US 20060223518A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2006/0223518 A1**

Haney (43) **Pub. Date: Oct. 5, 2006**

(54) **LOCATION SHARING AND TRACKING USING MOBILE PHONES OR OTHER WIRELESS DEVICES**

(76) Inventor: **Richard D. Haney**, Union City, CA (US)

Correspondence Address:
SCHWEGMAN, LUNDBERG, WOESSNER & KLUTH, P.A.
P.O. BOX 2938
MINNEAPOLIS, MN 55402 (US)

(21) Appl. No.: **11/099,362**

(22) Filed: **Apr. 4, 2005**

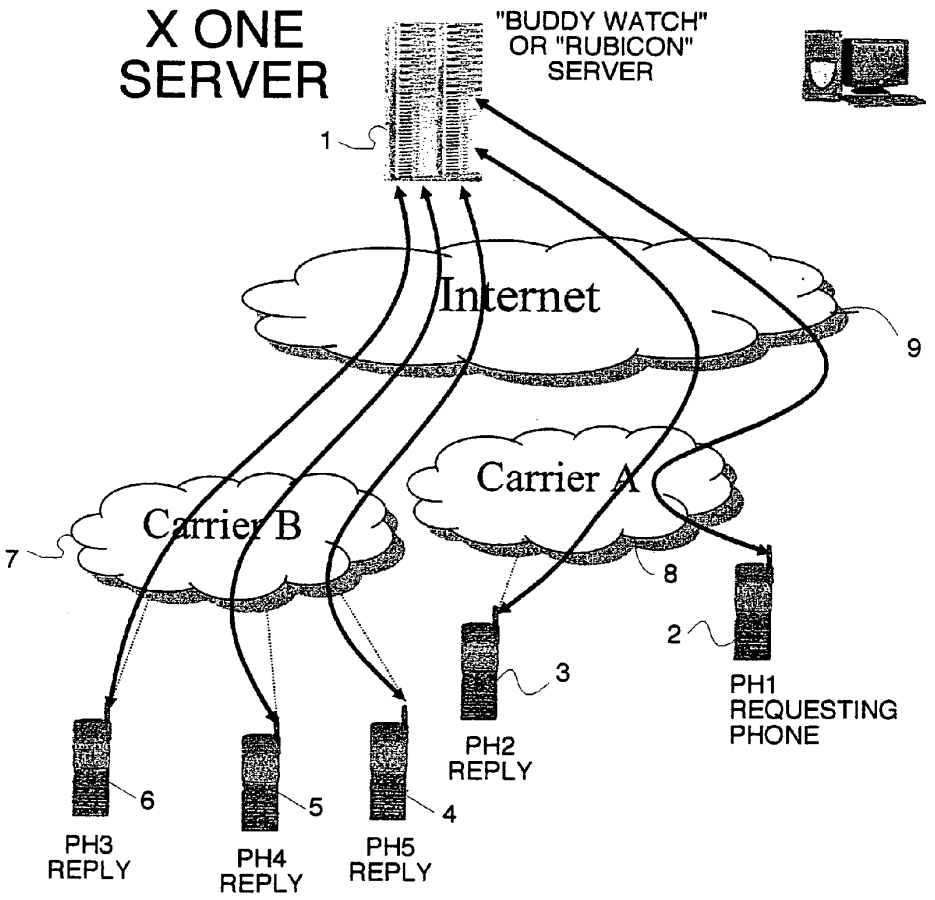
Publication Classification

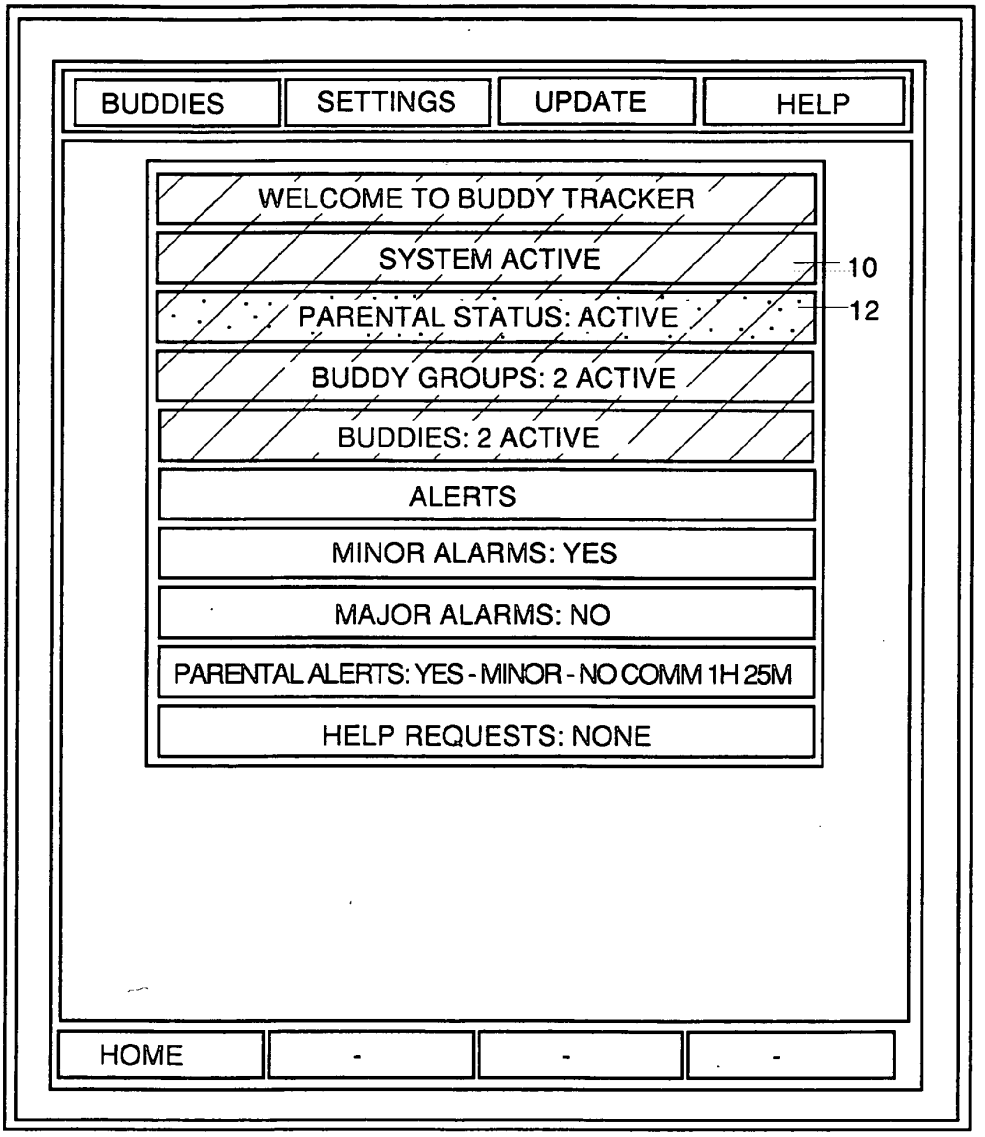
(51) **Int. Cl.**
H04M 3/00 (2006.01)

(52) **U.S. Cl.** **455/420**

(57) **ABSTRACT**

A system for exchanging GPS or other position data between wireless devices for purposes of group activities, child location monitoring, work group coordination, dispatching of employees etc. Cell phones and other wireless devices with GPS receivers have loaded therein a Buddy Watch application and a TalkControl application. The Buddy Watch application communicates with the GPS receiver and other wireless devices operated by buddies registered in the users phone as part of buddy groups or individually. GPS position data and historical GPS position data can be exchanged between cell phones of buddies and instant buddies such as tow truck drivers via a buddy watch server. Emergency monitoring services can be set up with notifications to programmable individuals in case an individual does not respond. Positions and tracks can be displayed. TalkControl simplifies and automates the process of joining talk groups for walkie talkie services such as that provided by Nextel.





OPENING SCREEN

FIG. 1

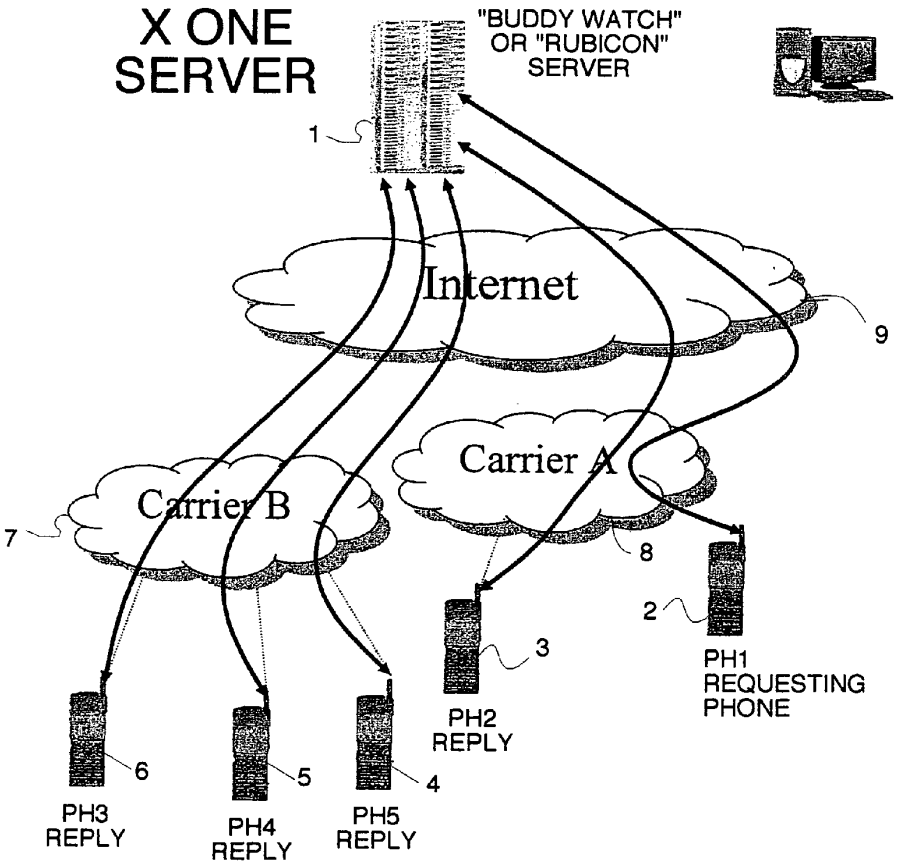
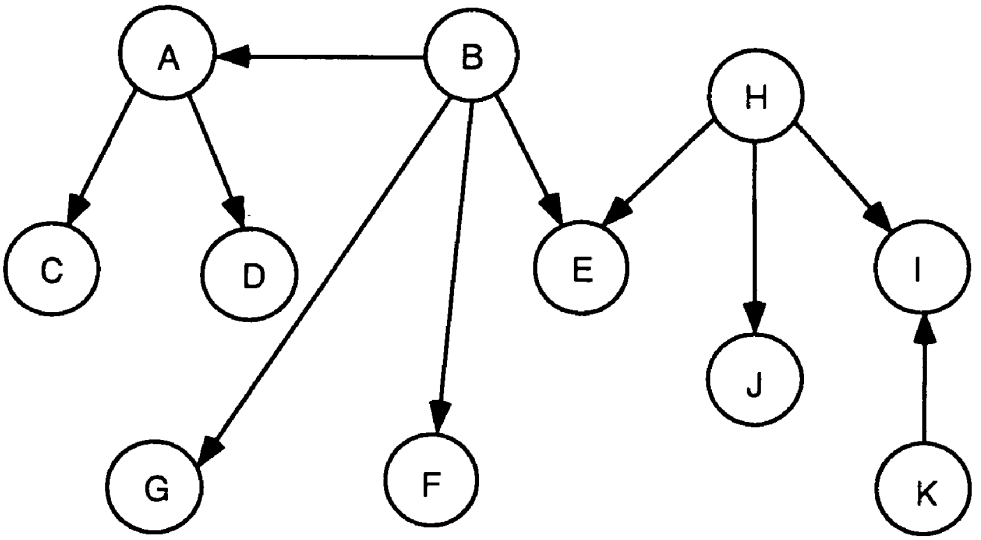

X ONE SERVER
"BUDDY WATCH" OR "RUBICON" SERVER
1
Internet
9
Carrier A
Carrier B
7
8
2
3
PH1 REQUESTING PHONE
PH2 REPLY
6
5
4
PH3 REPLY
PH4 REPLY
PH5 REPLY

FIG. 2A



MATRIX OF BUDDY LIST

FIG. 2B

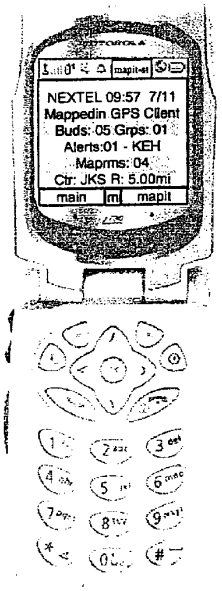


FIG. 2C

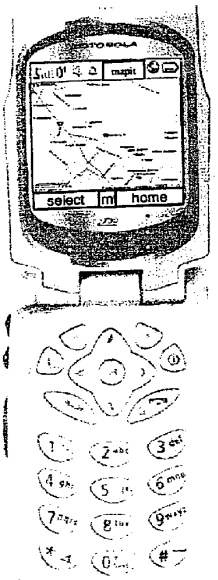


FIG. 2D

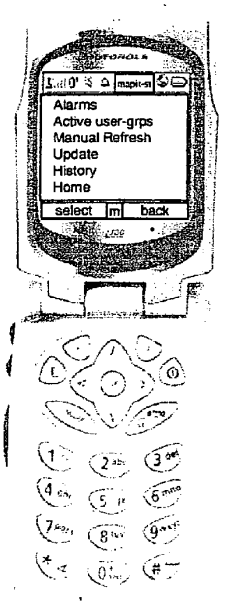


FIG. 2E

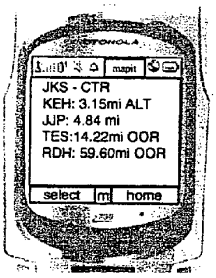
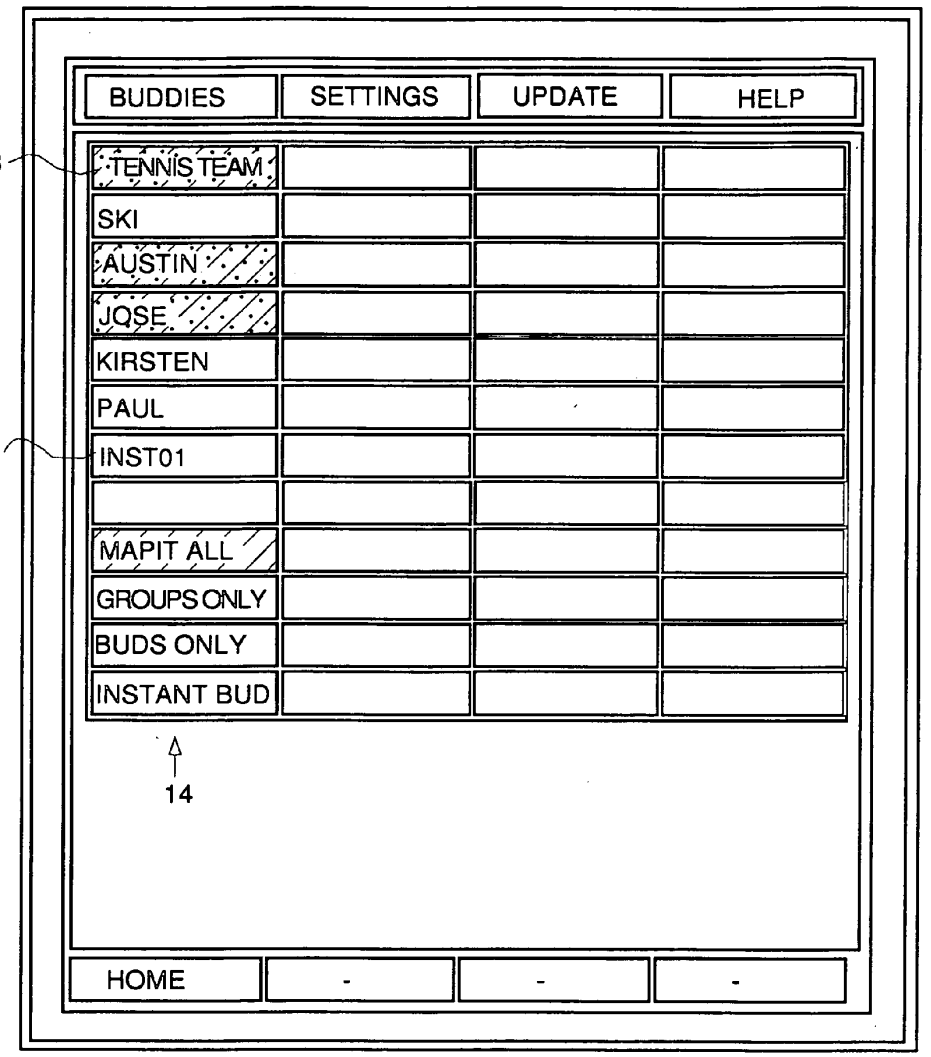
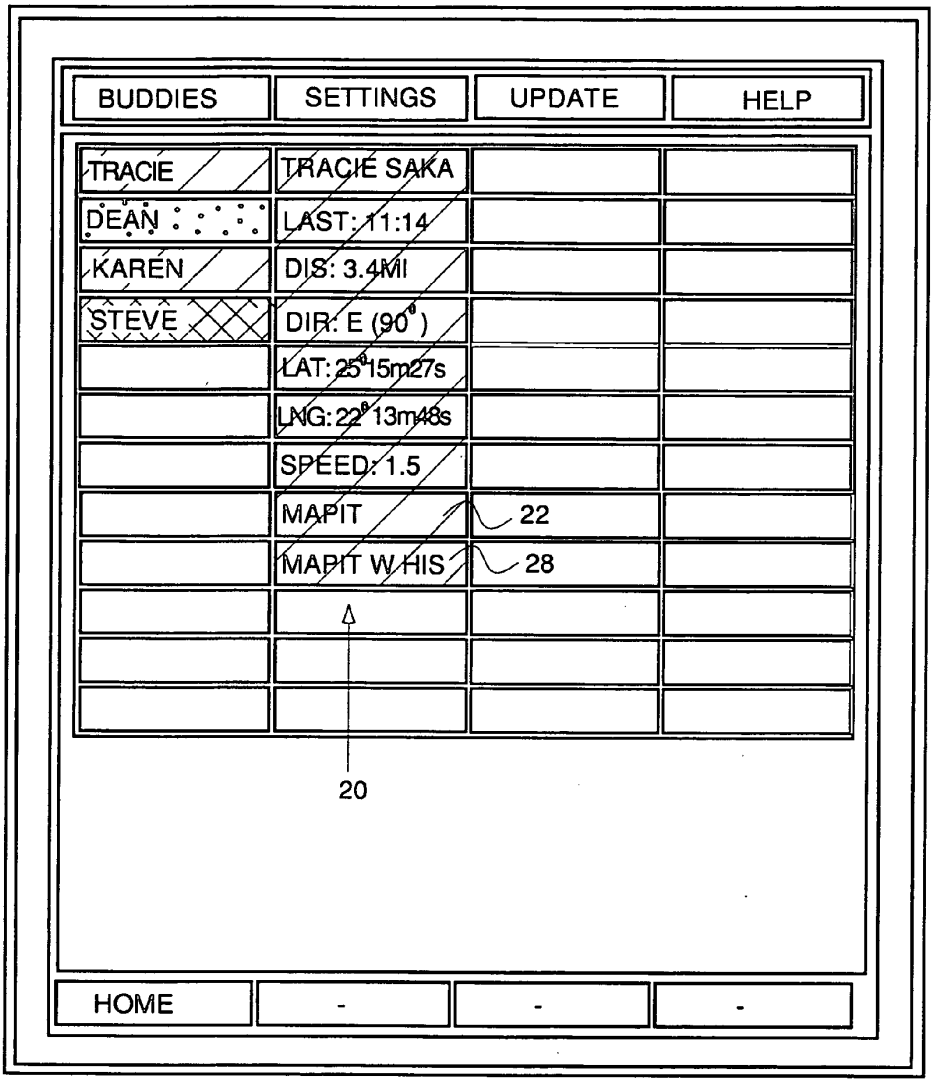


FIG. 2F



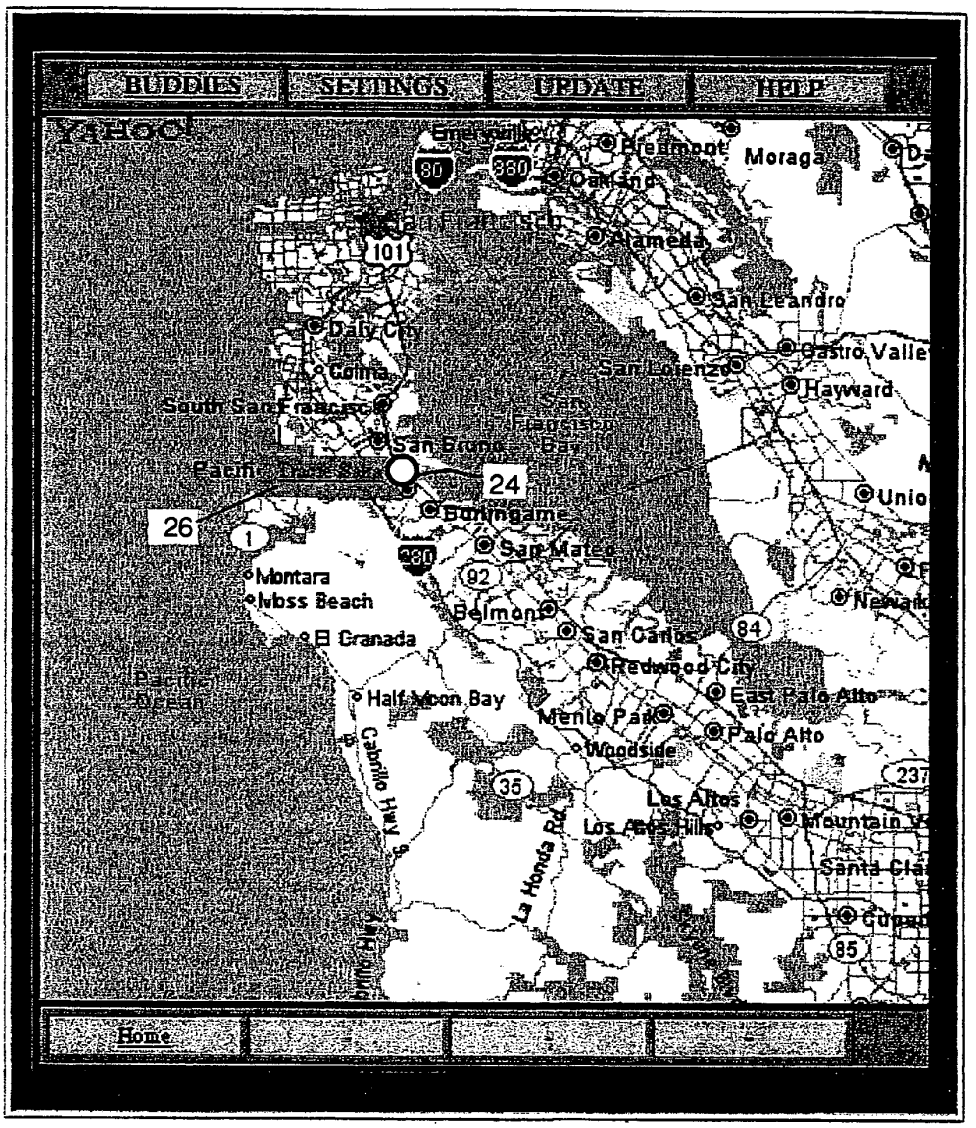
TYPICAL SCREEN SHOWING A NAMED BUDDY LIST'S CONTENTS

FIG. 3



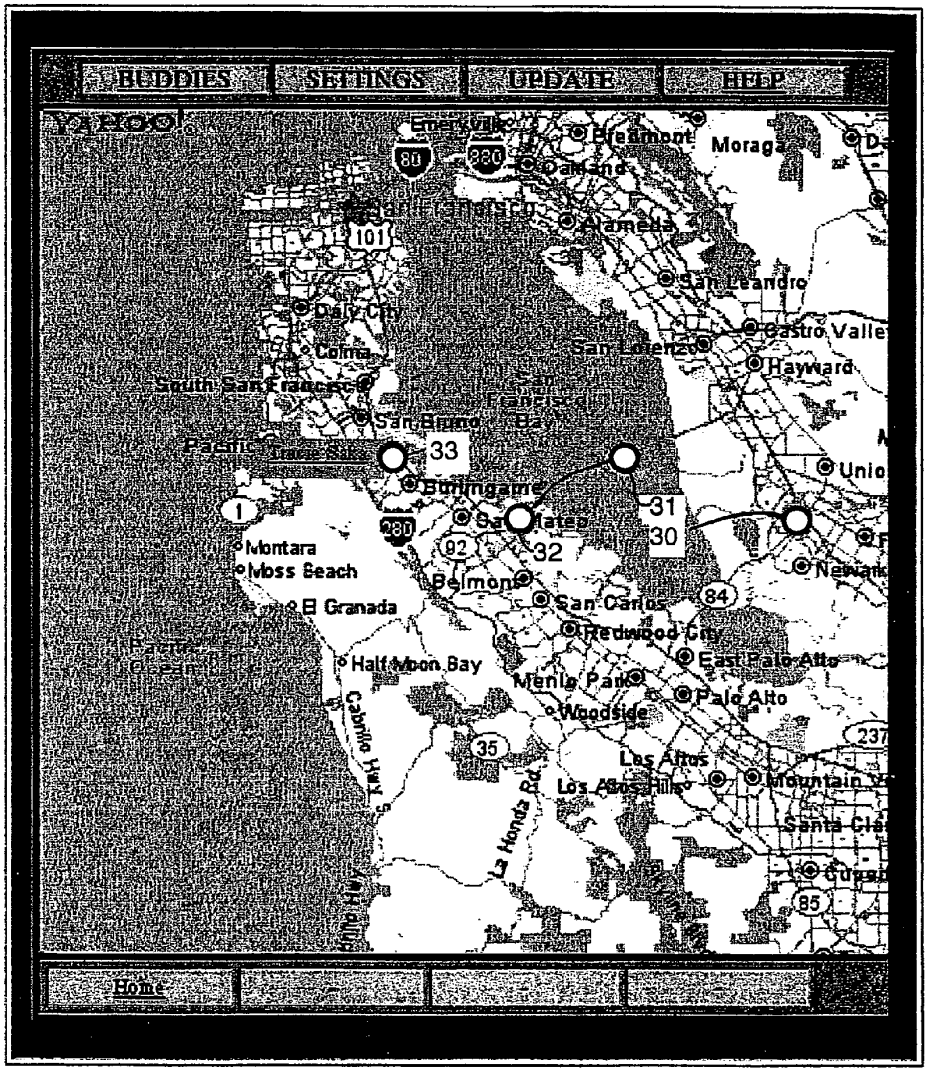
TYPICAL SCREEN SHOWING A BUDDY'S LOCATION ETC.

FIG. 4



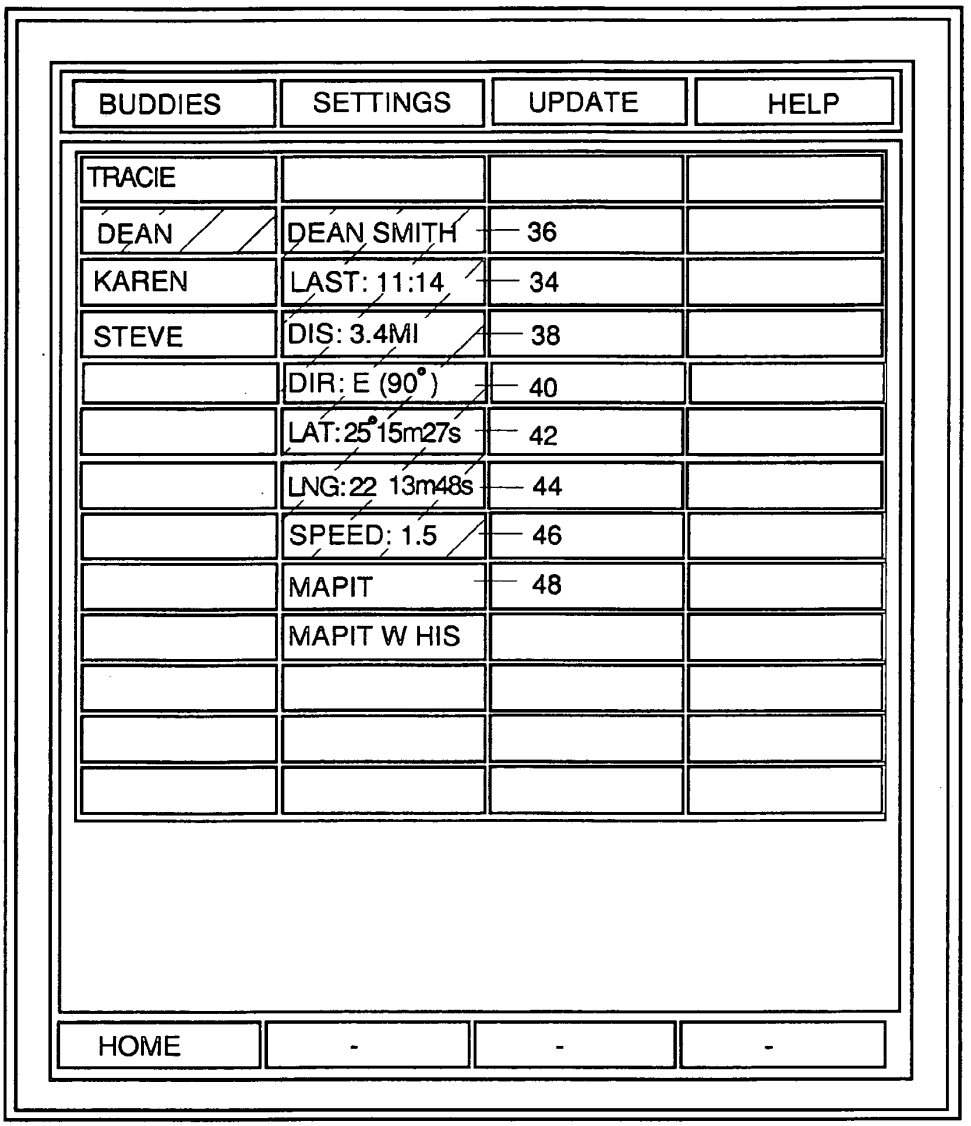
MAPIT™ DISPLAY

FIG. 5



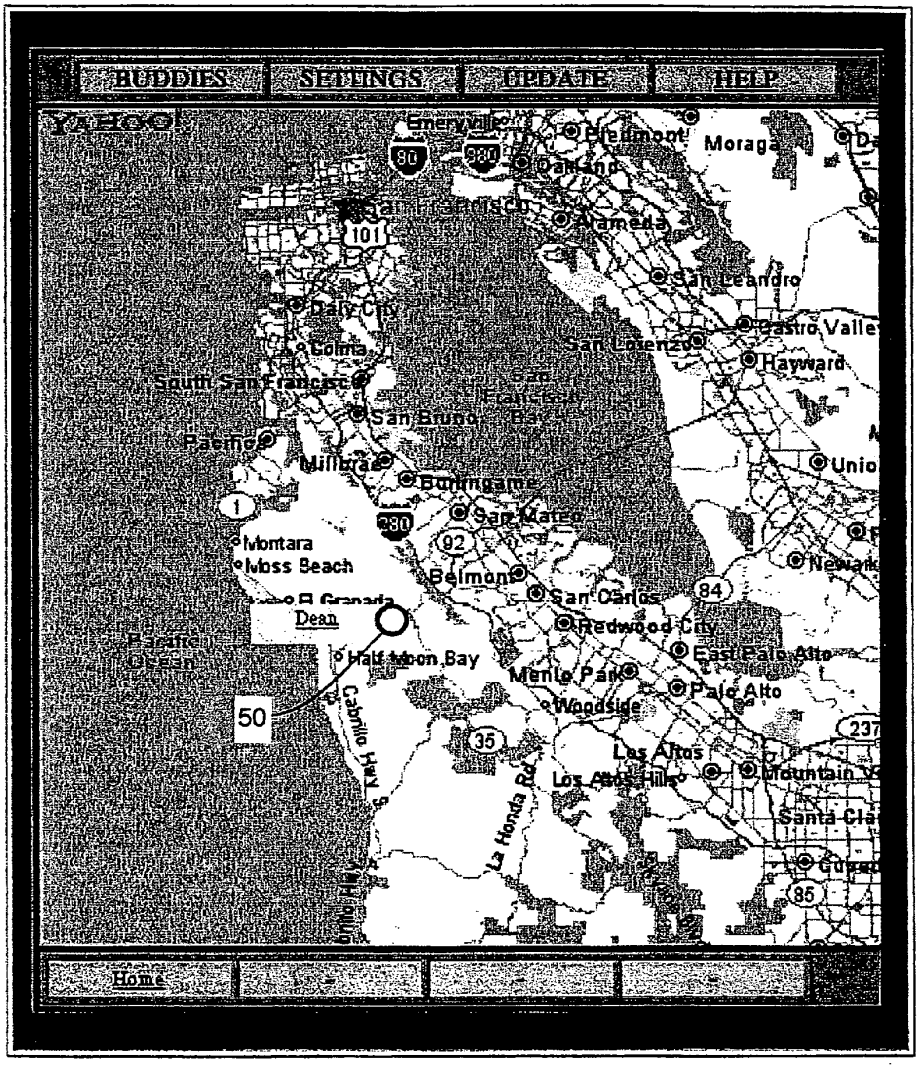
MAPIT DISPLAY SHOWING POSITION HISTORY OF A BUDDY

