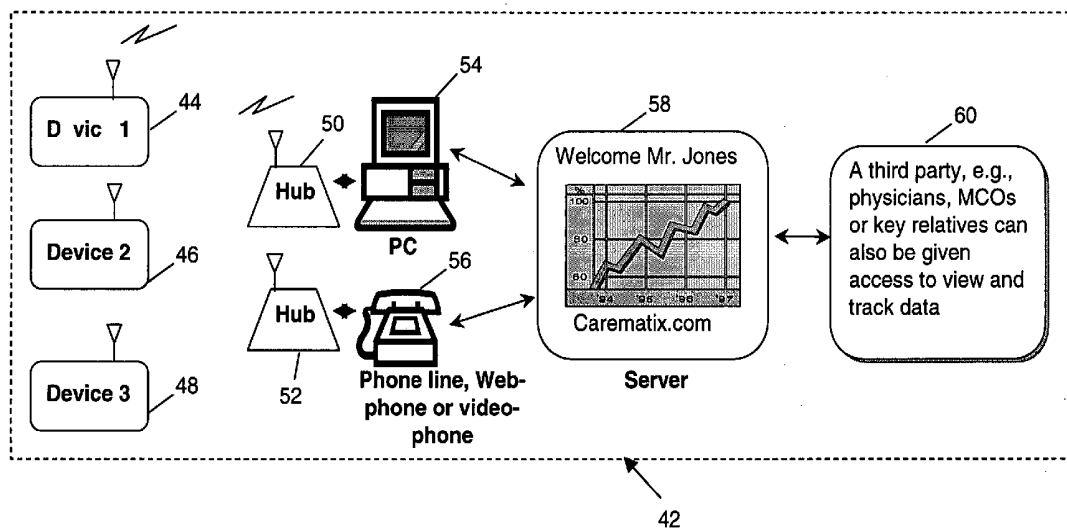


(19) **United States**(12) **Patent Application Publication** (10) **Pub. No.: US 2004/0102683 A1****Khanuja et al.**(43) **Pub. Date:****May 27, 2004**(54) **METHOD AND APPARATUS FOR
REMOTELY MONITORING THE
CONDITION OF A PATIENT**(76) Inventors: **Sukhwant Singh Khanuja**, Chicago, IL
(US); **Sandeep Garg**, Naperville, IL
(US); **Irwin Preet Singh**, Prospect
Heights, IL (US)

Correspondence Address:

John S. Paniaguas
KATTEN MUCHIN ZAVIS ROSENMAN
Suite 1600
525 West Monroe Street
Chicago, IL 60661 (US)(21) Appl. No.: **10/414,326**(22) Filed: **Apr. 15, 2003****Related U.S. Application Data**(60) Provisional application No. 60/372,894, filed on Apr.
16, 2002.**Publication Classification**(51) **Int. Cl.⁷** **A61B 5/00**(52) **U.S. Cl.** **600/300**(57) **ABSTRACT**

The present invention relates to a patient monitoring system which provides enhanced functional capability relative to known systems and provides a wireless communication link between a patient monitoring device, worn by a patient, and a local hub. The patient monitoring system in accordance with the present invention is adapted to monitor various patient physiological characteristics, such as blood pressure, pulse rate, blood glucose, weight, pulse oximetry and others. The data from the patient monitoring device is wirelessly transmitted to a local hub, which, in turn, is configured to automatically transfer the data to a remote server, for example, over a public or private communications network. In one embodiment of the invention, the server is configured as a web portal to selectively allow access to such patient physiological data by designated third parties, such as physicians, clinicians, relatives and the patient themselves. In accordance with another important aspect of the invention, the system provides for enhanced functionality relative to known systems and allows trends of the physiological data to be selectively generated. In addition, any third party can set thresholds to automatically notify the patient or other third parties by various communication methods including e-mail and/or pager when a particular physiological characteristic exceeds a predetermined threshold. The system is also configured to provide reminders to patients to take readings.



