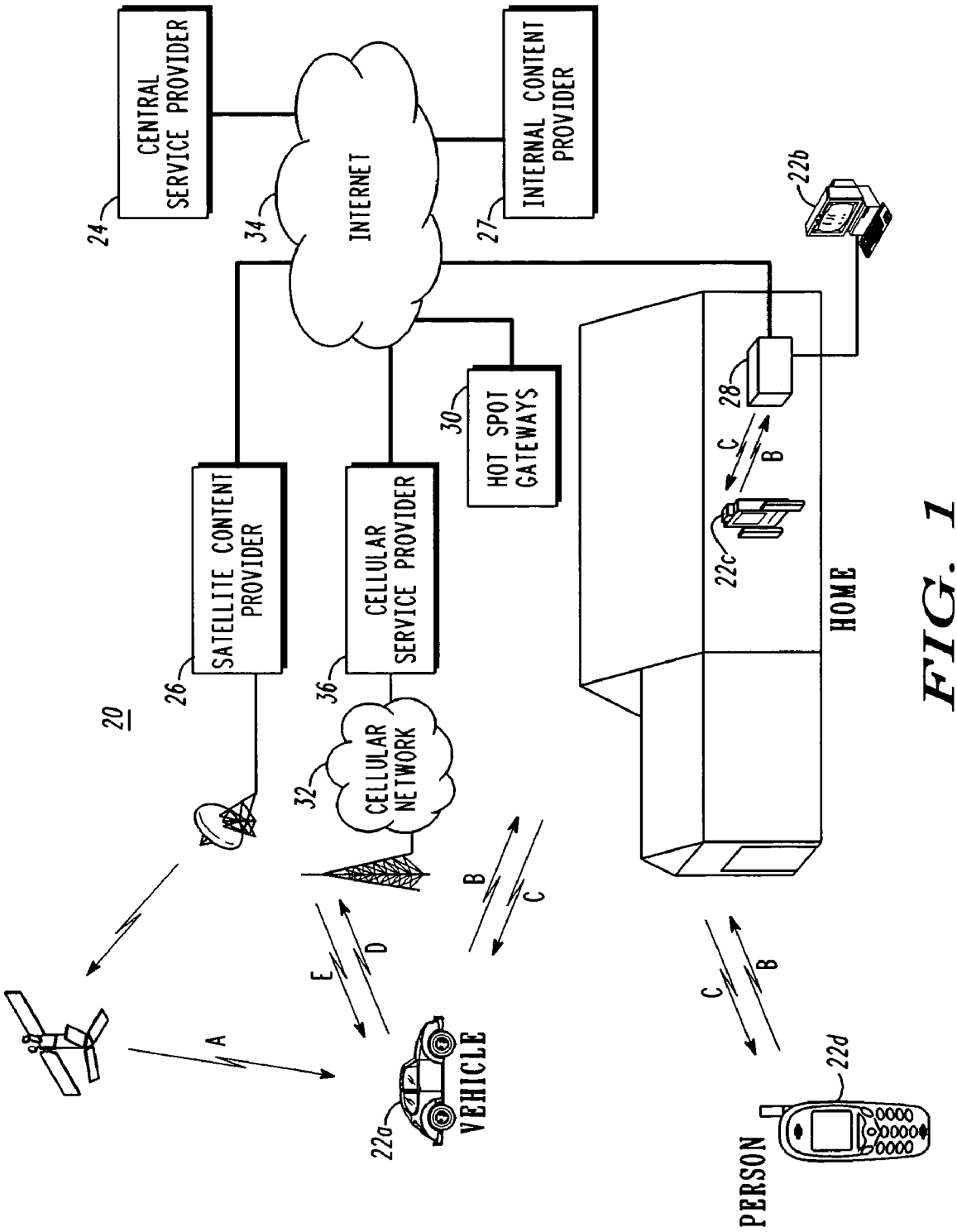
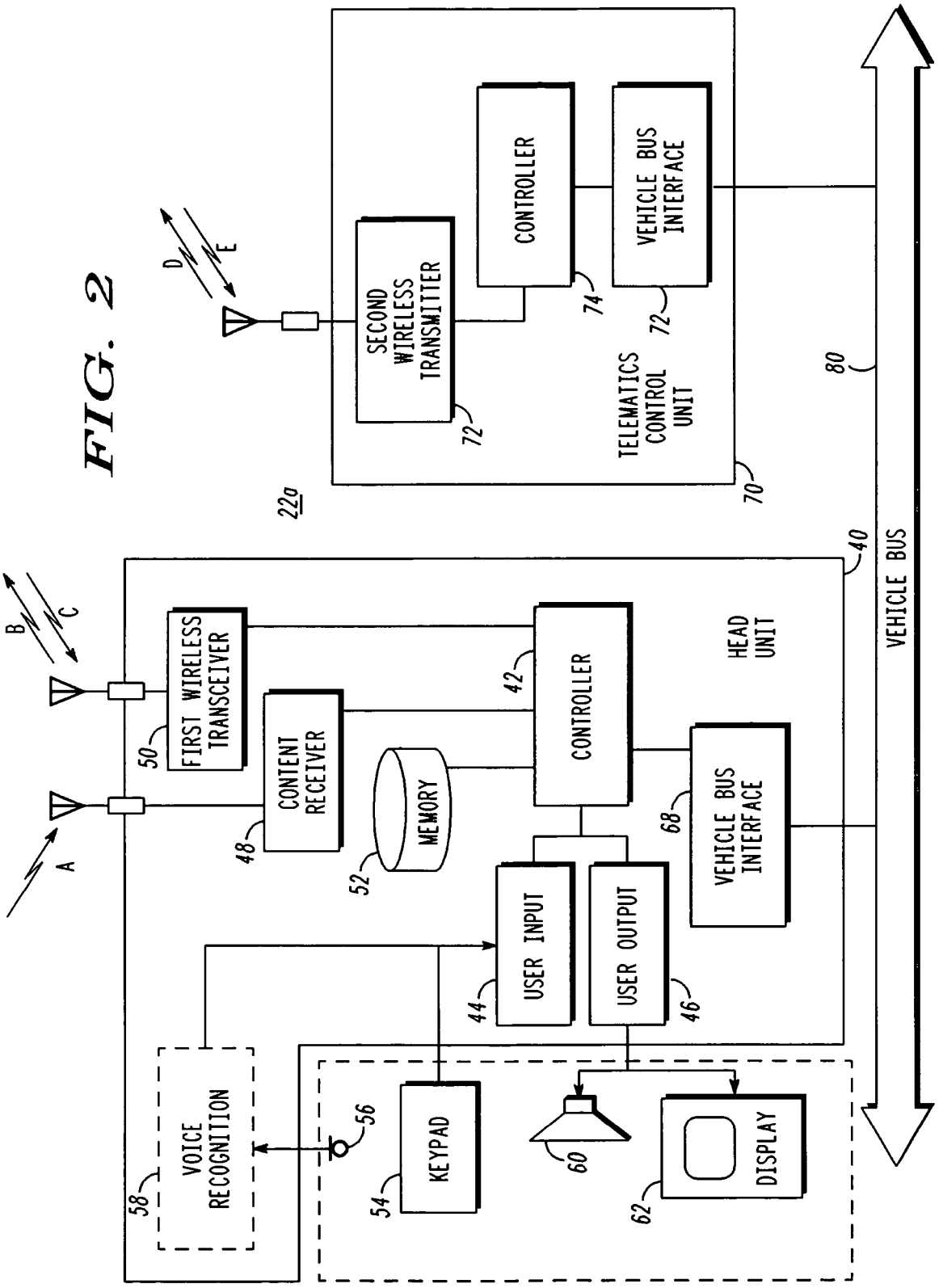