FIG. 6



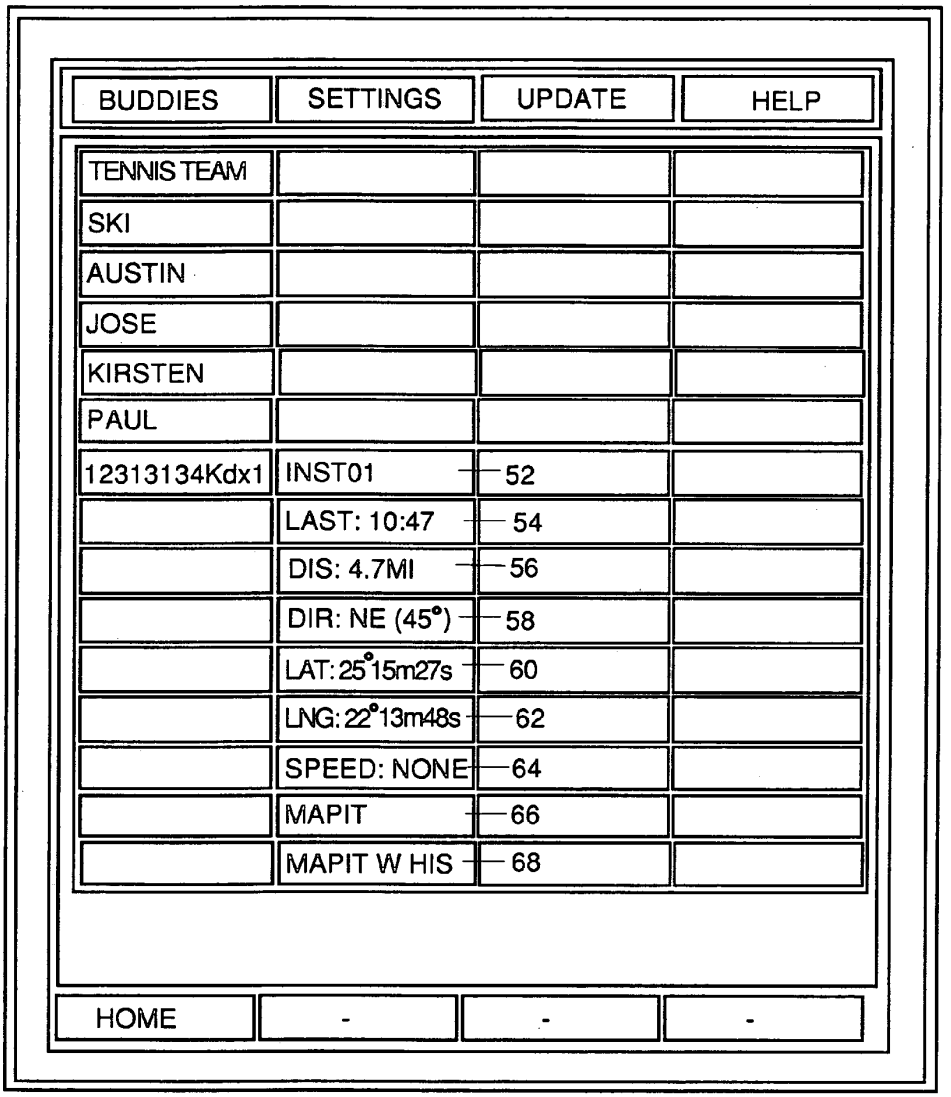
TYPICAL SCREEN SHOWING POSITION AND STATUS OF A MEMBER OF A GROUP

FIG. 7



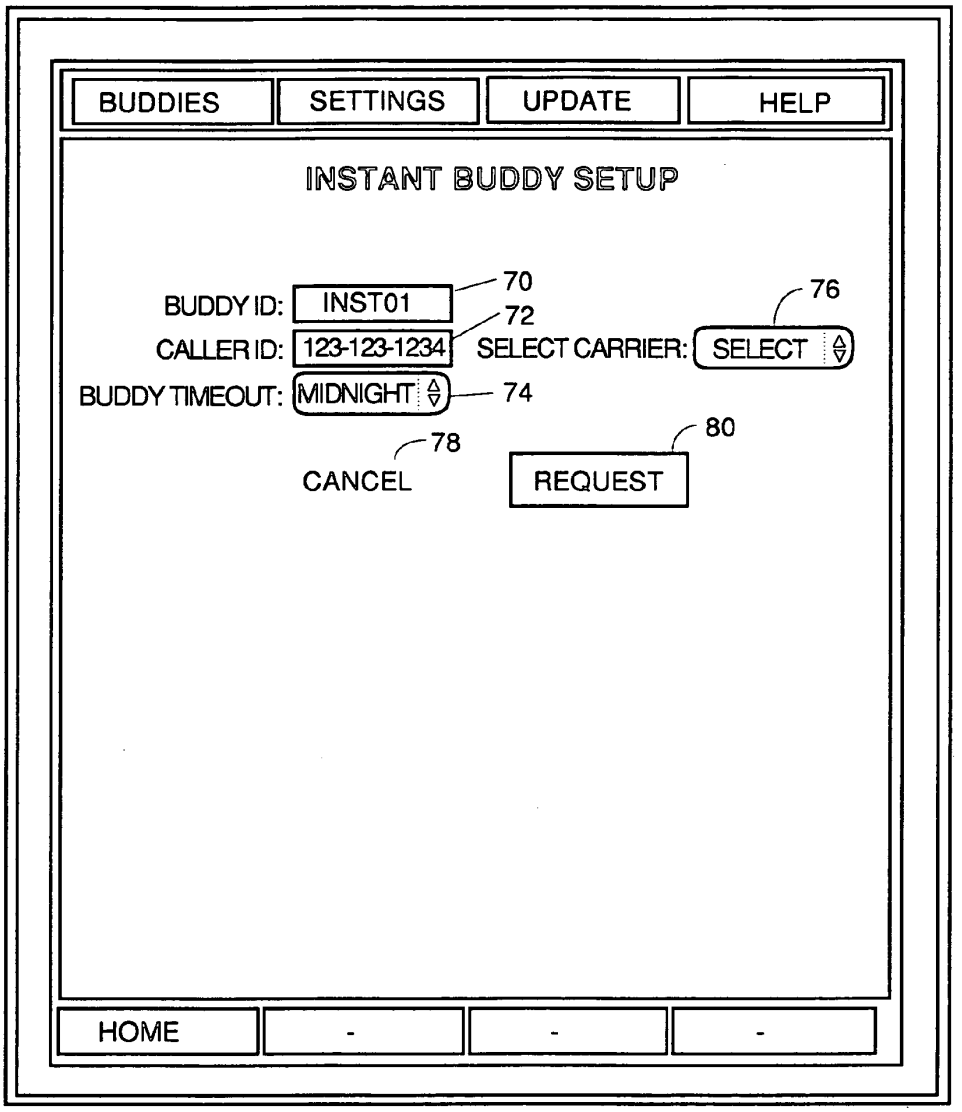
MAPIT DISPLAY WHEN THE POSITION OF A BUDDY IS REQUESTED

FIG. 8



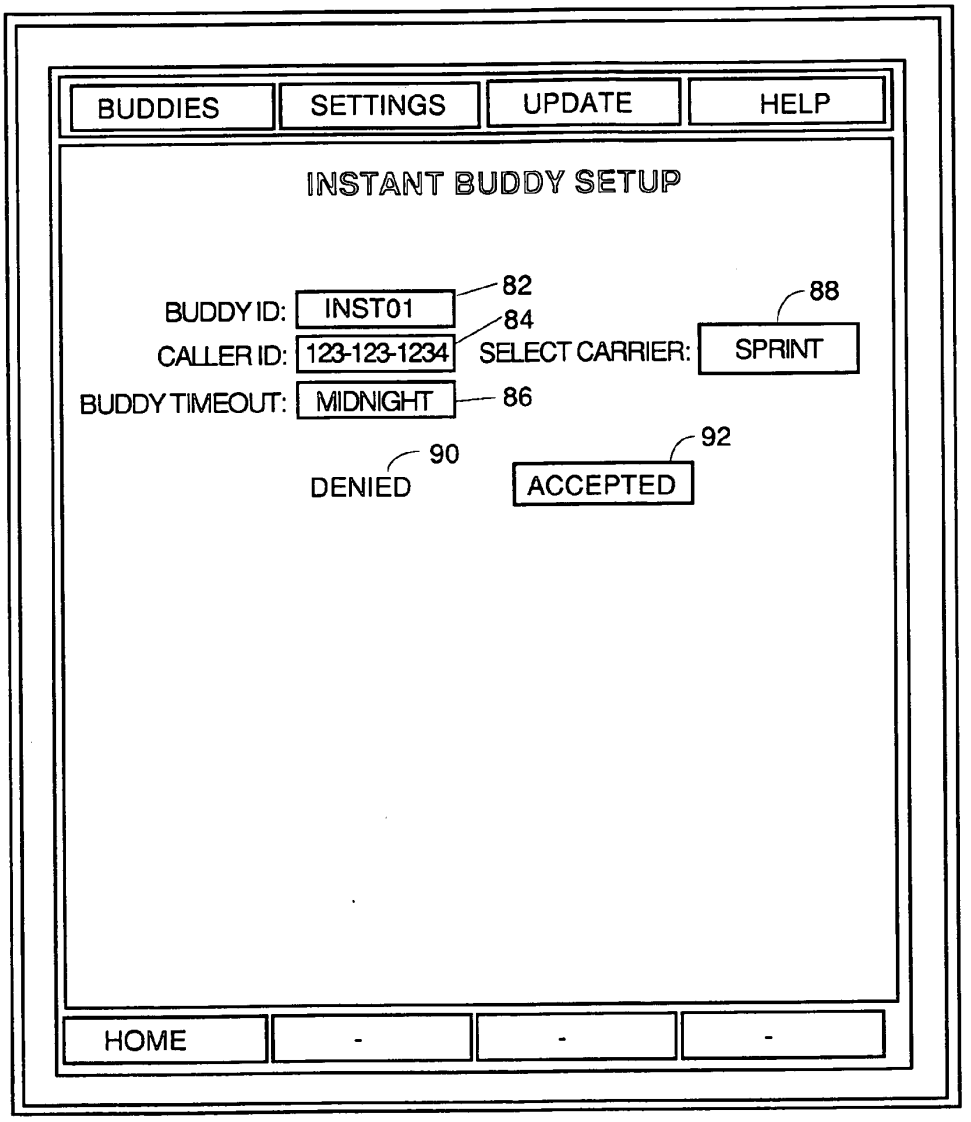
SCREEN SHOT SHOWING AN INSTANT BUDDIES LOCATION

FIG. 9



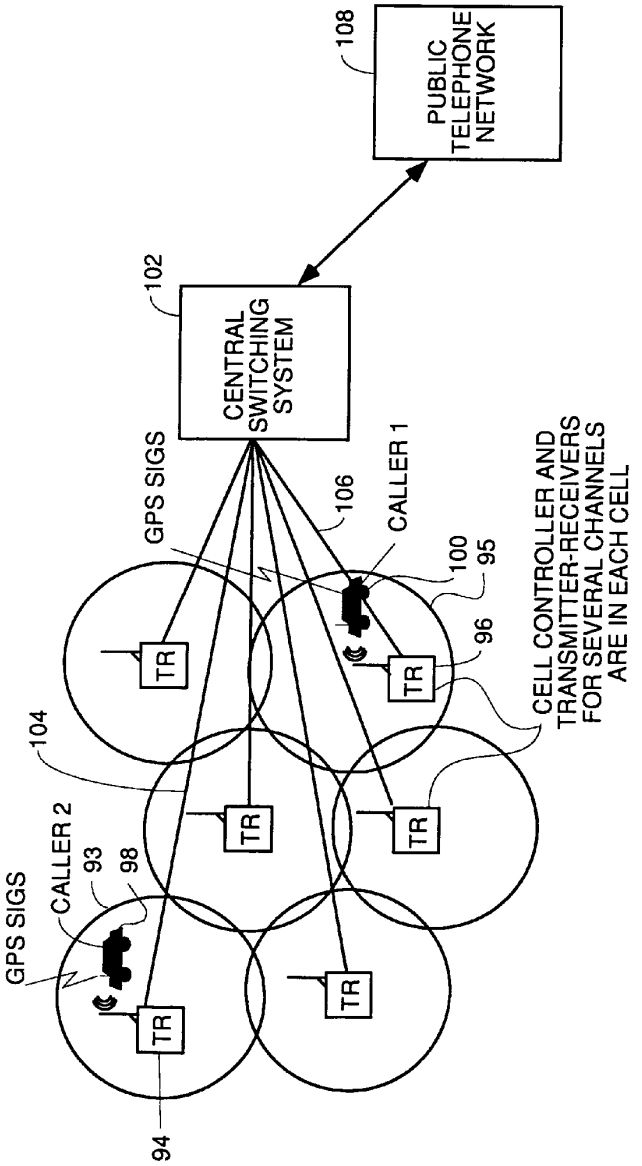
INSTANT BUDDY SETUP SCREEN

FIG. 10



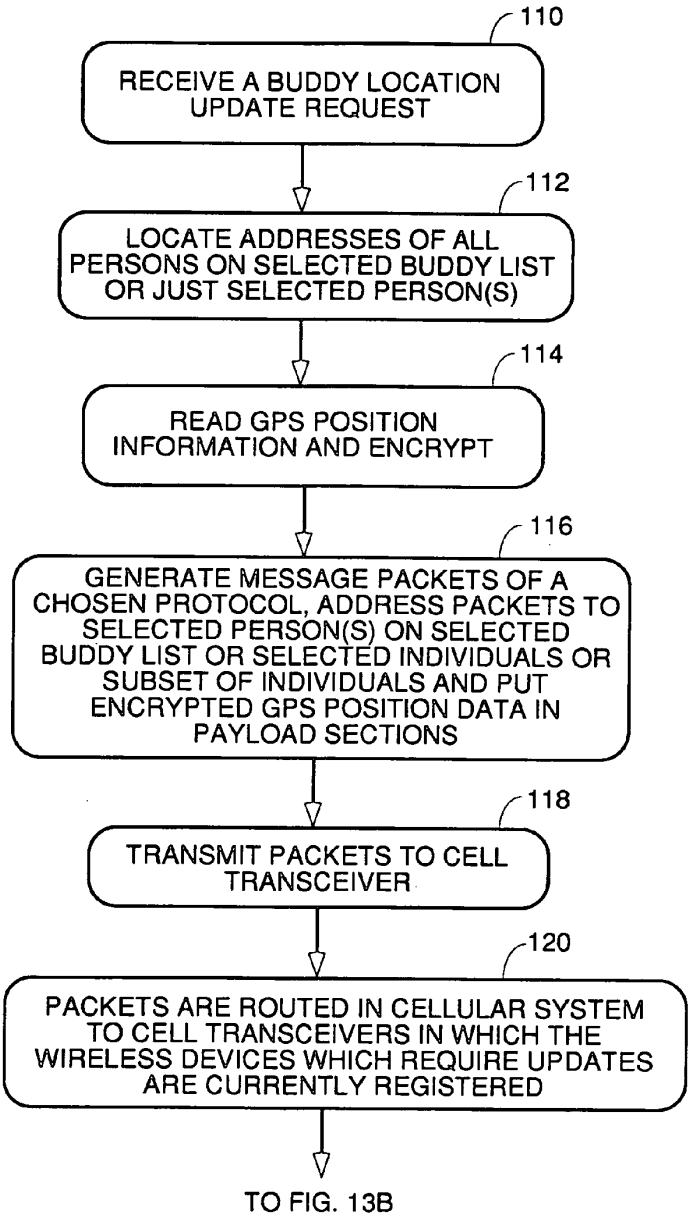
INSTANT BUDDY SETUP SCREEN DISPLAYED ON PHONE OF INSTANT BUDDY

FIG. 11



PRIOR ART CELL PHONE SYSTEM
PEER TO PEER EMBODIMENT

FIG. 12



BUDDY WATCH SERVER &
CELL PHONE PROCESS
TO EXCHANGE POSITION DATA

FIG. 13A

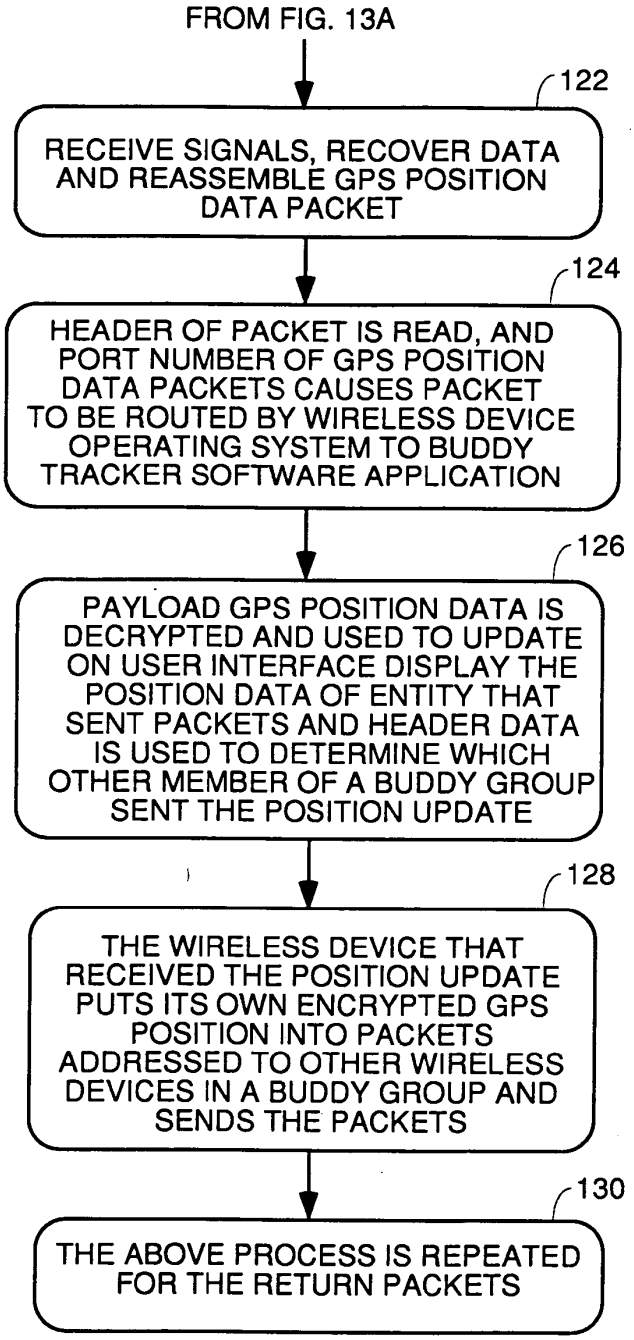
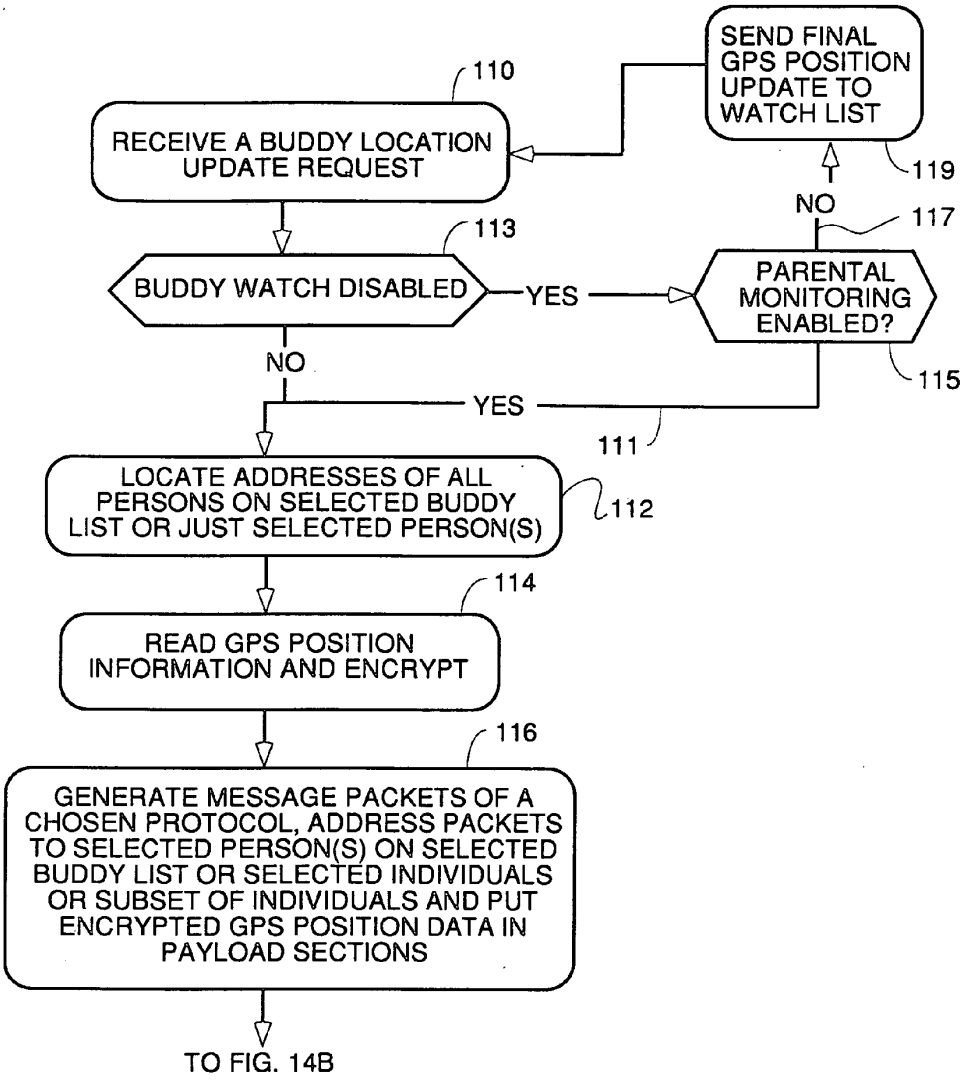

FROM FIG. 13A
122
RECEIVE SIGNALS, RECOVER DATA AND REASSEMBLE GPS POSITION DATA PACKET
124
HEADER OF PACKET IS READ, AND PORT NUMBER OF GPS POSITION DATA PACKETS CAUSES PACKET TO BE ROUTED BY WIRELESS DEVICE OPERATING SYSTEM TO BUDDY TRACKER SOFTWARE APPLICATION
126
PAYLOAD GPS POSITION DATA IS DECRYPTED AND USED TO UPDATE ON USER INTERFACE DISPLAY THE POSITION DATA OF ENTITY THAT SENT PACKETS AND HEADER DATA IS USED TO DETERMINE WHICH OTHER MEMBER OF A BUDDY GROUP SENT THE POSITION UPDATE
128
THE WIRELESS DEVICE THAT RECEIVED THE POSITION UPDATE PUTS ITS OWN ENCRYPTED GPS POSITION INTO PACKETS ADDRESSED TO OTHER WIRELESS DEVICES IN A BUDDY GROUP AND SENDS THE PACKETS
130
THE ABOVE PROCESS IS REPEATED FOR THE RETURN PACKETS

FIG. 13B



ALT. EMB. OF POSITION EXCHANGE PROCESS

FIG. 14A

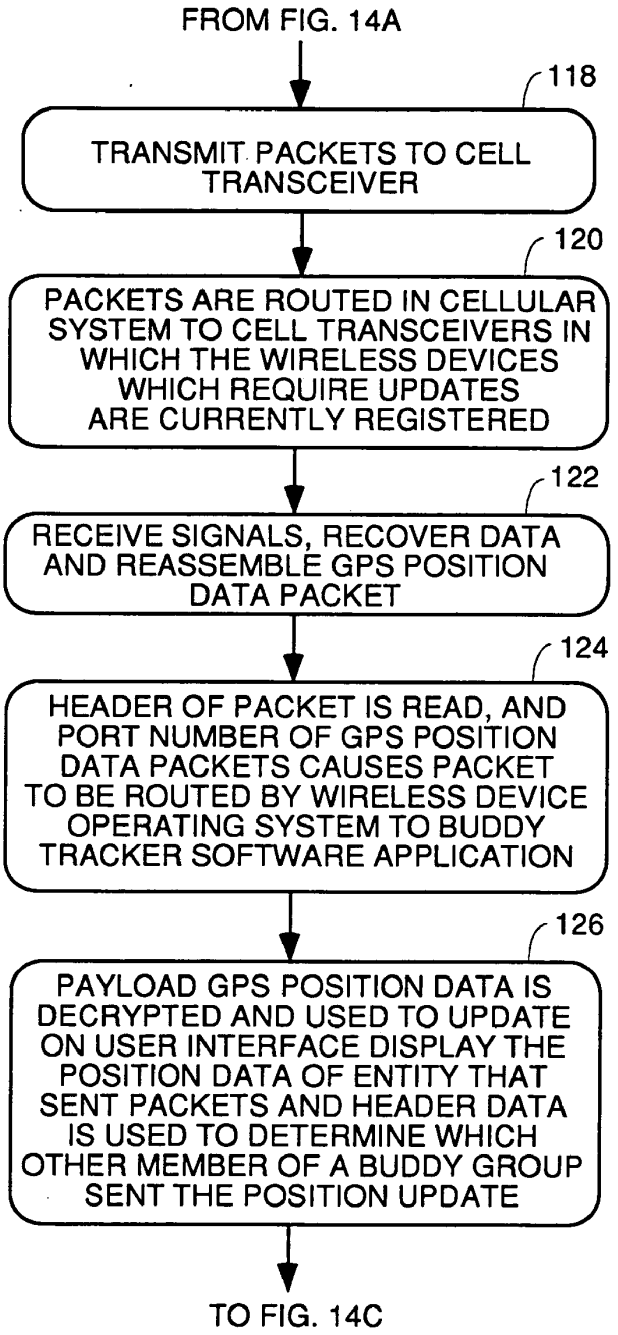

FROM FIG. 14A
118
TRANSMIT PACKETS TO CELL TRANSCEIVER
120
PACKETS ARE ROUTED IN CELLULAR SYSTEM TO CELL TRANSCEIVERS IN WHICH THE WIRELESS DEVICES WHICH REQUIRE UPDATES ARE CURRENTLY REGISTERED
122
RECEIVE SIGNALS, RECOVER DATA AND REASSEMBLE GPS POSITION DATA PACKET
124
HEADER OF PACKET IS READ, AND PORT NUMBER OF GPS POSITION DATA PACKETS CAUSES PACKET TO BE ROUTED BY WIRELESS DEVICE OPERATING SYSTEM TO BUDDY TRACKER SOFTWARE APPLICATION
126
PAYLOAD GPS POSITION DATA IS DECRYPTED AND USED TO UPDATE ON USER INTERFACE DISPLAY THE POSITION DATA OF ENTITY THAT SENT PACKETS AND HEADER DATA IS USED TO DETERMINE WHICH OTHER MEMBER OF A BUDDY GROUP SENT THE POSITION UPDATE
TO FIG. 14C

FIG. 14B

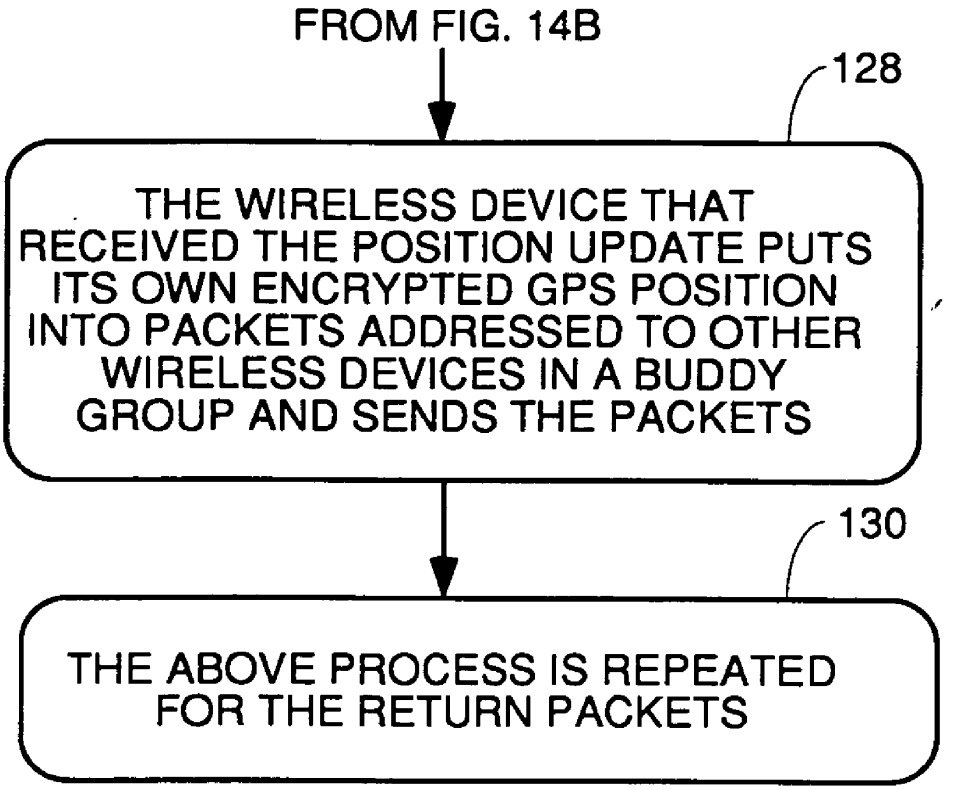

FROM FIG. 14B
128
THE WIRELESS DEVICE THAT RECEIVED THE POSITION UPDATE PUTS ITS OWN ENCRYPTED GPS POSITION INTO PACKETS ADDRESSED TO OTHER WIRELESS DEVICES IN A BUDDY GROUP AND SENDS THE PACKETS
130
THE ABOVE PROCESS IS REPEATED FOR THE RETURN PACKETS

FIG. 14C

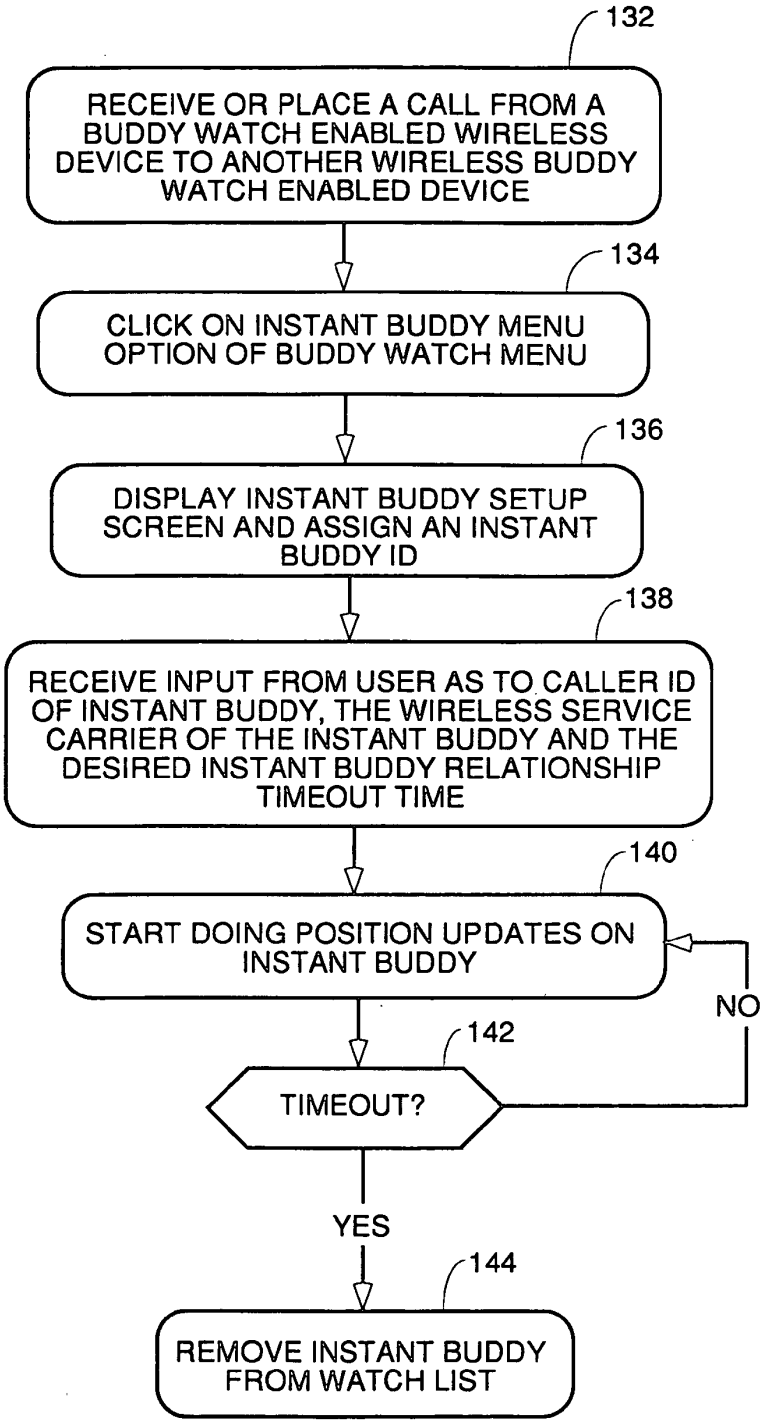

132
RECEIVE OR PLACE A CALL FROM A BUDDY WATCH ENABLED WIRELESS DEVICE TO ANOTHER WIRELESS BUDDY WATCH ENABLED DEVICE
134
CLICK ON INSTANT BUDDY MENU OPTION OF BUDDY WATCH MENU
136
DISPLAY INSTANT BUDDY SETUP SCREEN AND ASSIGN AN INSTANT BUDDY ID
138
RECEIVE INPUT FROM USER AS TO CALLER ID OF INSTANT BUDDY, THE WIRELESS SERVICE CARRIER OF THE INSTANT BUDDY AND THE DESIRED INSTANT BUDDY RELATIONSHIP TIMEOUT TIME
140
START DOING POSITION UPDATES ON INSTANT BUDDY
NO
142
TIMEOUT?
YES
144
REMOVE INSTANT BUDDY FROM WATCH LIST

FIG. 15

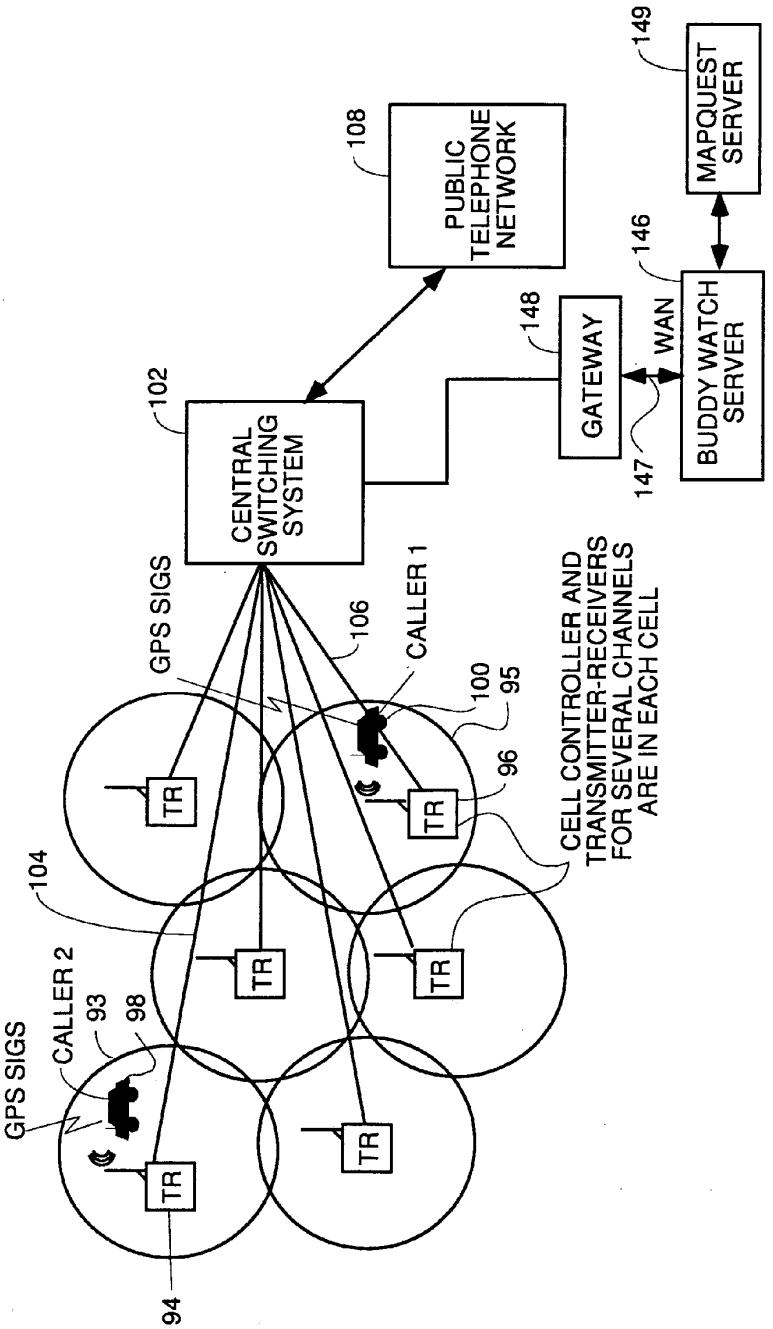


FIG. 16

SERVER-BASED BUDDY WATCH SYSTEM

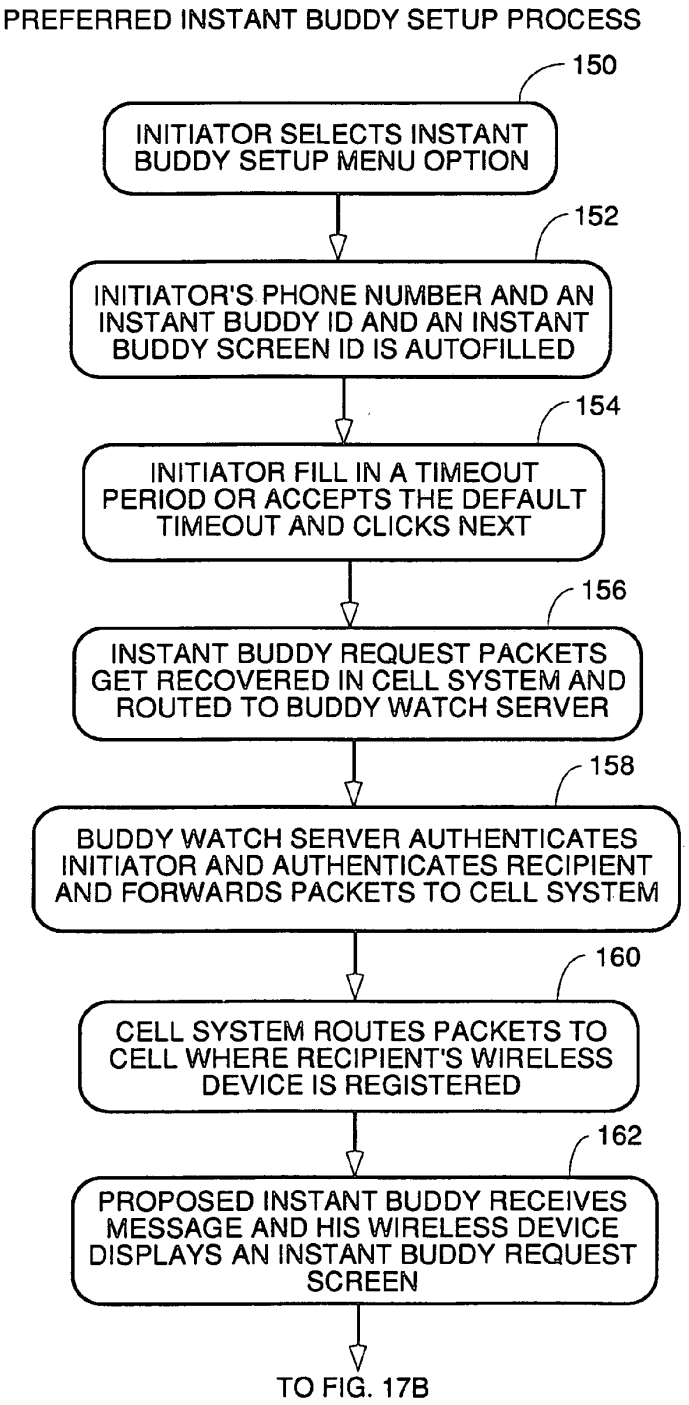

PREFERRED INSTANT BUDDY SETUP PROCESS
150
INITIATOR SELECTS INSTANT BUDDY SETUP MENU OPTION
152
INITIATOR'S PHONE NUMBER AND AN INSTANT BUDDY ID AND AN INSTANT BUDDY SCREEN ID IS AUTOFILLED
154
INITIATOR FILL IN A TIMEOUT PERIOD OR ACCEPTS THE DEFAULT TIMEOUT AND CLICKS NEXT
156
INSTANT BUDDY REQUEST PACKETS GET RECOVERED IN CELL SYSTEM AND ROUTED TO BUDDY WATCH SERVER
158
BUDDY WATCH SERVER AUTHENTICATES INITIATOR AND AUTHENTICATES RECIPIENT AND FORWARDS PACKETS TO CELL SYSTEM
160
CELL SYSTEM ROUTES PACKETS TO CELL WHERE RECIPIENT'S WIRELESS DEVICE IS REGISTERED
162
PROPOSED INSTANT BUDDY RECEIVES MESSAGE AND HIS WIRELESS DEVICE DISPLAYS AN INSTANT BUDDY REQUEST SCREEN
TO FIG. 17B