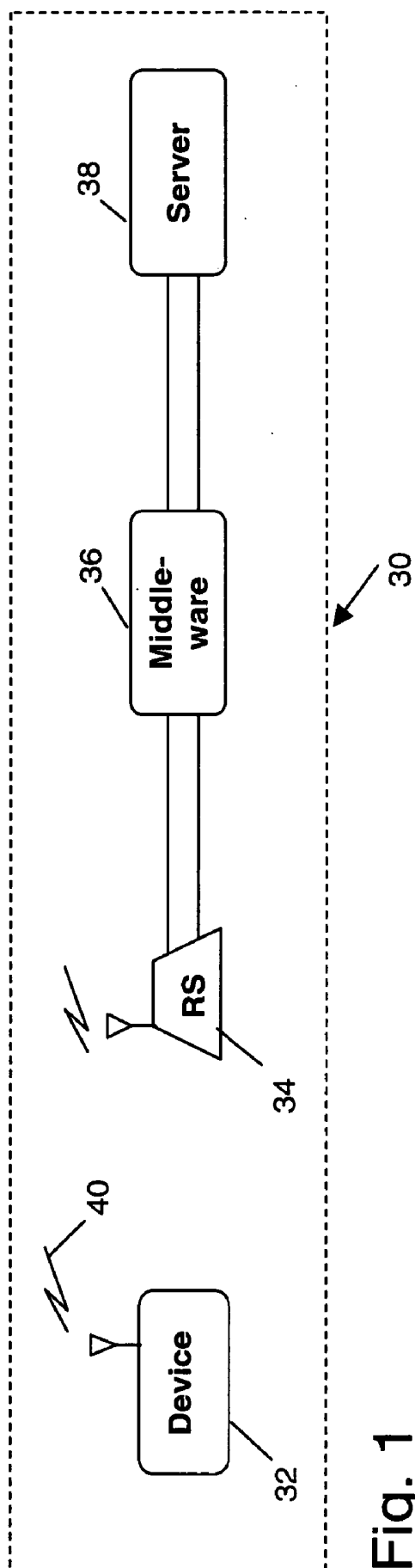


Fig. 1

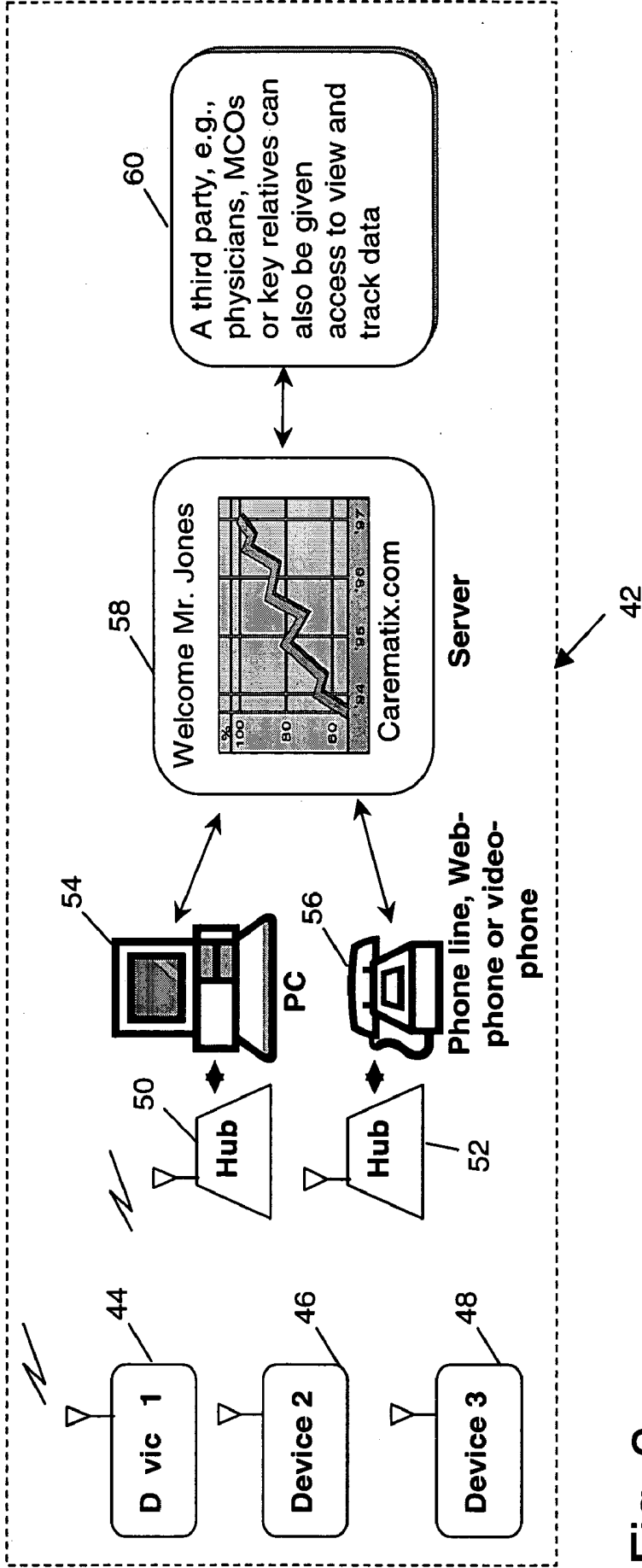