FIG. 1

FIG. 2



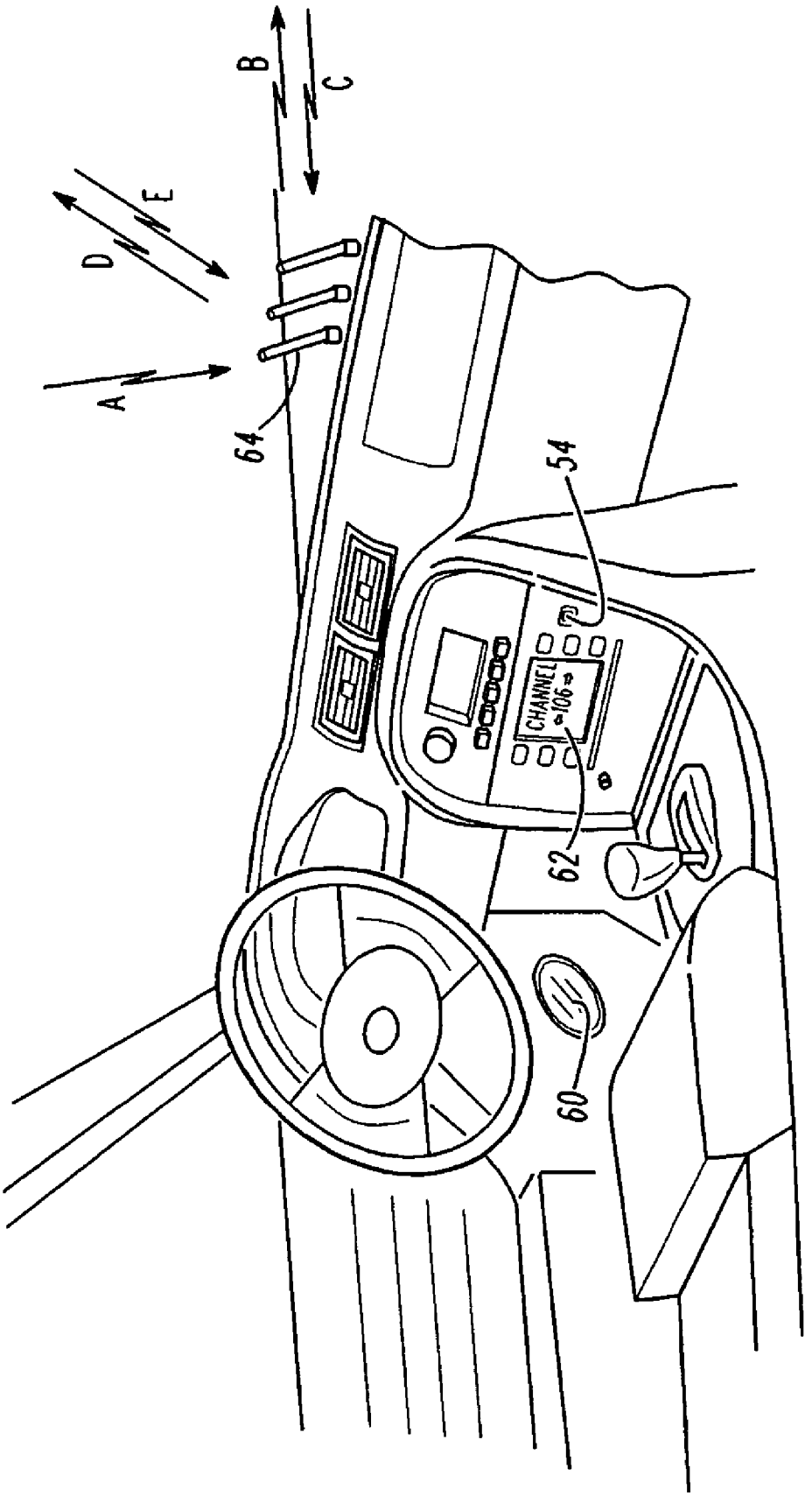


FIG. 3

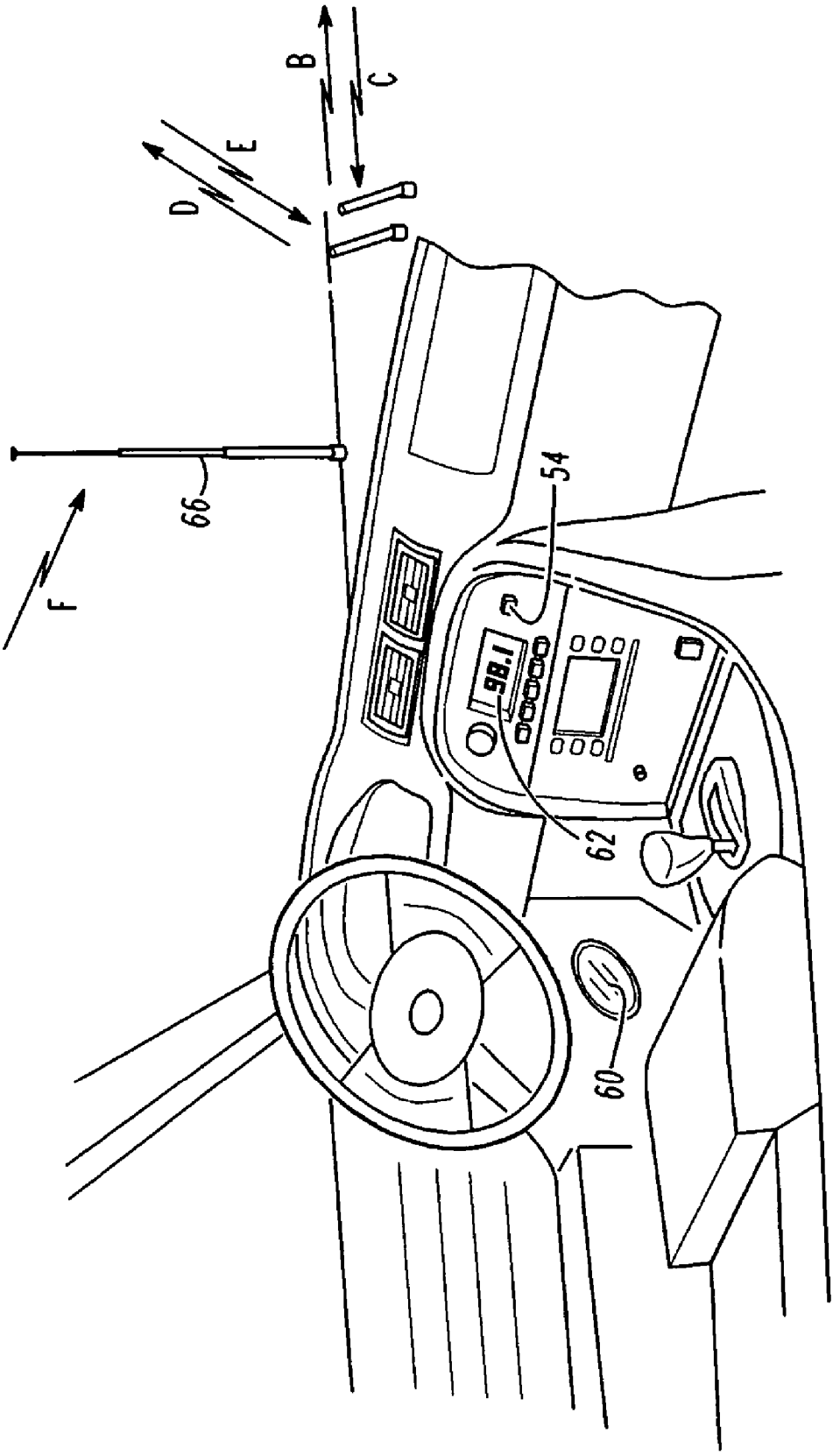


FIG. 4

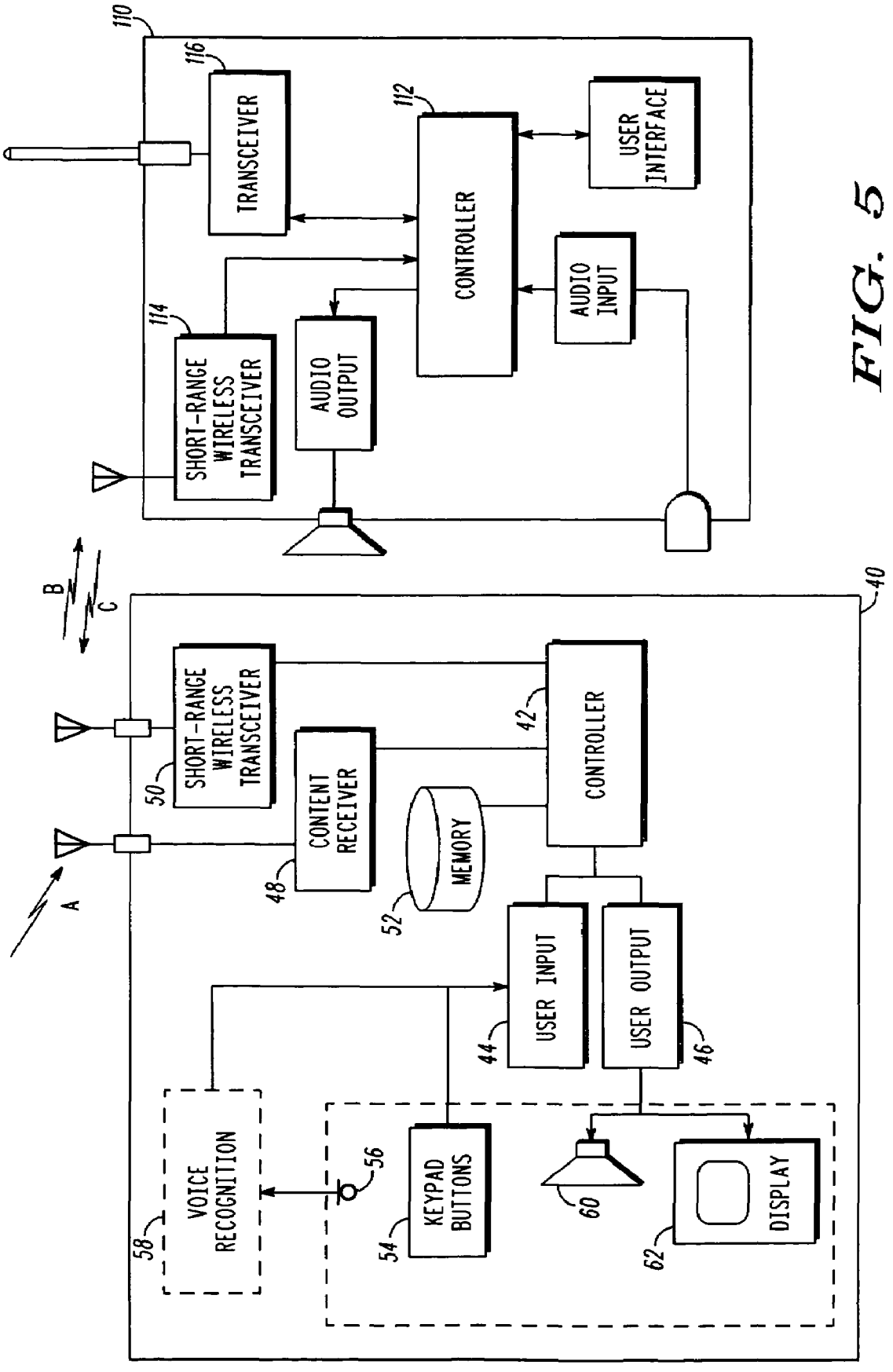


FIG. 5

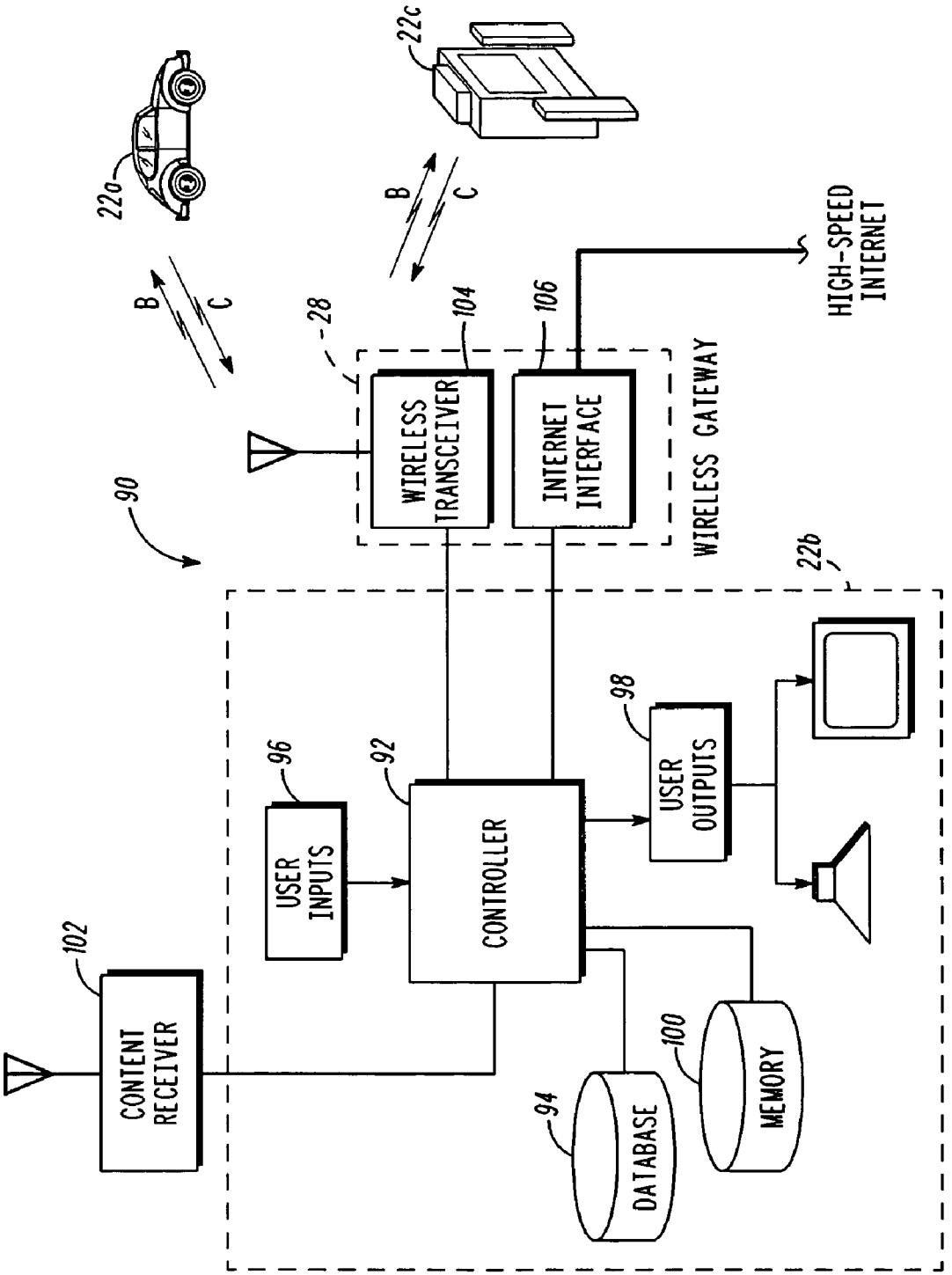


FIG. 6

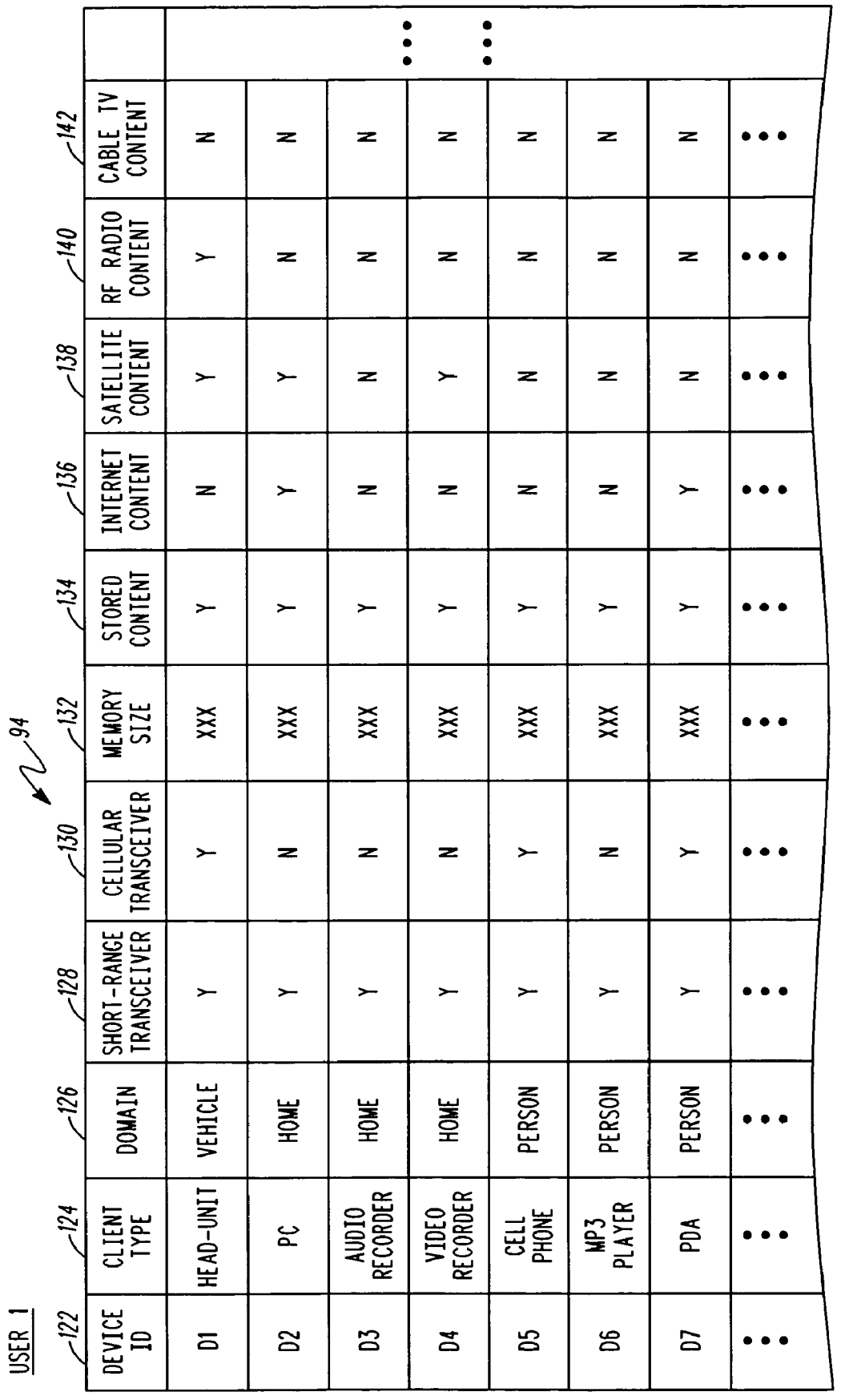


FIG. 7

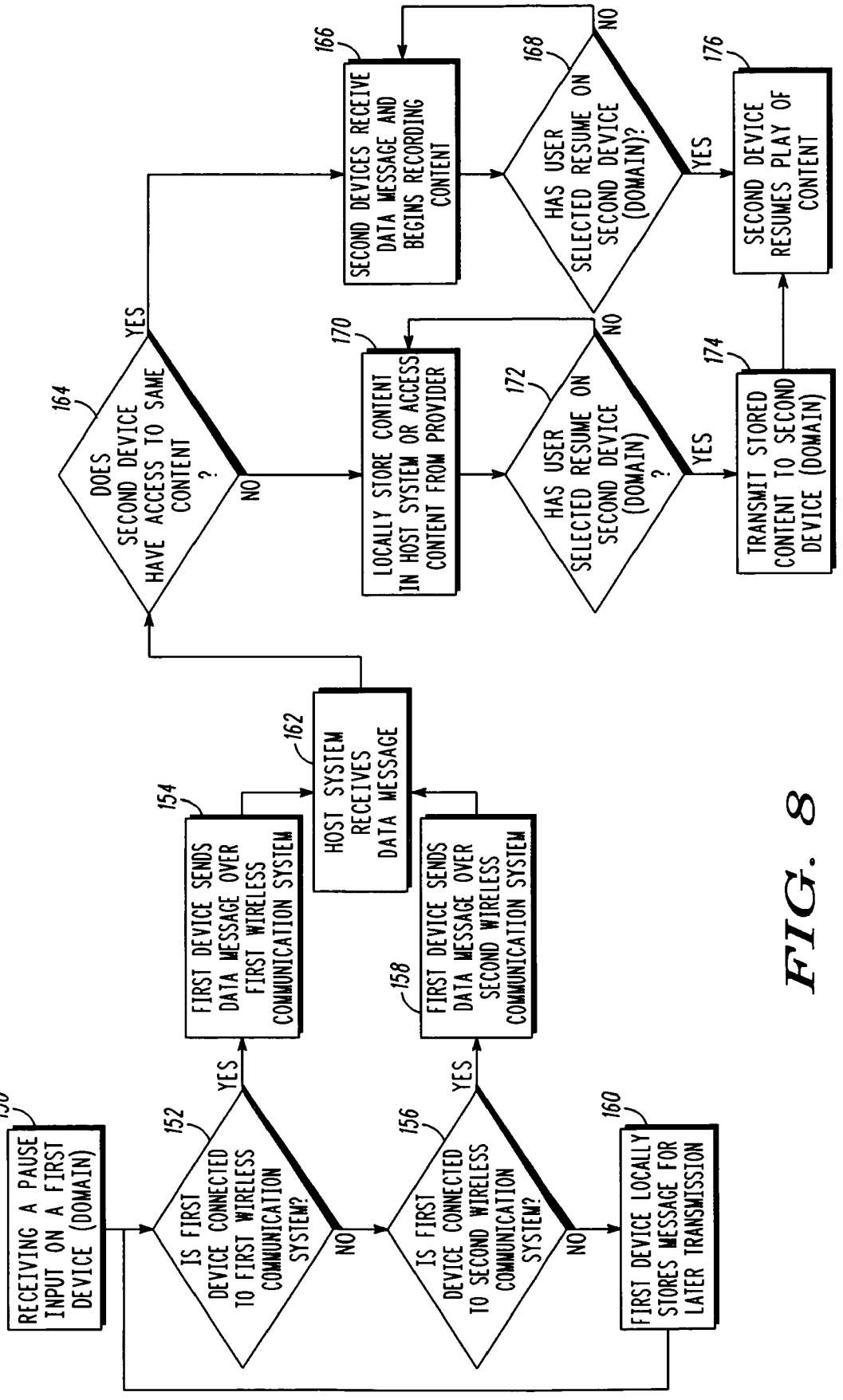