FIG. 17A

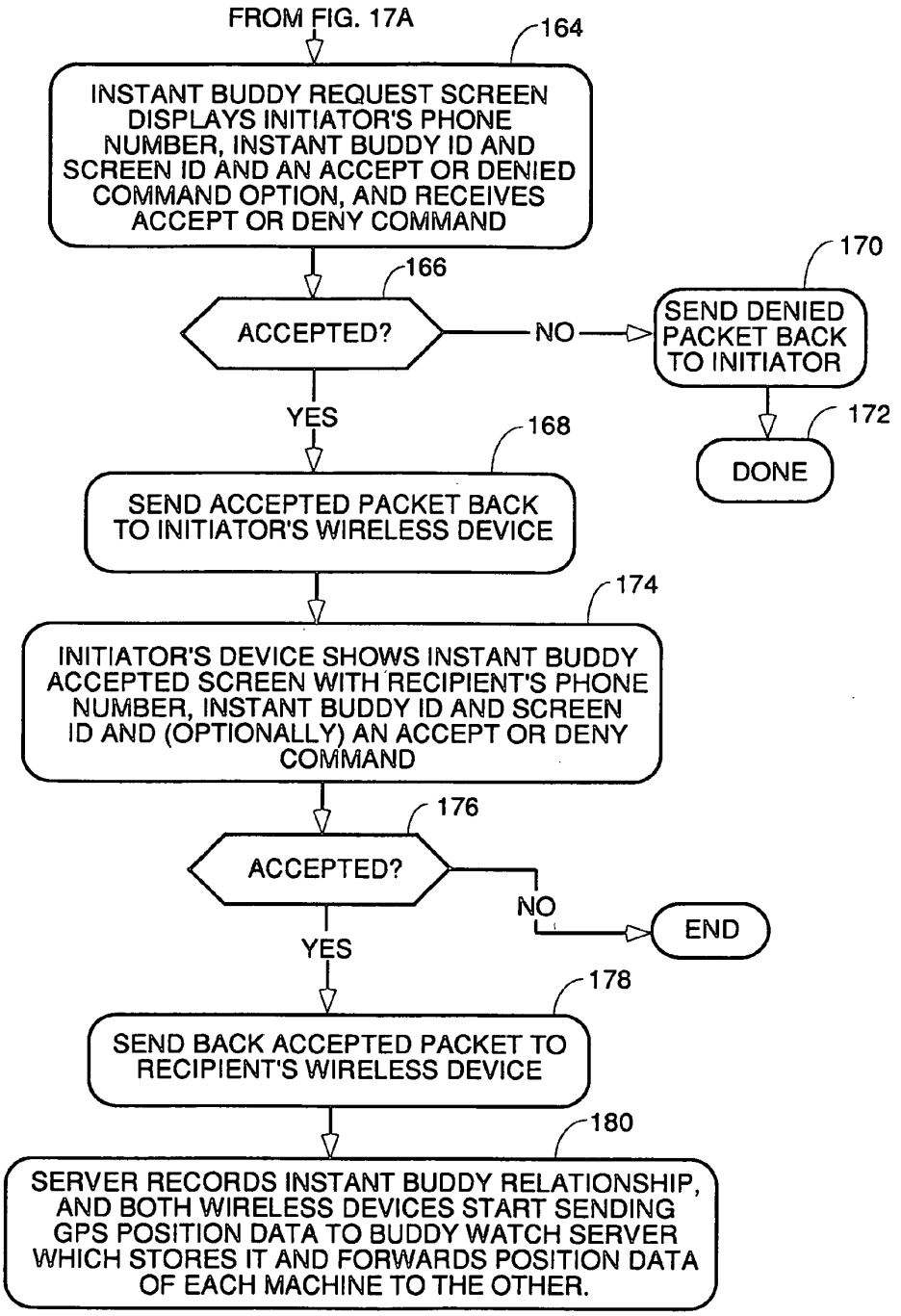

FROM FIG. 17A
164
INSTANT BUDDY REQUEST SCREEN DISPLAYS INITIATOR'S PHONE NUMBER, INSTANT BUDDY ID AND SCREEN ID AND AN ACCEPT OR DENIED COMMAND OPTION, AND RECEIVES ACCEPT OR DENY COMMAND
166
ACCEPTED?
NO
170
SEND DENIED PACKET BACK TO INITIATOR
172
DONE
YES
168
SEND ACCEPTED PACKET BACK TO INITIATOR'S WIRELESS DEVICE
174
INITIATOR'S DEVICE SHOWS INSTANT BUDDY ACCEPTED SCREEN WITH RECIPIENT'S PHONE NUMBER, INSTANT BUDDY ID AND SCREEN ID AND (OPTIONALLY) AN ACCEPT OR DENY COMMAND
176
ACCEPTED?
NO
END
YES
178
SEND BACK ACCEPTED PACKET TO RECIPIENT'S WIRELESS DEVICE
180
SERVER RECORDS INSTANT BUDDY RELATIONSHIP, AND BOTH WIRELESS DEVICES START SENDING GPS POSITION DATA TO BUDDY WATCH SERVER WHICH STORES IT AND FORWARDS POSITION DATA OF EACH MACHINE TO THE OTHER.

FIG. 17B

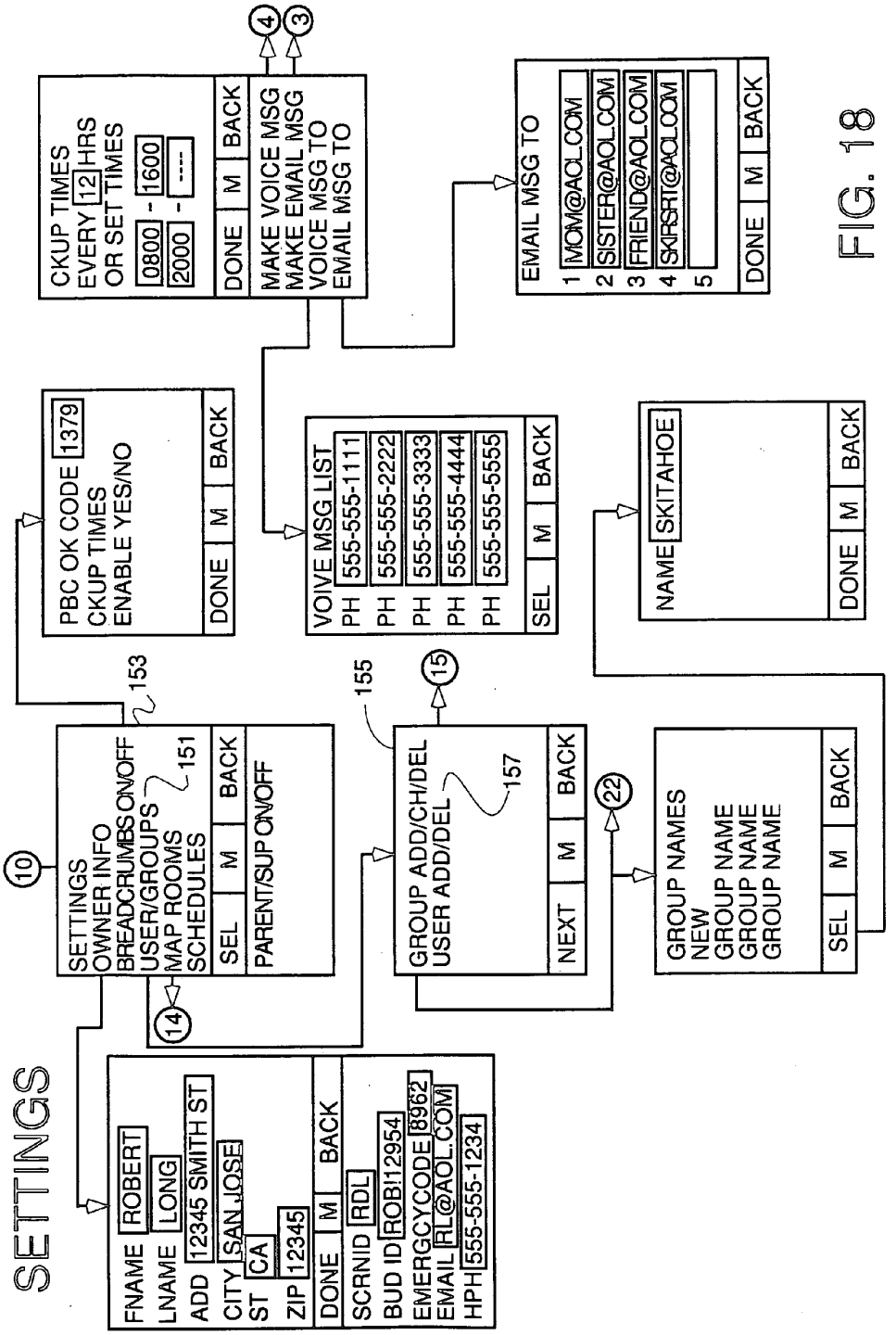

SETTINGS
FNAME ROBERT
LNAME LONG
ADD 12345 SMITH ST
CITY SAN JOSE
ST CA
ZIP 12345
DONE M BACK
SCRNID RDL
BUD ID ROB112954
EMERGCYCODE 8962
EMAIL RL@AOL.COM
HPH 555-555-1234
SETTINGS
OWNER INFO
BREADCRUMBS ON/OFF
USER/GROUPS
MAP ROOMS
SCHEDULES
SEL M BACK
PARENT/SUP ON/OFF
151
153
155
GROUP ADD/CH/DEL
USER ADD/DEL
157
NEXT M BACK
GROUP NAMES
NEW
GROUP NAME
GROUP NAME
GROUP NAME
SEL M BACK
NAME SKITAHOE
DONE M BACK
PBC OK CODE 1379
CKUP TIMES
ENABLE YES/NO
DONE M BACK
VOIVE MSG LIST
PH 555-555-1111
PH 555-555-2222
PH 555-555-3333
PH 555-555-4444
PH 555-555-5555
SEL M BACK
CKUP TIMES
EVERY 12 HRS
OR SET TIMES
0800 - 1600
2000 - ----
DONE M BACK
MAKE VOICE MSG
MAKE EMAIL MSG
VOICE MSG TO
EMAIL MSG TO
EMAIL MSG TO
1 MOM@AOL.COM
2 SISTER@AOL.COM
3 FRIEND@AOL.COM
4 SKIRSRT@AOL.COM
5
DONE M BACK
10
14
15
22
3
4

FIG. 18

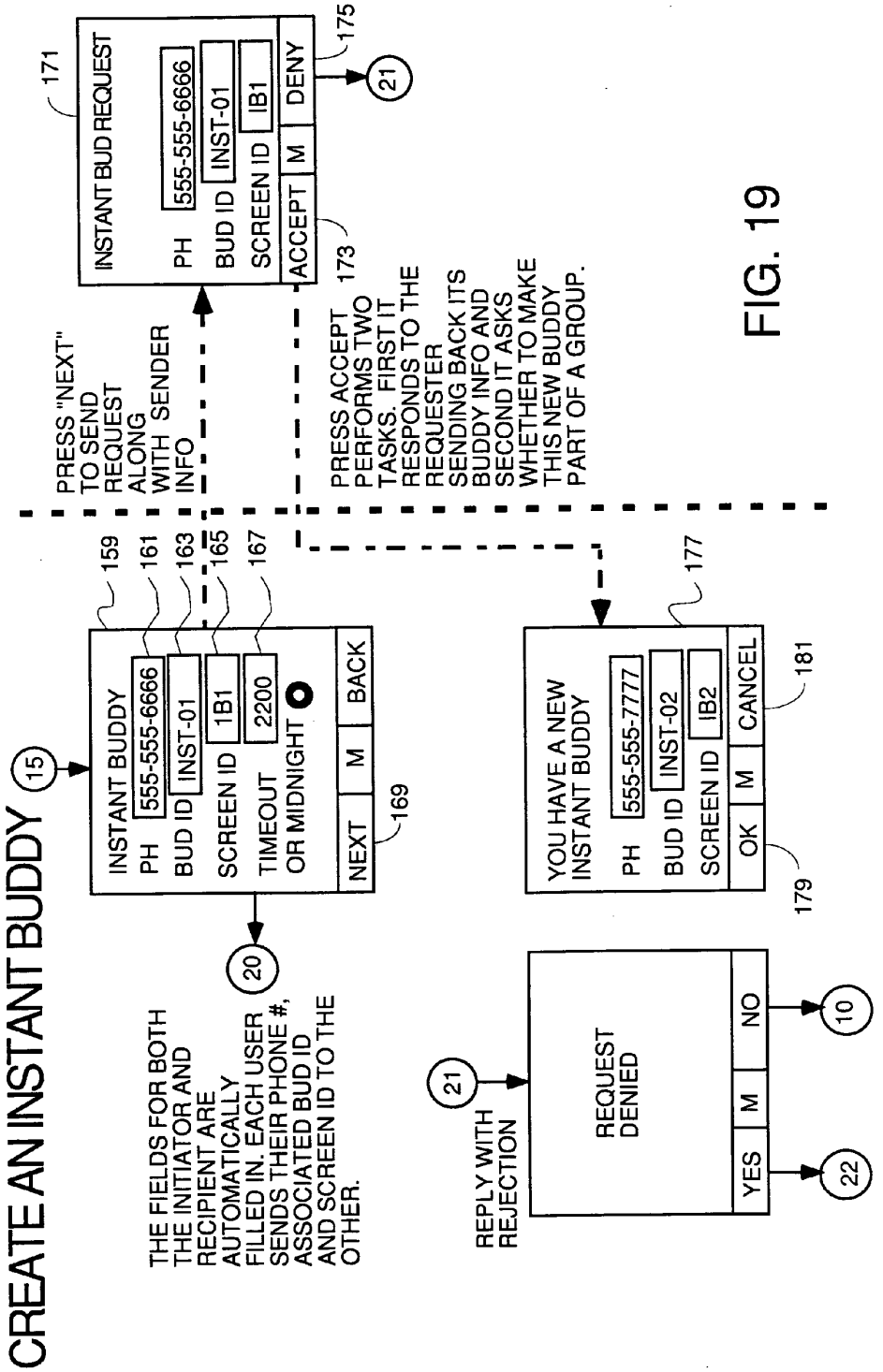

CREATE AN INSTANT BUDDY
15
INSTANT BUDDY
PH 555-555-6666
BUD ID INST-01
SCREEN ID 1B1
TIMEOUT 2200
OR MIDNIGHT
NEXT M BACK
159
161
163
165
167
169
PRESS "NEXT" TO SEND REQUEST ALONG WITH SENDER INFO
INSTANT BUD REQUEST
PH 555-555-6666
BUD ID INST-01
SCREEN ID IB1
ACCEPT M DENY
171
173
175
21
PRESS ACCEPT PERFORMS TWO TASKS. FIRST IT RESPONDS TO THE REQUESTER SENDING BACK ITS BUDDY INFO AND SECOND IT ASKS WHETHER TO MAKE THIS NEW BUDDY PART OF A GROUP.
YOU HAVE A NEW INSTANT BUDDY
PH 555-555-7777
BUD ID INST-02
SCREEN ID IB2
OK M CANCEL
177
179
181
THE FIELDS FOR BOTH THE INITIATOR AND RECIPIENT ARE AUTOMATICALLY FILLED IN. EACH USER SENDS THEIR PHONE #, ASSOCIATED BUD ID AND SCREEN ID TO THE OTHER.
20
REPLY WITH REJECTION
21
REQUEST DENIED
YES M NO
22
10

FIG. 19

PROCESS TO ESTABLISH AND USE PERSONAL BREAD CRUMBS MODE

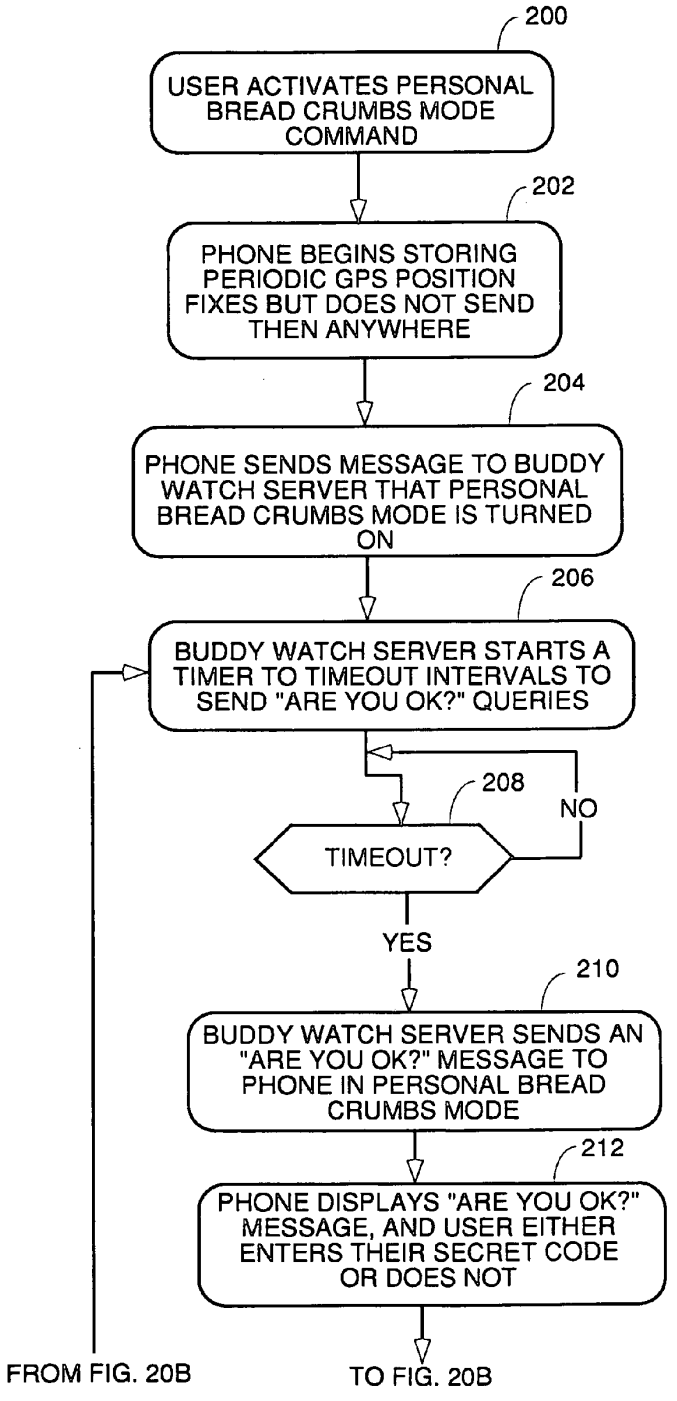


FIG. 20A

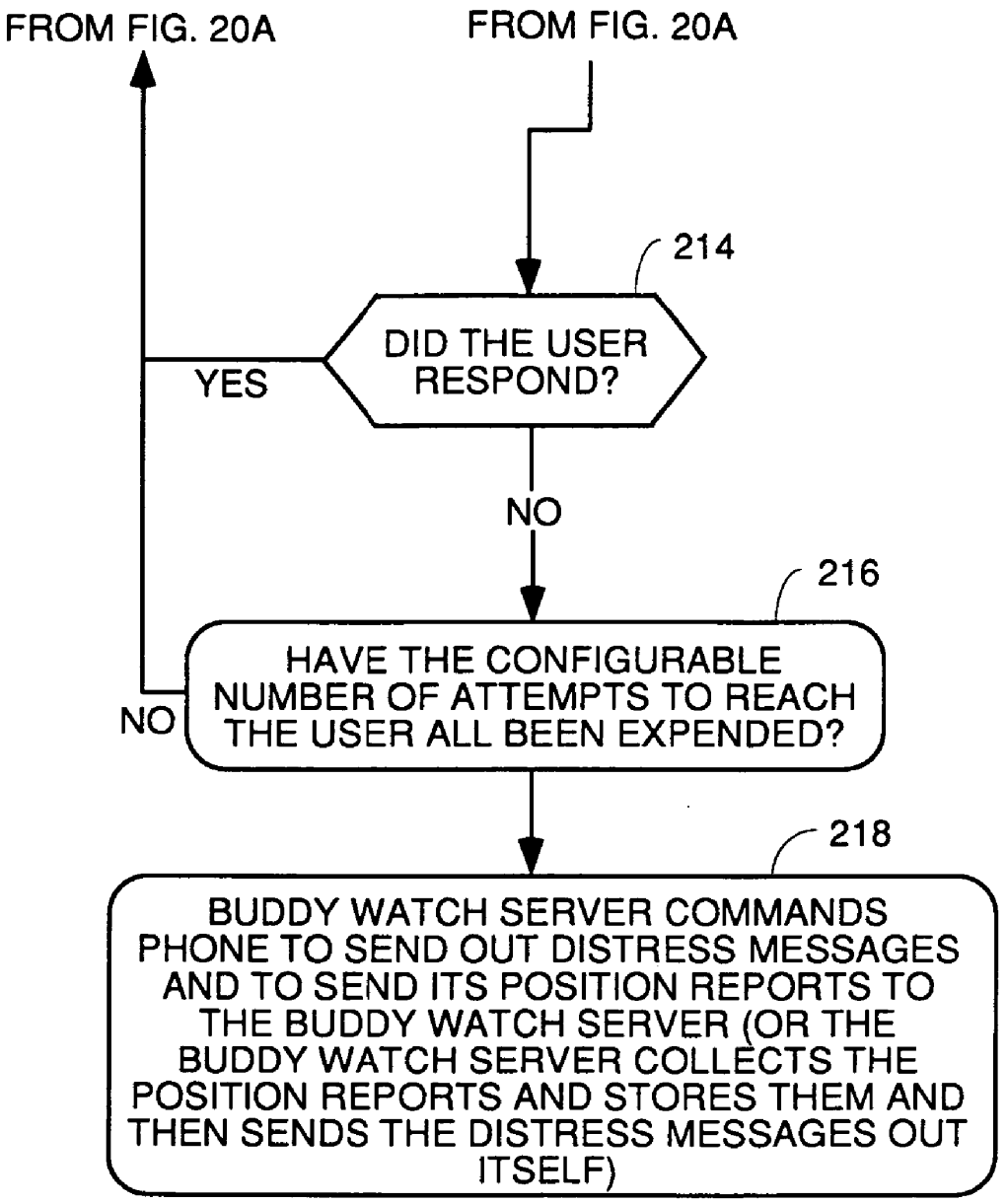

FROM FIG. 20A
FROM FIG. 20A
214
DID THE USER RESPOND?
YES
NO
216
HAVE THE CONFIGURABLE NUMBER OF ATTEMPTS TO REACH THE USER ALL BEEN EXPENDED?
NO
218
BUDDY WATCH SERVER COMMANDS PHONE TO SEND OUT DISTRESS MESSAGES AND TO SEND ITS POSITION REPORTS TO THE BUDDY WATCH SERVER (OR THE BUDDY WATCH SERVER COLLECTS THE POSITION REPORTS AND STORES THEM AND THEN SENDS THE DISTRESS MESSAGES OUT ITSELF)

FIG. 20B

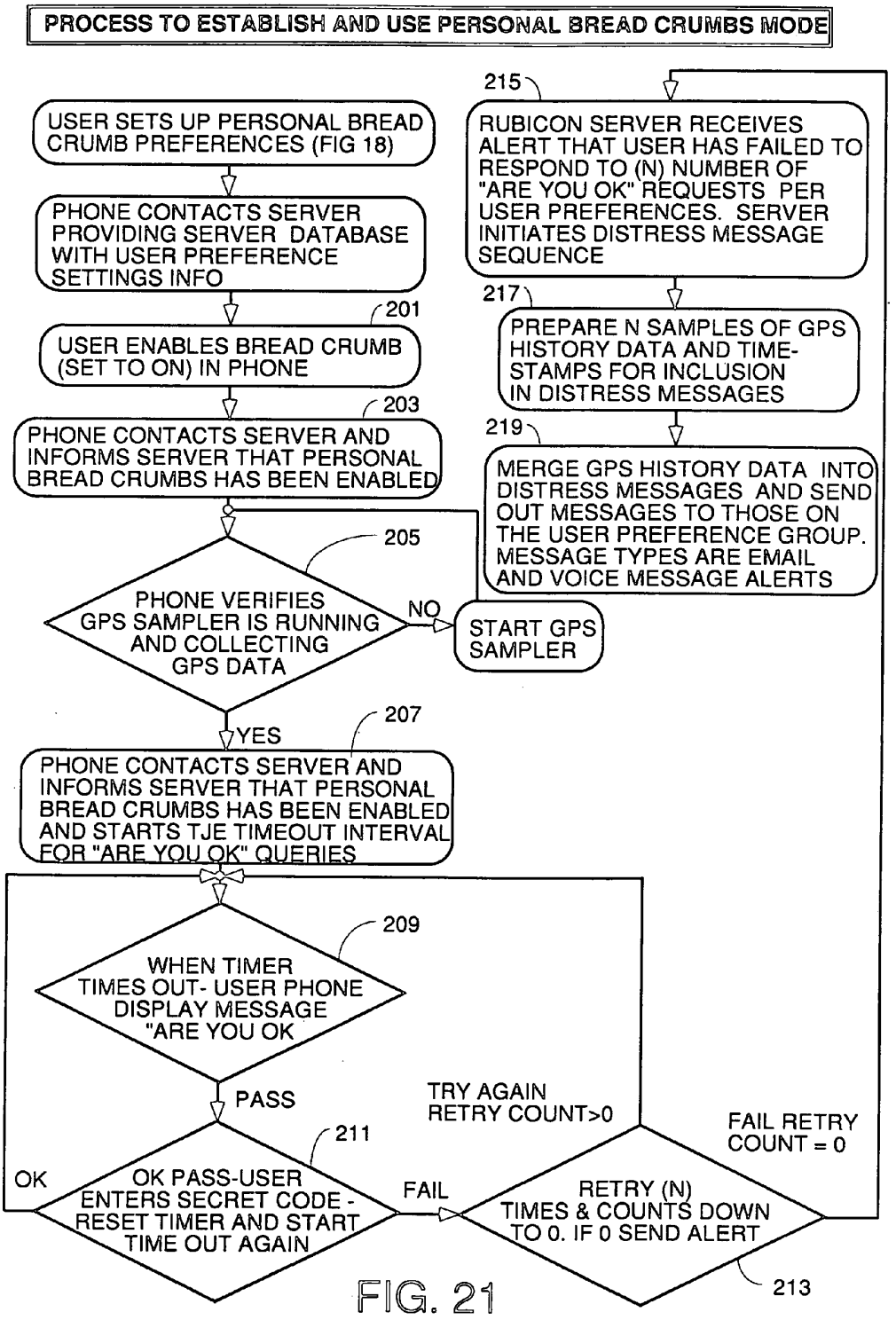

PROCESS TO ESTABLISH AND USE PERSONAL BREAD CRUMBS MODE
USER SETS UP PERSONAL BREAD CRUMB PREFERENCES (FIG 18)
PHONE CONTACTS SERVER PROVIDING SERVER DATABASE WITH USER PREFERENCE SETTINGS INFO
201
USER ENABLES BREAD CRUMB (SET TO ON) IN PHONE
203
PHONE CONTACTS SERVER AND INFORMS SERVER THAT PERSONAL BREAD CRUMBS HAS BEEN ENABLED
205
PHONE VERIFIES GPS SAMPLER IS RUNNING AND COLLECTING GPS DATA
NO
START GPS SAMPLER
YES
207
PHONE CONTACTS SERVER AND INFORMS SERVER THAT PERSONAL BREAD CRUMBS HAS BEEN ENABLED AND STARTS TJE TIMEOUT INTERVAL FOR "ARE YOU OK" QUERIES
209
WHEN TIMER TIMES OUT- USER PHONE DISPLAY MESSAGE "ARE YOU OK
PASS
211
OK
OK PASS-USER ENTERS SECRET CODE - RESET TIMER AND START TIME OUT AGAIN
FAIL
TRY AGAIN RETRY COUNT>0
RETRY (N) TIMES & COUNTS DOWN TO 0. IF 0 SEND ALERT
213
FAIL RETRY COUNT = 0
215
RUBICON SERVER RECEIVES ALERT THAT USER HAS FAILED TO RESPOND TO (N) NUMBER OF "ARE YOU OK" REQUESTS PER USER PREFERENCES. SERVER INITIATES DISTRESS MESSAGE SEQUENCE
217
PREPARE N SAMPLES OF GPS HISTORY DATA AND TIME-STAMPS FOR INCLUSION IN DISTRESS MESSAGES
219
MERGE GPS HISTORY DATA INTO DISTRESS MESSAGES AND SEND OUT MESSAGES TO THOSE ON THE USER PREFERENCE GROUP. MESSAGE TYPES ARE EMAIL AND VOICE MESSAGE ALERTS

FIG. 21

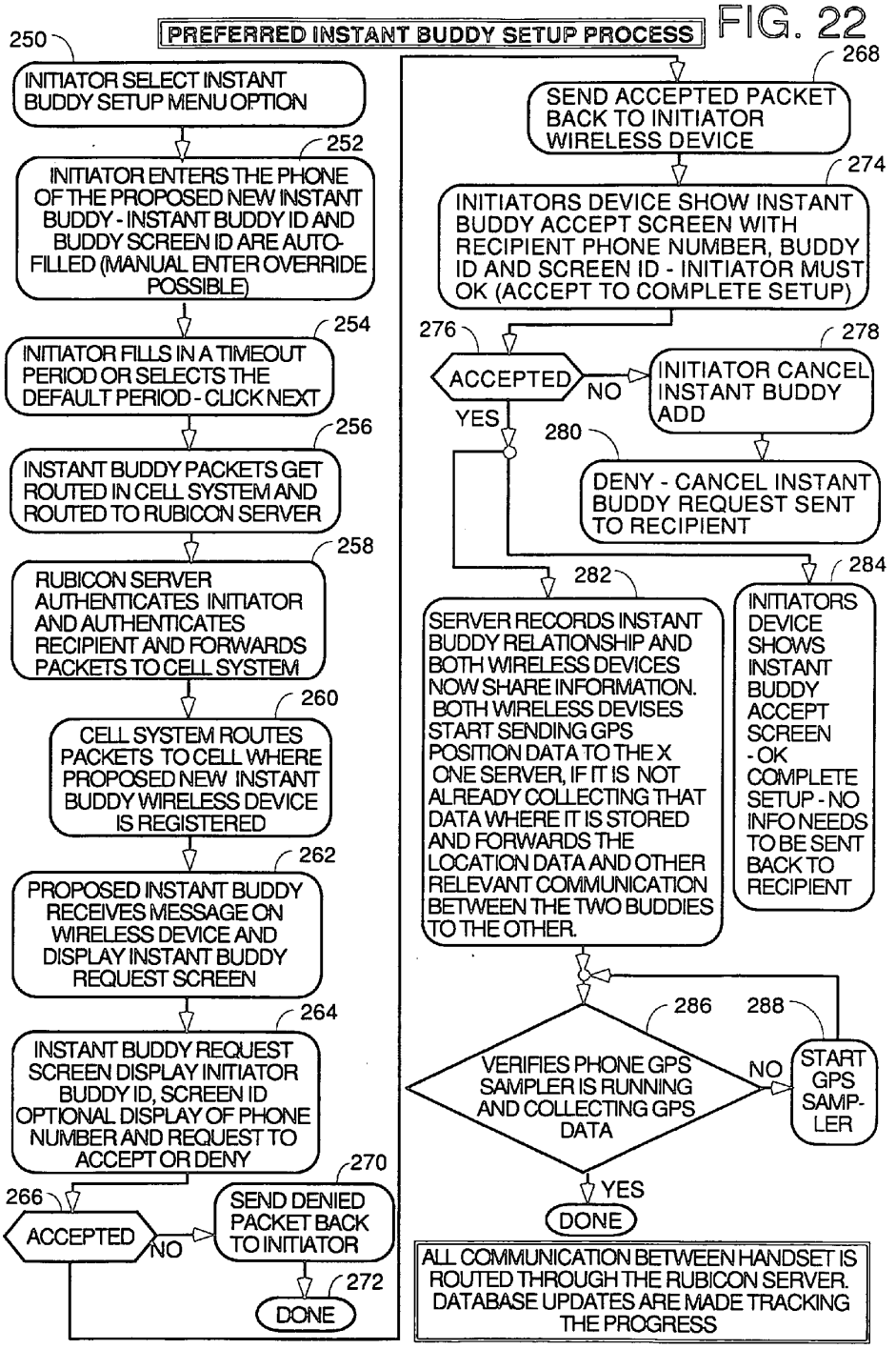

PREFERRED INSTANT BUDDY SETUP PROCESS
250
INITIATOR SELECT INSTANT BUDDY SETUP MENU OPTION
252
INITIATOR ENTERS THE PHONE OF THE PROPOSED NEW INSTANT BUDDY - INSTANT BUDDY ID AND BUDDY SCREEN ID ARE AUTO-FILLED (MANUAL ENTER OVERRIDE POSSIBLE)
254
INITIATOR FILLS IN A TIMEOUT PERIOD OR SELECTS THE DEFAULT PERIOD - CLICK NEXT
256
INSTANT BUDDY PACKETS GET ROUTED IN CELL SYSTEM AND ROUTED TO RUBICON SERVER
258
RUBICON SERVER AUTHENTICATES INITIATOR AND AUTHENTICATES RECIPIENT AND FORWARDS PACKETS TO CELL SYSTEM
260
CELL SYSTEM ROUTES PACKETS TO CELL WHERE PROPOSED NEW INSTANT BUDDY WIRELESS DEVICE IS REGISTERED
262
PROPOSED INSTANT BUDDY RECEIVES MESSAGE ON WIRELESS DEVICE AND DISPLAY INSTANT BUDDY REQUEST SCREEN
264
INSTANT BUDDY REQUEST SCREEN DISPLAY INITIATOR BUDDY ID, SCREEN ID OPTIONAL DISPLAY OF PHONE NUMBER AND REQUEST TO ACCEPT OR DENY
266
ACCEPTED
NO
270
SEND DENIED PACKET BACK TO INITIATOR
272
DONE
268
SEND ACCEPTED PACKET BACK TO INITIATOR WIRELESS DEVICE
274
INITIATORS DEVICE SHOW INSTANT BUDDY ACCEPT SCREEN WITH RECIPIENT PHONE NUMBER, BUDDY ID AND SCREEN ID - INITIATOR MUST OK (ACCEPT TO COMPLETE SETUP)
276
ACCEPTED
NO
278
INITIATOR CANCEL INSTANT BUDDY ADD
YES
280
DENY - CANCEL INSTANT BUDDY REQUEST SENT TO RECIPIENT
282
SERVER RECORDS INSTANT BUDDY RELATIONSHIP AND BOTH WIRELESS DEVICES NOW SHARE INFORMATION. BOTH WIRELESS DEVISES START SENDING GPS POSITION DATA TO THE X ONE SERVER, IF IT IS NOT ALREADY COLLECTING THAT DATA WHERE IT IS STORED AND FORWARDS THE LOCATION DATA AND OTHER RELEVANT COMMUNICATION BETWEEN THE TWO BUDDIES TO THE OTHER.
284
INITIATORS DEVICE SHOWS INSTANT BUDDY ACCEPT SCREEN - OK COMPLETE SETUP - NO INFO NEEDS TO BE SENT BACK TO RECIPIENT
286
VERIFIES PHONE GPS SAMPLER IS RUNNING AND COLLECTING GPS DATA
288
NO
START GPS SAMP-LER
YES
DONE
ALL COMMUNICATION BETWEEN HANDSET IS ROUTED THROUGH THE RUBICON SERVER. DATABASE UPDATES ARE MADE TRACKING THE PROGRESS