Fig. 2

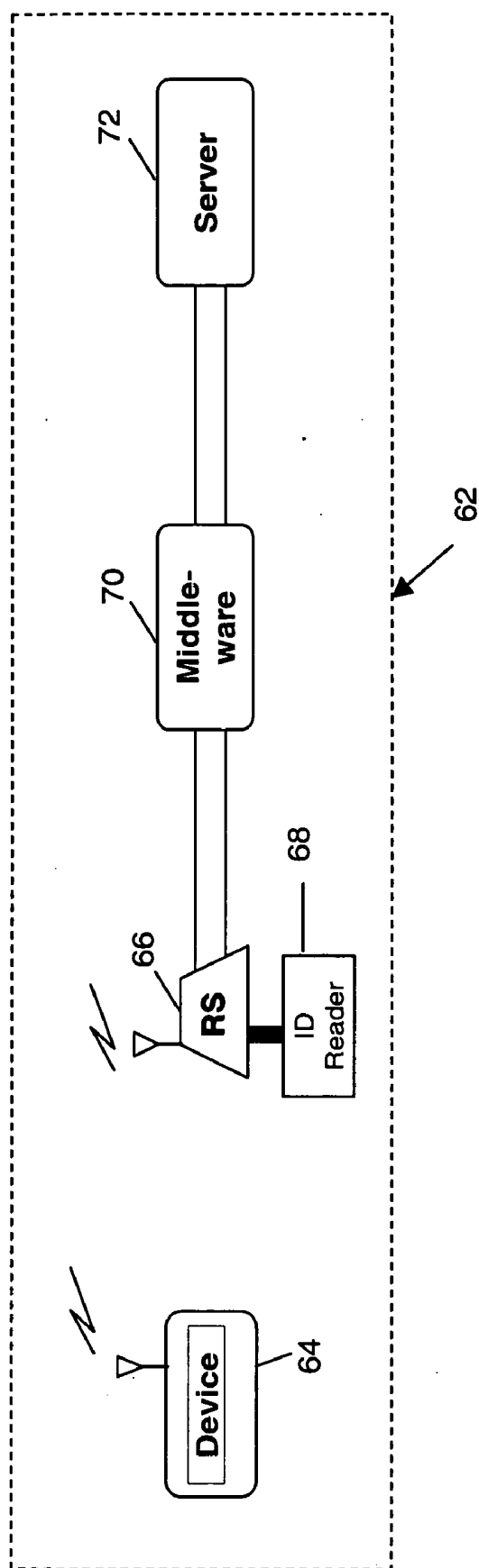


Fig. 3