FIG. 8

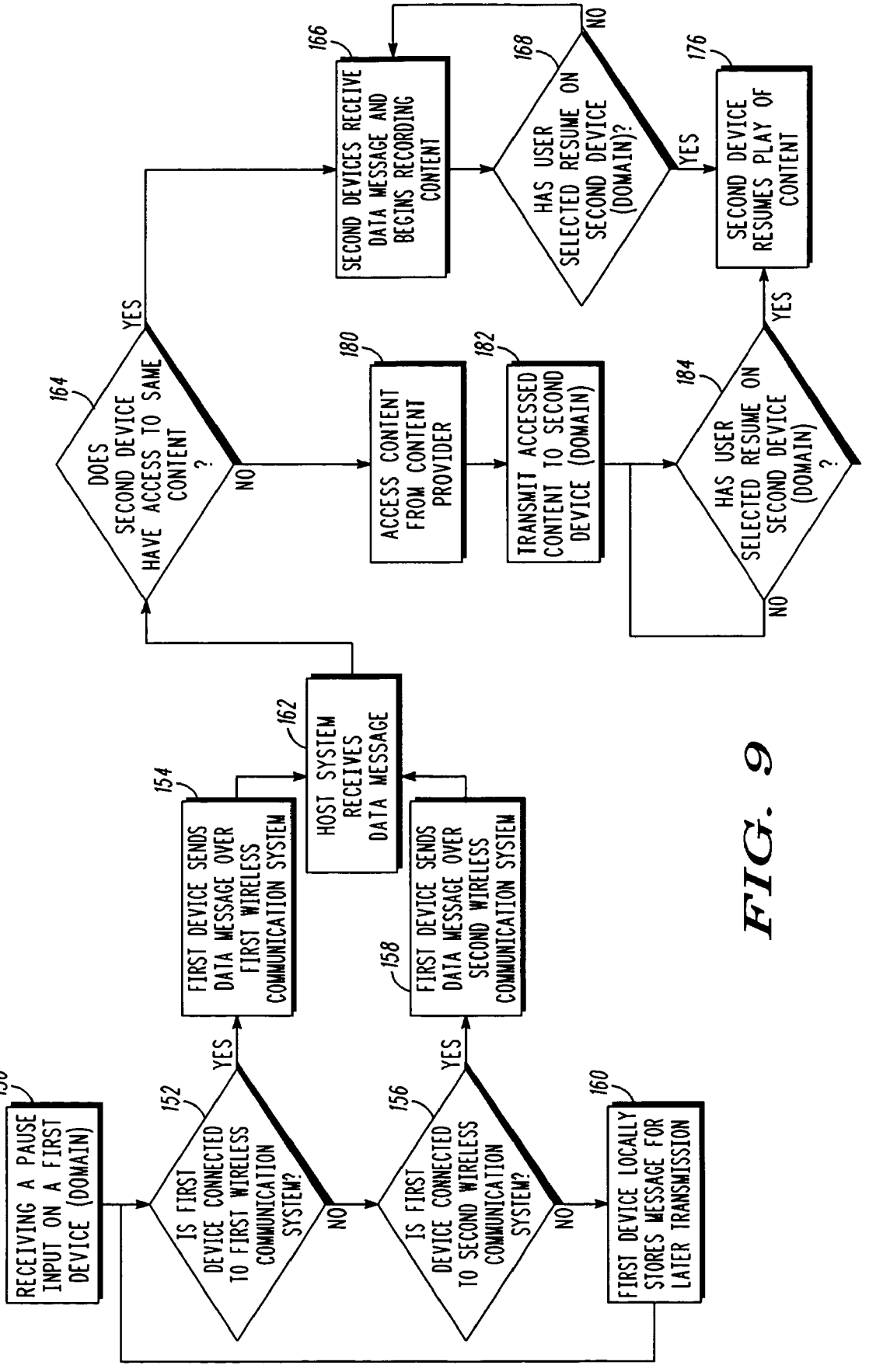


FIG. 9

SYSTEM AND METHOD FOR MANAGING CONTENT BETWEEN DEVICES IN VARIOUS DOMAINS

CROSS-REFERENCE TO RELATED APPLICATION

The instant application claims priority to U.S. Provisional Patent Application No. 60/574,601, filed May 7, 2004.

FIELD OF THE INVENTION

This invention in general relates to managing content between devices in various domains and, more particularly, to a system and method for pausing content in one device and resuming playback of the content in another device that may be in a different domain.