FIG. 22

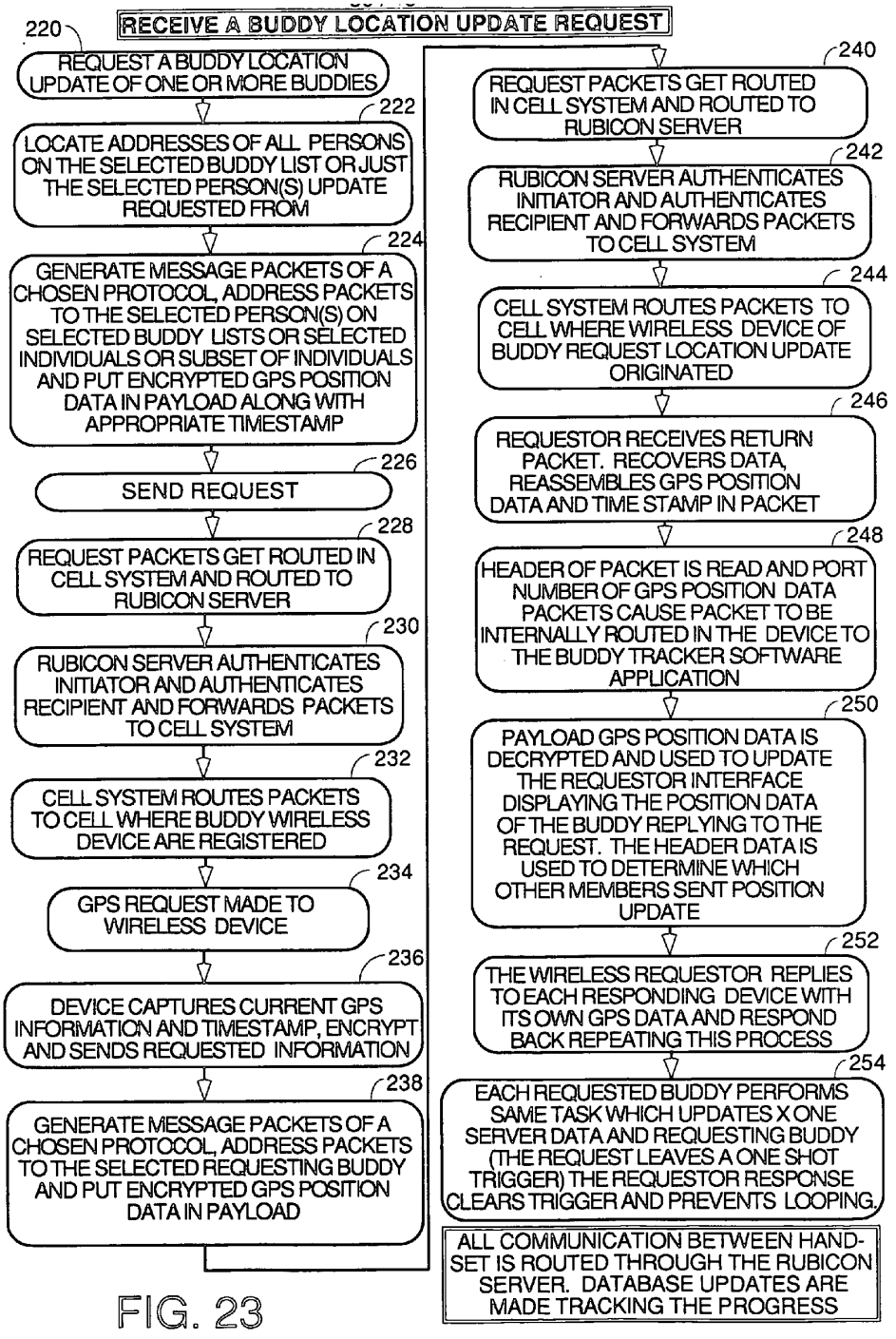

RECEIVE A BUDDY LOCATION UPDATE REQUEST
220
REQUEST A BUDDY LOCATION UPDATE OF ONE OR MORE BUDDIES
222
LOCATE ADDRESSES OF ALL PERSONS ON THE SELECTED BUDDY LIST OR JUST THE SELECTED PERSON(S) UPDATE REQUESTED FROM
224
GENERATE MESSAGE PACKETS OF A CHOSEN PROTOCOL, ADDRESS PACKETS TO THE SELECTED PERSON(S) ON SELECTED BUDDY LISTS OR SELECTED INDIVIDUALS OR SUBSET OF INDIVIDUALS AND PUT ENCRYPTED GPS POSITION DATA IN PAYLOAD ALONG WITH APPROPRIATE TIMESTAMP
226
SEND REQUEST
228
REQUEST PACKETS GET ROUTED IN CELL SYSTEM AND ROUTED TO RUBICON SERVER
230
RUBICON SERVER AUTHENTICATES INITIATOR AND AUTHENTICATES RECIPIENT AND FORWARDS PACKETS TO CELL SYSTEM
232
CELL SYSTEM ROUTES PACKETS TO CELL WHERE BUDDY WIRELESS DEVICE ARE REGISTERED
234
GPS REQUEST MADE TO WIRELESS DEVICE
236
DEVICE CAPTURES CURRENT GPS INFORMATION AND TIMESTAMP, ENCRYPT AND SENDS REQUESTED INFORMATION
238
GENERATE MESSAGE PACKETS OF A CHOSEN PROTOCOL, ADDRESS PACKETS TO THE SELECTED REQUESTING BUDDY AND PUT ENCRYPTED GPS POSITION DATA IN PAYLOAD
240
REQUEST PACKETS GET ROUTED IN CELL SYSTEM AND ROUTED TO RUBICON SERVER
242
RUBICON SERVER AUTHENTICATES INITIATOR AND AUTHENTICATES RECIPIENT AND FORWARDS PACKETS TO CELL SYSTEM
244
CELL SYSTEM ROUTES PACKETS TO CELL WHERE WIRELESS DEVICE OF BUDDY REQUEST LOCATION UPDATE ORIGINATED
246
REQUESTOR RECEIVES RETURN PACKET. RECOVERS DATA, REASSEMBLES GPS POSITION DATA AND TIME STAMP IN PACKET
248
HEADER OF PACKET IS READ AND PORT NUMBER OF GPS POSITION DATA PACKETS CAUSE PACKET TO BE INTERNALLY ROUTED IN THE DEVICE TO THE BUDDY TRACKER SOFTWARE APPLICATION
250
PAYLOAD GPS POSITION DATA IS DECRYPTED AND USED TO UPDATE THE REQUESTOR INTERFACE DISPLAYING THE POSITION DATA OF THE BUDDY REPLYING TO THE REQUEST. THE HEADER DATA IS USED TO DETERMINE WHICH OTHER MEMBERS SENT POSITION UPDATE
252
THE WIRELESS REQUESTOR REPLIES TO EACH RESPONDING DEVICE WITH ITS OWN GPS DATA AND RESPOND BACK REPEATING THIS PROCESS
254
EACH REQUESTED BUDDY PERFORMS SAME TASK WHICH UPDATES X ONE SERVER DATA AND REQUESTING BUDDY (THE REQUEST LEAVES A ONE SHOT TRIGGER) THE REQUESTOR RESPONSE CLEARS TRIGGER AND PREVENTS LOOPING.
ALL COMMUNICATION BETWEEN HAND-SET IS ROUTED THROUGH THE RUBICON SERVER. DATABASE UPDATES ARE MADE TRACKING THE PROGRESS

FIG. 23

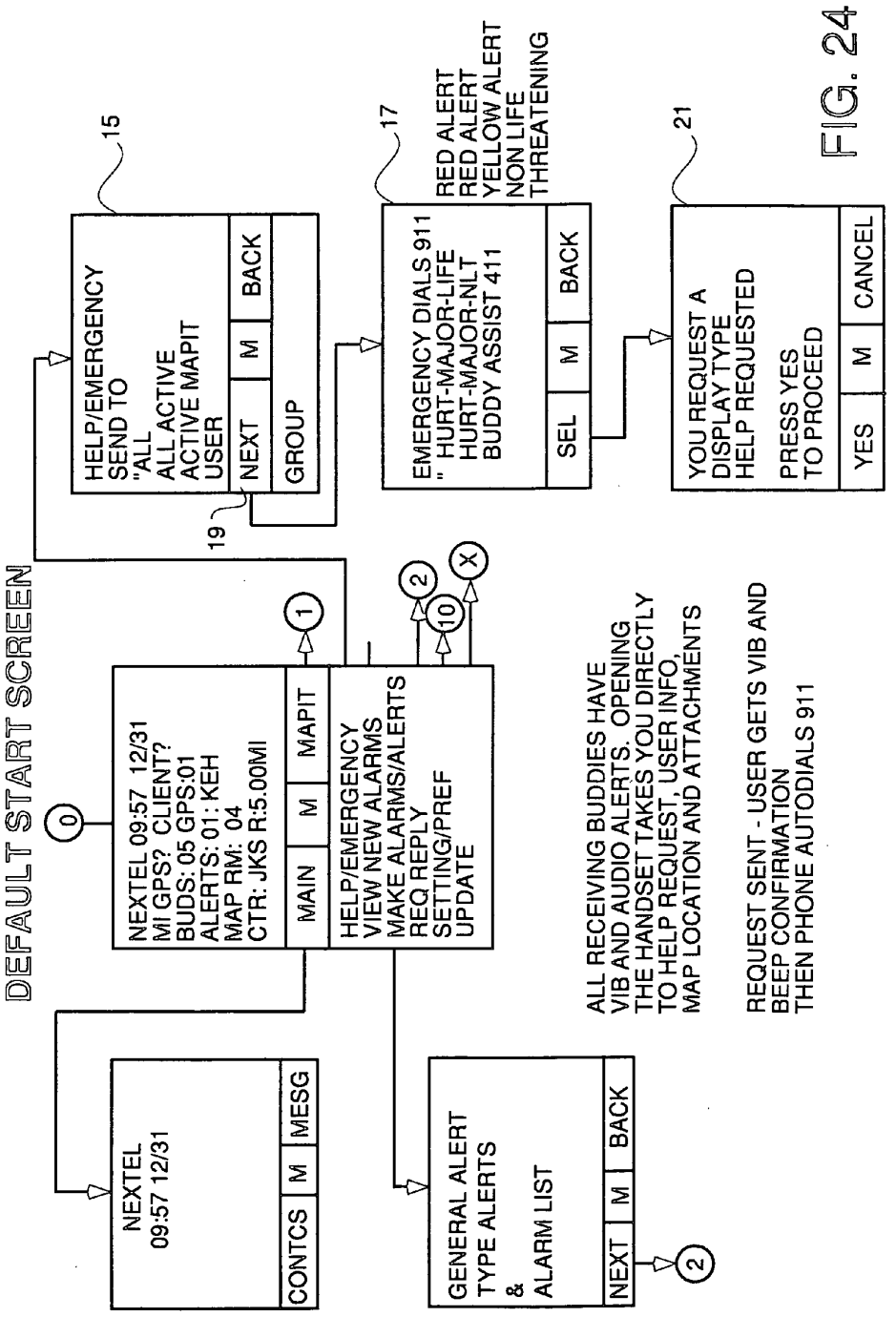

DEFAULT START SCREEN
0
NEXTEL 09:57 12/31
MI GPS? CLIENT?
BUDS: 05 GPS:01
ALERTS: 01: KEH
MAP RM: 04
CTR: JKS R:5.00MI
MAIN
M
MAPIT
1
HELP/EMERGENCY
VIEW NEW ALARMS
MAKE ALARMS/ALERTS
REQ REPLY
SETTING/PREF
UPDATE
2
10
X
NEXTEL
09:57 12/31
CONTCS
M
MESG
GENERAL ALERT
TYPE ALERTS
&
ALARM LIST
NEXT
M
BACK
2
15
HELP/EMERGENCY
SEND TO
"ALL
ALL ACTIVE
ACTIVE MAPIT
USER
NEXT
M
BACK
GROUP
19
17
EMERGENCY DIALS 911
" HURT-MAJOR-LIFE
HURT-MAJOR-NLT
BUDDY ASSIST 411
RED ALERT
RED ALERT
YELLOW ALERT
NON LIFE
THREATENING
SEL
M
BACK
21
YOU REQUEST A
DISPLAY TYPE
HELP REQUESTED
PRESS YES
TO PROCEED
YES
M
CANCEL
ALL RECEIVING BUDDIES HAVE
VIB AND AUDIO ALERTS. OPENING
THE HANDSET TAKES YOU DIRECTLY
TO HELP REQUEST, USER INFO,
MAP LOCATION AND ATTACHMENTS
REQUEST SENT - USER GETS VIB AND
BEEP CONFIRMATION
THEN PHONE AUTODIALS 911

FIG. 24

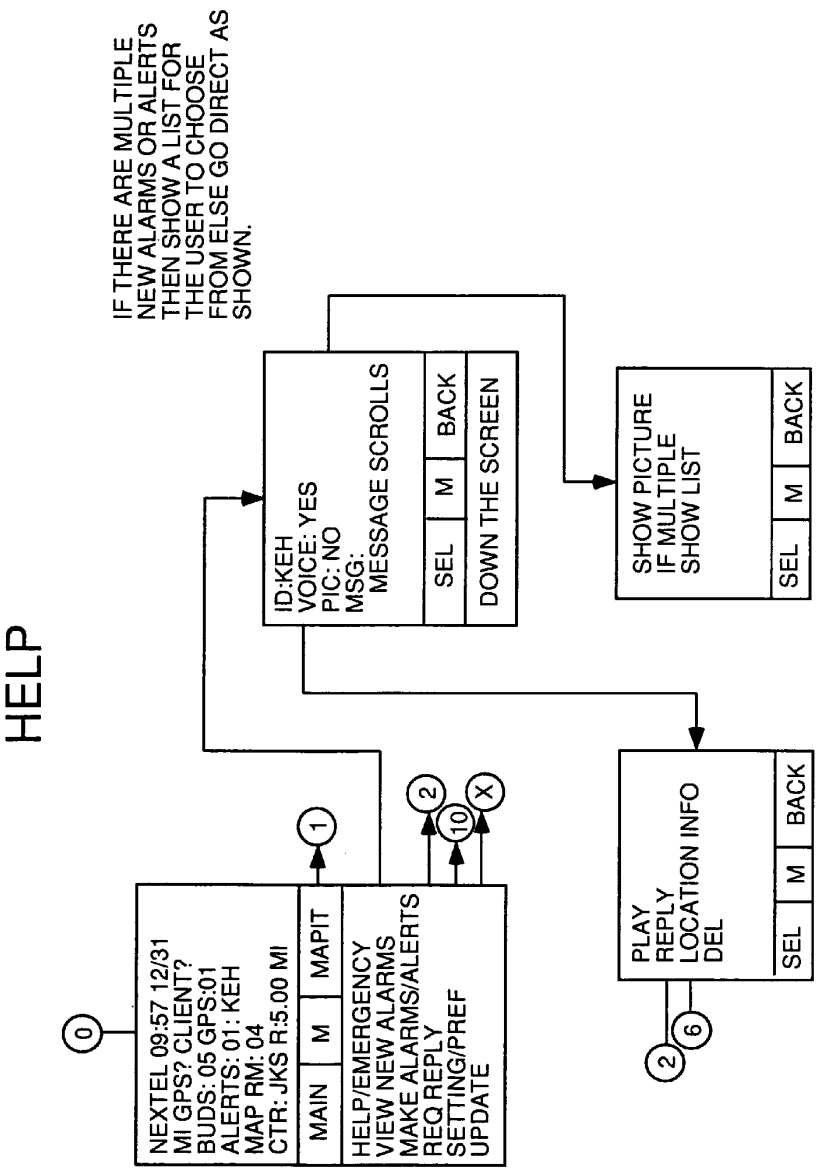

HELP
IF THERE ARE MULTIPLE NEW ALARMS OR ALERTS THEN SHOW A LIST FOR THE USER TO CHOOSE FROM ELSE GO DIRECT AS SHOWN.
NEXTEL 09:57 12/31
MI GPS? CLIENT?
BUDS: 05 GPS:01
ALERTS: 01: KEH
MAP RM: 04
CTR: JKS R:5.00 MI
MAIN
M
MAPIT
HELP/EMERGENCY
VIEW NEW ALARMS
MAKE ALARMS/ALERTS
REQ REPLY
SETTING/PREF
UPDATE
ID:KEH
VOICE: YES
PIC: NO
MSG:
MESSAGE SCROLLS
SEL
M
BACK
DOWN THE SCREEN
SHOW PICTURE
IF MULTIPLE
SHOW LIST
SEL
M
BACK
PLAY
REPLY
LOCATION INFO
DEL
SEL
M
BACK
0
1
2
10
X
2
6

FIG. 25

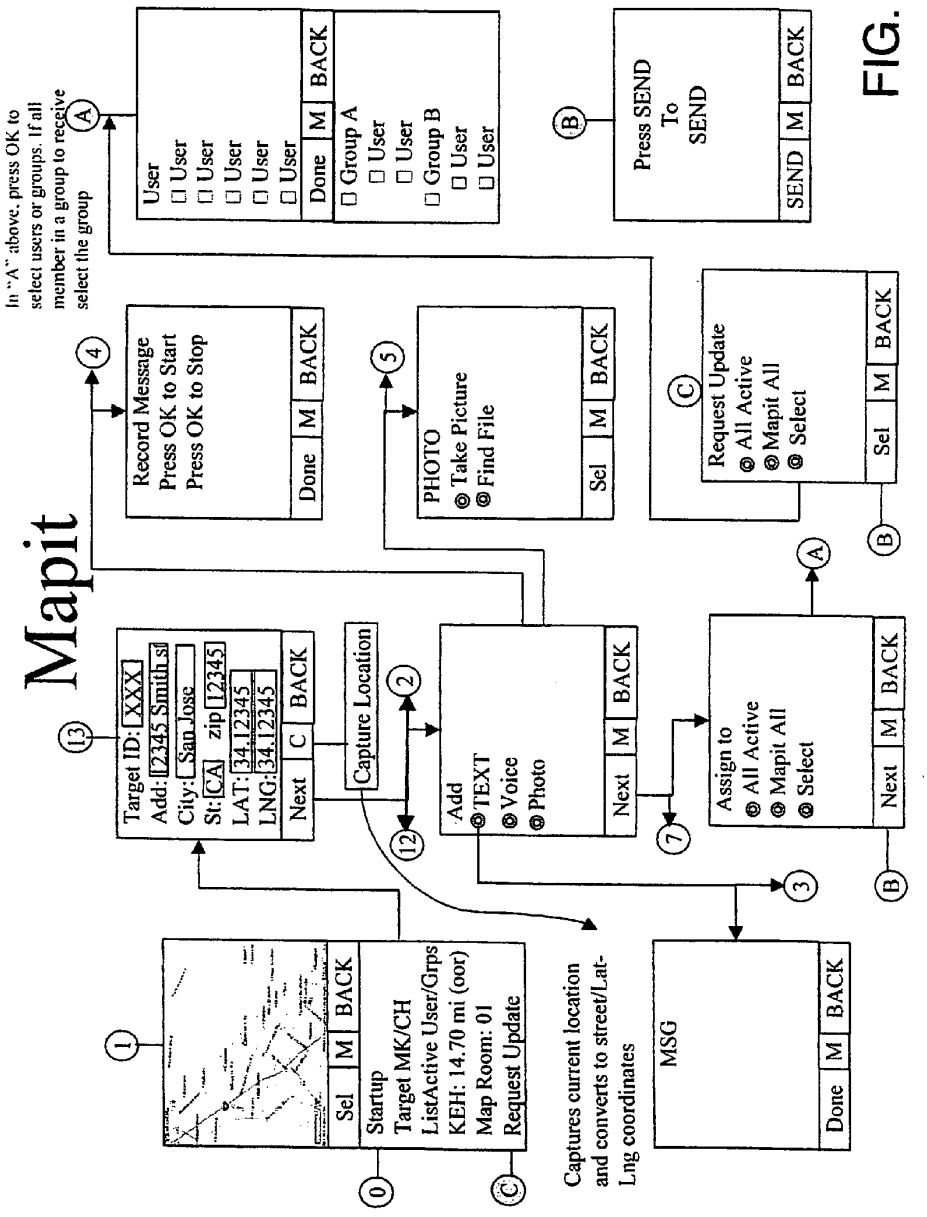

Mapit
Startup
Target MK/CH
ListActive User/Grps
KEH: 14.70 mi (oor)
Map Room: 01
Request Update
Sel
M
BACK
Captures current location and converts to street/Lat-Lng coordinates
Target ID: XXX
Add: 12345 Smith st
City: San Jose
St: CA zip 12345
LAT: 34.12345
LNG: 34.12345
Next
C
BACK
Capture Location
Add
TEXT
Voice
Photo
Next
M
BACK
MSG
Done
M
BACK
Assign to
All Active
Mapit All
Select
Next
M
BACK
Request Update
All Active
Mapit All
Select
Sel
M
BACK
Record Message
Press OK to Start
Press OK to Stop
Done
M
BACK
PHOTO
Take Picture
Find File
Sel
M
BACK
In "A" above, press OK to select users or groups. If all member in a group to receive select the group
User
User
User
User
User
User
Done
M
BACK
Group A
User
User
Group B
User
User
Press SEND
To
SEND
SEND
M
BACK

FIG. 26

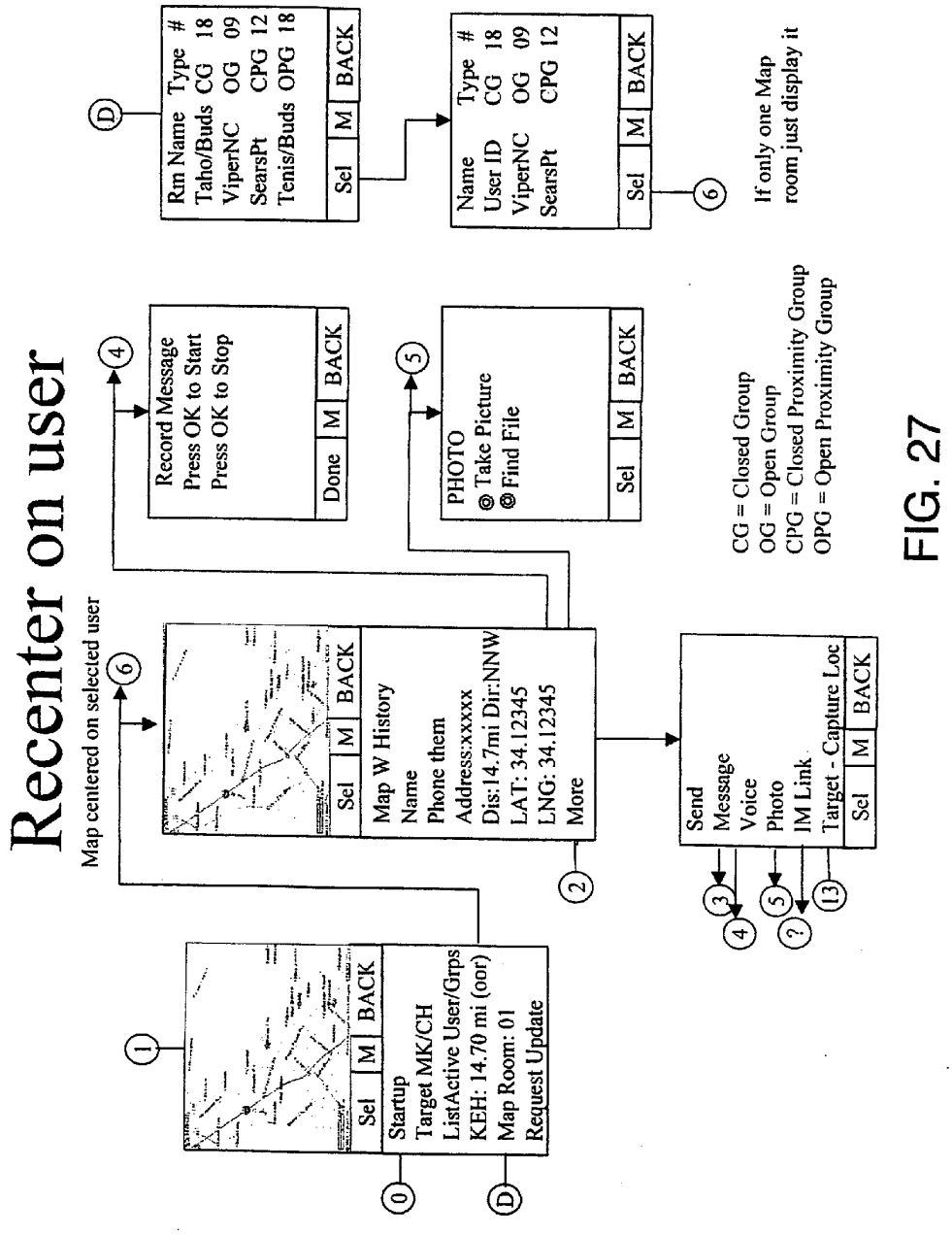

Recenter on user
Map centered on selected user
Sel | M | BACK
Startup
Target MK/CH
ListActive User/Grps
KEH: 14.70 mi (oor)
Map Room: 01
Request Update
Map W History
Name
Phone them
Address:xxxxx
Dis:14.7mi Dir:NNW
LAT: 34.12345
LNG: 34.12345
More
Record Message
Press OK to Start
Press OK to Stop
Done | M | BACK
PHOTO
@ Take Picture
@ Find File
Send
Message
Voice
Photo
IM Link
Target - Capture Loc
Rm Name Type #
Taho/Buds CG 18
ViperNC OG 09
SearsPt CPG 12
Tenis/Buds OPG 18
Name Type #
User ID CG 18
ViperNC OG 09
SearsPt CPG 12
CG = Closed Group
OG = Open Group
CPG = Closed Proximity Group
OPG = Open Proximity Group
If only one Map
room just display it

FIG. 27

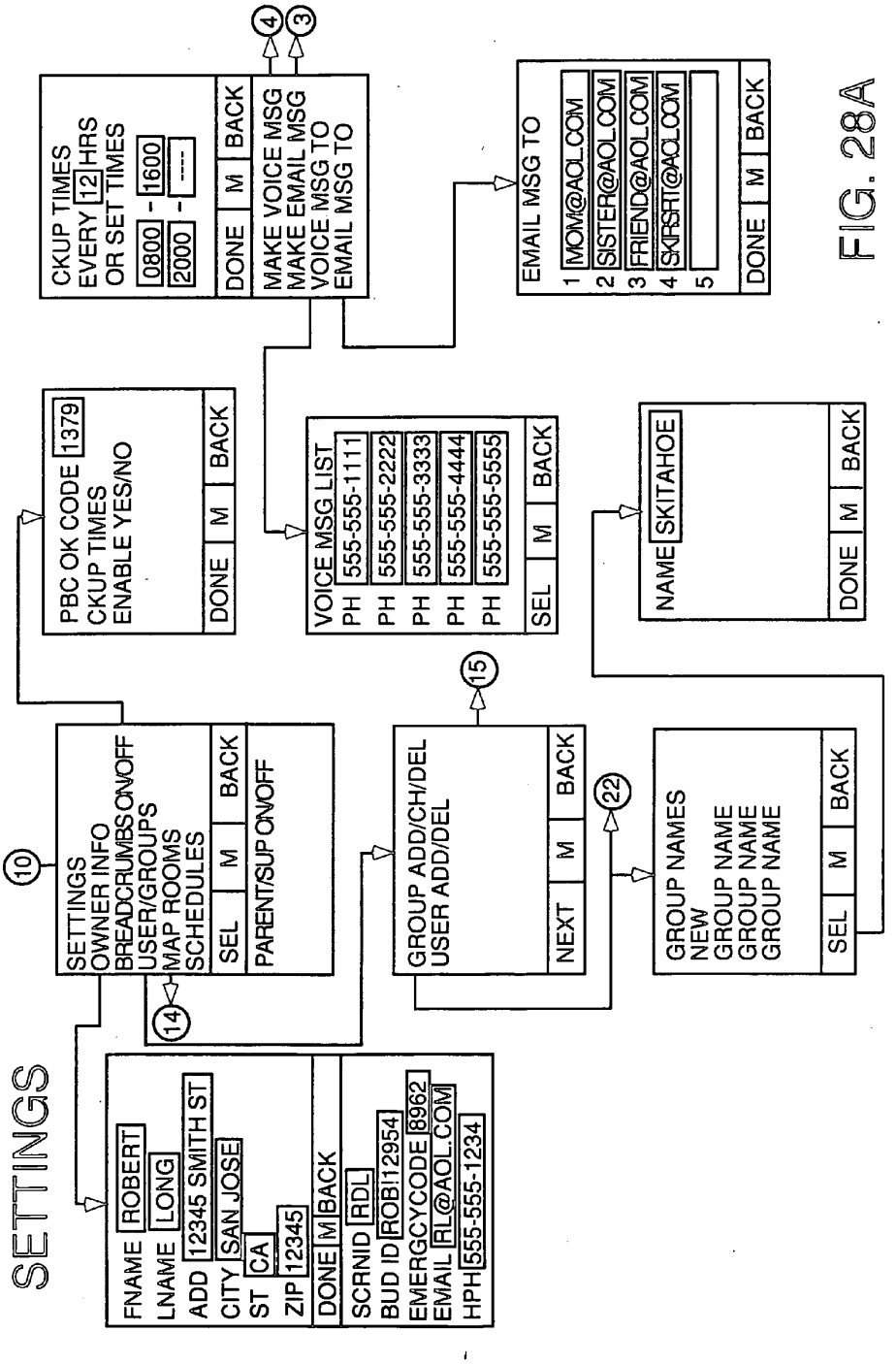

SETTINGS
FNAME ROBERT
LNAME LONG
ADD 12345 SMITH ST
CITY SAN JOSE
ST CA
ZIP 12345
DONE M BACK
SCRNID RDL
BUD ID ROBI12954
EMERGCYCODE 8962
EMAIL RL@AOL.COM
HPH 555-555-1234
10
SETTINGS
OWNER INFO
BREADCRUMBS ON/OFF
USER/GROUPS
MAP ROOMS
SCHEDULES
SEL M BACK
PARENT/SUP ON/OFF
14
PBC OK CODE 1379
CKUP TIMES
ENABLE YES/NO
DONE M BACK
CKUP TIMES
EVERY 12 HRS
OR SET TIMES
0800 - 1600
2000 - ---
DONE M BACK
MAKE VOICE MSG
MAKE EMAIL MSG
VOICE MSG TO
EMAIL MSG TO
4
3
EMAIL MSG TO
1 MOM@AOL.COM
2 SISTER@AOL.COM
3 FRIEND@AOL.COM
4 SKRSHT@AOL.COM
5
DONE M BACK
VOICE MSG LIST
PH 555-555-1111
PH 555-555-2222
PH 555-555-3333
PH 555-555-4444
PH 555-555-5555
SEL M BACK
15
GROUP ADD/CH/DEL
USER ADD/DEL
NEXT M BACK
22
GROUP NAMES
NEW
GROUP NAME
GROUP NAME
GROUP NAME
SEL M BACK
NAME SKITAHOE
DONE M BACK

FIG. 28A

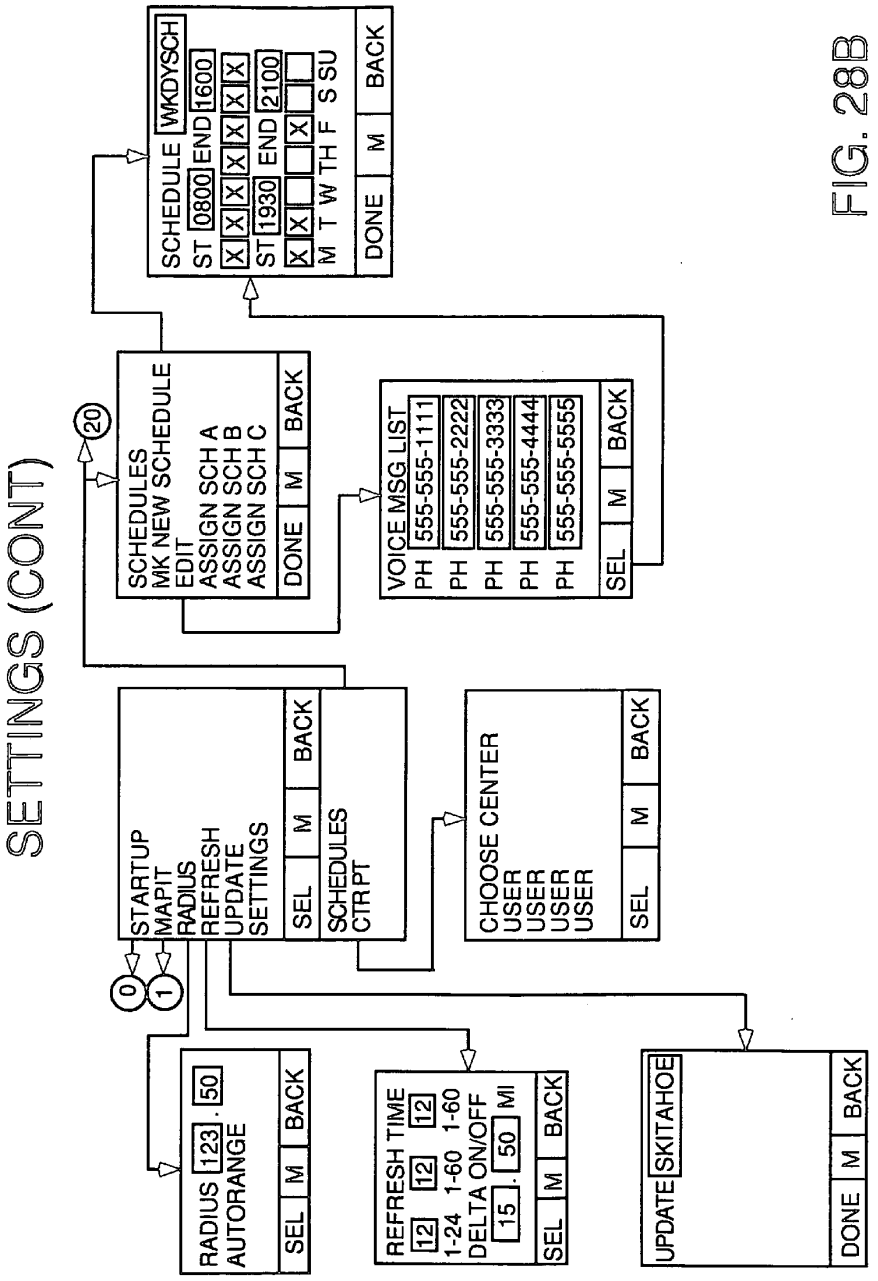

SETTINGS (CONT)
STARTUP
MAPIT
RADIUS
REFRESH
UPDATE
SETTINGS
SEL
M
BACK
SCHEDULES
CTR PT
CHOOSE CENTER
USER
USER
USER
USER
SEL
M
BACK
SCHEDULES
MK NEW SCHEDULE
EDIT
ASSIGN SCH A
ASSIGN SCH B
ASSIGN SCH C
DONE
M
BACK
SCHEDULE WKDYSCH
ST 0800 END 1600
ST 1930 END 2100
M T W TH F S SU
DONE
M
BACK
VOICE MSG LIST
PH 555-555-1111
PH 555-555-2222
PH 555-555-3333
PH 555-555-4444
PH 555-555-5555
SEL
M
BACK
RADIUS 123 . 50
AUTORANGE
SEL
M
BACK
REFRESH TIME 12
12 12 1-60 1-60
DELTA ON/OFF
15 . 50 MI
SEL
M
BACK
UPDATE SKITAHOE
DONE
M
BACK
0
1
20