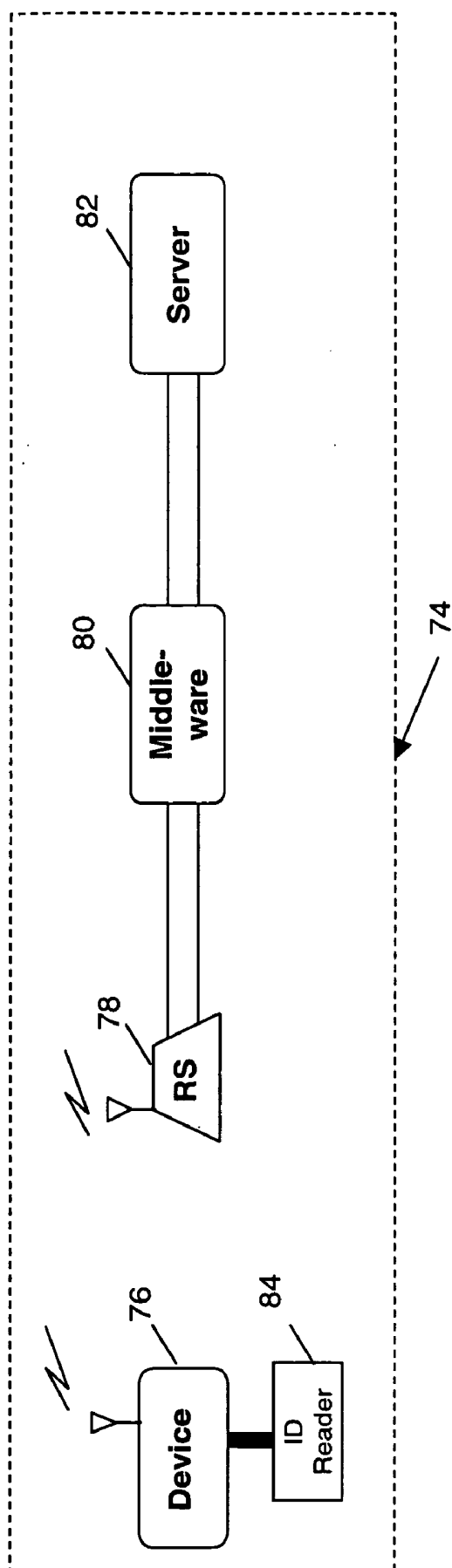


Fig. 4

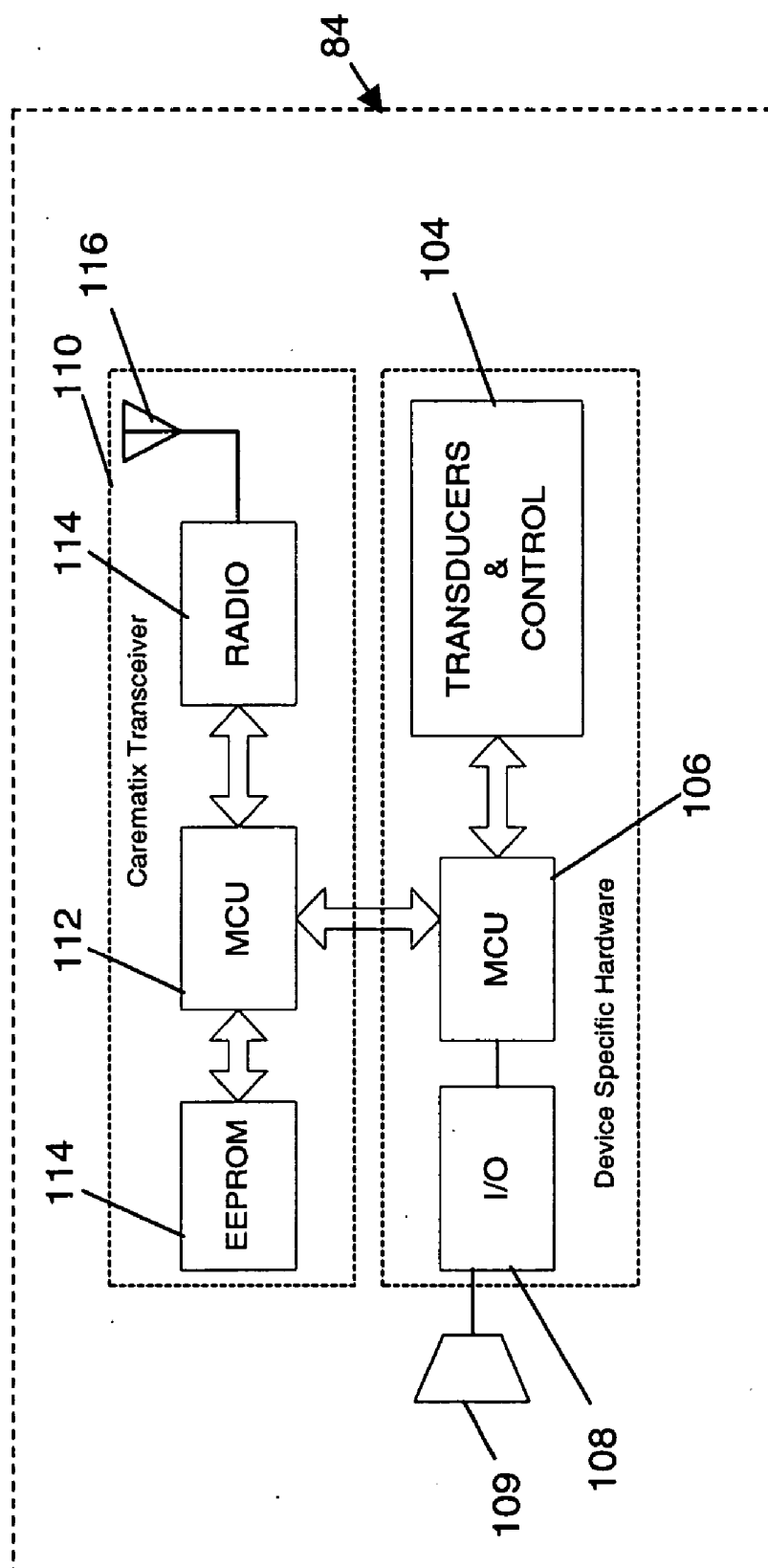