BACKGROUND OF THE INVENTION

Digital video recorders exist that allow a user to pause and store video content and playback the video content at a later time. Conventional systems, however, are limited in that they only deal with storing video content and playing back the content in a single domain.

A need exists for a user to seamlessly listen to (or watch) audio (or video) content when moving from one domain (such as a vehicle) to a different domain (such as a home) without missing a portion of that content. For instance, assume that a user is driving in their vehicle and listening to an audio broadcast content such as a talk show or a sporting event. When the user arrives at their home and parks the vehicle, the user may wish to continue listening to the broadcast content at a later time and, preferably, in a different domain (such as their home). Currently, the user cannot pause the broadcast content in the vehicle and resume the broadcast content at the exact spot later in their home. The same applies for video, e.g., pause a digital video recorder at home and resume playing the video content on an entertainment system in the vehicle.

It is, therefore, desirable to provide a system and method to overcome or minimize most, if not all, of the preceding problems especially in the area of managing content in different domains.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is a block diagram of a system having client devices in different domains supported by a server at a central service provider;

FIG. **2** is a block diagram of one embodiment of a client device;

FIG. **3** is a perspective view of the inside of a vehicle illustrating one embodiment of a client device in the vehicular domain;

FIG. **4** is a perspective view of the inside of a vehicle illustrating another embodiment of a client device in the vehicular domain;

FIG. **5** is a block diagram of another embodiment of a client device in communication with a separate local wireless communication device;

FIG. **6** is a block diagram of one embodiment of a communication system that communicates with different client devices in different domains;

FIG. **7** is a diagram of one embodiment of a database that may reside in a host system to access information and characteristics about a particular client device;

FIG. **8** is a flow diagram of one method for managing content between a first client device and a second client device; and

FIG. **9** is a flow diagram of another method for managing content between a first client device and a second client device.

While the invention is susceptible to various modifications and alternative forms, specific embodiments have been shown by way of example in the drawings and will be described in detail herein. However, it should be understood that the invention is not intended to be limited to the particular forms disclosed. Rather, the invention is to cover all modifications, equivalents and alternatives falling within the spirit and scope of the invention as defined by the appended claims.

DETAILED DESCRIPTION

What is described is a system and method for managing content between different client devices in various domains (such as vehicle, home, person, hotspot, workplace, or school). To this end, in one embodiment there is a method for managing content between a first client device and a second client device. The method comprises the steps of: receiving an input from a user on the first client device to pause the content, determining whether the first client device is connected to a first wireless communication system, sending a data message to the second client device if the first client device is connected to the first wireless communication system; and storing the content in the second client device after receiving the data message to permit the user to resume playback of the content on the second client device.

The content may be a variety of types including broadcast content and stored content. Broadcast content may include digital content such as from a digital satellite communication system. Broadcast content may also include analog or digital content from local radio broadcasters over a radio tuner. The content may also be stored content in that a user may have the system store and manage personal content and content that is accessible from Internet content providers.

In one embodiment, the first communication system is a short-range wireless communication system such as a Bluetooth™ communication system or an IEEE 802.11 communication system. The first device may also include a second transceiver for communication with a second wireless communication system such as a cellular communication network. In a further embodiment, where the first device has a second transceiver, the method may further include the steps of: determining whether the first device is connected to the second wireless communication system and sending the data message over the second wireless communication system if a connection exists. If the first client device is not connected to either the first or second wireless communication system, the first client device may store the data message for later transmission.

In another embodiment, there is a method for managing content between a first client device and a second client device, where the first client device is in a first domain (such as a vehicle) and the second client device is in a second domain (such as in a home or on the person). The method comprises the steps of: receiving an input from a user on the first device to pause the content; determining whether the first device is connected to a first wireless communication system; and sending a data message to a wireless gateway if the first device is connected to the first wireless communication system. Here, the data message sent to the wireless gateway comprises a plurality of information elements that

includes at least a pause content instruction. If the content is broadcast content, the information elements in the data message may further include a content identification (such as a satellite channel identification or a radio frequency identification). The information elements may also include a host address, a content type identification, a user identification, and a date and a time that the user selected the pause content command. If the content is stored content, the information elements in the data message may further include a content identification and a pause location identification as well as other information to enhance user functionality.

In a further embodiment, there is a client device in a communication system where the system is capable of managing content between the client device and other client devices in various domains. The client device includes at least a user interface, a first wireless transceiver, and a controller. The user interface is used to receive an input from a user to pause the content. The first wireless transceiver is capable of wireless connecting the client device to a first wireless communication system. The controller, in response to the input from the user to pause the content, determines whether the client device is connected to the first wireless communication system. If so, the controller generates and sends a data message to a wireless gateway over the first communication system. The data message sent to the wireless gateway includes at least a pause content instruction and any additional information as described briefly above and in more detail below.

In another embodiment, there is a communication system for managing content between a first client device and a second client device. The communication system includes at least a wireless gateway, a database, and a controller. The wireless gateway is configured to receive a first data message from the first client device. The data message comprises a plurality of information elements including at least a pause content instruction. The database is configured to store information regarding the first client device and the second client device. The controller is connected to the wireless gateway and the database. The controller is capable of identifying the second device from the database and sending a second data message (or the content itself) to the second client device in response to receiving the first data message from the first client device.

Now, turning to the drawings, FIG. **1** illustrates a top-level block diagram of an example use of a communication system **20** for the present invention. Generally, the communication system **20** may include a plurality of client devices **22***a*, **22***b*, **22***c*, **22***d* that exist in various domains such as the vehicle, home, and person. For instance, a client device **22***a* in the vehicular domain may be incorporated into a vehicle's head unit and/or entertainment system. A client device **22***b*, **22***c* in the home domain may include a personal computer, a home entertainment system, a digital audio recorder, and/or a digital video recorder. A client device **22***d* in the personal domain may include a portable electronic device such as a personal digital assistant (PDA), a digital music player, and/or a portable phone. Client devices may also exist in other domains such as a hotspot, workplace, or school.

The communication system **20** also includes a central service provider **24** that can communicate with the client devices **22***a-d* through a combination of wireless and wired links. In one embodiment, the central service provider **24** is connected to a high-speed Internet network **34**. The central service provider **24** may assist in managing the distribution and control of content between different client devices **22***a-d*. The central service provider **24** may further provide additional services or be incorporated into the services of other service providers such as a cellular service provider, a satellite broadcast content provider, a cable television content provider, or a stored Internet content provider. Moreover, some or all of the functions of managing the distribution and control of the content between client devices **22***a-d* may reside locally with a user in the home domain.

In one embodiment, where the client device **22***a-d* has a content receiver, the client devices **22***a-d* may receive broadcast content (audio and/or video) from a satellite content provider **26**. This is shown in FIG. **1** through an exemplary satellite content provider and the receipt of a communication link A to the client devices **22***a-d*. In other embodiments, the client device **22***a-d* may also have a receiver to receive broadcast content via radio signals from local content broadcasters (not shown). The client device **22***a-d* may also receive stored content from an Internet content provider **27**. The Internet content provider **27** may provide stored music or video content to users or be part of a cable television provider. If the client device is a portable or mobile unit (such as a client device **22***a* in the vehicular domain or a client device **22***d* in the person domain), as explained in more detail below, the client device may receive stored content from a home gateway **28** or a hot spot gateway **30** through a short-range communication system.

As illustrated in FIG. **1**, the client devices **22***a-d* may wirelessly communicate in the communication system **20** through different communication links (see communication arrows B-E). The wireless communication links B-E may be divided into individual sets (B-C, D-E) for different types of wireless communication protocols. For instance, the client devices **22***a-d* may include a first wireless transceiver that is capable of establishing a wireless communication link B-C through a short-range wireless communication system or network. In this embodiment, the short-range wireless communication system or network may include a Bluetooth™ communication system or an IEEE 802.11 communication system. The short-range wireless transceiver in a client device **22***a-d* may provide direct communication to another client device **22***a-d* through a home wireless gateway **28** (such as from the client device **22***a* in the vehicle to the client device **22***b*, **22***c* in the home). Alternatively, the short-range wireless transceiver in a client device **22***a-d* may provide indirect wireless communication to another client device through a hot spot gateway **30** (such as from a client device **22***a* in the vehicle, through the hot spot gateway **30**, to the client device **22***b*, **22***c* in the home). The wireless communication links over the short-range communication system can provide for the exchange of data messages as well as the transfer of stored content to client devices.