FIG. 28B

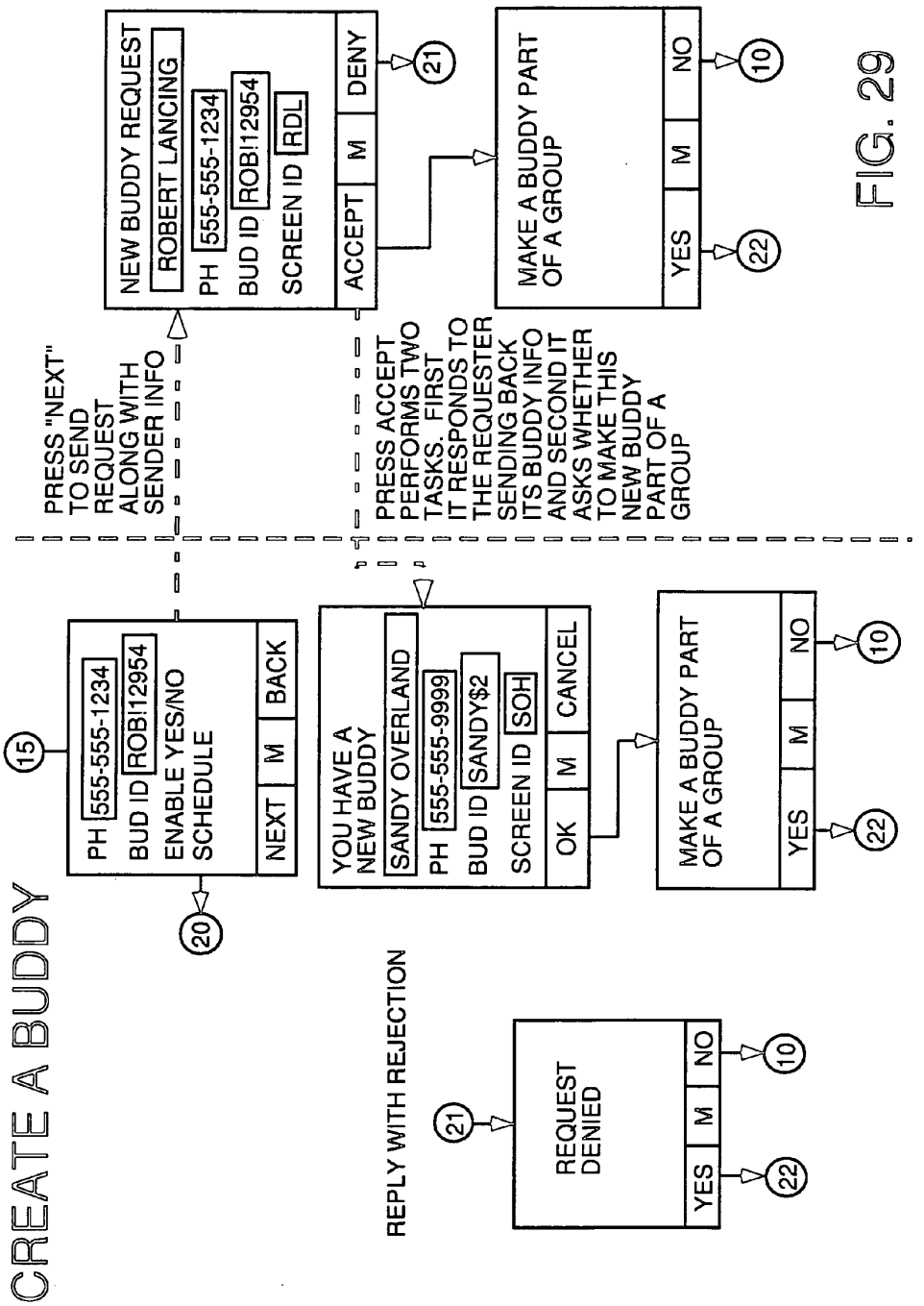

CREATE A BUDDY
15
PH 555-555-1234
BUD ID ROB!12954
ENABLE YES/NO SCHEDULE
NEXT | M | BACK
20
PRESS "NEXT" TO SEND REQUEST ALONG WITH SENDER INFO
NEW BUDDY REQUEST
ROBERT LANCING
PH 555-555-1234
BUD ID ROB!12954
SCREEN ID RDL
ACCEPT | M | DENY
21
PRESS ACCEPT PERFORMS TWO TASKS. FIRST IT RESPONDS TO THE REQUESTER SENDING BACK ITS BUDDY INFO AND SECOND IT ASKS WHETHER TO MAKE THIS NEW BUDDY PART OF A GROUP
MAKE A BUDDY PART OF A GROUP
YES | M | NO
22
10
YOU HAVE A NEW BUDDY
SANDY OVERLAND
PH 555-555-9999
BUD ID SANDY$2
SCREEN ID SOH
OK | M | CANCEL
MAKE A BUDDY PART OF A GROUP
YES | M | NO
22
10
REPLY WITH REJECTION
21
REQUEST DENIED
YES | M | NO
22
10

FIG. 29

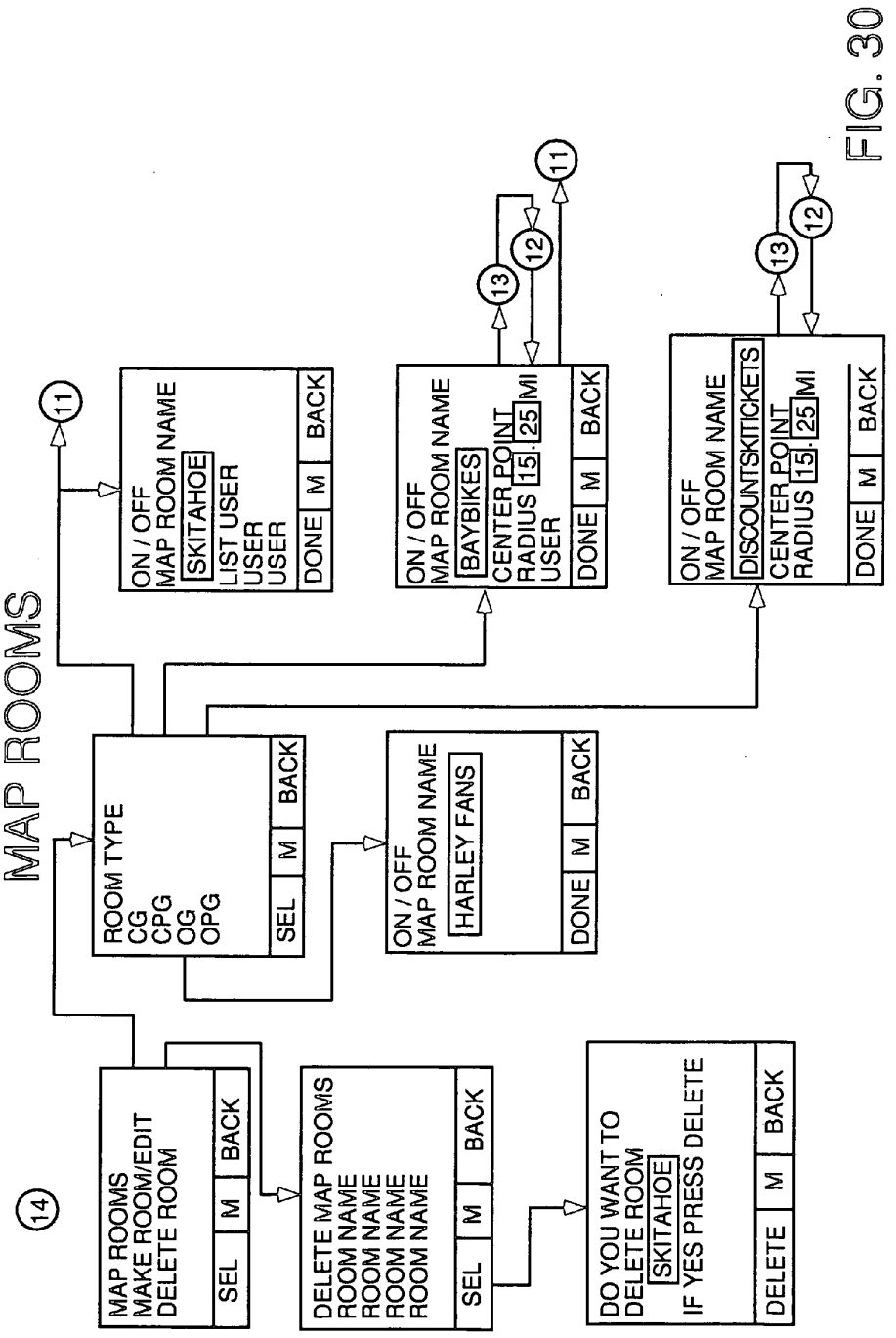

MAP ROOMS
14
MAP ROOMS
MAKE ROOM/EDIT
DELETE ROOM
SEL
M
BACK
DELETE MAP ROOMS
ROOM NAME
ROOM NAME
ROOM NAME
ROOM NAME
SEL
M
BACK
DO YOU WANT TO
DELETE ROOM
SKITAHOE
IF YES PRESS DELETE
DELETE
M
BACK
ROOM TYPE
CG
CPG
OG
OPG
SEL
M
BACK
11
ON / OFF
MAP ROOM NAME
SKITAHOE
LIST USER
USER
USER
DONE
M
BACK
ON / OFF
MAP ROOM NAME
HARLEY FANS
DONE
M
BACK
ON / OFF
MAP ROOM NAME
BAYBIKES
CENTER POINT
RADIUS 15 25 MI
USER
DONE
M
BACK
ON / OFF
MAP ROOM NAME
DISCOUNTSKITICKETS
CENTER POINT
RADIUS 15 25 MI
DONE
M
BACK
11
12
13
12
13

FIG. 30

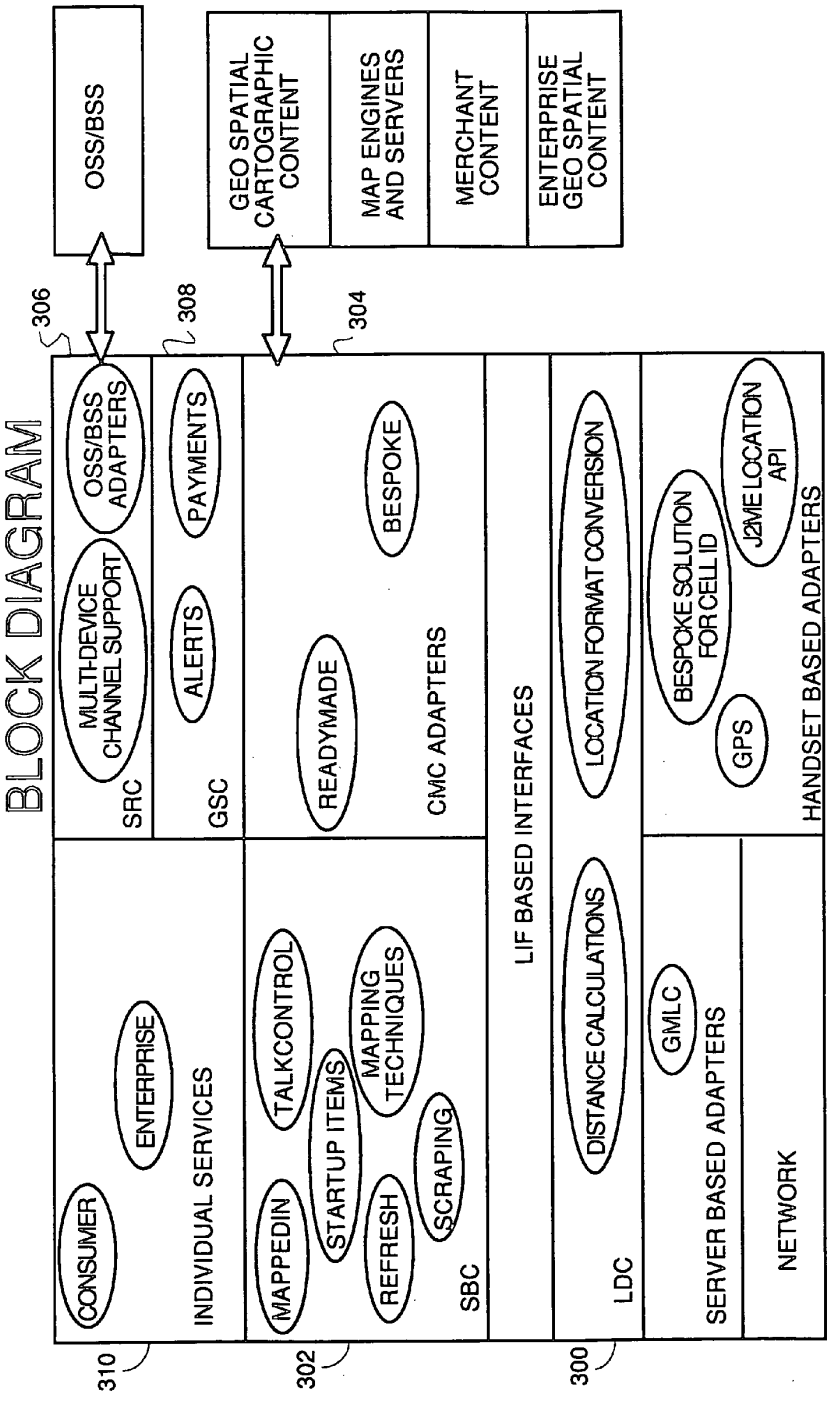

BLOCK DIAGRAM
OSS/BSS
GEO SPATIAL CARTOGRAPHIC CONTENT
MAP ENGINES AND SERVERS
MERCHANT CONTENT
ENTERPRISE GEO SPATIAL CONTENT
306
308
304
SRC
MULTI-DEVICE CHANNEL SUPPORT
OSS/BSS ADAPTERS
GSC
ALERTS
PAYMENTS
CMC ADAPTERS
READYMADE
BESPOKE
LIF BASED INTERFACES
LDC
DISTANCE CALCULATIONS
LOCATION FORMAT CONVERSION
SERVER BASED ADAPTERS
GMLC
BESPOKE SOLUTION FOR CELL ID
GPS
J2ME LOCATION API
HANDSET BASED ADAPTERS
NETWORK
INDIVIDUAL SERVICES
CONSUMER
ENTERPRISE
SBC
MAPPEDIN
TALKCONTROL
STARTUP ITEMS
MAPPING TECHNIQUES
REFRESH
SCRAPING
310
302
300

FIG. 31

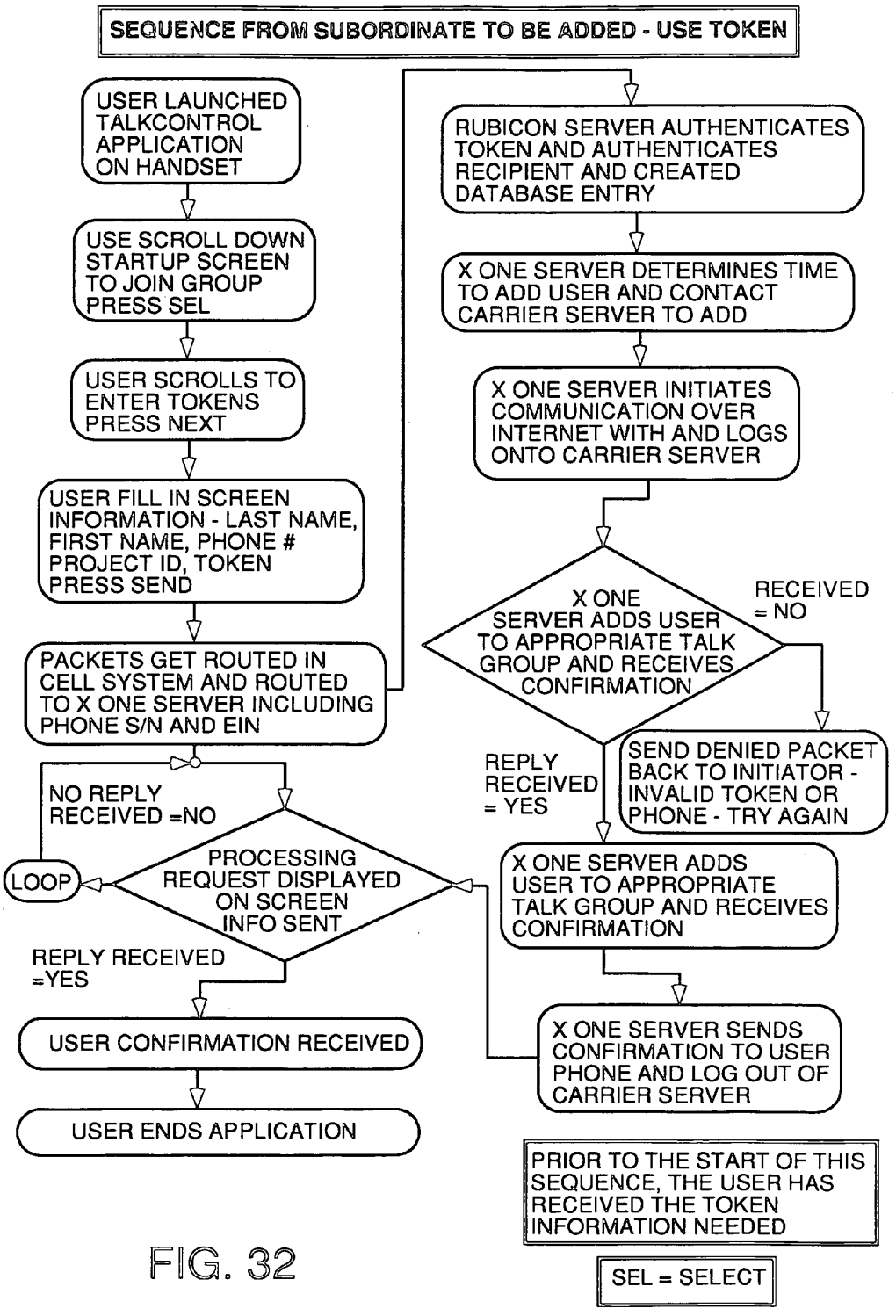

SEQUENCE FROM SUBORDINATE TO BE ADDED - USE TOKEN
USER LAUNCHED TALKCONTROL APPLICATION ON HANDSET
USE SCROLL DOWN STARTUP SCREEN TO JOIN GROUP PRESS SEL
USER SCROLLS TO ENTER TOKENS PRESS NEXT
USER FILL IN SCREEN INFORMATION - LAST NAME, FIRST NAME, PHONE # PROJECT ID, TOKEN PRESS SEND
PACKETS GET ROUTED IN CELL SYSTEM AND ROUTED TO X ONE SERVER INCLUDING PHONE S/N AND EIN
NO REPLY RECEIVED =NO
LOOP
PROCESSING REQUEST DISPLAYED ON SCREEN INFO SENT
REPLY RECEIVED =YES
USER CONFIRMATION RECEIVED
USER ENDS APPLICATION
RUBICON SERVER AUTHENTICATES TOKEN AND AUTHENTICATES RECIPIENT AND CREATED DATABASE ENTRY
X ONE SERVER DETERMINES TIME TO ADD USER AND CONTACT CARRIER SERVER TO ADD
X ONE SERVER INITIATES COMMUNICATION OVER INTERNET WITH AND LOGS ONTO CARRIER SERVER
X ONE SERVER ADDS USER TO APPROPRIATE TALK GROUP AND RECEIVES CONFIRMATION
RECEIVED = NO
SEND DENIED PACKET BACK TO INITIATOR - INVALID TOKEN OR PHONE - TRY AGAIN
REPLY RECEIVED = YES
X ONE SERVER ADDS USER TO APPROPRIATE TALK GROUP AND RECEIVES CONFIRMATION
X ONE SERVER SENDS CONFIRMATION TO USER PHONE AND LOG OUT OF CARRIER SERVER
PRIOR TO THE START OF THIS SEQUENCE, THE USER HAS RECEIVED THE TOKEN INFORMATION NEEDED
SEL = SELECT

FIG. 32

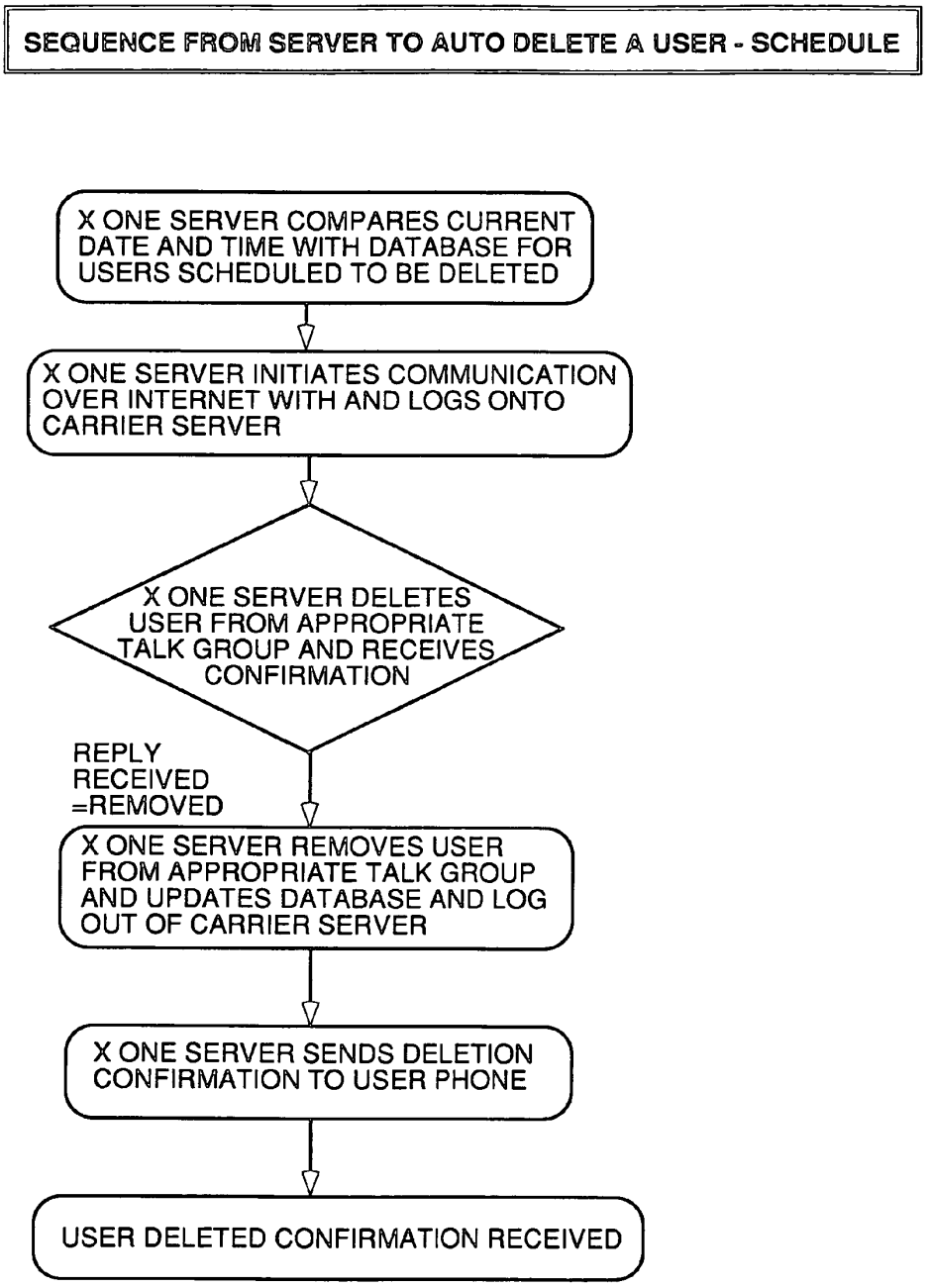

SEQUENCE FROM SERVER TO AUTO DELETE A USER - SCHEDULE
X ONE SERVER COMPARES CURRENT DATE AND TIME WITH DATABASE FOR USERS SCHEDULED TO BE DELETED
X ONE SERVER INITIATES COMMUNICATION OVER INTERNET WITH AND LOGS ONTO CARRIER SERVER
X ONE SERVER DELETES USER FROM APPROPRIATE TALK GROUP AND RECEIVES CONFIRMATION
REPLY RECEIVED =REMOVED
X ONE SERVER REMOVES USER FROM APPROPRIATE TALK GROUP AND UPDATES DATABASE AND LOG OUT OF CARRIER SERVER
X ONE SERVER SENDS DELETION CONFIRMATION TO USER PHONE
USER DELETED CONFIRMATION RECEIVED

FIG. 33

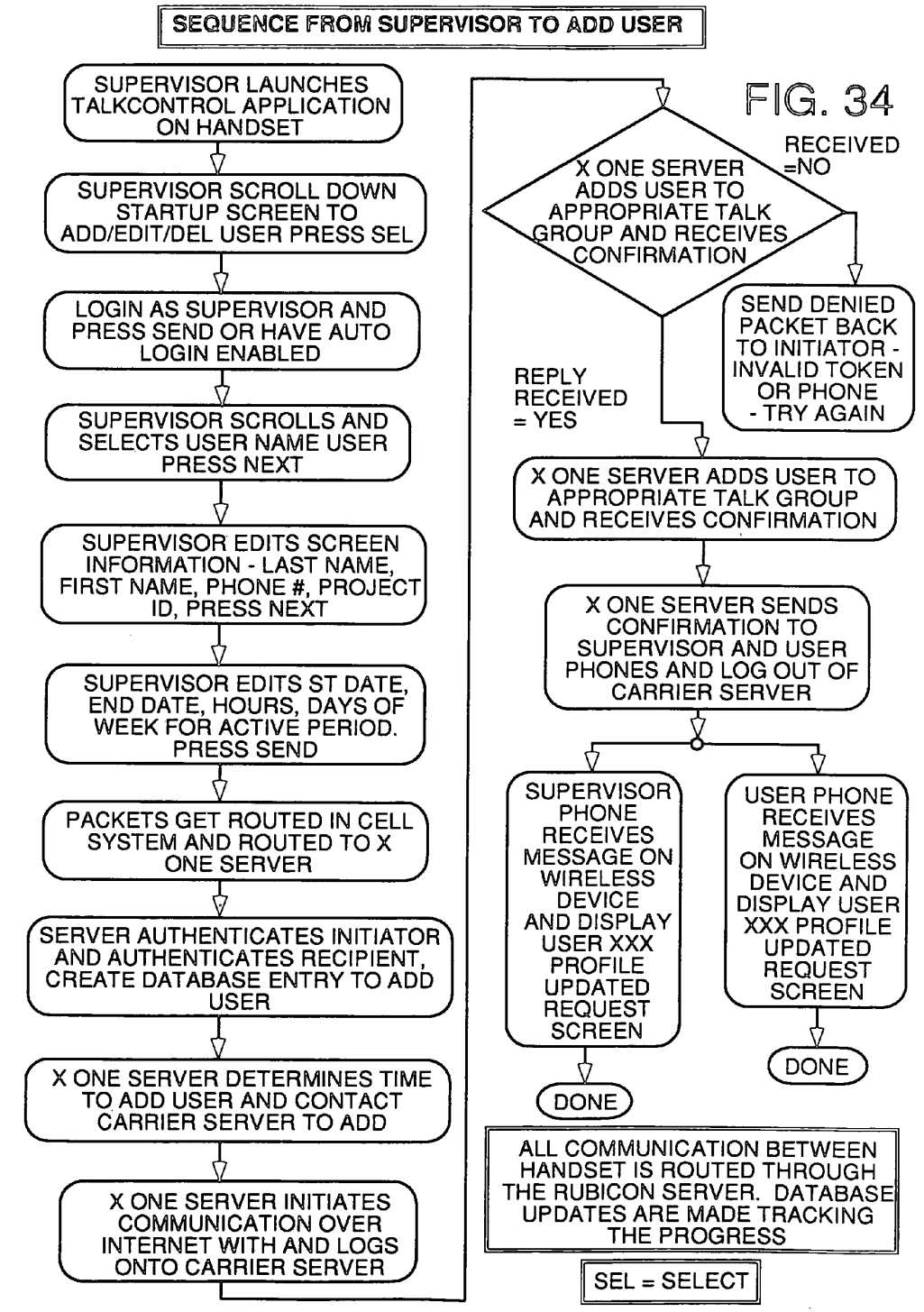

SEQUENCE FROM SUPERVISOR TO ADD USER
SUPERVISOR LAUNCHES TALKCONTROL APPLICATION ON HANDSET
SUPERVISOR SCROLL DOWN STARTUP SCREEN TO ADD/EDIT/DEL USER PRESS SEL
LOGIN AS SUPERVISOR AND PRESS SEND OR HAVE AUTO LOGIN ENABLED
SUPERVISOR SCROLLS AND SELECTS USER NAME USER PRESS NEXT
SUPERVISOR EDITS SCREEN INFORMATION - LAST NAME, FIRST NAME, PHONE #, PROJECT ID, PRESS NEXT
SUPERVISOR EDITS ST DATE, END DATE, HOURS, DAYS OF WEEK FOR ACTIVE PERIOD. PRESS SEND
PACKETS GET ROUTED IN CELL SYSTEM AND ROUTED TO X ONE SERVER
SERVER AUTHENTICATES INITIATOR AND AUTHENTICATES RECIPIENT, CREATE DATABASE ENTRY TO ADD USER
X ONE SERVER DETERMINES TIME TO ADD USER AND CONTACT CARRIER SERVER TO ADD
X ONE SERVER INITIATES COMMUNICATION OVER INTERNET WITH AND LOGS ONTO CARRIER SERVER
X ONE SERVER ADDS USER TO APPROPRIATE TALK GROUP AND RECEIVES CONFIRMATION
RECEIVED =NO
SEND DENIED PACKET BACK TO INITIATOR - INVALID TOKEN OR PHONE - TRY AGAIN
REPLY RECEIVED = YES
X ONE SERVER ADDS USER TO APPROPRIATE TALK GROUP AND RECEIVES CONFIRMATION
X ONE SERVER SENDS CONFIRMATION TO SUPERVISOR AND USER PHONES AND LOG OUT OF CARRIER SERVER
SUPERVISOR PHONE RECEIVES MESSAGE ON WIRELESS DEVICE AND DISPLAY USER XXX PROFILE UPDATED REQUEST SCREEN
USER PHONE RECEIVES MESSAGE ON WIRELESS DEVICE AND DISPLAY USER XXX PROFILE UPDATED REQUEST SCREEN
DONE
DONE
ALL COMMUNICATION BETWEEN HANDSET IS ROUTED THROUGH THE RUBICON SERVER. DATABASE UPDATES ARE MADE TRACKING THE PROGRESS
SEL = SELECT

FIG. 34

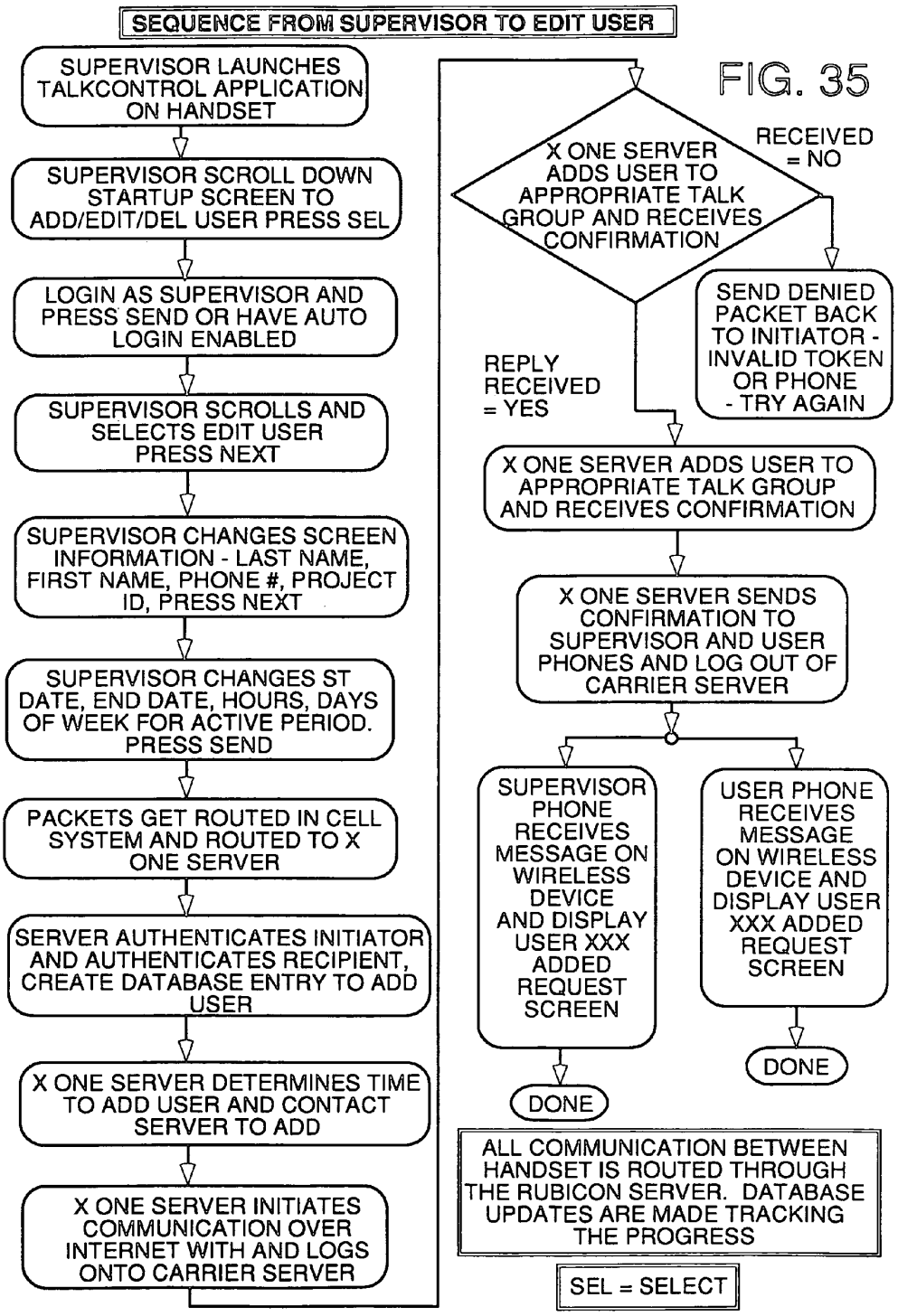

SEQUENCE FROM SUPERVISOR TO EDIT USER
SUPERVISOR LAUNCHES TALKCONTROL APPLICATION ON HANDSET
SUPERVISOR SCROLL DOWN STARTUP SCREEN TO ADD/EDIT/DEL USER PRESS SEL
LOGIN AS SUPERVISOR AND PRESS SEND OR HAVE AUTO LOGIN ENABLED
SUPERVISOR SCROLLS AND SELECTS EDIT USER PRESS NEXT
SUPERVISOR CHANGES SCREEN INFORMATION - LAST NAME, FIRST NAME, PHONE #, PROJECT ID, PRESS NEXT
SUPERVISOR CHANGES ST DATE, END DATE, HOURS, DAYS OF WEEK FOR ACTIVE PERIOD. PRESS SEND
PACKETS GET ROUTED IN CELL SYSTEM AND ROUTED TO X ONE SERVER
SERVER AUTHENTICATES INITIATOR AND AUTHENTICATES RECIPIENT, CREATE DATABASE ENTRY TO ADD USER
X ONE SERVER DETERMINES TIME TO ADD USER AND CONTACT SERVER TO ADD
X ONE SERVER INITIATES COMMUNICATION OVER INTERNET WITH AND LOGS ONTO CARRIER SERVER
X ONE SERVER ADDS USER TO APPROPRIATE TALK GROUP AND RECEIVES CONFIRMATION
RECEIVED = NO
SEND DENIED PACKET BACK TO INITIATOR - INVALID TOKEN OR PHONE - TRY AGAIN
REPLY RECEIVED = YES
X ONE SERVER ADDS USER TO APPROPRIATE TALK GROUP AND RECEIVES CONFIRMATION
X ONE SERVER SENDS CONFIRMATION TO SUPERVISOR AND USER PHONES AND LOG OUT OF CARRIER SERVER
SUPERVISOR PHONE RECEIVES MESSAGE ON WIRELESS DEVICE AND DISPLAY USER XXX ADDED REQUEST SCREEN
USER PHONE RECEIVES MESSAGE ON WIRELESS DEVICE AND DISPLAY USER XXX ADDED REQUEST SCREEN
DONE
DONE
ALL COMMUNICATION BETWEEN HANDSET IS ROUTED THROUGH THE RUBICON SERVER. DATABASE UPDATES ARE MADE TRACKING THE PROGRESS
SEL = SELECT