Fig. 5

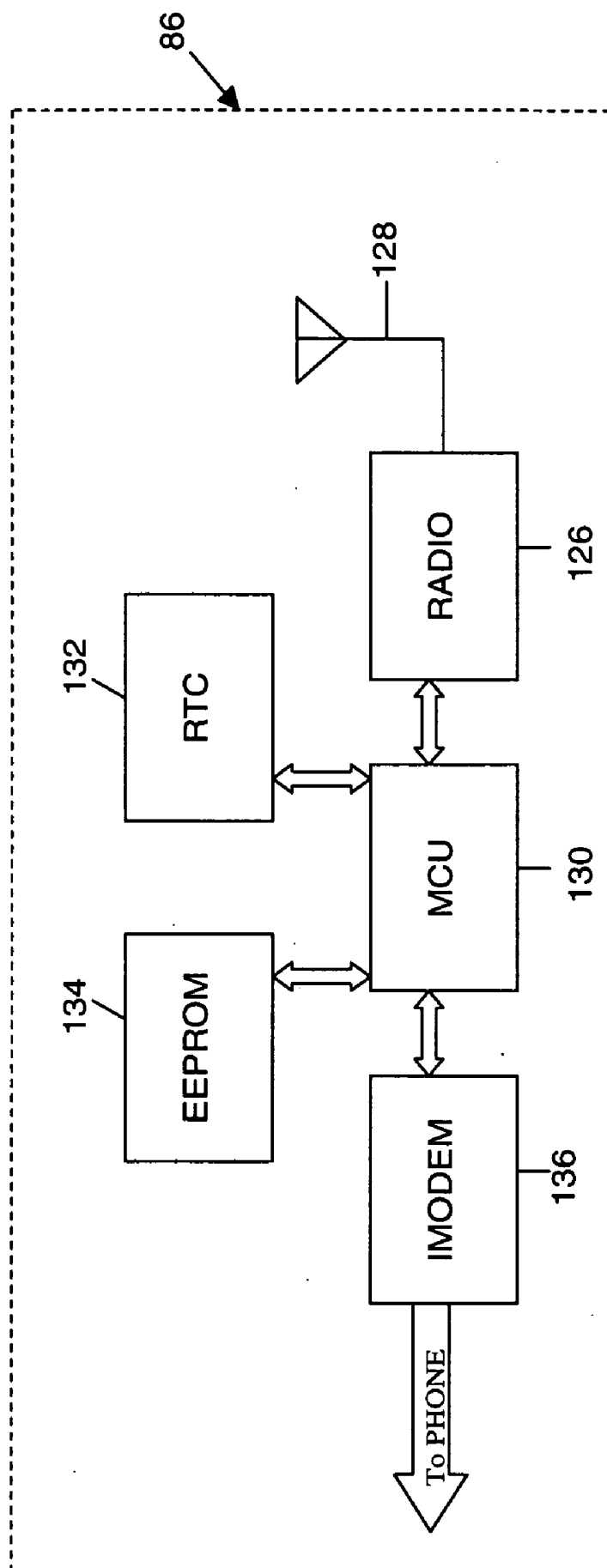


Fig. 6