Additionally, the client devices **22***a-d* may include a second wireless transceiver that is capable of establishing a wireless communication link D-E through a second wireless communication system, such as a cellular communication system and network **32**. The cellular communication system and network **32** can operate according to a wireless communication protocol such as a Global System for Mobile Communications (GSM) protocol, a Code Division Multiple Access (CDMA) protocol, or a Time Division Multiple Access (TDMA) protocol. Here, the cellular system or network **32** is further coupled to the Internet **34** by the cellular service provider **36** or other wired network on route to the central service provider **24**, which may ultimately act as the host for data message communications between client devices **22***a-d*. Alternatively, the cellular system or network **32** is coupled to the Internet **34** or other wired network on

route to another client device (such as from the client device **22***a* in the vehicle, through to the cellular network provider **36**, to the client device **22***b*, **22***c* in the home). The above described wireless communication protocols are merely representative of existing protocols that could be used in the present invention.

An exemplary client device **22***a* in a vehicular domain will now be described in further detail with relation to FIGS. **2-4**. In one embodiment, a client device **22***a* in a vehicle may comprise of two main components: a head unit **40** and a Telematics control unit **70**. Although shown as separate components for purposes of illustration, one skilled in the art having the benefit of this disclosure will recognize that aspects of the head unit **40** and the Telematics control unit **70**, and components thereof, can be combined or swapped. In any event, in the embodiment as shown in FIG. **2**, the head unit **40** may include a controller **42**, a user interface **44**, **46**, a content receiver **48**, a first wireless transceiver **50**, a memory **52**.

The user interface includes a user input **44** and a user output **46**. The user input **44** may include a keypad or a specific user dedicated set of buttons **54**. The user may use the keypad or dedicated buttons to perform particular functions of the present invention, including a request to pause content or a request to resume playback of content. Additionally, a microphone **56** may also be used to pick up a speaker's voice in the vehicle, and/or possibly to give commands to the head unit **40** if it is equipped with a voice recognition module **58**.

Ultimately, user inputs **44** are processed by the controller **42** in the head unit **40**. The controller **42** also executes processes to provide outputs to the occupants in the vehicle through the user output **46**, such as through a speaker **60** and/or a display **62**. The speakers **60** employed can be the audio (radio) speakers normally present in the vehicle, of which there are typically four or more, although only one is shown for convenience.

The content receiver **48** in the client device **22***a* is capable of receiving broadcast content (audio and/or video) from a content provider **26**. This is shown in FIG. **3** through an exemplary satellite content provider by the receipt of a communication link A to the client device **22***a*. The user may use the user interface to select one of a plurality of satellite channels that are received by the satellite receiver antenna **64**. In other embodiments, as shown in FIG. **4**, the client device **22***a* may also have an antenna **66** and a radio receiver to receive broadcast content via radio signals F from local content broadcasters in the geographic area. This type of broadcast content may be obtained through the content receiver **48** by tuning a radio receiver to a specific radio frequency.

Referring back to FIG. **2**, the client device **22***a* may also obtain stored content (as well as exchange data messages with a host system or other client devices) through the use of the first wireless transceiver **50**. The first wireless transceiver **50** is used for establishing wireless communications B-C over a short-range wireless communication system or network. For instance, as mentioned above, the short-range wireless communication system or network may include a Bluetooth™ communication system or an IEEE 802.11 communication system. Although shown as part of the head unit **40**, the first wireless transceiver **50** could also be included as part of the Telematics control unit **70** or other vehicle control unit. In any event, the short-range wireless transceiver **50** may provide wireless communication to another client device **22***b-d* over a home gateway **28** (such as a data message from the client device **22***a* in the vehicle to the client device **22***b*, **22***c* in the home) or may provide wireless communication to another client device or content provider through a hot spot gateway **30** (such as a data message from the client device **22***a* in the vehicle, through the hot spot gateway **30**, to the client device **22***b*, **22***c* in the home; or such as obtaining stored content from the internet content provider **27**, through the hot spot gateway **30**, to the client device **22***a*).

Additionally, the client device **22***a* includes a memory **52** for storing content. The memory **52** is controlled by the controller **42** and is responsive to user inputs **44** and to certain data messages that may be received by the controller **42** from other client devices **22***b-d*. For instance, assume that a user of the vehicular client device **22***a* is listening to broadcast content on a specific satellite channel over the content receiver **48**. The user may desire to pause the broadcast content while the user talks to another occupant in the vehicle. The user may then select an input on the keypad or other dedicated button **54** to pause the content. The system may also be configured to automatically generate a pause command upon the initiation of a user action such as the changing of a channel, selecting a mute button on the entertainment system, or turning off the entertainment system or vehicle.

In response to that input (generated directly or indirectly by the user), the controller **42** would use the memory **52** to begin storing the broadcast content on the specific satellite channel to enable the user to play back the content at a later time. Additionally, the memory **52** may be used for storing specific programs of the broadcast content that a user desires to playback at a time that is different from the original broadcast time. For example, as explained below, the controller **42** in the vehicular client device **22***a* may receive data messages from another client device **22***b-d* in a different domain (i.e. home or person). That data message may include an instruction to start the recording of a program of the broadcast content (in whole or in part) on a specific satellite channel. Moreover, the memory **52** may be used to download specific content from an Internet content provider **27** through a home gateway **28** or a hot spot gateway **30**.

In the embodiment where the client device **22***a* is incorporated into the head unit **40** of a vehicle, the controller **42** may also be configured to communicate via a vehicle bus interface **68** to a vehicle bus **80**, which carries communication information and other operational data throughout the vehicle. This connection may be important to allow the controller **42** to utilize a cellular communication transceiver in the Telematics control unit **70** to transmit and receive data messages. In particular, the Telematics control unit **70** is similarly coupled to the vehicle bus **80**, via a vehicle bus interface **72**, and hence the head unit **40**. The Telematics control unit **70** is responsible for sending and receiving voice or data communications to and from the vehicle over a cellular communication network. As such, it comprises a Telematics controller **74** to organize such communications, and a network access device (NAD) that includes a cellular wireless transceiver **76**, which may be used as a second wireless transceiver for purposes of the present invention as described further below.

In an alternative embodiment, the client device **22***a* may be configured to seek out a local wireless communication device to determine whether a data message may be sent over a second wireless communication system. For instance, in FIG. **5**, the controller **42** in the client device **22***a* may determine whether it has access to a second wireless communication network through another wireless communication device **110** that is within proximity of the short-range

wireless transceiver **50**. Here, the wireless communication device **110**, such as a cellular phone, has its own controller **112**, a short-range wireless transceiver **114** and a cellular wireless transceiver **116**. The short-range wireless transceivers **50** and **114** may communication with each other through a short-range communication protocol such as is set forth in the Bluetooth™ communication system and an IEEE 802.11 communication system. The benefit of this design is that it will allow the controller **42** to utilize a cellular transceiver in the local wireless communication device **110** to transmit and receive data messages, which may be used as a second wireless transceiver for purposes of the present invention as described further below.

The client device **22***a* illustrated above can provide a great deal of communicative flexibility within vehicle to manage and control content with other client devices **22***b-d* owned by a user. For example, assume for purposes of illustration that a driver of the vehicle, using client device **22***a*, is listening to a program of audio broadcast content such as a talk show or a sporting event on the content receiver **48**. When the driver arrives at their home, the driver may select an input on the keypad or other dedicated button **54** to pause the playback of the content. Additionally, as mentioned above, the system may also be configured to automatically generate a pause command input upon the initiation of a user action such as the changing of a channel, selecting a mute button on the entertainment system, or turning off the entertainment system or vehicle. In either event, in response to that input, the controller **42** would use the memory **52** to begin storing the broadcast content from the content receiver **48**. This would permit the driver to playback the stored content at a later time in the vehicle. Additionally, in response to that input, the controller **42** would generate a data message for transmission to other client devices **22***b-d*.

In one embodiment, before sending a data message, the controller **42** would determine whether the client device **22***a* in the vehicle is connected to, or capable of sending messages over, a first wireless communication system. This may be done by having the controller **42** determine whether the first wireless transceiver **50** is connected to a wireless gateway **28** in the home or a hot spot gateway **30** mentioned above. If the client device **22***a* in the vehicle is not connected to, or not capable of sending data messages over, a first wireless communication system, the controller **42** may then make a determination whether the client device **22***a* in the vehicle is connected to, or capable of sending data message over, a second wireless communication system. The second wireless communication system in FIG. **2** is shown as a cellular wireless transceiver **72** in the Telematics control unit **70** and in FIG. **5** as a cellular wireless transceiver **116** in a portable wireless communication device **110**. If the client device **22***a* is not connected to either the first or second communication system, then the controller **74** may store the data message for later transmission.

The data message may be formatted in a number of ways. In one embodiment, the information in the data message will depend on the type of content being played by the driver of the vehicle. For instance, assume that the content receiver **48** is a digital satellite receiver and that the type of content to be paused is broadcast content. In this case, the data message may include a plurality of information elements or fields that includes at least a pause content instruction and a satellite channel identification. To enhance the functionality of the system, the data message may also include other information elements or fields such as an address (for identifying a host for the driver's content), a client device identification, a user identification, and a date and a time that the user selected the pause content command. This later information may be used to access content that is not readily storable as well as provide specific information to a user about stored content for later selection.