FIG. 35

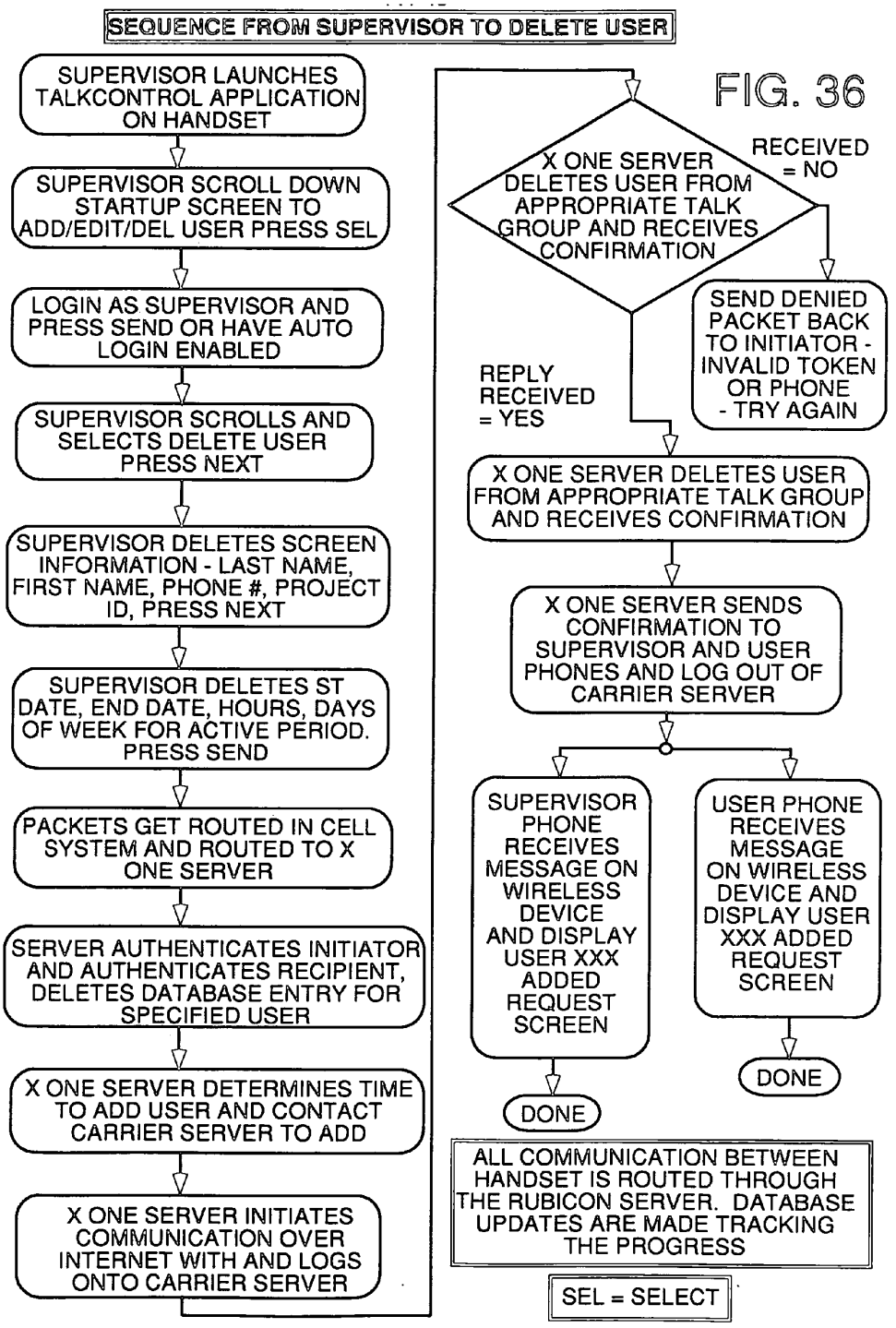

SEQUENCE FROM SUPERVISOR TO DELETE USER
SUPERVISOR LAUNCHES TALKCONTROL APPLICATION ON HANDSET
SUPERVISOR SCROLL DOWN STARTUP SCREEN TO ADD/EDIT/DEL USER PRESS SEL
LOGIN AS SUPERVISOR AND PRESS SEND OR HAVE AUTO LOGIN ENABLED
SUPERVISOR SCROLLS AND SELECTS DELETE USER PRESS NEXT
SUPERVISOR DELETES SCREEN INFORMATION - LAST NAME, FIRST NAME, PHONE #, PROJECT ID, PRESS NEXT
SUPERVISOR DELETES ST DATE, END DATE, HOURS, DAYS OF WEEK FOR ACTIVE PERIOD. PRESS SEND
PACKETS GET ROUTED IN CELL SYSTEM AND ROUTED TO X ONE SERVER
SERVER AUTHENTICATES INITIATOR AND AUTHENTICATES RECIPIENT, DELETES DATABASE ENTRY FOR SPECIFIED USER
X ONE SERVER DETERMINES TIME TO ADD USER AND CONTACT CARRIER SERVER TO ADD
X ONE SERVER INITIATES COMMUNICATION OVER INTERNET WITH AND LOGS ONTO CARRIER SERVER
X ONE SERVER DELETES USER FROM APPROPRIATE TALK GROUP AND RECEIVES CONFIRMATION
RECEIVED = NO
SEND DENIED PACKET BACK TO INITIATOR - INVALID TOKEN OR PHONE - TRY AGAIN
REPLY RECEIVED = YES
X ONE SERVER DELETES USER FROM APPROPRIATE TALK GROUP AND RECEIVES CONFIRMATION
X ONE SERVER SENDS CONFIRMATION TO SUPERVISOR AND USER PHONES AND LOG OUT OF CARRIER SERVER
SUPERVISOR PHONE RECEIVES MESSAGE ON WIRELESS DEVICE AND DISPLAY USER XXX ADDED REQUEST SCREEN
USER PHONE RECEIVES MESSAGE ON WIRELESS DEVICE AND DISPLAY USER XXX ADDED REQUEST SCREEN
DONE
DONE
ALL COMMUNICATION BETWEEN HANDSET IS ROUTED THROUGH THE RUBICON SERVER. DATABASE UPDATES ARE MADE TRACKING THE PROGRESS
SEL = SELECT

FIG. 36

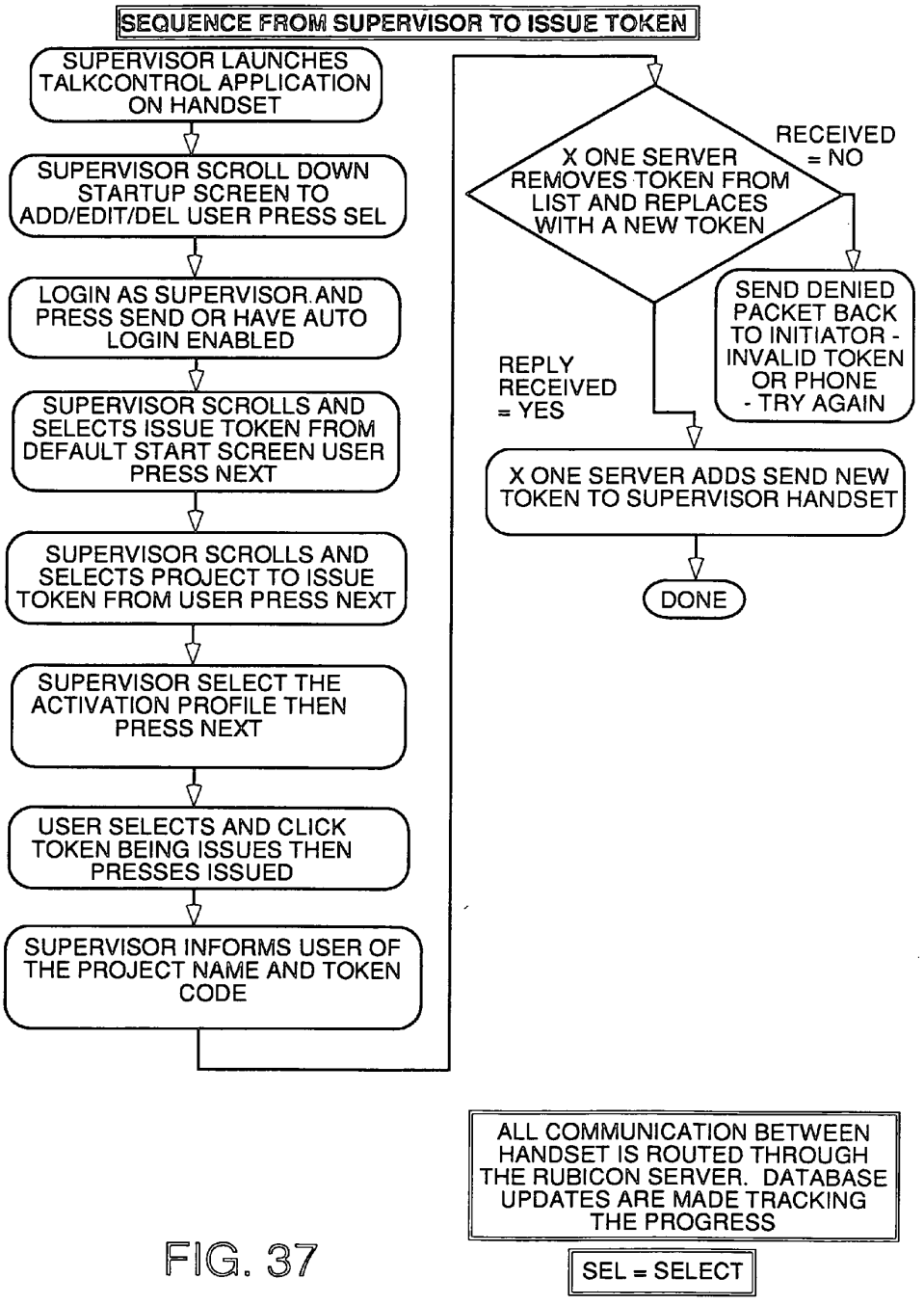

SEQUENCE FROM SUPERVISOR TO ISSUE TOKEN
SUPERVISOR LAUNCHES TALKCONTROL APPLICATION ON HANDSET
SUPERVISOR SCROLL DOWN STARTUP SCREEN TO ADD/EDIT/DEL USER PRESS SEL
LOGIN AS SUPERVISOR AND PRESS SEND OR HAVE AUTO LOGIN ENABLED
SUPERVISOR SCROLLS AND SELECTS ISSUE TOKEN FROM DEFAULT START SCREEN USER PRESS NEXT
SUPERVISOR SCROLLS AND SELECTS PROJECT TO ISSUE TOKEN FROM USER PRESS NEXT
SUPERVISOR SELECT THE ACTIVATION PROFILE THEN PRESS NEXT
USER SELECTS AND CLICK TOKEN BEING ISSUES THEN PRESSES ISSUED
SUPERVISOR INFORMS USER OF THE PROJECT NAME AND TOKEN CODE
X ONE SERVER REMOVES TOKEN FROM LIST AND REPLACES WITH A NEW TOKEN
RECEIVED = NO
SEND DENIED PACKET BACK TO INITIATOR - INVALID TOKEN OR PHONE - TRY AGAIN
REPLY RECEIVED = YES
X ONE SERVER ADDS SEND NEW TOKEN TO SUPERVISOR HANDSET
DONE
ALL COMMUNICATION BETWEEN HANDSET IS ROUTED THROUGH THE RUBICON SERVER. DATABASE UPDATES ARE MADE TRACKING THE PROGRESS
SEL = SELECT

FIG. 37

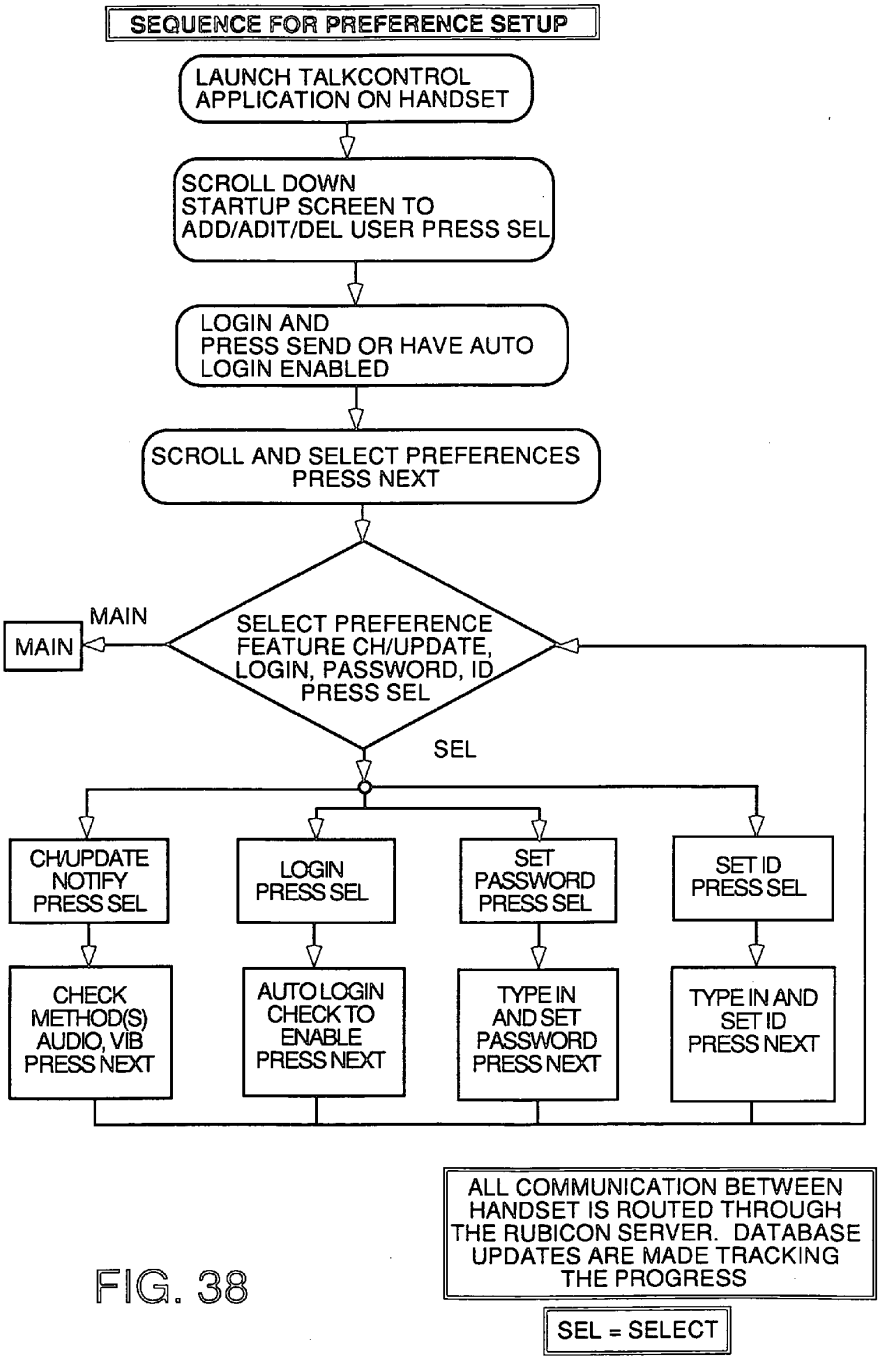

SEQUENCE FOR PREFERENCE SETUP
LAUNCH TALKCONTROL APPLICATION ON HANDSET
SCROLL DOWN STARTUP SCREEN TO ADD/ADIT/DEL USER PRESS SEL
LOGIN AND PRESS SEND OR HAVE AUTO LOGIN ENABLED
SCROLL AND SELECT PREFERENCES PRESS NEXT
SELECT PREFERENCE FEATURE CH/UPDATE, LOGIN, PASSWORD, ID PRESS SEL
MAIN
MAIN
SEL
CH/UPDATE NOTIFY PRESS SEL
LOGIN PRESS SEL
SET PASSWORD PRESS SEL
SET ID PRESS SEL
CHECK METHOD(S) AUDIO, VIB PRESS NEXT
AUTO LOGIN CHECK TO ENABLE PRESS NEXT
TYPE IN AND SET PASSWORD PRESS NEXT
TYPE IN AND SET ID PRESS NEXT
ALL COMMUNICATION BETWEEN HANDSET IS ROUTED THROUGH THE RUBICON SERVER. DATABASE UPDATES ARE MADE TRACKING THE PROGRESS
SEL = SELECT

FIG. 38

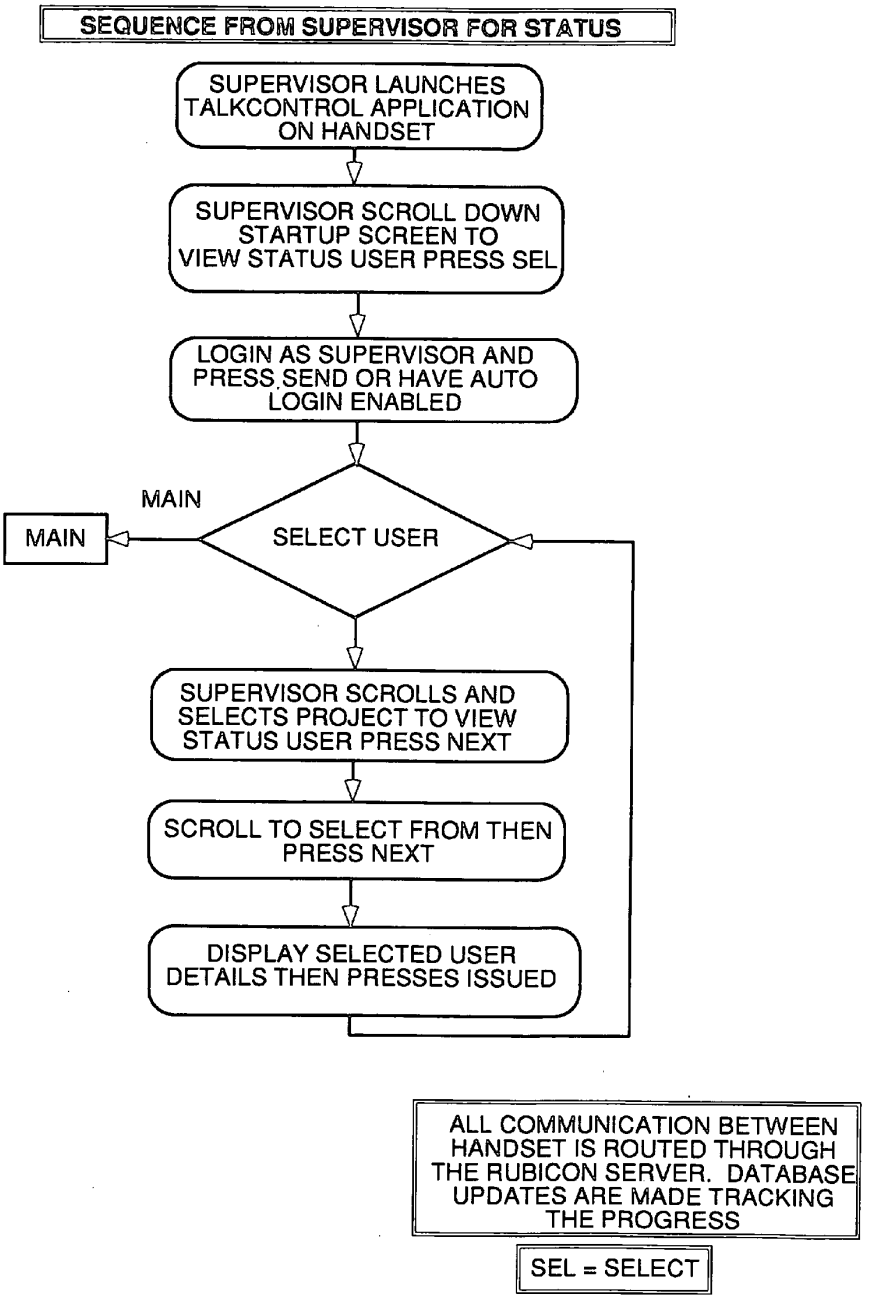

SEQUENCE FROM SUPERVISOR FOR STATUS
SUPERVISOR LAUNCHES TALKCONTROL APPLICATION ON HANDSET
SUPERVISOR SCROLL DOWN STARTUP SCREEN TO VIEW STATUS USER PRESS SEL
LOGIN AS SUPERVISOR AND PRESS SEND OR HAVE AUTO LOGIN ENABLED
MAIN
MAIN
SELECT USER
SUPERVISOR SCROLLS AND SELECTS PROJECT TO VIEW STATUS USER PRESS NEXT
SCROLL TO SELECT FROM THEN PRESS NEXT
DISPLAY SELECTED USER DETAILS THEN PRESSES ISSUED
ALL COMMUNICATION BETWEEN HANDSET IS ROUTED THROUGH THE RUBICON SERVER. DATABASE UPDATES ARE MADE TRACKING THE PROGRESS
SEL = SELECT

FIG. 39

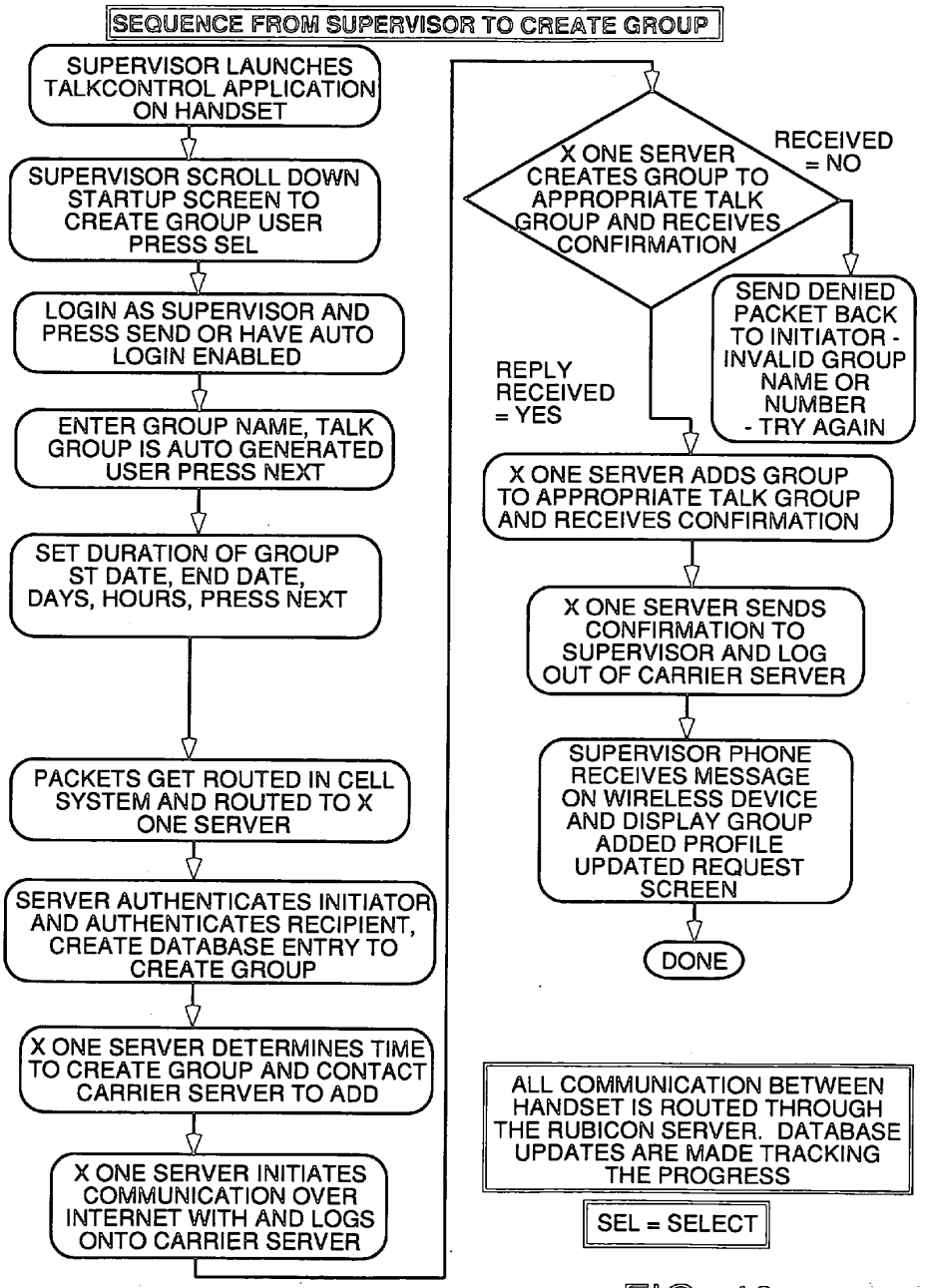

SEQUENCE FROM SUPERVISOR TO CREATE GROUP
SUPERVISOR LAUNCHES TALKCONTROL APPLICATION ON HANDSET
SUPERVISOR SCROLL DOWN STARTUP SCREEN TO CREATE GROUP USER PRESS SEL
LOGIN AS SUPERVISOR AND PRESS SEND OR HAVE AUTO LOGIN ENABLED
ENTER GROUP NAME, TALK GROUP IS AUTO GENERATED USER PRESS NEXT
SET DURATION OF GROUP ST DATE, END DATE, DAYS, HOURS, PRESS NEXT
PACKETS GET ROUTED IN CELL SYSTEM AND ROUTED TO X ONE SERVER
SERVER AUTHENTICATES INITIATOR AND AUTHENTICATES RECIPIENT, CREATE DATABASE ENTRY TO CREATE GROUP
X ONE SERVER DETERMINES TIME TO CREATE GROUP AND CONTACT CARRIER SERVER TO ADD
X ONE SERVER INITIATES COMMUNICATION OVER INTERNET WITH AND LOGS ONTO CARRIER SERVER
X ONE SERVER CREATES GROUP TO APPROPRIATE TALK GROUP AND RECEIVES CONFIRMATION
RECEIVED = NO
SEND DENIED PACKET BACK TO INITIATOR - INVALID GROUP NAME OR NUMBER - TRY AGAIN
REPLY RECEIVED = YES
X ONE SERVER ADDS GROUP TO APPROPRIATE TALK GROUP AND RECEIVES CONFIRMATION
X ONE SERVER SENDS CONFIRMATION TO SUPERVISOR AND LOG OUT OF CARRIER SERVER
SUPERVISOR PHONE RECEIVES MESSAGE ON WIRELESS DEVICE AND DISPLAY GROUP ADDED PROFILE UPDATED REQUEST SCREEN
DONE
ALL COMMUNICATION BETWEEN HANDSET IS ROUTED THROUGH THE RUBICON SERVER. DATABASE UPDATES ARE MADE TRACKING THE PROGRESS
SEL = SELECT

FIG. 40

LOCATION SHARING AND TRACKING USING MOBILE PHONES OR OTHER WIRELESS DEVICES

FIELD OF USE AND BACKGROUND OF THE INVENTION

[0001] The cellular market in 2003 was around 150-160 million devices in the US and this number is growing at over 10% per year at least. Europe already has 320 million cell phone users, and the global market is over 1.4 billion devices.

[0002] Cell phone carriers are looking for opportunities to both increase revenue and profits by providing new services. For example, recently cell phones have been provided with browsers to allow surfing the internet from the phone. One of the needs businesses and families and individuals have is the need to know where their employees, children and friends are. No two way position information sharing technology currently exists as far as the applicant is aware.

[0003] The one way location sharing prior art includes On Star and the Mercedes Benz TeleAid services where, via GPS receivers and cellular phone capability built into a car, an aid center can track cars all over the world and speak with the occupants and sense when the cars airbags have deployed. Other commerical services allow parents to track the locations of their children in a one way location sharing manner. None of these services allow the occupants of the car to know where the aid center is or allows the children to know where their parents are.

[0004] Another need is for a system for use by motorists, hikers, pilots and boatsmen to allow them to be able to contact rescuers and know the location of the rescuers as they come to the aid of the stranded person and to allow the rescuers to know the location of the victims they are trying to rescue. This need requires that the cell phones have the capability to be reconfigured in the field to add an "instant buddy" to the list of people with whom location information is shared. The prior art kid tracking systems could not be reconfigured in the field to add new individuals with whom location information was to be shared.

Differences Over Kid Tracking Prior Art

[0005] In the prior art, one could buy phones that were set up at the manufacturer to enable parents to locate their children. One such service allows up to eight phones to be used and allows parents to monitor the locations of their kids. But these services do not allow the kids see the locations of their parents because the service is not set up to share location information between phones. In other words, it is a one way service with the kids locations being sent to the parents phones for display but not vice versa. Further, there is no mechanism to add groups and members of groups, and there is no mechanism to set up "instant buddies" as that term is used below (temporary location sharing between phones on an ask and accept basis which automatically expires after a configurable interval terminates). The kid locator phones are set up at the factory and nothing can be changed in the field by the users and they are always on and cannot be disabled.

[0006] It is useful to be able to share locations among multiple cell phones which have GPS locator ability. Such an ability would be useful for people in groups who have made plans to meet at specific locations at specific times. When one person is late, the others in the group would be able to ascertain the tardy person's location. To alleviate privacy concerns, it would be useful to be able to turn off location sharing or to program location sharing so that it turns itself on automatically at some date and time and turns itself off at some other programmable date and time. It would also be useful to have a map display on cell phones which are picture enabled and to plot the locations on the map of persons in a group who have their location sharing capability turned on.

SUMMARY OF THE INVENTION

[0007] The invention contemplates 2.5 GHz and 3 GHz Java enabled, web enabled (or similar) cell phones and Personal Digital Assistants or other web enabled wireless products with global positioning system (GPS) receivers and sufficiently large liquid crystal displays for the preferred embodiment. The phones must be web enabled to be able to send and receive TCP/IP or other protocol packets over the internet to the Buddy Watch server.

[0008] In some embodiments where push-to-talk enablement is implemented, GPS receivers are not necessary in the cell phones but they must be web enabled to be able to send and receive TCP/IP or other packets over the internet to the Buddy Watch server.

[0009] These phones and other wireless devices are programmed with software (programmed at the factory or downloaded from the assignee of the present invention) to have the user interface and functionality described herein to allow mutual tracking and optional position mapping displays of members of groups and of instant buddies coming to the rescue of stranded motorists, hikers, pilots and boatsmen. These phones work with a Buddy Watch™ server coupled to the internet. This server is not limited to any specific language or technology nor is it limited to any specific wired or wireless solution or any particular transmission physical layer or protocol.

[0010] The teachings of the invention do not require development of new cell phone or PDA technology nor do they require development of new cellular communication infrastructure. The functionality implemented by the software of the invention utilizes existing platforms and infrastructure. In the preferred embodiment, the software of the invention is developed to JAVA specifications.

[0011] In its primary mode, the process of the invention only allows exchanging and mapping of position data with persons on a Buddy List™ programmed into a Buddy Watch™ (synonym for Buddy Tracker™) device (defined as any of the devices mentioned anywhere in this specification when programmed to operate in Buddy Watch mode or coupled to another device operating in Buddy Watch mode). The user must allow others on his Buddy Lists to "see" his location (location sharing may be turned off), and user must request to see the location of others on her Buddy Lists to be able to have their positions reported and/or mapped. Position information exchanged via radio transmission on the cellular infrastructure is encrypted so that outsiders cannot see or use location information that is transmitted. A simple menu structure allows easy setup and management of Buddy Watch application programs. The keypad of the phone or PDA is used to enter information into the Buddy Watch

enabled device. Online help is available to setup and use the Buddy Tracker application program(s).

[0012] The teachings of the invention can also be integrated into other products and services such as autos with GPS based navigation systems. This would be done by expanding the navigation system to have a cellular transceiver capable of sending and receiving digital data including position data to the Buddy Tracker server. It could also be done by expanding the GPS navigation system product to have a USB or other interface port to couple the system to a cell phone or PDA of the type described above. This interface would allow the GPS navigation system to receive position data from the wireless digital data transceiver and map the position data on the GPS navigation system display of the auto. Handheld GPS navigation devices can also be expanded by integrating a cell phone therein or providing a port to interface to a cell phone to exchange position information with the Buddy Tracker server.

[0013] In a system employing the teachings of the invention, the users can change things on the fly in the field such as: adding groups and members; adding instant buddies, changing the size of the area in which their buddies can be tracked, enabling or disabling the location information sharing function without disabling the phone, etc.

[0014] Some of the benefits of the Buddy Tracker technology are that it allows businesses to easily identify which service persons are closest to the next job and to provide personnel in the field to know the positions of their co-workers and to share their location with their co-workers. Parents can keep track of where their kids are. Friends can keep track of where their buddies and share their position with their buddies. Location information will be shared only so long as the phone is on and in an area where the device can receive a GPS signal and send the phone's coordinates out on the cellular network (and the location sharing capability is enabled).

[0015] Further, the cellular carriers do not have to invest in engineering or infrastructure to offer the Buddy Watch functionality. The software that implements the Buddy Watch functionality can be downloaded from the web or installed at the point of sale of a cell phone or PDA. Use and sale of an application that makes use of the on-board GPS capability of cell phones and PDAs built to comply with the E911 requirement allows the carriers to recoup some of the costs imposed upon them by the E911 requirement.

[0016] Enhancements to cellular phones in recent years such as the addition of cameras and web browsers have lost track of one of the basic reasons for cell phones in the first place—people want to communicate with and know where other people are. This is applicable to parental monitoring and increasing the efficiency of business and increasing the effectiveness of law enforcement. The Buddy Watch system also functions to decrease the load on the 911 system since not every situation requires the help of 911 authorities.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017] **FIG. 1** is a screen shot of a typical opening screen which would be displayed on a cellphone with the BuddyTracker™ software enabled.

[0018] **FIG. 2A** shows a block diagram of the Buddy Watch system.

[0019] **FIG. 2B** illustrates a matrix or web of supervisorial relationships and Buddy Lists.

[0020] **FIG. 2C** is a diagram of the start up screen.

[0021] **FIG. 2D** shows the mapit page.

[0022] **FIG. 2E** shows additions options for manual refresh, etc. which can be reached by scrolling down the mapit page below the list of active users. **FIG. 2F** is a screen of active users.

[0023] **FIG. 3** represents a display in the user interface which shows individuals on the phone's Buddy List as well as a group of buddies which has been given the name Tennis Team.

[0024] **FIG. 4** is a user interface display showing the result when the tennis team Buddy List entry is selected and the information that is displayed when one of the members of the tennis team is selected for display of location information.

[0025] **FIG. 5** is a user interface display showing a map rendering with the location of a selected member of the tennis teach group displayed thereon.

[0026] **FIG. 6** is a user interface display showing a map rendering with the location history of a selected member of the tennis team rendered thereon.

[0027] **FIG. 7** is a screen shot of a display in a typical system employing the invention showing positions and status of members of a selected group.

[0028] **FIG. 8** is a screen display showing what is displayed when Dean is selected and the Mapit command in box **48** is given by double clicking on the box or by any other means.

[0029] **FIG. 9** is a screen shot of an instant buddy display after an instant buddy relationship has been set up.

[0030] **FIG. 10** is a screen shot of a typical display in a system employing the teachings of the invention to establish an instant buddy ID in box, and give the instant buddy a caller ID in box **72** (the instant buddy's caller ID or phone number is used by default).

[0031] **FIG. 11** is a screen shot of the display which appears on at least the instant buddy's phone after a stranded motorist, pilot or hiker has contacted 911 and entered a caller ID and carrier for a proposed instant buddy.

[0032] **FIG. 12** is a block diagram of a typical prior art cellular system infrastructure in which the method and apparatus of the invention work in a peer-to-peer embodiment.

[0033] **FIG. 13**, comprised of **FIGS. 13A and 13B**, is a flowchart of the method of exchanging GPS position data among cell phones of a watch list.

[0034] **FIG. 14**

[0035] **FIG. 15** is a flowchart of the process of establishing an Instant Buddy Relationship.

[0036] **FIG. 16** is a block diagram of a typical cellular system coupled by a gateway and a Wide Area Network such as the internet to a Buddy Watch server to provide the infrastructure of the invention.

[0037] **FIG. 17** is a flowchart of the preferred Instant Buddy Setup process.

[0038] **FIGS. 18 and 19** are diagrams of some of the user interface display screens involved in the Instant Buddy Setup process.