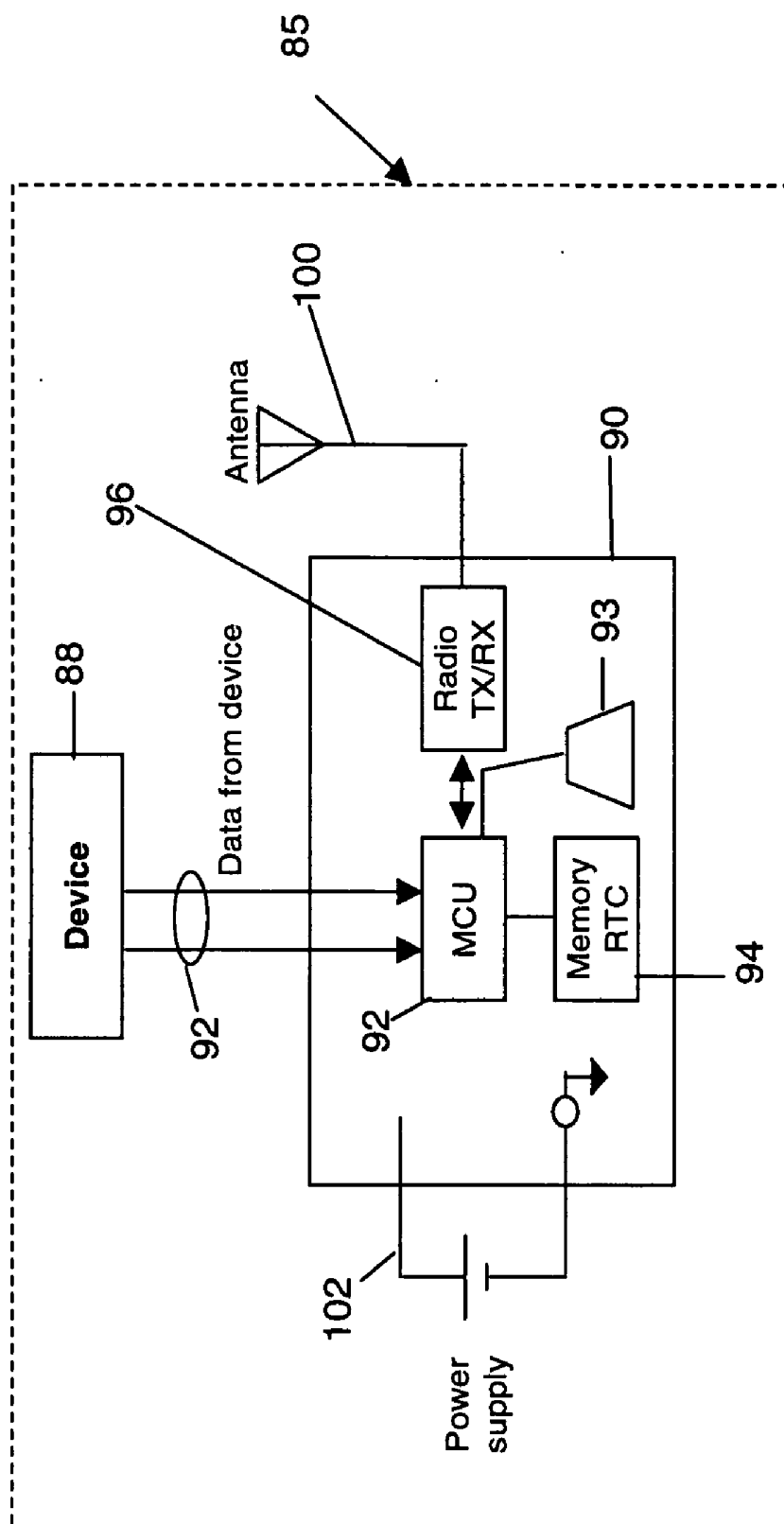


Fig. 7

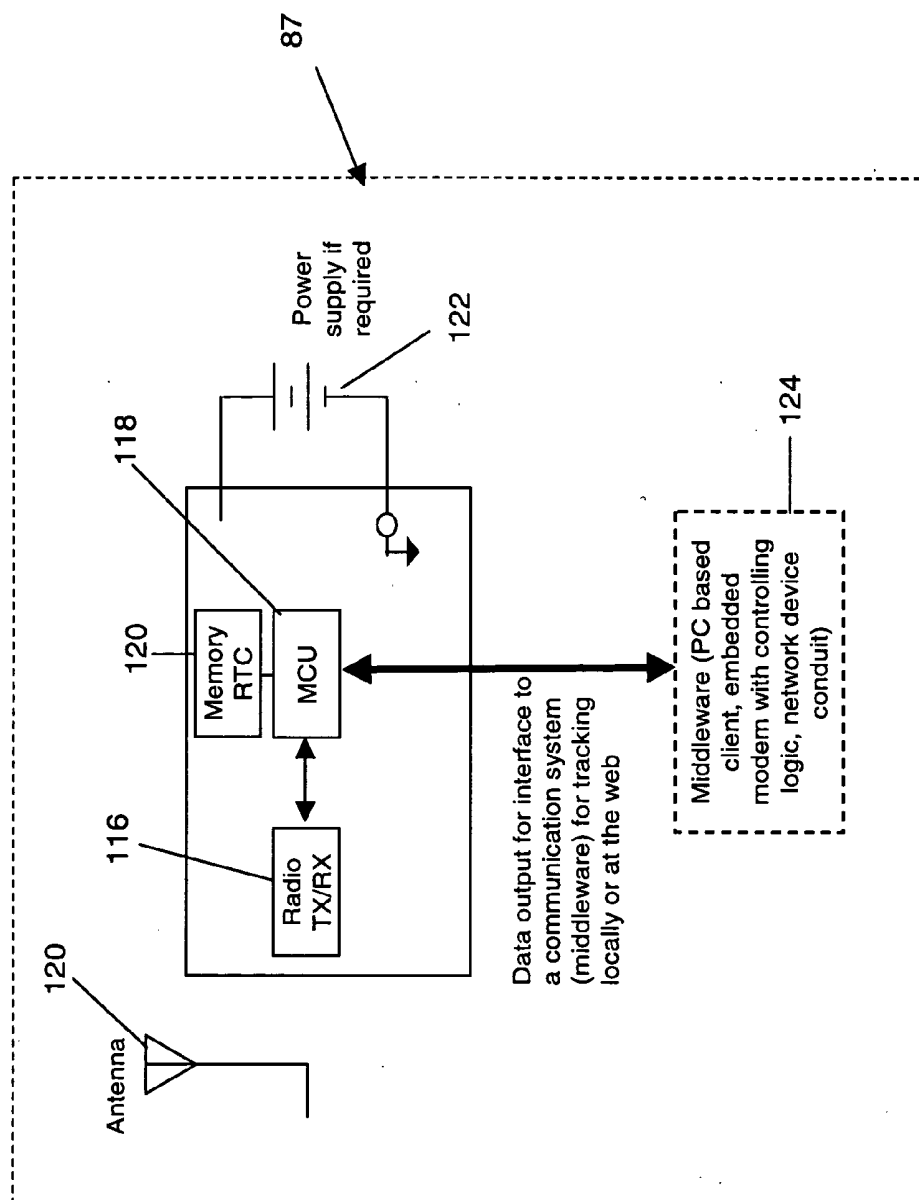