In another instance, assume that the content receiver **48** is a radio tuner and the type of content to be paused is broadcast content. In this case, the data message may include a plurality of information elements or fields that includes at least a pause content instruction and a radio frequency identification. Additionally, to enhance functionality, the data message may also include other information elements or fields such as an address (for identifying a host for the driver's content), a client device identification, a content type identification, a user identification, and a date and a time that the user selected the pause content command.

In a further instance, assume that the client device **22***a* contains stored content and the user was listening to a specific song of the stored content or a previously stored broadcast program. In this case, the data message may include a plurality of information elements or fields that includes at least a pause content instruction, a content identification, and a pause location identification. Other information elements or fields that may be included, for enhancing functionality, include an address (for identifying a host for the driver's content), a client device identification, a user identification, and a date and a time that the user selected the pause content command.

FIG. **6** shows a block diagram of one embodiment of a host system **90** for managing content between a first client device (such a client device **22***a* in a vehicle) and a second client device (such as a client device **22***c* in the home). In one embodiment, the host system includes a wireless gateway **28**, a controller **92**, and a database **94**. Additionally, the system **90** may include other components such as a user inputs **96**, user outputs **98**, other memory **100**, and a content receiver **102**. In one embodiment, components of the host system may be included in one of the client devices **22***b*, such as a home personal computer. However, one skilled in the art having the benefit of this disclosure will recognize that aspects of the host system, and components thereof, can be combined or swapped with other types of devices and systems. For instance, instead of having the host system located in the home, the controller **92**, database **94**, and memory **100** may be located and managed remotely by the central service provider **24**.

The wireless gateway **28** may include a wireless transceiver **104** and an Internet interface **106**. In one embodiment, the wireless transceiver **104** is capable of wirelessly connecting to the first and second client devices **22***a*, **22***c* over a short-range wireless communication system such as a Bluetooth™ communication system and an IEEE 802.11 communication system. The Internet interface **106** may be used for communicating with a central service provider **24**. The connection with the central service provider **24** may also be used to facilitate communications with the first and second client device **22***a*, **22***c*, if the client devices are connected to other wireless gateways (such as a hot spot gateway **30**) or connected to another wireless communication system (such as a cellular communication system). Moreover, the connection with the central service provider **24** may be used to facilitate access to content providers in addition to, or separate from, the content receiver **102**.

In any event, in addition to transferring stored content to client devices, the wireless gateway **28** is used to receive data messages from the first and second client devices **22***a*, **22***c*, including any data messages that have an instruction to pause broadcast content or stored content. In one embodi-

ment, as described above, data messages that are received from a client device include a plurality of information elements or fields that include at least a pause content instruction. If the content is broadcast content, the data message may further include information elements or fields that identify the type of client device, identify the content (a specific satellite channel or a radio frequency), and identify the time and date of the pause instruction. If the content is stored content, the data message may further include information elements or fields that identify the type of client device, identify the content (a specific artist, album, song, or stored broadcast program), and identify the location of the pause instruction.

The database **94** is used by the system to store information regarding features and operations of the different client devices **22***a-d*. The database **94** may also be used to store user preferences and keep track of user stored content. The database **94** may be configurable by the user to facilitate the storage and playback of content between different client devices **22***a-d* in various domains. For instance, the database **94** may identify the different types of client devices **22***a-d*, associated with a particular user. Referring to FIG. **7**, the database **94** may also include information specific to individual client devices **22***a-d* such as: the client device identification (**122**); the client type (**124**); the domain (**126**); whether the client device has a short-range transceiver (**128**); whether the client device has a cellular transceiver (**130**); the size of memory on the client device (**132**); whether the client device retains stored content (**134**); whether the client device has access to Internet content (**136**); whether the client device has access to satellite content (**138**); whether the client device has access to RF radio content (**140**); and whether the client device has access to cable television (**142**).

Referring back to FIG. **6**, the controller **92** is connected to the wireless gateway **28** and the database **94**. The controller **92** is capable of receiving data messages from a first client device **22***a* and then using the database **94** to identify other client devices **22***b-d* associated with a specific user. As mentioned above, the data message will include an instruction to pause content and associated information about the paused content. In one embodiment, the controller **92** is capable of generating and sending a second set of data messages to other client devices **22***b-d* after receiving the data message from the first client device **22***a*. This may be beneficial if the other client devices **22***b-d* have their own content receiver and memory.

In another embodiment, the controller **94** is capable of accessing content from the central service provider **24** in response to receiving the data message (containing the pause instruction) from the first client device **22***a*. This feature may be beneficial if the other client devices **22***b-d* do not have their own content receiver. Additionally, after accessing content from the central service provider **24**, the controller **92** may locally store the content in memory **100** or pass the content onto the other client devices **22***c*, **22***d* connected to the host system **90**.

FIG. **8** shows a flow diagram illustrating one embodiment of a method for managing content between a first client device and a second client device. For purposes of illustration, the first client device will be in a first domain and the second client device will be in a second domain. In one embodiment, the method includes a process block **150** that receives an input from a user on the first device to pause the content. As mentioned above, this input may be a direct instruction by the user to pause the content (e.g., by pressing a button on a keypad or a dedicated pause button **54**) or the input may be a result of some action taken by the user (e.g., changing a channel, selecting a mute button on the entertainment system, or turning off the entertainment system or vehicle). Moreover, the content may include either broadcast content or stored content. The process then proceeds to decision block **152**.

At decision block **152**, the first device will make a determination whether the first device is connected to a first wireless communication system. In the context of embodiment described in FIGS. **2** and **3**, this may include having the controller **42** determine whether a short-range wireless transceiver **50** is connected to another short-range communication device such as to a home wireless gateway **28** or a hot spot gateway **30**. If the first device is connected to the first wireless communication system, then the process proceeds to process block **154**.

At process block **154**, the first device will generate and transmit a data message over the first communication system to the second device. In one embodiment, as described above, the data message may depend on the type of content being played by the user of the first device. For instance, assume that the type of content to be paused is broadcast content. In this case, the data message may include a plurality of information elements or fields that includes at least a pause content instruction and a broadcast content identification (such as a satellite channel or a radio frequency). To enhance the functionality of the system, the data message may also include other information elements or fields such as an address, a client device identification, a content type identification, a user identification, and a date and a time that the user selected the pause content command.

In another instance, assume that the first device contains stored content and the user was listening to a specific song of the stored content or a previously stored broadcast program. In this case, the data message may include a plurality of information elements or fields that includes at least a pause content instruction, a content identification, and a pause location identification. Other information elements or fields that may be included, for enhancing functionality, include an address, a client device identification, a user identification, and a date and a time that the user selected the pause content command.

In any event, if the first device is not connected to the first wireless communication system, then the process proceeds to decision block **156**. In one embodiment, a determination is then made whether the user is capable of connection through a second wireless communication system. As described above, this may be accomplished by having the controller **42** determine whether a data message may be transmitted via a cellular wireless transceiver **72** in a connected Telematics control unit **70** (see FIG. **2**). Alternatively, this may be accomplished by having the controller **42** determine whether a data message may be transmitted via a cellular wireless transceiver **116** in a wirelessly connected communication device **110** (see FIG. **5**). Moreover, the decision on whether the user is capable of connecting to a second wireless communication system may include a determination of whether the user has subscribed to preferred services plan of the central service provider **24**. If so, the process proceeds to process block **158** and where the first device will generate and send a data message over the second communication system.

If the first device is not connected to the first wireless communication system or the second wireless communication system, then the first device **22***a* will generate a data message but will store the data message instead of immediately transmitting the data message (block **160**). In this

case, it is preferred that the information in the data message include a paused location for the content such as a date and a time or a record and a track. The process will return to decision blocks **152** and **156** to wait until the first communication device is connected to the first or second wireless communication system.

Once the data message is transmitted, at process block **162**, the host system **90** will receive the data message from the first device over the first communication system or the second communication system. As mentioned above, the first communication system may be a short-range wireless communication system transmitted directly to a home gateway **28** or to the home gateway **28** through a remote hot spot gateway **30**. The second communication system may be a cellular system that transmits the message to the host system **90** through a cellular network. In response, the host system **90** will generate and send a second data message to other second devices. As explained below, the second data message may be a modified data message that is addressed directly to the second devices associated with the user of the first device and based on the features and capabilities of the second device.

For instance, at decision block **164**, the host system **90** may determine from the database **94** whether the second device **22***c* has access to the content that the user desires to be stored. If the second device has direct access to the same content, then at process block **166**, the second device will receive a data message from the host system **90**. In response to the data message from the host system, the second device will then begin to record the content based on the information contained in the data message. As illustrated in decision block **168**, in one embodiment, the second device will continue to store the content on a particular satellite channel or a radio frequency until the second device receives an input from the user to resume playback of the content on the second device. Alternatively, the second device may be configured to store the content for a predetermined period of time after receiving the pause instruction (such as 2 hours) or may be configured to store the content until a particular program on the broadcast content is completed.