[0039] **FIG. 20** is a flowchart of the process of enabling the personal bread crumbs mode and how it works.

[0040] **FIG. 21** is a flowchart of another embodiment of a process to establish and use personal bread crumbs mode.

[0041] **FIG. 22** is a flowchart of the preferred embodiment for the instant buddy setup process.

[0042] **FIG. 23** is a flowchart of another embodiment of a process to receive buddy location update requests and process them.

[0043] **FIG. 24** is a diagram of the default start screen and some of the other user interface screens that the user can navigate to from the start screen.

[0044] **FIG. 25** is the help screen and showing how navigation to a view new alarms screen can be accomplished.

[0045] **FIG. 26** shows the tree structure of a plurality of other screens which can be used to add target locations, annotate the target locations with text, voice or photo notes, add a text message, give commands to take a picture or find a picture file, record a voice message to be appended to the target, request position updates for all active buddies, map the positions of all active buddies or select particular buddies for mapping or requesting a position update.

[0046] **FIG. 27** shows a number of screens which can be displayed to map the position of a selected user with history and give information about the user as well as send short text messages, record and send voice messages, photos, Instant Messenger links, target positions, etc.

[0047] **FIGS. 28A and 28B** shows user interface screens created by Buddy Tracker software to create settings such as bread crumbs on or off, security codes for personal bread crumbs tracking and verification that a user is OK, set checkup timeout intervals, establish phone numbers and email addresses of other users to call in case of emergency in personal bread crumbs mode, add, change or delete group names, set the mapit screen radius, refresh rate and update setting, refresh time, delta position change for refresh.

[0048] **FIG. 29** shows the user interface screens to create a new buddy and showing the communication paths and accept protocol to do this. **FIG. 19**, is discussed more below, and is a representation of the screens and protocols to initiate and accept an instant buddy relationship.

[0049] **FIG. 30** is a diagram of the user interface screens for defining, deleting and using map rooms for closed proximity groups, open proximity groups, etc. For closed proximity group map rooms, listed user can set their preferences to automatically enter or be alerted that they in the Zone and manually decide to enter.

[0050] **FIG. 31** is a block diagram of the system for TalkControl to simplify cell phone walkie talkie operations.

[0051] **FIG. 32** is a flow diagram of a process a user of a walkie talkie enabled phone can initiate to joint a talk group to enable subsequent walkie talkie operations.

[0052] **FIG. 33** is a flowchart of the process the Rubicon server carries out to automatically delete a user.

[0053] **FIG. 34** is a flowchart of the process the Rubicon server carries out to allow a supervisor to add a user.

[0054] **FIG. 35** is a flowchart of the process for a supervisor to edit a user in a talk group.

[0055] **FIG. 36** is a flowchart of a process for a supervisor to delete a user from a talk group.

[0056] **FIG. 37** is a flowchart of a process for a supervisor to issue a token.

[0057] **FIG. 38** is a flowchart for the process of setting up preferences.

[0058] **FIG. 40** is a process flowchart of the process for a supervisor to create a group.

DETAILED DESCRIPTION OF THE PREFERRED AND ALTERNATIVE EMBODIMENTS

[0059] **FIG. 1** is a screen shot of a typical opening screen which would be displayed on a cellphone with the Buddy Tracker™ software enabled on the phone. **FIG. 2A** is a block diagram of the Buddy Watch system. A Buddy Watch or Rubicon server communicates with wireless devices **2** through **6** via the internet **9** and wireless carrier systems **7** and **8**. In the claims, the Buddy Tracker software is called the GPS position data sharing software application and it is resident on each of wireless devices **2** through **6**. Generally, communication between the handsets and the Rubicon (Buddy Watch) server occurs as follows. Each handset communicates data packets through its local cellular carrier network via TCP/IP compliant data packets encapsulated in cell system packets. The carrier network tower receives the packets and strips off the cellular encapsulation and forwards the TCP/IP packet to an appropriate gateways connected to the internet **9**. Routers in the internet route the packet to its destination, generally the Buddy Watch server **1**. The receiving server validates the content of the IP packet to authenticate the sender as a register Rubicon user and to verify that the sending phone EIN matches the phone EIN stored in the server. Once authenticated, the packet content is processed by the server. A response to the request in the packet is prepared using information from a database maintained by the Rubicon server and any associated map needed for the response is requested from a map server. The complete response is compiled, including any data needed to render a map on the recipient wireless device display and packetized into a TCP/IP packet and sent back to the originator of the request via internet routers and carrier gateways that couple the wireless carrier systems to the internet. The gateway of the carrier identifies the correct tower for the cell in which the recipient's phone is currently resident and the packet is encapsulated in a cell system packet and forwarded to the appropriate tower where it is transmitted wirelessly to the cell phone or other wireless device of the recipient. The wireless device then recovers the data in the TC/IP packet and the port address in the TCP/IP packet header causes the packet to be routed to the Buddy Watch software where it is processed.

[0060] **FIG. 2C** is a diagram of the start up screen. On startup, each handset starts its GPS sampler and the Buddy

Watch application program. If Main is pressed, the user is taken to the Nextel default page (or whatever other carrier is being used). If Mapit is selected, the user is take to the mapit page shown in **FIG. 2D**. **FIG. 2D** shows the mapit page where the positions of active users within the radius set up in the preferences of the center point XXX within radius YYY is shown. Scrolling down the mapit page below the map is the list of active users including those outside the radius. **FIG. 2E** shows additions options for manual refresh, etc. which can be reached by scrolling down the mapit page below the list of active users. **FIG. 2F** is a screen of active users. Color is used to highlight items. Scrolling to a user and pressing OK gives two options: recentering the map on the user and displaying details of that user.

[**0061**] The Buddy Tracker software creates the displays such as that shown in **FIG. 1** and other user interface displayed discussed elsewhere herein. **FIG. 24** is a diagram of the default start screen and some of the other user interface screens that the user can navigate to from the start screen. There is a help/emergency screen **15** which has a next command **19** which can be selected to take the user to the 911 screen **17** which can be used to take the user to a screen **21** wherein the user can select the type of help requested. **FIG. 25** is the help screen and showing how navigation to a view new alarms screen can be accomplished. **FIG. 26** shows the tree structure of a plurality of other screens which can be used to add target locations, annotate the target locations with text, voice or photo notes, add a text message, give commands to take a picture or find a picture file, record a voice message to be appended to the target, request position updates for all active buddies, map the positions of all active buddies or select particular buddies for mapping or requesting a position update. **FIG. 27** shows a number of screens which can be displayed to map the position of a selected user with history and give information about the user as well as send short text messages, record and send voice messages, photos, Instant Messenger links, target positions, etc. **FIGS. 28A and 28B** shows user interface screens created by Buddy Tracker software to create settings such as bread crumbs on or off, security codes for personal bread crumbs tracking and verification that a user is OK, set checkup timeout intervals, establish phone numbers and email addresses of other users to call in case of emergency in personal bread crumbs mode, add, change or delete group names, set the mapit screen radius, refresh rate and update setting, refresh time, delta position change for refresh. **FIG. 29** shows the user interface screens to create a new buddy and showing the communication paths and accept protocol to do this. **FIG. 19**, is discussed more below, and is a representation of the screens and protocols to initiate and accept an instant buddy relationship. **FIG. 30** is a diagram of the user interface screens for defining, deleting and using map rooms for closed proximity groups, open proximity groups, etc. For closed proximity group map rooms, listed user can set their preferences to automatically enter or be alerted that they in the Zone and manually decide to enter. For Open Proximity Group Map Rooms, anyone can join by opting in from their phone or from a sponsor's website. Upon entry, they can view and be viewed by all other members in the map room. Proximity rooms are useful to find and be found by friends attending an event.

[**0062**] In **FIG. 1**, area **10** discloses that the Buddy Tracker location sharing application software is active and is sharing the location of the phone with other members of a designated group. Area **12** indicates that parental status is active which means that the employer of the employee carrying the phone or the parent of the kid carrying the phone can see the location of the employee or kid if the phone is on. When parental status is active, the supervisory function cannot be turned off or evaded. This supervisory location sharing can be hierarchical such that an employer can see the location of all its employees, and each of the employees can be set up as supervisor of their children such that the employees can see the locations of their children, but the employer of each employee cannot see the locations of the children of each employee. The supervisorial relationships can be set up to define a matrix or web of Buddy List and supervisorial relationships, such as is illustrated in **FIG. 2B**.

[**0063**] In **FIG. 2B**, phone A has phones C and D on its Buddy List and is set up as the supervisor of those two phones. Phone B has phones A, G, F and E on its Buddy List and is set up as the supervisor of those phones. Phone H has phones E, J and I on its Buddy List and is set up to supervise those. Phone K has phone I on its Buddy List and is set up to supervise phone I.

[**0064**] Each of the phones in **FIG. 2** is coupled to the cellular carrier infrastructure in a conventional manner and can send phone calls or short text messages or email messages to any other phone including the cell phones represented by lettered circles in **FIG. 2**. **FIG. 12** is a block diagram of a typical prior art cellular system infrastructure in which the teachings of the invention in a peer-to-peer embodiment can be practiced. An area of the country is divided into several cells represented by circles such as **93** and **95**. Inside each cell is a transceiver tower, represented by blocks **94** and **96** which carries out time division multiple access or code division multiple access digital radio communications with cell phones in its cell. The cell phones or PDA are represented by autos **98** and **100**. Data recovered from the cell phone transmissions is transmitted to a central switching system **102** by data paths such as **104** and **106**. The central switching system **102** is coupled to a public service telephone network **108**.

[**0065**] Transmissions from one cell phone to another take place via the towers such as **94** and **96** and the central switching system **102**. For example, suppose cell phone **98** wants to send its GPS location data to cell phone **100** and cell phone **100** wants to send its GPS location data to cell phone **98**. The system of the invention uses some communication protocol such as XML, modified short text messages or other methods to send GPS location information to all cell phones on a Watch list. XML is a slimmed down version of SGML and enables Web authors to create their own tags so that they can more accurately capture the structure of their data. Because this structure can be read by SML-compliant browsers, the information encoded in these tags can be made available to programs such as Java applets or it can be displayed by formatting the XML tags with a style sheet.

[**0066**] In the preferred embodiment, the wireless devices in a group which has location tracking turned on all periodically send their GPS position data to all the other members in the group. The process for each wireless device to send its position data to any other wireless device in the group is as shown in **FIGS. 13A and 13B**. Basically, **FIG. 13** is a flow chart of the process of two or more cell phones

exchanging encrypted GPS position data. **FIG. 23**, discussed below, is a flowchart of another embodiment of a process to receive and process Buddy location update requests. The process of **FIG. 13** starts at step **110** with a request for a position update. In the preferred embodiment, this happens when a user of a Buddy Tracker phone uses his phone to make a request for a location update. In other embodiments, location updates can be requested automatically and periodically by the Buddy Tracker software on a device that is reporting its position. In other embodiments, a position update can be automatically generated by a device which is reporting its position to other members of a group whenever the position of the device has changed from its last reported position by a programmable amount. The requested position update may be sent to everybody on a selected Buddy List or just a single person's wireless device. In some embodiments, the position update is sent to some subset of persons on a selected Buddy List. Step **112** represents the process of looking up the addresses for all people on the selected Buddy List, a watch list, just selected individual or a subset of individuals from a watch list, as the case may be. Some embodiments may be limited to position updates on entire Buddy Lists.

[**0067**] Step **114** represents the process of reading the GPS position data from the built in GPS receiver of the phone (or the GPS receiver of the navigation system) and encrypting the position data.

[**0068**] In step **116**, cell phone **98** puts its encrypted GPS location data into a message according to the chosen communication protocol (assume short text message—SMS for short) and addresses the message packets to the one or more phones of the selected persons with which position information is to be shared.

[**0069**] In step **118**, the SMS message packets are transmitted to tower **94** using whatever physical layer protocol the cellular system uses such as TDMA or CDMA. The header of the SMS packets contains data indicating the payload data is to be sent to the Buddy Watch software of a particular cell phone and not to the inbox of the phone's SMS software. The payload data of these packets is the encrypted GPS position data. The physical layer protocol typically involves the following steps. First, the packets are disassembled into groups of bits of some predetermined size called codewords the size of which depends upon the particular configuration of the forward error correction software. The codewords are then interleaved to help defeat burst errors. Each codeword is then be encoded with error detection and correction bits such as by using Reed Solomon encoding. The codewords are then broken down into groups of bits called constellation points. The number of bits in each group depends upon the type of modulation scheme in use. In some embodiments, the groups of bits are then further encoded in a Trellis encoder.

[**0070**] The encrypted GPS position data packets would be addressed such that they would be routed in the cellular system to all the other wireless devices using the Buddy Tracker software which requested a position update. This is done by routing the packets to the cell transceivers in the cells in which the wireless devices which require updates are currently registered, as represented by step **120**. For example, if cell phone **100** in **FIG. 12** is on the Watch list or is being automatically updated or has requested a position update manually, it will have transmitted a packet to transceiver **96** indicating it needs a position update and, when the wireless device entered cell **95**, it will have exchanged packets with transceiver **96** to achieve synchronization with transceiver **96** and to register in cell **95**. Each wireless device that is registered in a cell will have done this, and the transceivers in each cell will communicate with the central switching system **102** to indicate which wireless devices are registered in their cells. Therefore, the routers in central switching system **102** will know which wireless devices are registered in each cell and will know which wireless devices are to receive position updates. Step **120** represents the process of receiving the signals from each wireless device that are modulated with constellation points that contain the data of packets that contain GPS positions, recovering the data from the constellation points and doing error detection and correction and recovering the GPS position data packets. These packets are then routed to the central switching system which uses the destination addresses in the packets and its routing tables to route them to the transceivers in whatever cell or cells the wireless devices that require position updates are registered. There, the packets are disassembled, encoded with error correction and detection data, and assembled into symbols or constellation points in whatever type modulation (QAM, QPSK, etc.) is being used and transmitted to the wireless device. This happens for every wireless device on a watch list or which has requested a position update.

[**0071**] At each wireless device which receives the signals modulated with the constellation points bearing the GPS position data, the data of each packet is recovered and the packet is reassembled, as symbolized by step **122**. The header data of the packet (the port number in the case of TCP/IP) packets causes the wireless device to pass the packet to the Buddy Tracker software which is monitoring a particular port, step **124**. When a packet is passed to that port (or just the payload data thereof), the payload data is decrypted and the position data recovered, step **126**. Step **126** also represents the process of reading the header data of the incoming packets and determining which other member of a buddy group sent the position update so that the position information for the proper member of the Buddy Group is displayed. The position data is then used to display the position of the other party in the group who sent the packet, and, if the user gives the "map it" command, the position data will be converted to a waypoint on a displayed map so as to graphically display the position of the wireless device which sent the packet.

[**0072**] Step **128** represents the process of the device which received the position update encrypting its own GPS position into short message or email packets addressed to the other members of a Buddy Watch group or to a single other wireless device. These packets are then sent to all the other wireless devices by the same process described in steps **116**, **118**, **120**, **122** and **124** of **FIGS. 13A and 13B**, as represented by step **130**.

[**0073**] **FIG. 23** is a flowchart of another embodiment of a process to receive buddy location update requests and process them. Step **220** represents requesting a buddy location update. Addresses of all persons on the buddy list or just a selected buddy are located in step **222**. Message packets are generated in **224** addressed to the selected Buddy List or individuals, and encrypted position data is put in them. A

request is sent—**226**, and these packets get routed to the Rubicon server—**228**. The initiator and recipient are authenticated—**230**, and the packets are forwarded to the recipients via the cell system. The cell system forwards the packets to whatever cell each recipient is in—**232**. The packet arrives and causes a GPS position request to be made in each wireless device—**234**. The device captures its current GPS position, and encrypts it and packetizes the GPS position in the payload portion of a packet addressed to the Rubicon server with information as to the requester in some embodiments. The packet is then sent with a timestamp to the Rubicon server for forwarding to the requesting Buddy—**238** and **240**. The Rubicon server authenticates the initiator and the recipient and forwards the packet to the initiator via the cell system—**242**. The cell system forwards the packet to the cell where the initiator recipient is located—**244**. The initiators cell phone receives the packet, and recovers the timestamp—**246**, and reads the packet header and port number of the GPS information and uses the port number to route the packet to the Buddy Tracker software—**248**. The Buddy Tracker application program on the initiator's cell phone receives and decrypts the GPS information from the packet, displays the position of the Buddy, and uses information in the header to determine which other Buddies sent position updates—**250**. The wireless device of the initiator responds to the position update of each Buddy by sending a reply packet with the encrypted position of the initiator—**252**. Each requested Buddy device updates the Rubicon server and the wireless device of the requesting buddy with its position—**254**.

Out of Coverage Update Response

[**0074**] In some embodiments, when a wireless device requests an update from another wireless device, and the other wireless device is out of cellular coverage, a timeout occurs. When a timeout occurs without receiving a position update, the wireless device expecting the update changes its display to yellow or some other color for the wireless device which is out of coverage. The Buddy Watch system only works when the phone is on and in a cellular coverage area.

Buddy Watch Modes

[**0075**] **FIG. 14**, comprised of **FIGS. 14A, 14B** and **14**C is a flowchart of processing of an embodiment that implements the several modes described below. The steps that are numbered with like numbers to steps in **FIG. 13** represent the same functionality.

[**0076**] 1) Disable:

[**0077**] The Buddy Watch application can be disabled by the user. When disabled, the wireless device does not share its GPS position data with any other wireless device so no other buddy can see your position. There is an exception for the parental monitoring function. When parental monitoring is turned on, as symbolized by line **111** in **FIG. 14A**, the wireless device always shares its GPS position and cannot be disabled. The disable functionality is represented by test **113** in **FIG. 14A** which determines when a position update is requested whether the Buddy Watch function has been turned off. If so, processing proceeds to test **115** which determines if parental monitoring is turned on. If not, processing proceeds back to step **110** along line **117** and GPS position sharing does not happen. If the Buddy Watch function has not been disabled, processing proceeds to step **112** to look up the addresses of the wireless devices to which the position update is to be sent. When the Buddy Watch application is disabled and it has been operating and parental monitoring is not turned on, a final position update is sent is sent to those wireless devices on the current Watch List (the wireless devices which are active and monitoring each other's positions).

[**0078**] 2. Enable:

[**0079**] This is the normal mode of operation. Buddies can be added or deleted from the Watch List at any time. Any wireless device that is operating and on the Watch List can find the location of any other device on the Watch List by issuing a position update request. If a buddy is removed from a Buddy List, he or she is not able to receive position updates. Multiple lists can be joined to form a group.

[**0080**] 3.

[**0081**] Supervisor—Parental/Corporate Code: In this mode, as long as the wireless device of the worker or child is on, the supervisor will be able to monitor position by GPS position updates. The worker or child will not be able to disable the Buddy Watch feature nor remove the parent or supervisor from the Watch List. Only the supervisor or parent will have the necessary password to remove himself from a Buddy List or watch list.

[**0082**] 4. Buddy Lists:

[**0083**] This is the normal mode of operation. Buddies can be added or deleted from a list at any time. If a buddy is off a Buddy List, he cannot receive position updates from any other wireless device on the list. Multiple Buddy Lists can be joined into a group and entire groups may be enabled and disabled. Workgroup lists are lists of buddies which need to be aware of the location of their coworkers during working hours but not after. Buddy Lists or Buddy Groups are a means to have a single icon, folder or some similar graphic user interface (GUI) mechanism to represent a list of people and enables single commands to turn on or turn off tracking of a group of people. If a folder is used for each Buddy Group, a drop down list showing the specific names and locations of each person on the list can be displayed when the folder or icon representing the group is selected. If an icon is used, the Buddies would be groupled in and shown on the phone display. Groups receive a color on the GUI and the members of the group are connected by a translucent shaped outline encompassing all the points representing positions on the mapit display. If the group is spread too far apart to be shown on a single mapit display, then the shaped outline for the group is not shown and on the specific color coded Buddy positions that can be shown are shown. If the mapit display is zoomed out, the translucent group outline returns when all members of the group can be shown on a single screen. Buddies that are in multiple groups are colored a non group color or the color of any of the displayed groups. If groups overlay and when Buddies that are in two active groups are shown, the translucent outline shall overlap as needed, and only cover Buddies that exist in both lists.

[**0084**] 5. Instant Buddies:

[**0085**] Instant Buddies can be created when a call is placed between two cell phones, phone enabled PDAs or other wireless Buddy Watch enabled devices. **FIG. 15** is a flowchart of the process of establishing an Instant Buddy

Relationship. The first step is **132** where the wireless device places or receives a call from a Buddy Watch enabled wireless device to or from another Buddy Watch enabled wireless device. The two users such as a stranded motorist and a tow truck driver discuss the situation and decide to establish an Instant Buddy releationship. After the call is established, and the two agree to allow it, the two wireless device users can click on the Instant Buddy menu choice in the Buddy Watch menu, as symbolized by step **134**. The wireless devices then each display an Instant Buddy Setup screen like that shown in **FIG. 11** (step **136**) and fill in the appropriate data (fields **84**, **88** and **86**) in step **138**. Both users then indicate their acceptance (field **92** or deny the relationship (field **90**), or in some embodiments, only the recipient of the call needs to accept or deny the relationship. Once the Instant Buddy relationship is accepted, the two wireless devices start exchanging position data (step **140**). After 24 hours, or some other time set in the Instant Buddy preference menu (timeout checked in step **142**), Instant Buddies are discarded (step **144**). The Instant Buddy preference menu allows the time period for the Instant Buddy relationship to be set to something other than the default value of 24 hours.

Preferred Instant Buddy Setup Process

[**0086**] **FIG. 16** is a block diagram of a typical cellular system coupled by a gateway and a Wide Area Network such as the internet to a Buddy Watch server to provide the infrastructure of the invention. The cellular system shown in **FIG. 16** is typical and has the same structure and operation as the cellular system of **FIG. 12**. What is new is the connection between the central switching system and a Buddy Watch server **146** through gateway **148**. The purpose of this will be made clear in the following discussion of the preferred Instant Buddy setup process.

Buddy Watch Server Functions

[**0087**] The Buddy Watch server's main function is to serve maps to the cell phones registered in the Buddy Watch system and implement GPS position data exchanges between itself and the phones on a buddy list to enable members of a buddy list to view the locations of other members of the list. In some embodiments, the Buddy Watch server also downloads application software as needed to phones registered in the system as the phones send packets to the Buddy Watch server indicating a particular command has been given which requires an application program on the phone which is not present.

[**0088**] In the preferred embodiment, the Buddy Watch server runs all the application programs on the server and just sends pages to be displayed on the phone to solicit the user to enter data needed to implement a function.

[**0089**] If the phones had as much memory as the Personal Digital Assistants, the application programs could be loaded and run on the phones themselves.

[**0090**] Other functions of the Buddy Watch server are: setup of user profiles, billing and database access and maintenance.

Purchase/Payment Activate Deactivate Key

[**0091**] The functions of the Buddy Watch server will be made clear in discussions which follow. But one of its functions will be to manage activate and deactivate codes. The Buddy Watch application will be a service which a cellular carrier offers on a subscription basis. When a subscriber buys a Buddy Watch enabled phone, he will be issued an activation code and the Buddy Watch server will also be given the activation code. This activation code will be kept in active status as long as the subscriber has paid for the service. Subsequent communications of packet data between the Buddy Watch server and the phone such as downstream position updates of positions of buddies on a Buddy List, receipt of phone position for use in updating other buddies on a Buddy List, etc. will only be enabled as long as the activation code is in active status. When the subscriber stops paying for the service, the activation code will be changed to a deactivation code status, and subsequent communication between the phone and the Buddy Watch server will be impossible. The Buddy Watch server implements this functionality by checking the activation code status each time before communication with a phone is carried out.

[**0092**] The Buddy Watch application is downloaded via the internet for subscribers who do not already have it on their phones. The customer receives an activate code to key into the phone, or an activation application on the Buddy Watch server receives confirmation of the purchase and automatically sends the activate code to the phone/Personal Digital Assistant and receives back a confirmation. Each month, payment for the service is required. Failure to make the payment results in an application receiving a request to deactivate the Buddy Watch application on the phone/PDA. A deactivate code is sent and a response is received back confirming the phone application has been deactivated. Further attempts to use the application are met with a simple display indicating the service subscription has expired.

[**0093**] The protocol to activate and deactivate the Buddy Watch application is secure in the preferred embodiment.

[**0094**] **FIG. 17** is a flowchart of one embodiment for an Instant Buddy Setup process. **FIGS. 18 and 19** are diagrams of some of the user interface display screens involved in the Instant Buddy Setup process. All three figures will be referred to in the following discussion. In the preferred embodiment, the Instant Buddy relationship is set up in the following manner.

[**0095**] 1) An initiator selects the Instant Buddy menu choice options in step **150**. This is done by selecting menu option **151** of screen **153** in **FIG. 18**. This causes a transition to display screen **155** on the initiator's device where the user selects menu option **157**. This causes a transition to screen **159** on **FIG. 19** which is the Instant Buddy setup screen. This screen shows the initiator's phone number, Instant Buddy ID and Screen ID in autofilled fields **161**, **163** and **165**, respectively (step **152**). There is also a timeout field **167** which the initiator can set to some time if the default time of midnight is not acceptable (step **154**). After filling in the timeout value, the Initiator clicks Next in field **169**.

[**0096**] 2) Instant Buddy request packets are generated and sent to the cell transceiver of whatever cell the wireless device of the initiator is registered. These packets contain data which identifies the initiator and the recipient (proposed Instant Buddy) and are addressed to the IP address of the Buddy Watch server **146** in **FIG. 16**. The packets are recovered by the cell

transceiver, sent to the central switching system **102** and routed from there to gateway **148** where they are routed over wide area network **147** to the Buddy Watch server (step **156**).

[**0097**] 3) Buddy Watch server authenticates the initiator and the recipient from data in the packet as a Buddy Watch subscribers. If either is not a Buddy Watch subscriber, the server blocks the transaction by not forwarding the packets to the recipient. Assuming both are subscribers, the server forwards the Instant Buddy request packets to the recipient's wireless device and these packets get routed in the cell system (step **160**) via the gateway, central switching system and cell transceiver of the cell in which the recipient's wireless device is registered.

[**0098**] 4) The proposed Instant Buddy's wireless device receives the packets and displays an Instant Buddy Request screen (step **164**) like that shown at **171** in **FIG. 19** (step **162**). This screen shows the phone number, Instant Buddy ID and Screen ID of the Initiator so the recipient knows who has requested the Instant Buddy relationship. The recipient can either accept or deny the relationship using commands displayed at **173** and **175**.

[**0099**] 5) If the Instant Buddy relationship is accepted (step **166**), processing proceeds to step **168** where an accepted packet is sent back to the initiator's wireless device. If the Instant Buddy relationship is denited, step **170** sends a denied packet back to the Initiator device and the process is over (step **172**) save for a display on the Initiator device that the Instant Buddy relationship has been denied.

[**0100**] 6) When the accepted packet arrives at the Initiator device, the device shows an Instant Buddy accepted screen as shown at **177** in **FIG. 19** (step **174**). This screen shows the phone number, Instant Buddy ID and Screen ID of the recipient and provides commands to accept or cancel the relationship at fields **179** and **181**.

[**0101**] 7) If the Initiator accepts the Instant Buddy relationship (step **176**), step **178** occurs where an accepted packet is sent back to the Recipient through the Buddy Watch server.

[**0102**] 8) The Buddy Watch server records the existence of the new Instant Buddy relationship (step **180**), and both wireless devices start sending their GPS position data in packets addressed to the Buddy Watch server. The Buddy Watch server stores the position data from each wireless device and forwards the packets to the other device for updating of their displays. In the preferred embodiment, the Buddy Watch server pulls an appropriate map from the MapQuest server **149** in **FIG. 16** based upon the GPS position data of the Initiator and sends that map and the GPS position data in packets addressed to the Recipient. The Buddy Watch server then pulls an appropriate map from the MapQuest server based upon the Recipient's position, and sends that map and the Recipient's GPS position to the Initiator. Each wireless device then displays the position of the other Instant Buddy on the map provided by the Buddy Watch server.

[**0103**] An alternative Instant Buddy setup process is described next:

[**0104**] 1) A call from one wireless device to another is initiated;

[**0105**] 2) After agreeing to establish an Instant Buddy relationship, the inititator clicks on the Instant Buddy menu option;

[**0106**] 3) This causes an Instant Buddy Setup screen to be shown on the initiator's device which has a first field which is autofilled with the initiator's phone number, a second field which is autofilled with an Instant Buddy ID, and a third field which is autofilled with a Screen ID for the Instant Buddy (this screen ID is a three digit number which will be displayed with the position of the Instant Buddy and is shorter than the Instant Buddy ID);

[**0107**] 4) The initiator fills in a timeout period for the Instant Buddy relationship or accepts the default value of midnight and clicks a Next command;

[**0108**] 5) The recipient's wireless device receives the instant buddy request and displays an Instant Buddy Request screen that shows the initiator's phone number, Instant Buddy ID and Screen ID and displays an accept or deny command;

[**0109**] 6) The recipient either accepts or denies the Instant Buddy relationship;

[**0110**] 7) If the recipient accepts the Instant Buddy relationship, this fact is communicated to the initiator's wireless device which then displays a screen which displays the recipient's phone number and the recipient's Instant Buddy ID and his or her Screen ID and displays an accept or deny command which the initiator can click on;

[**0111**] 8) If the initiator selects the accept command, both wireless devices start exchanging GPS position data, but they do not if the initiator decides to deny the Instant Buddy relationship.

[**0112**] **FIG. 22** is a flowchart of the preferred embodiment for the instant buddy setup process. The initiator selects the instant buddy setup process—**250**, and enters the phone of the proposed new instant buddy—**252**. Initiator fills in timeout period—**254**, and instant buddy packets get routed to the Rubicon (Buddy Watch) server through the cell system—**256**. Rubicon server authenticates initiator and recipient and forwards packets to cell system—**258**. The cell system routes packets to the cell where the proposed new instant buddy is—**260**, and the proposed instant buddy receives a messages on her wireless device and displays the instant buddy request screen—**262**. The instant buddy sees the initiator buddy ID, screen ID and, optionally, his phone number—**264**, and accepts or denies the relationship—**266**, **270**, **272**. If accepted, a packet is sent back to the initiator wireless device—**268**, which causes the initiator's device to show an Instant Buddy accept screen with recipient phone number, buddy ID and screen ID which the initiator must OK to establish the relationship—**274**. Steps **276**, **278** and **280** handle the acceptance or rejection. In **282**, if accepted, the Rubicon server records the new instant buddy relationship and both wireless devices start sharing location information with the Rubicon (X One) server where it is stored

and forwarded to the other Instant Buddy. In **284**, the initiator's device shows Instant Buddy Accept screen. Steps **286**, **288** verify the phone is collecting GPS data using the GPS sampler program.

User Interface Displays for Buddy Lists

[0113] **FIG. 3** represents a display in the user interface which shows individuals on the phone's Buddy List as well as a group of buddies which has been given the name Tennis Team. In all the user interface screen shots in the figures of this patent application, a cross hatched area indicates an active status and is typically colored green on the phone display. For example, the cross hatched buddies in column **14** indicate that buddy's location sharing is active and his position can be seen if the user clicks on that buddy using whatever navigation or pointing mechanism that is built into the cell phone user interface.

[0114] The Buddy Tracker software also has the ability to set up instant buddies with, for example, tow truck drivers. Display area **16** shows an instant buddy entry for an instant buddy named Inst**01**. For example, the user's car breaks down. The user calls a towing service, and find out the two truck driver has a cell phone with Buddy Tracker on it. The user dials the tow truck driver's cell phone and requests to be an instant buddy of the tow truck driver's phone. His phone is then set up as an instant buddy on the user's phone. After both phones are set up as instant buddies, each phone shows the location of the other phone on its moving map. This allows the tow truck driver to find the user tow truck customer and the user customer to know where the tow truck driver is.

[0115] **FIG. 4** shows another user interface display that results from selecting the tennis team entry **18** on the Buddy List of **FIG. 4** and then clicking on the Tracie entry. When the Tracie entry is clicked, the information in column **20** appears showing her full name, position, the time of her last fix, her distance from the user and her speed. A green status (cross hatched) means a buddy has his phone on with location sharing turned on and the phone is within range. A yellow status for a buddy (stippled) means the buddy was active and had his location sharing turned on, but contact with him has been lost for one reason or another. A darker green status (double cross hatched), means the buddy is active and has his location sharing turned on but his is out of the immediate area that can easily be shown on the phone's map display. For example, suppose most of the tennis teach group are in the Northern California area, but one member of the group is in Los Angeles. If the member in Los Angeles has his phone turned on with location sharing on, his entry in the tennis teach list will be shown as dark green meaning his position cannot be mapped.

[0116] The mapit function shown at **22** in **FIG. 4** is a function that can be invoked to map the location of Tracie Saka on the phone's display. If Tracie is within range, and the Mapit function is clicked, a display such as the one shown in **FIG. 5** is rendered on the phone's display showing the general area and showing Tracie's position at **24** with a text box **26** superimposed on the map with Tracie's name rendered therein.

[0117] **FIG. 6** is a user interface display showing a map rendering with the location history of a selected member of the tennis team rendered thereon. This display is rendered when the Mapit with History function **28** in **FIG. 4** is selected. This display shows the path Tracie took to get to her current location by way of waypoints **30**, **31**, **32** and **33**. In some embodiments, when a user wishes to record a waypoint for their current position, a command can be given that causes the current position of the phone to be reported and saved as a waypoint on the Buddy Watch server **146** in **FIG. 16**.

[0118] In other embodiments, a particular position such as the phones current position or a position selected by moving crosshairs on a map display on a phone can be sent as a meeting place to all buddies on a Buddy List. When such a command is given and a Buddy List is selected, the position of the meeting place and the designated Buddy List is put into packets addressed to the Buddy Watch server **146** and transmitted thereto where the information is stored. The meeting position is then packetized in packets addressed to all the buddies of the designated Buddy List, and those packets are addressed to the phones of the buddies on the designated Buddy List and sent thereto.

[0119] Referring to **FIG. 7**, there is shown a display of a screen showing positions and status of members of a selected group. In this example, Tracie and Karen's positions are known and their name boxes in the left column are displayed in some color such as green indicating they are within cellular coverage and their positions are known. On the other hand, Dean's name box is shown in some other distinctive color such as yellow (represented by single cross hatch) to indicate contact with Dean has been lost. This happens when a user travels outside cellular coverage. Because Dean's name box is currently selected by the cursor, the settings column has the last known information about Dean also displayed in the distinctive color and represented by a single cross hatch. These boxes show Dean's last known position fix time (box **34**), his full name (box **36**), his last known distance (box **38**), and his last known direction, latitude, longitude and speed (boxes **40**, **42**, **44** and **46**, respectively).

[0120] **FIG. 8** is a screen display showing what is displayed when Dean is selected and the Mapit command in box **48** is given by double clicking on the box or by any other means. When this map it command is given, Dean's last known position is displayed with a circle of a distinctive color (such as red), as illustrated at **50**.

Instant Buddy Display with Mapit Position Mapping

[0121] An instant buddy relationship also allows the location of the motorist, lost or injured hiker or other user to appear on the tow truck or ambulance driver's cell phone mapit display.

[0122] **FIG. 9** is an instant buddy display showing the instant buddy position. This display can be selected after an instant buddy relationship has been set up. This display shows the ID of the instant buddy in box **52**, the time of the last position fix in box **54**, the distance to the instant buddy in box **56**. The direction to the intant buddy, latitude and longitude and speed of the instant buddy are shown in boxes **58**, **60**, **62** and **64**, respectively. If the user selects the mapit command in box **66** or the mapit with history command in box **68**, the phone display will change to a display like that shown in **FIG. 8** or **FIG. 6**, respectively, with the current position of the instant buddy shown and the prior positions shown if the history option is selected.

[0123] Alternative Instant Buddy Setup Process:

[0124] To set up an instant buddy relationship, the phone is given a command to display an instant buddy setup screen like that shown in **FIG. 10**. The display of **FIG. 10** is used to establish an instant buddy ID in box, give the instant buddy a caller ID in box **72** (the instant buddy's caller ID or phone number is used by default). Box **74** is used to establish a timeout period **74** at the end of which the instant buddy relationship is automatically terminated. The timeout period can be set to any interval in some embodiments, or to some selected interval from a drop down menu. Box **76** is used to establish the carrier the instant buddy is using. A cancel command is shown at **78** and a request command is shown at **80**.