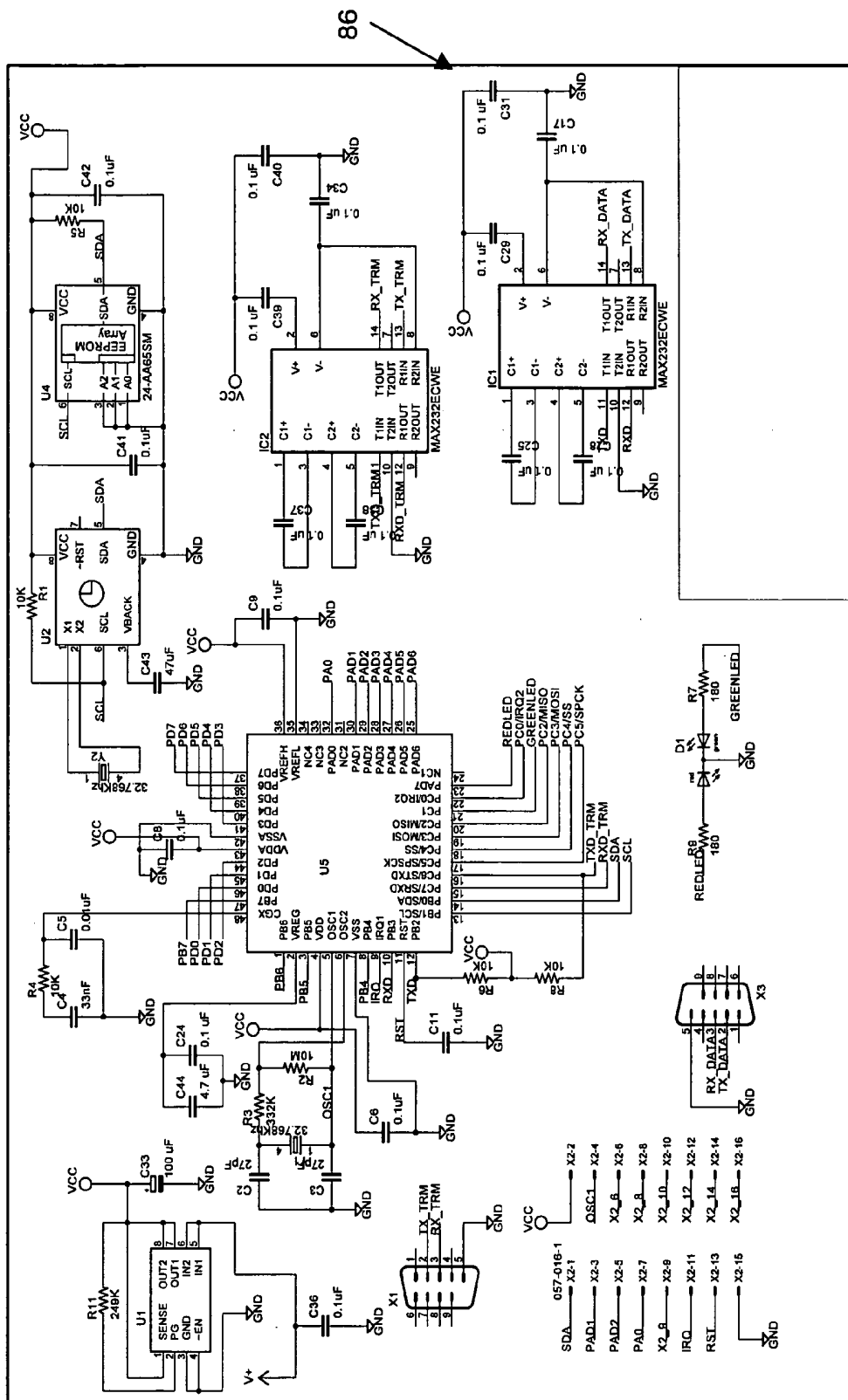
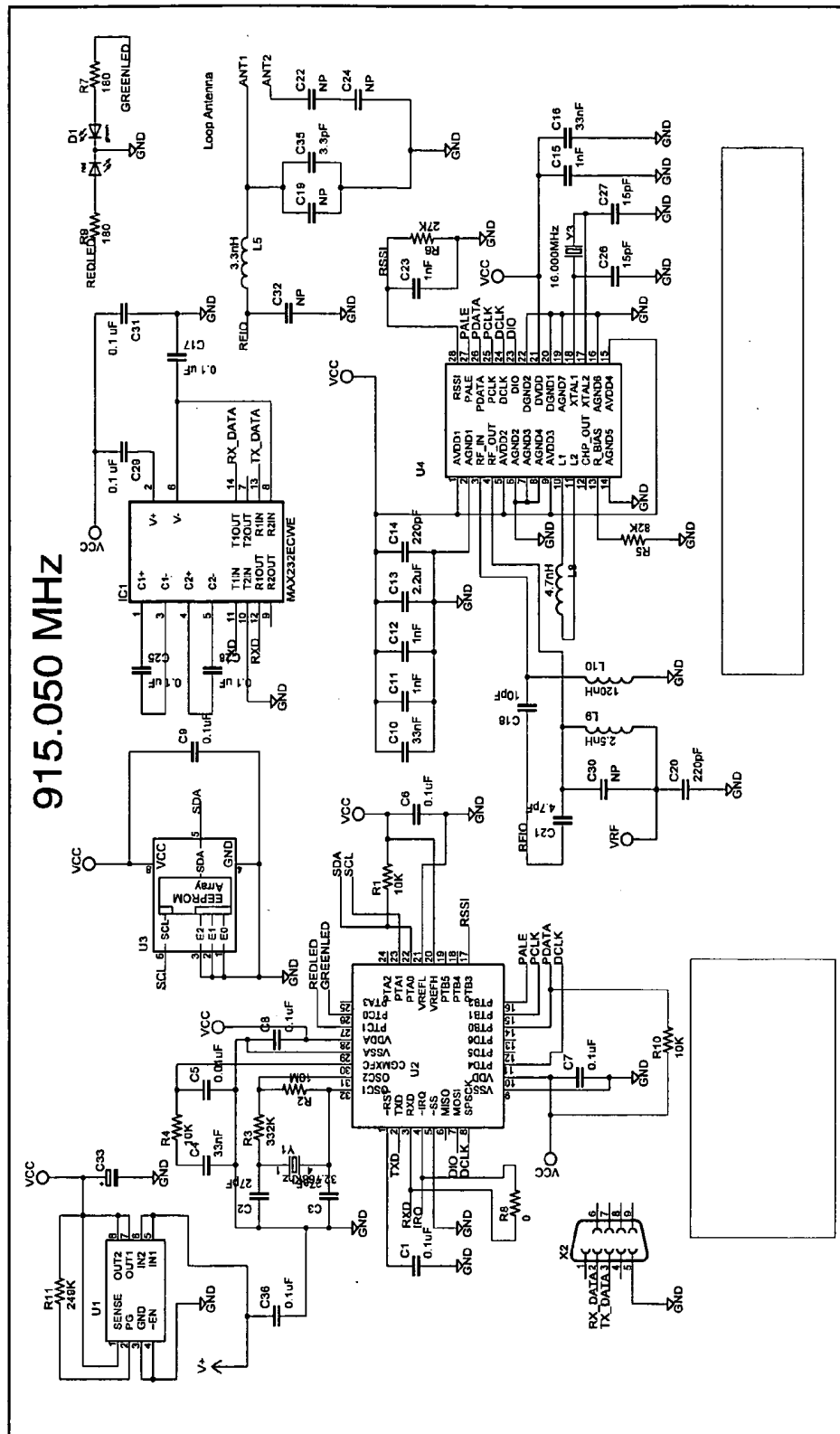