Alternatively, in process block **170**, if the second device **22***c* does not have access to the same content, then the host system may record the content itself or access the content from a content service provider. The ability to access content from a content service provider can provide significant benefits to the user. For instance, if the data message originally sent to the host system is late (e.g., if the data message with the paused instruction was not sent immediately), the host system **90** may connect to a service provider to download the requested content. As illustrated in decision block **172**, in one embodiment, the host system **90** will continue to store the content or access the content until the second device receives an input from the user to resume playback of the content on the second device. At process block **174**, once the user desires to playback the content, the host system **90** will transmit the stored or accessed content to the second device. The process then proceeds to block **176** where the second device resumes playback of the content.

FIG. **9** shows a flow diagram illustrating a further embodiment of a method for managing content between a first client device and a second client device. This method is similar to the one described with relation to FIG. **8**, but adds additional steps **180**, **182**, and **184**. In particular, if it is determined that the second device does not have access to the same content (decision block **164**), the host system **90** will then access the content from another source (block **180**). For instance, the host system **90** could obtain any paused content from the Internet content provider **27** shown in FIG. **1**. In process block **182**, the host system **90** will then transfer the accessed content to the second device **22***c*. The transferred content should also include a data message that identifies the content and paused location of the content. Then, in decision block **184**, the second device will wait until the user has selected to resume the content on the second device **22***c*. At that point, the process proceeds to block **176** where the second device resumes playback of the content.

What has been described is a communication system in a vehicle that includes the capability of managing and controlling content between different devices in different domains. The system and method allow a user to seamlessly listen (or watch) audio (or video) content when moving from one domain (such as a vehicle) to a different domain (such as home) without missing a portion of that content.

Although the invention has been described herein by reference to an exemplary embodiment thereof, it will be understood that such embodiment is susceptible of modification and variation without departing from the inventive concepts disclosed. Specifically, any and all wireless communication(s) or wireless communication system(s) may be substituted with wired communication(s) or wired communication system(s), respectively, or a "wireless and wired" hybrid system. In addition, similar to Bluetooth™ and IEEE 802.11, wireless communication protocols such as wireless USB, Zigbee, or any other wireless communication protocol, may be used for wireless communication(s) and/or wireless communication system(s) described above. All such modifications and variations, therefore, are intended to be encompassed within the spirit and scope of the appended claims.

What is claimed is:

1. A method for managing content between a first device and a second device, the method comprising the steps of:

(a) receiving an input in a first device to pause a content;

(b) determining whether said first device is connected to a first wireless communication system;

(c) sending a data message to said second device if said first device is connected to the first wireless communication system; and

(d) resuming playback of the content on said second device, wherein said first device being in a first domain and said second device being in a second domain, said first domain and said second domain selected from a group consisting of at least the home, vehicle, and person.

2. The method of claim **1** further comprising the step of storing said content in said second device prior to step (a).

3. The method of claim **1** further comprising the step of storing said content in said second device after receiving said data message from step (c).

4. The method of claim **1**, wherein said first device and said second device is selected from a group consisting of a vehicular entertainment system, a home entertainment system, and a portable electronic device.

5. The method of claim **1**, wherein said content is broadcast content and received by a digital satellite communication system.

6. The method of claim **5**, wherein said data message comprises a plurality of information elements including at least a pause content instruction and a satellite channel identification.

7. The method of claim **1**, wherein said content is broadcast content received by a radio tuner.

8. The method of claim 7, wherein said data message comprises a plurality of information elements including at least a pause content instruction and a radio frequency identification.

9. The method of claim 1, wherein said content is stored content received by a content provider through a wireless gateway.

10. The method of claim 1, wherein said content is video and said first device is a digital video recorder and said second device is a vehicular video entertainment system.

11. The method of claim 1, wherein said content is video and said first device is a digital video recorder and said second device is a portable digital video recorder.

12. The method of claim 1 further comprising the step of storing said data message in the first device if it is determined that said first device is not connected to the first wireless communication system.

13. The method of claim 1, wherein said first wireless communication system is at least one of a short range wireless communication system, an IEEE 802.11 communication system, a wireless USB communication system, and a low power radio frequency communication system.

14. The method of claim 13, wherein said first device has a first transceiver to communicate with said first wireless communication system and a second transceiver to communicate with a second wireless communication system.

15. The method of claim 14, wherein said second wireless communication system is a cellular communication system.

16. The method of claim 15 further comprising the steps of determining whether said first device is connected to said second wireless communication system, and sending said data message to said second device if said first device is connected to said second wireless communication system.

17. The method of claim 16 further comprising the step of storing said data message in said first device if it is determined that said first device is not connected to said second wireless communication system.

18. The method of claim 14 further comprising the steps of determining whether said first device is subscribed to a preferred service plan, and connecting to said second wireless communication system if it is determined that said first device is subscribed to the preferred service plan.

19. The method of claim 1 further comprising the step of determining whether said content is broadcast content or stored content prior to sending said data message to said second device.

20. The method of claim 19 wherein, if it is determined that said content is broadcast content, said data message comprising a plurality of information elements including at least a pause content instruction and a satellite channel identification.

21. The method of claim 19 wherein, if it is determined that said content is stored content, said data message comprising a plurality of information elements including at least a pause content instruction, a content identification, and a pause location identification.

22. The method of claim 1 further comprising the steps of:

receiving said data message in a host system, said host system connected to a wireless gateway;

determining in said host system whether said second device has access to the content;

sending a second data message to said second device if it is determined that said second device has access to the content.

23. The method of claim 22 further comprising the step of:

accessing said content from a central service provider in response to receiving said data message in the host system.

24. The method of claim 23 further comprising the step of sending said content accessed from said central service provider to said second device in response to a playback instruction.

25. A client device in a communication system, the communication system being capable of managing content between the client device and other client devices, the client device and other client devices being in other domains, the domains selected from a group consisting of at least the home, vehicle, and person, the client device comprising:

a user interface for receiving an input to pause said content;

a first wireless transceiver that is capable of wirelessly connecting the client device to a first wireless communication system;

a controller, responsive to the input to pause said content, for determining whether said client device is connected to said first wireless communication system and sending a data message to a wireless gateway if said client device is connected to said first wireless communication system, wherein said data message sent to said wireless gateway comprises a plurality of information elements including at least a pause content instruction.

26. The client device of claim 25, wherein said client device is selected from a group consisting of a vehicular entertainment system, a home entertainment system, and a portable electronic device.

27. The client device of claim 25, wherein said client device further comprises a satellite radio receiver, the content being broadcast content received from the satellite radio receiver.

28. The client device of claim 27, wherein the plurality of information elements of said data message further includes at least a satellite channel identification.

29. The client device of claim 25, wherein the client device further comprises a radio tuner, the content being broadcast content received from the radio tuner.

30. The client device of claim 29, wherein the plurality of information elements of the data message further includes at least a radio frequency identification.

31. The client device of claim 25, wherein said content is stored content received by a content provider through a wireless gateway.

32. The client device of claim 25, wherein said content is video and said client device is a home digital video recorder.

33. The client device of claim 25, wherein said first wireless communication system is at least one of a short range wireless communication system, an IEEE 802.11 communication system, a wireless USB communication system, and a low power radio frequency communication system.

34. The client device of claim 33, wherein said client device further comprises a second wireless transceiver that is capable of wirelessly connecting said client device to a cellular wireless communication system.

35. The client device of claim 34, wherein said controller sends said data message through said second wireless transceiver if said client device is not connected to the first wireless communication system.

36. The client device of claim 35, wherein said controller further determines whether said client device is subscribed to a preferred service plan and sends said data message

through said second wireless transceiver if said client device is subscribed to said preferred service plan.

37. The client device of claim **25**, wherein said controller further determines whether said content is broadcast content or stored content prior to sending said data message to said wireless gateway.

38. The client device of claim **37**, wherein the plurality of information elements of said data message further includes at least a satellite channel identification if it is determined that the content is broadcast content.

39. The client device of claim **37**, wherein the plurality of information elements of said data message further includes at least a content identification and a pause location identification if it is determined that said content is stored content.

40. The client device of claim **25**, wherein said wireless gateway is in a home domain and connected to a host system, said host system having a database that contains information about other client devices associated with a user of said client device.

41. A communication system for managing content between a first device and a second device, the communication system comprising:

a wireless gateway for receiving a first data message from a first device, said first data message comprising a plurality of information elements including at least a pause content instruction;

a database for storing information regarding said first device and a second device; and

a controller connected to said wireless gateway, said controller capable of identifying said second device from said database and sending a second data message to said second device in response to receiving said first data message from said first device;

wherein said first device being in a first domain and said second device being in a second domain, said first domain and said second domain selected from a group consisting of at least the home, vehicle, and person.

* * * * *