[0125] To start the instant buddy relationship, the request command is issued after the other boxes are filled in. Typically, a stranded motorist or hiker will call a tow truck or 911 and get the caller ID and carrier of the tow truck driver or rescuer. The stranded motorist or hiker will then enter this information in boxes **72** and **76**. Box **70** shows an instant buddy ID which is automatically assigned by the system. After entering the information, the request command shown at **80** is selected. The screen of the phone of the rescuer's phone will then change to the display shown in **FIG. 11**. The information the requester filled in on the **FIG. 10** screen will appear in boxes **82**, **84**, **86** and **88** on the stranded motorist or hiker's phone as well as on the instant buddy's phone (the tow truck or 911 rescuer). Commands for Denied and Accepted will also appear at **90** and **92** of the instant buddy's phone. If the instant buddy desires to accept the instant buddy relationship, he or she selects the accept command, and the tracking of the two instant buddies positions will begin. Upon acceptance of the instant buddy relationship, each instant buddy's phone displays changes to the display shown in **FIG. 9** from which the mapit or mapit with history command can be issued.

Corporate Supervision Setup Via Passcode

[0126] Corporations that wish to monitor the locations of their employees can use the system of the invention by using a corporate passcode. In this mode of operations, corporate employees are set up as a group with their supervisor as one member of the group. Each employee in the group can have his own buddies but he cannot delete the supervisor from the group. Only the supervisor can delete himself from the group of each employee's phone since only the supervisor has the passcode to change the group's members to delete himself. In one embodiment, the location information sharing is unidirectional from employees to supervisor but each employee can see the location of other employees on their phones but not the location of the supervisor. In this embodiment, the location sharing can be configured to be on only during working hours Monday to Friday. In other embodiments, the employees can see the locations of the supervisor as well as the locations of the other employees.

Timed Updates

[0127] The teachings of the invention contemplate doing position updates peridodically at configurable intervals as well as a configuration option to do periodic updates as well as an update every x miles if a buddy in a group being monitored moves more than x miles between periodic updates. In some embodiments, the velocity at which a Buddy is moving or the amount of distance since the last update a Buddy has moved controls the frequency of the updates. Timed updates are handy for parents to monitor the positions of their children to make sure they do not move more than X miles from their home base. Position updates can be requested by a member of a Buddy List for position updates from the Buddy Watch server. The server receives positions reports from all the Buddy Watch phones registered with it and stores them and knows the Buddy Lists for each phone. When a request for a position update is received, positions of all the buddies on Buddy Lists of which that phone is a part will be transmitted as packets addressed to all the phones on all the Buddy Lists of which the requester is a part. In alternative embodiments, the position updates will be sent for all members of all Buddy Lists of which the requesting phone is a part, but will only be sent to the requesting phone to avoid excess network traffic. In other alternative embodiments, the requesting phone can designate a particular member of a particular Buddy List and request an update only for the position of the designated buddy. The position update will be sent only to the requesting phone.

Follow Me Mode

[0128] In some applications such as construction sites with large construction crews and one supervisor, it is useful for everybody working on the job to be able to find the supervisor but the supervisor does not care where anybody else is. In embodiments with this capability, the supervisor turns on the follow me mode, typically my making a menu selection. This causes the supervisor's position to be reported to the Buddy Watch server on a regular basis in packets that have information in their headers or elsewhere which indicate they are Follow Me packets and which designates to which Buddy List this information is pertinent. The Buddy Watch server takes these position updates and packetizes them into packets addressed to each of the phones on the designated Buddy List and sends those packets to the Buddy List phones. Position updates from the phones on the Buddy List are not sent to the supervisor phone or any of the other phones on the Buddy List.

[0129] This Follow Me mode can also be done in a blind code mode. This means that the supervisor does not need to list everyone on his buddy list. This is an "open channel" mode. Any "follower" who wants to track the position of the supervisor only needs to list the supervisor's name and phone number on a buddy list of the "follower" phone. The supervisor enters a blind code in follow me mode, and this code is published to all phones that have Buddy Watch software. This blind code entry and publication allows any follower to enter the blind code in a buddy list on the follower phone and thereafter to receive the supervisor's position reports. This entry of the blind code will give any follower the ability to receive position reports from the supervisor's phone, and the supervisor will not have to approve each buddy individually. This can be a great convenience since on some job sites, there may be hundreds or thousands of workers. The follower phone sends a packet to the Buddy Watch server telling it that the follower phone is in follow me mode for the particular supervisor. This causes the Buddy Watch server to send position reports it receives from the supervisor phone to the follower phone, but the server does not send position reports from the follower phone to the supervisor phone. The follower phone does not

DK Crown Holdings Inc.
Ex. 1012, Page 59

send position reports to the Buddy Watch server when in follow me mode. Disabling, removing or changing the blind code, stops Follow Me mode.

Buddies Only Mode

[0130] The Buddies only mode differs from the All On Follow Me mode and the Blind Code Buddies modes in that position reports are only received from Buddies on a specifically named Buddy List with specifically named Buddies. No blind code Buddies or Instant Buddy position reports can be received in this mode.

Waypoint Store Mode

[0131] This mode is useful for parents to monitor the travels of their children. In this mode, the child's phone periodically reports the child's position, and the parent can have the position reports sent to his phone (or computer in some embodiments). In some embodiments, position alert data can be configured to send an alarm signal to a parent if a child's position gets too close to a specified location or too far from the home location or some other location.

Request Update

[0132] This mode provides a specific user to request an update on the position of a specific Buddy. The requesting phone sends a request packet to the Buddy Watch server identifying itself and requesting a position update on a specified Buddy. The Buddy phone need not do anything other than do its normal operation of sending position updates to the Buddy Watch server. The update request causes the Buddy Watch server to provide a two-way update so that the requesting phone's location is sent by the Buddy Watch server to the Buddy phone and the Buddy phone's location is sent by the Buddy Watch server to the requesting phone. If the requester is part of a group, then the reply as to where the Buddy is is sent to all phones in the group.

Timed Update

[0133] In this mode, periodic updates from the phone of a person such as a child or other person being cared for can be periodically sent to a list of parental or other supervisor destinations such as the parent's cell phone or email address. The sender phone may also be configured to send its location periodically to all others on a list. Updates on position can be every 15 minutes or some other configurable interval. In addition, each supervisorial user can request an update and the updates will be sent to every phone on the supervisorial list. If a phone on the list is not available, the update will indicate that no update is available, change the display to yellow and the status to unavailable but keep displaying the last waypoint.

Personal Bread Crumbs

[0134] This is an emergency feature which allows tracking down children or elderly people who are no longer responding to inquiries sent to their phone. This mode is useful for children who do not want to be watched but want a safety line to their friends and family in case something happens. A user who wishes to use this feature sets up their profile such that the Buddy Watch server checks in with them via their Buddy Watch enabled phone on a daily basis to determine if all is OK. The user must enter their secret code to confirm that all is OK. The phone prompts them to enter this code, and a certain number of prompts can be ignored before the system raises any alarms.

[0135] **FIG. 20** is a flowchart of one embodiment of the process of enabling the personal bread crumbs mode and how it works. Step **200** represents the process of enabling this mode. Typically, this is done by the user in selecting a menu command, but in some embodiments, it may be permanently configured to be on by the phone manufacturer. When this mode is enabled, the phone stores waypoints of the position of the holder of the phone periodically (step **202**). The phone does not send the waypoints to anybody, but it does send data or a message to the Buddy Watch server that the the personal bread crumbs feature has been enabled (step **204**), so the Buddy Watch server starts a timer (step **206**). The purpose of starting this timer is to establish intervals at the end of which an "Are you OK?" message will be sent to the phone which is in Personal Bread Crumbs™ mode.

[0136] Step **208** represents the process of monitoring the timer for a timeout event. This may take the form of a hardware or software interrupt. When a timeout occurs, the Buddy Watch server sends an inquiry to the phone inquiring if the user is OK (step **210**). The phone then displays the "Are you OK?" message, and the user either enters his or her secret code to say they are OK or does not. If the user does not respond, processing proceeds back to step **206** to start the timer again as the user may simply be busy, have their phone off, be asleep, etc. However, after a configurable number of attempts to establish contact with no response, step **216** will conclude that the user may be in trouble and need rescue. In that case, processing is vectored by step **216** to step **218**. In step **218**, the phone is commanded by the Buddy Watch server to send distress messages out to predetermined phone numbers (five in the preferred embodiment) and/or email addresses. The voice mail message may indicate to check email for details. The email contains a content of a position report file that contains all the waypoints since the last OK was received. If there are no stored waypoints, at least one set of stored waypoints previously recorded are sent. The waypoints all provide latitude, longitude, date and time of recording.

[0137] The personal bread crumbs profile includes:

[0138] 1) a list of emails to which messages should be sent;

[0139] 2) a list of phone numbers to which the prerecorded voicemails are to be sent;

[0140] 3) frequency of OK confirmation user need to agree to (default is daily at noon);

[0141] 4) the text of an email to describe the emergency situation to readers and should include the mobile phone number, home phone number, work phone number, home address and other pertinent information;

[0142] 5) whether or not auto attachment of waypoints to emails is to be carried out;

[0143] In an alternative embodiment, step **218** represents the Buddy Watch server itself sending out the distress messages. In some embodiments, the distress messages are prerecorded voicemail messages which indicate the user may be in trouble and giving instructions to the recipient how to retrieve the position reports from the Buddy Watch

server. Step **218** also represents the process of the phone sending its GPS position waypoints to the Buddy Watch server. In some embodiments, the prerecorded voicemails are sent to pre-determined phone numbers and the predetermined emails are sent to predetermined email addresses and include the GPS position reports in the text of the message. The email messages at least will include the personal breadcrumb position reports. These messages indicate to the recipients that there may be trouble and that they should start looking for the person who owns the phone.

[**0144**] **FIG. 21** is a flowchart of another embodiment of a process to establish and use personal bread crumbs mode. In step **201**, the user enables the bread crumbs mode, and in step **203** the mobile phone contacts the Buddy Watch server (also called the Rubicon server herein) and informs it that personal bread crumbs mode is on. GPS sample data is collected (**205**) and the server is contacted to start the "Are you OK" timeout interval (**207**). Timeout causes the phone to display an are you OK message (**209**). Steps **211** and **213** handle the situation where the user does not enter a secret code and retries. Step **215** represents the Rubicon server response if the user does not respond to the are you OK message properly and timely, said response involving sending whatever distress messages are set up in the preferences file. GPS location samples and timestamps are included in the distress messages (**217**) and the messages are sent to the users listed in the preferences file (**219**).

Relational Database Compatibility

[**0145**] The Buddy Watch server is configured and programmed to be compatible with business applications where the customer may desire to find individuals based upon their capabilities, certifications or the equipment they are carrying. By making the Buddy Watch fields of the Buddy Watch database available for search and/or integration into other business databases, a company such as a service based organization can determine which individuals have the proper certification to work on a specific problem and/or who have the appropriate tools and where those individuals are located relative to a site to which the company wishes them to be dispatched. The Buddy Watch server is programmed to provide information about the subscribers and their locations in a format which is compatible with the other business database structures of customers who are interested in having this data. Each position update received by the Buddy Watch server then is exported and automatically updates the customer database. This can be done over the Internet or over a dedicated local area or wide area network.

Radar Inclusion

[**0146**] The radar inclusion mode is a mode which allows police departments or fire departments or any other emergency response type organizations to instantly expand their buddy lists to predetermined lists of all available personnel. This is useful when it is necessary to know the whereabouts of persons to assist in an emergency situation or other situation. This feature may be used by police or other groups where the formation of a group may vary throughout the day. This feature can be used in conjunction with standard groups. How this feature differs is that a user does not need to be identified and only when the user comes within the "radar" range or radius do the user get included within the radar inclusion group.

[**0147**] City, County, State or Federal law enforcement or other agencies can offer two capabilities with radar inclusion. The first capability is to send an alert with a fixed target or to add a moving target to any individuals or groups without any input from the field officers. The target could be a suspect on the move. The target affords all the officers a better view of what is going on. The second capability allows the agency using the radar inclusion feature to "light up" the positions of other individuals or groups of individuals on a mapit display so that one or more officers/firemen responding to an emergency can see the positions of possible reinforcements relative to their position. This is useful when groups that normally do not work together such as perhaps the fire and police need to work together. Details about each Buddy which is lit up on the mapit display can be sent to any other Buddy in need thereof by a command to the Buddy Watch server issued by the controlling personnel of the agency.

[**0148**] In the instant messaging protocol packets transmitted from a phone to the Buddy Watch server, there is a field that can be left blank or a prefix can be put in. An agency using radar inclusion can put a code in this field and then all Buddy Watch phones/PDAs operating in radar inclusion mode is sent these packets and retains the Buddy whose information is in the packets in a group. This new group can be retained for a user programmable time up to 24 hours beyond the radar inclusion Buddy display disappearing.

[**0149**] The Buddy Watch server determines if a matching radar code is in range of a user and is not currently part of their active buddy list. If not they are added if the radar inclusion mode is active.

Split Groups

[**0150**] When a member of a group specified by a Buddy in that group for mapit display is outside the radius set up in a Group Map Size configuration entry, then that member is split from the group and will not appear on the map of the group. However, that member which has been split from the group will have an entry in a distinctive color such as dark green on the list of active users in the group. Changing the Group Map Size configuration entry to a larger size may allow the split member to displayed. If the location of the slit member must be viewed but the Group Map Size is not to be changed, clicking on the member of the group which has been split from the Buddy List will cause the mapit display to change to the locale of the split member and display the member's location on the map so long as the split member's Buddy Watch status is active.

Power Off or Disable Buddy Tracker

[**0151**] When the phone is turned off or the Buddy Tracker application is disabled, a final transmission to the Buddy Watch server of the location of the Buddy is made. The Buddy Watch server distributes this location in packets addressed to all the members of the group of the Buddy who just went to inactive status.

Targets

[**0152**] The Buddy Tracker software allows targets to be designated to specify meeting points, sites of emergencies or service call locations. Law enforcement agencies can use this feature to silently redirect personnel to the site of a crime or emergency without broadcasting the location on the radio for persons using police scanners to hear. Each target can have a user defined label associated with it and a

message, photo(s) or other document(s) can be attached to the target. All the data defining the target, any label associated therewith and any photos or other documents is packetized in packets identifying the data therein as target data or attachments to the target. These packets are received by the Buddy Watch server and re-packetized addressed to all members of a group or a radar inclusion group or specific Buddies.

[0153] A target can be specified by any member of a group or by a dispatcher of a law enforcement or other agency. Targets can be specified using a web browser. The target is a forward looking waypoint. This can be useful if groups are to meet at a predefined location and the first to arrive may find this is either not the right location or for some reason the meeting point should be changed. The target can be moved, and then packets containing the data of the new target location are sent by the Buddy Watch server to all members of the group with an alert message indicating the target has moved. Targets can be moved simply by dragging and dropping the target to a new location on the display on the web browser which is logged into the Buddy Watch server and which has invoked the target specification command. Once the target has been initially set, moving it to a new location creates a waypoint history. Each target can have a description associated with it, and if the target has been moved, the history can be viewed.

Out of Coverage Operation

[0154] When devices are out of cellular coverage, some limited operations are still possible based on the device. For devices with a full GPS recevier, the user can set targets or force waypoints that are stored. Each device may differ based on the amount of available memory.

[0155] If a Buddy takes his phone into areas of intermittent coverage, it offers a means of some contact. Additionally, one may visit a site on a rural road or other location out of coverage. Setting a target or forcing a waypoint from a phone or desktop computer which is not located at the target provides the location, but does not provide any idea regarding what is at the location. A picture phone at the location can capture a picture of the location, and this picture be associated with the target to give other Buddies in the group some idea of what to expect when they get to a meeting point or target.

[0156] When a user wants to return to the site, the saved target can be recalled and sent to other Buddies in a group or individually designated so a return trip can be planned. This provides the ability to return to spots not located on roads or at intersections such as pastoral settings.

[0157] Local maps when out of coverage would not show up on the user's phone when the mapit command is issued. This is because the map pixels are sent from the Buddy Watch server to the phone after being retrieved from a mapping server such as Yahoo maps. When the phone is out of coverage, the map pixel packets cannot reach the phone and it cannot render a map. However, if the phone has a GPS receiver, it can store the point the user indicate he would like to capture, and, later when the phone is back in coverage, it can send the GPS location to the Buddy Watch server in a mapit command packet, and get the map pixels back from the Buddy Watch server along with any attachments.

[0158] Phones with limited memory will decrease the frequency of position updates so as to not exceed the memory capacity.

Attachments to Targets and Waypoints

[0159] Attachments such as photos can be appended to targets and waypoints even with travelling outside a coverage area. Once the phone is back in coverage, the attachment to a waypoint for example will be sent to the Buddy Watch server and can be distributed to other users. Documents created with phone apps or pictures captured by the phone's built in camera can be attached, and, if the phone has a USB port, pictures captured by a digital camera or camcorder can be imported and attached.

Encryption of Data

[0160] The Buddy Watch software application is disabled and encrypted when it is downloaded to prevent others unauthorized users from installing and using it. The Buddy Watch application program is decrypted and enabled with the access code is downloaded after a subscription is purchased since the decryption key is or is part of or is encoded into the access code.

Access Codes

[0161] Access codes to enable the Buddy Watch application are designed to incorporate the phone number or phone serial number as part of the encyrption key so that the access code can only enable one phone. Large groups with many phones, can ask for and receive access codes that allow operation across a large number of phones.

[0162] Access codes are downloaded to the phone from the cell provider's server or emailed to the user when the user provides name, phone number, phone serial number and a form of payment. The application may be downloaded to a MAC or PC, and then configured on the personal computer before being uploaded to the phone by a computer-to-phone USB connection.

Targets

[0163] A member of a buddy group can market a target on a mapit display, and that target location can be shared to all the members of the group and show up on their mapit displays so they all know where to meet. Marking targets is done using cursors on the mapit display on the phone. The user then designates the buddy list to which the marked target is to be published. Packets are generated in the Buddy Watch application on the phone which include the GPS location, any name assigned to the target and the identification of the buddy list to which the location is to be published. These packets are sent to the Buddy Watch server which then extracts the data and packetizes it into packets addressed to all the phones on the designated buddy list. These packets are then sent to the buddies on the list and the location of the target is extracted and posted on a mapit display.

User Waypoints

[0164] The users can mark particular waypoints as they travel using the mapit displays on their phones, and pictures or memos can be attached to these waypoints. In one embodiment, this is done by sending a packet with the location marked by the user to the Buddy Watch server and in that packet giving an identifier or pointer that will be

contained in other packets which record the memo or photo to be attached to the waypoint. The Buddy Watch server then extracts the data from these packets and stores the user waypoint location with a pointer to the file in which the memo or photo is stored.

SOS Support

[0165] Each user of Buddy Watch can define a profile of buddies to which an SOS alert is to be sent in the case of emergency. The SOS alert message includes location, time and phone number (caller ID) and a preset message for email or Instant Message service and a prerecorded voice message. This data is sent in packets addressed to the Buddy Watch server when the user gives a command to send the SOS message. The Buddy Watch server then receives the SOS message, determines who it is from, retrieves the SOS profile stored on the server for that user and generates packets for email and IM and sends them on the internet and generates packets containing the digitized voice message and addresses them to the phones listed in the SOS profile and sends those packets to the cellular system central switching system **102** in **FIG. 16** via internet gateway **148**.

[0166] The SOS message protocol can be carried out by the Buddy Watch server either on demand from the user, or automatically in conjunction with any 911 call made from a phone which has a stored SOS profile. The SOS support configuration file contains data which defines which way the phone will act, and the buddies receiving the SOS messages will be aware of whether an 911 call was made or not. The buddies are actually in a better condition to help the caller since they can see the caller's position on their mapit displays, and they may be closer to the caller and be able to act quicker than the 911 support personnel.

[0167] The preferred embodiment causes the SOS messages to be sent when the user dials **911**. A **411** dialed call will send the SOS messages to only active buddies whose phones are registered in the system and on with Buddy Watch activated.

The User Interface Genus

[0168] All species within the genus of user interfaces according to the teachings of the invention will display buddy lists and a list of buddies on each buddy list when that buddy list is selected. All species will display the specific information about a buddy when a particular buddy is selected including at least 'their current location and the time of the fix. All species will display a command or icon or menu choice that can be invoked to allow a user to turn off location sharing. All species will display commands, icons or menu choices to add, delete or edit buddy lists to add or delete or edit buddies.

[0169] Some species within this genus will also display one or more of the following items of information about individual buddies: speed, last contact, altitude or direction. Some species within the genus will provide icons, menu choices, etc. which a user can invoke to allow the user to select a map display with the location of a buddy displayed thereon. Some species within the genus will allow a user to give a command to request historical fixes which trace a path to the buddy's current position. Some species within the genus will allow instant buddy relationships to be set up to allow location sharing between a person in trouble and a rescuer.

The Server Genus

[0170] All servers programmed with Buddy Watch software will have functionality to:

[0171] 1. either store map data for entire geographical areas that they serve or to obtain pertinent map data from another server such as a Mapquest™ server and pick the appropriate maplet that surrounds the positions of buddies to be displayed and serve the maplet data to Buddy Watch enabled phones;

[0172] 2. pick the appropriate maplet for each buddy list or buddy based upon the center of gravity of the buddy positions of the buddies within the selected buddy list and exclude buddies which are out of the coverage area;

[0173] 3. render buddy locations on maplets based upon GPS location data gathered from Buddy Watch applications running on GPS enabled cell phone and PDAs;

[0174] 4. store user defined data that embodies each user's buddy lists and buddies and configuration data;

[0175] 5. store at least some preference data that defines at who can use the server, i.e., only those with a valid Buddy Watch user ID and password);

[0176] 6. request and receive update and regularly scheduled GPS location data from users who have their Buddy Watch application turned on on their phones or PDAs and to distribute location data and maplets to the phones and PDA of the buddies on buddy lists who have their Buddy Watch capability turned on;

[0177] 7. turn Buddy Watch functionality on or off in terms of receiving location data from users who have indicated they want their Buddy Watch application turned off and turn off sharing location data of buddies who have turned off their Buddy Watch application.

[0178] Various species within this genus: can calculate the center of gravity of the best fit for the maximum number of buddies that are within the coverage of one maplet; determine the proper maplet size to send to the client phone or PDA based upon configuration data which defines the screen size of the device; send the same size maplet to all clients; allow each client to determine its own maplet size; send maplets with buddies color coded to show who is out of bounds and who is in lost contact status; implement a permissive buddy list wherein a person cannot be added to a buddy list until they consent; implement timed updates for GPS position and scheduled cutoff times for for position sharing; store auxiliary information about each buddy such as phone numbers, etc.; offer the functionality to allow each user to specify the maplet size they receiver or specify a maximum maplet size for a buddy list; off the functionality to request updates whenever a programmable delta time or delta position difference over the last update occurs; offer a user preference to turn on or turn off GPS position updates; the ability to cross communicate with other carriers cellular systems to send maplets to and receive location data from users on other systems; function to enable or disable the Buddy Watch application without disabling location sharing with parental or supervisor units; storing as a preference or configuration data SOS emails and voicemail messages which can be sent out to email addresses and/or phone numbers specified in a configuration data file on demand or automatically when a 911 call is made.

The Client Application Genus

[0179] The client Buddy Watch application and phone or PDA platform genus collectively provide the following functionality:

[0180] 1. the programmed phone or PDA must be able to retrieve GPS position data directly or indirectly from a GPs receiver in the phone or PDA, and it should be able to wirelessly send the GPS position data to the Buddy Watch server either periodically or on demand from the server, but one or another, it must be able to exchange position information data with the server;

[0181] 2. the phone or PDA must have a display large enough to display maplets and be able to download maplets from the Buddy Watch server;

[0182] 3. it must have Java or similar software to exchange digital data with the Buddy Watch server using a wireless web application program;

[0183] 4. it must be able to communicate with the phone's application programmatic interface and any application programmatic interface of the cell phone service provider to:

[0184] be able to receive maplets from the Buddy Watch server with location data rendered thereon and display the maplets;

[0185] send location data and receive downstream messages and requests from the Buddy Watch server;

An important species with this genus will be able to request software needed to execute commands given by the user from the Buddy Watch server, receive a download of the software requested, install it into random access memory and execute it to carry out the requested command. In other species, the software

Apparatus and Process to Simplify Push to Talk Walkie Talkie Operations in Cell Phones

[0186] **FIG. 31** is a block diagram of the system for TalkControl to simplify cell phone walkie talkie operations. Block **300** is a location determination component which functions to determine user locations. This can be done in the cell phones or the Rubicon server and provides a generic solution to extract location from GPS, J2ME location API or bespoke development for extracting Cell ID. If done on the server, the GMLC based solution us used. Block **300** also does distance calculations, location format conversion etc. LDG can expose a LIF based interface to location based services when applicable. This provides location determination flexibility as needed.

[0187] SBC component **302** functions to do buddy group/list management, mapping techniques, refresh based upon time or delta movement, geo coding, reverse geo coding, routing, etc.

[0188] CMC block **304** functions to provide local content to location based services. The content can be local maps or commercia/enterprise specifi content. Multiple parties like commercial content providers, government establishments or enterprises will provide the content. This CMS component will provide a common API to extract content from multiple providers and provide the flexibility to choose any content provider based upon parameters such as accuracy, availiability of content, rates, whether the content is the latest, etc.

[0189] The SRC block **306** is a software rendering component which provides multiple channel and device rendering, mobile application provisioning, service creation environment, OSS/BSS integration in both pre-paid and post paid modes, usage analysis reports and SNMP based system management software.

[0190] GSC block **308** provides alert and notification systems, personalization, payment integration etc.

[0191] Individual services block **310** provides tools and generic components to build individual applications in consumer and enterprise domains. Consumer services like child tracking, buddy location, location based advertisements for target user groups can be built. Enterprise services such as work force management, fleet tracking, emergency services, etc. can use the generic components.

[0192] **FIG. 32** is a flow diagram of a process a user of a walkie talkie enabled phone can initiate to joint a talk group to enable subsequent walkie talkie operations. This process greatly simplifies the process of signing up for walkie talkie operations of a wireless carrier. A user who wishes to join a walkie talkie talk group launches the TalkControl application, scrolls down to Join Group menu option, selects an Enter Tokens options and fills in her name, phone number, project ID, and Token and presses send. One or more packets are sent to the Rubicon server which authenticates the token and the recipient and creates a database entry. The Rubicon server then determines a time to add the user to the talk group and contacts a server of the wireless carrier to add a user. The Rubicon server logs onto the Carrier Server and adds the user to the appropriate talk group and receives a confirmation. The confirmation is send to the user who initiated the process, and the Rubicon server logs out of the carrier server.

[0193] **FIG. 33** is a flowchart of the process the Rubicon server carries out to automatically delete a user. Users in talk groups can be deleted automatically based upon a scheduled deletion time using this process. The Rubicon (X One) server compares the current date and time with a database for users scheduled to be deleted. If a user is to be deleted per schedule, the Rubicon server logs onto a carrier server and deletes the user from the appropriate talk group and receives confirmation. The Rubicon server then deletes the user from the talk group in its database and logs out of the carrier server. The Rubicon server then sends the deletion confirmation to the user phone.

[0194] **FIG. 34** is a flowchart of the process the Rubicon server carries out to allow a supervisor to add a user. The supervisor launches the TalkControl application program and scrolls down to the add/edit/delete user menu option and logs in as a supervisor and presses send. The supervisor then selects User Name and selects next to take him to the user screen where the user's name, phone number, project ID are entered. The supervisor then edits the start date, end date, hours, days of the week for the active period when the user being added will be part of the talk group so that walkie talkie service can only be had during the specified times. One or more packets are then generated addressed to the Rubicon server and encapsulated in a cellular system packet

and sent. These packets get routed to the Rubicon server which authenticates the initiator and recipient, creates a database entry for the user and contacts the Carrier server and logs on. The Rubicon server then adds the user to the appropriate talk group and receives confirmation. The Rubicon server then adds the user to the appropriate talk group and updates its database and receives the confirmation. The confirmation is send to the supervisor who added the user and to the user phone which was added to the talk group.

[0195] **FIG. 35** is a flowchart of the process for a supervisor to edit a user in a talk group. The supervisor lauches the TalkControl application and scrolls down to add/edit/delete user and selects that option. The supervisor logs in as the supervisor and selects edit user and selects a user already in a talk group and edits data in fields for name, phone number, project ID of the user to be edited and presses next. Start date, end date, hours, days of the week are then changed as desired. From that point, the process is the same as adding a new user.

[0196] **FIG. 36** is a flowchart of a process for a supervisor to delete a user from a talk group. The supervisor lauches TalkControl and scrolls down to add/edit/delete user and selects that. She logs in as a supervisor and scrolls down to delete a user and deletes data in name, phone number, project ID, start date, end date, hours and days of week field and presses send. One or more packets get routed to the Rubicon servers which authenticates the initiator and recipient. The Rubicon server then logs onto the Carrier server and deletes the user from the appropriate talk group and receives a confirmation. The Rubicon server receives the confirmation and updates its database to delete the user from a talk group. Confirmation is then sent from the Rubicon server to the supervisor phone and the user's phone, and the Rubicon server then logs out of the carrier server.

[0197] **FIG. 37** is a flowchart of a process for a supervisor to issue a token. The supervisor launches TalkControl and scrolls down to add/edit/delete a user. She logs in as a supervisor and scrolls down to issue token menu option and selects it. She then scrolls down to select project to issue token menu option and presses next. The user selects token being issued and presses issue. The supervisor then informs the user of the project name and the token code. Packets are sent to the Rubicon server which removes the token from the list and replaces it with a new token. The Rubicon server then send a message to the supervisor phone to add the message send new token to the supervisor's handset.

[0198] **FIG. 38** is a flowchart for the process of setting up preferences. The TalkControl application is launched on the handset and the user scrolls down to the add/edit/delete user option and presses select. the user logs in and presses send. The user then scrolls down to preferences and presses next. The user then selects the preference feature to be updated and presses select. This vectors processing to one of the four illustrated lines of processing to set the methods of notification as audio, vibrate or select auto login or set the new password or set a new ID. Processing then loops back to allow another preference to be selected and edited.

[0199] **FIG. 39** is a flowchart for the process to allow a supervisor to request status. The supervisor launches TalkControl and scrolls down to view status. She logs in as a supervisor and presses send and then selects a user. The supervisor then scrolls and selects project to view status and presses next and scrolls down to select the project and the user and views the user's details.

[0200] **FIG. 40** is a process flowchart of the process for a supervisor to create a group. The supervisor launches TalkControl application and scrolls down to create a group and presses select. She then logs in as supervisor and presses send and enters group name. A talk group is then automatically created. The supervisor sets the duration of the group, its start date, end date, days, hours and presses next. One or more packets addressed to the Rubicon server are then created and sent to the Rubicon server. The Rubicon server then authenticates the initiator and recipient and creates a database entry for a new group. The Rubicon server then contacts the carrier server and logs in and creates a talk group in the carrier server and receives a confirmation. The Rubicon server then adds the group to the appropriate talk group and receives confirmation. The confirmation is then sent from the Rubicon server to the supervisor and the Rubicon server logs out of the carrier server. The supervisor phone then receives a message on the wireless device displaying the group added profile updated request screen.

[0201] Although the invention has been disclosed in terms of the preferred and alternative embodiments disclosed herein, those skilled in the art will appreciate possible alternative embodiments and other modifications to the teachings disclosed herein which do not depart from the spirit and scope of the invention. All such alternative embodiments and other modifications are intended to be included within the scope of the claims appended hereto.

What is claimed is:

1. A process for sending a GPS position from a first wireless device to one or more second wireless devices, comprising:

A) receiving in a first wireless device a position update request;

B) looking up addresses of one or more second wireless devices to which a position update is to be sent;

C) reading GPS information and encrypting it;

D) putting encrypted GPS information into payload portions of one or more packets of short message or email protocol format and addressing said packets to one or more wireless devices to which position updates are to be sent;

E) transmitting the packet data to a transceiver in a cell in which said first wireless device is registered;

F) receiving the signals containing said packet data and recovering said packets at said transceiver;

G) routing said packets to a central switching unit;

H) in said central switching unit routing each packet to a transceiver in a cell in which the wireless device to which said packet is addressed is registered;

I) in each said transceiver which received one or more packets from said central switching unit, processing the data of said packets to generate radio frequency signals and transmitting said radio frequency signals to the wireless devices in said cell;

J) in the wireless device to which the packet is addressed, recovering said the data of said one or more packets and reassembling said packets;

K) reading the header data of said packets and routing said packets to the GPS position data sharing software application;

L) in said GPS position data sharing software application decrypting said GPS position data and using the header information of said packets to determine the identity of the person whose GPS position data was just decrypted and using said decrypted position data to update the displayed position data of said person.

2. The process of claim 1 wherein step E comprises the steps:

A2) disassembling each said packet into groups of bits of a predetermined size codewords;

B2) interleaving said codewords;

C2) encoding each codeword with a plurality of error detection and correction bits;

D2) generating symbols to send from said codewords

3. The process of claim 2 wherein step D2 comprises breaking each codeword into groups of bits of a predetermined size, the number of bits in each group being dependent upon the modulation scheme in use.

4. The process of claim 2 wherein step D2 comprises breaking each codeword into groups of bits of a predetermined size, the number of bits in each group being dependent upon the modulation scheme in use, and encoding each group of bits in a Trellis encoder.

5. The process of claim 1 wherein step L further comprises the steps:

receiving a map it command from a user of said wireless device;

displaying a map of a geographic area in which positions of said wireless devices can be displayed;

using said GPS position data decrypted in step L to place a contrasting point or other indicator of the position of said entity which sent said position update.

6. The process of claim 1 wherein step A comprises receiving a position update request from a timer in said first wireless device that initiates a position update periodically.

7. The process of claim 1 wherein step A comprises receiving a position update request from a process running on said first wireless device which constantly compares the current GPS position to the last reported GPS position and generates a position update request whenever said position changes by more than a predetermined amount.

8. The process of claim 1 wherein step A comprises receiving a position update request from one or more of said second wireless devices.

9. The process of claim 1 further comprising the steps:

6A) reading GPS information of a second wireless device which just received a GPS position update and encrypting said GPS information;

6B) putting said encrypted GPS information into payload portions of one or more packets of short message or email protocol format and addressing said packets to said first wireless device;

6C) transmitting said packet data to a transceiver in a cell in which said second wireless device is registered;

6D) receiving the signals containing said packet data and recovering said packets at said transceiver;

6E) routing said packets to a central switching unit;

6F) in said central switching unit routing each packet to a transceiver in a cell in which said first wireless device to which said one or more packets is addressed is registered;

6G) in said transceiver which received one or more packets from said central switching unit address to said first wireless device, processing the data of said packets to generate radio frequency signals and transmitting said radio frequency signals to the wireless devices in said cell;

6H) in said first wireless device to which said one or more packets is addressed, recovering said the data of said one or more packets and reassembling said packets;

6I) reading the header data of said one or more packets and routing said packets to GPS position data sharing software application;

L) in said GPS position data sharing software application decrypting said GPS position data and using the header information of said packets to determine the identity of the person whose GPS position data was just decrypted and using said decrypted position data to update the displayed position data of said person.

10. A process for receiving a GPS position from a second wireless device at a first wireless devices, comprising:

A) in a first wireless device to which one or more GPS position data bearing packet is addressed, recovering said the data of said one or more packets and reassembling said packets;

B) reading header data of said one or more recovered packets and routing said packets to the GPS position data sharing software application;

C) in said GPS position data sharing software application decrypting said GPS position data from payload sections of said one or more packets and using header information of said packets to determine the identity of the person whose GPS position data was just decrypted and using said decrypted position data to update the displayed position data of said person;

D) reading GPS position data from a GPS receiver in said first wireless device and encrypting said GPS position data;

E) encapsulating said GPS position data into payload portions of packets and addressing said packets to said second wireless device; and

F) transmitting said one or more packets containing GPS position data of said first wireless device to a transceiver in a cell in which said first wireless device is registered.

11. The process of claim 10 further comprising the steps:

making a position update request from said first wireless device to said second wireless device and starting a timer in said first wireless device at the time said request is made;

if said timer reaches a predetermined time without having received a position update from said second wireless device, changing the color of a display of a position of said second wireless device to some color which indicates said second wireless device is either out of cellular coverage or not responding for some other reason.

12. An apparatus comprising a wireless cell phone, PCS device, Blackberry or other device capable of receiving and making wireless phone calls and having a GPS receiver, and programmed to communicate with a server so as to be able to exchange GPS location data with another wireless cell phone, PCS device, Blackberry or other device capable of receiving and making wireless phone calls such that the locations of individuals or groups can be known.

13. A process carried out by a server comprising:

registering users of a buddy watch service;

receiving and storing position data giving the location of wireless devices operated by said registered users, said wireless devices being capable of making and receiving phone calls;

sending position data to wireless devices being capable of making and receiving phone calls to other registered users.

* * * * *