Fig. 8





126

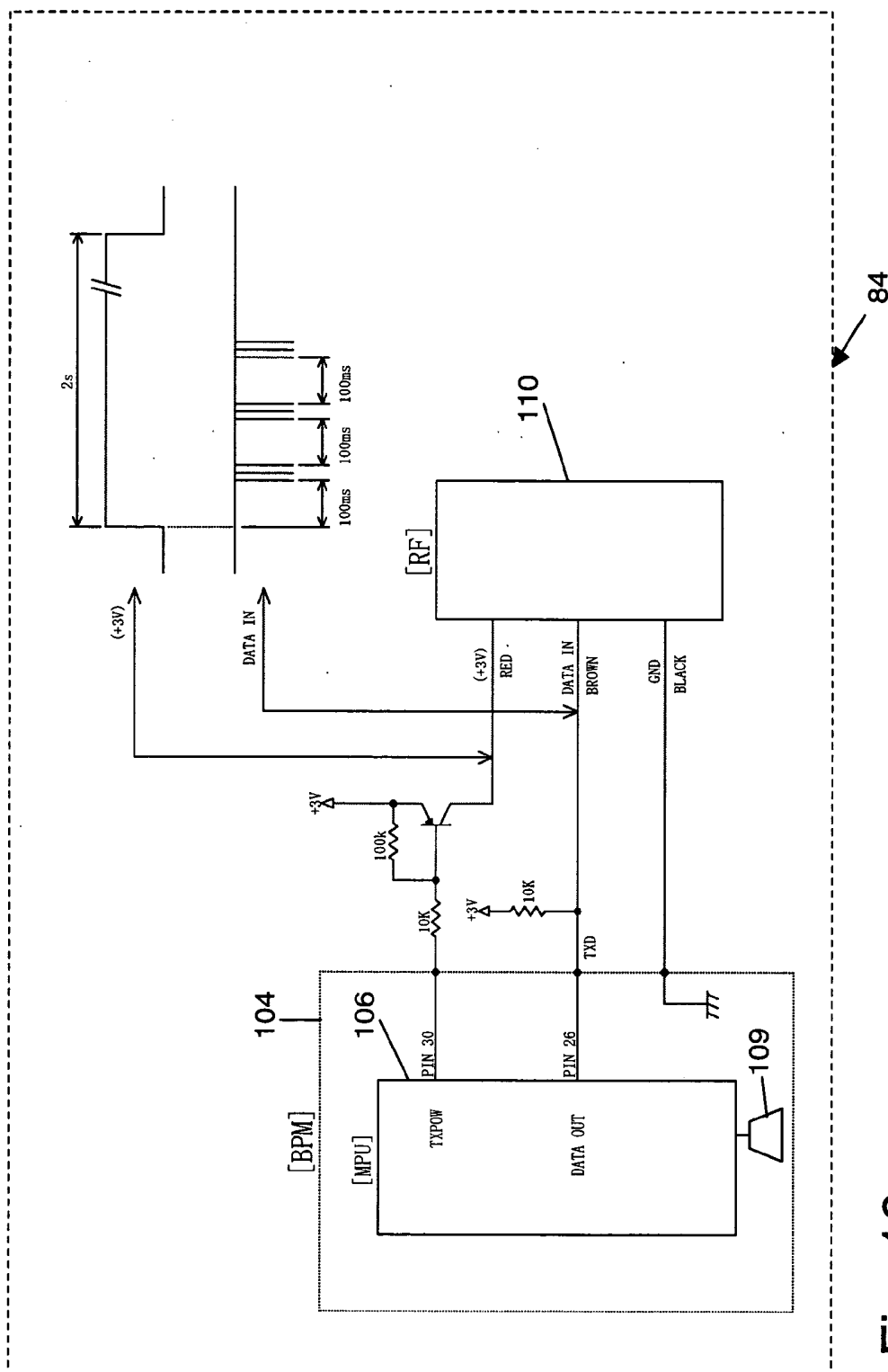


Fig. 10

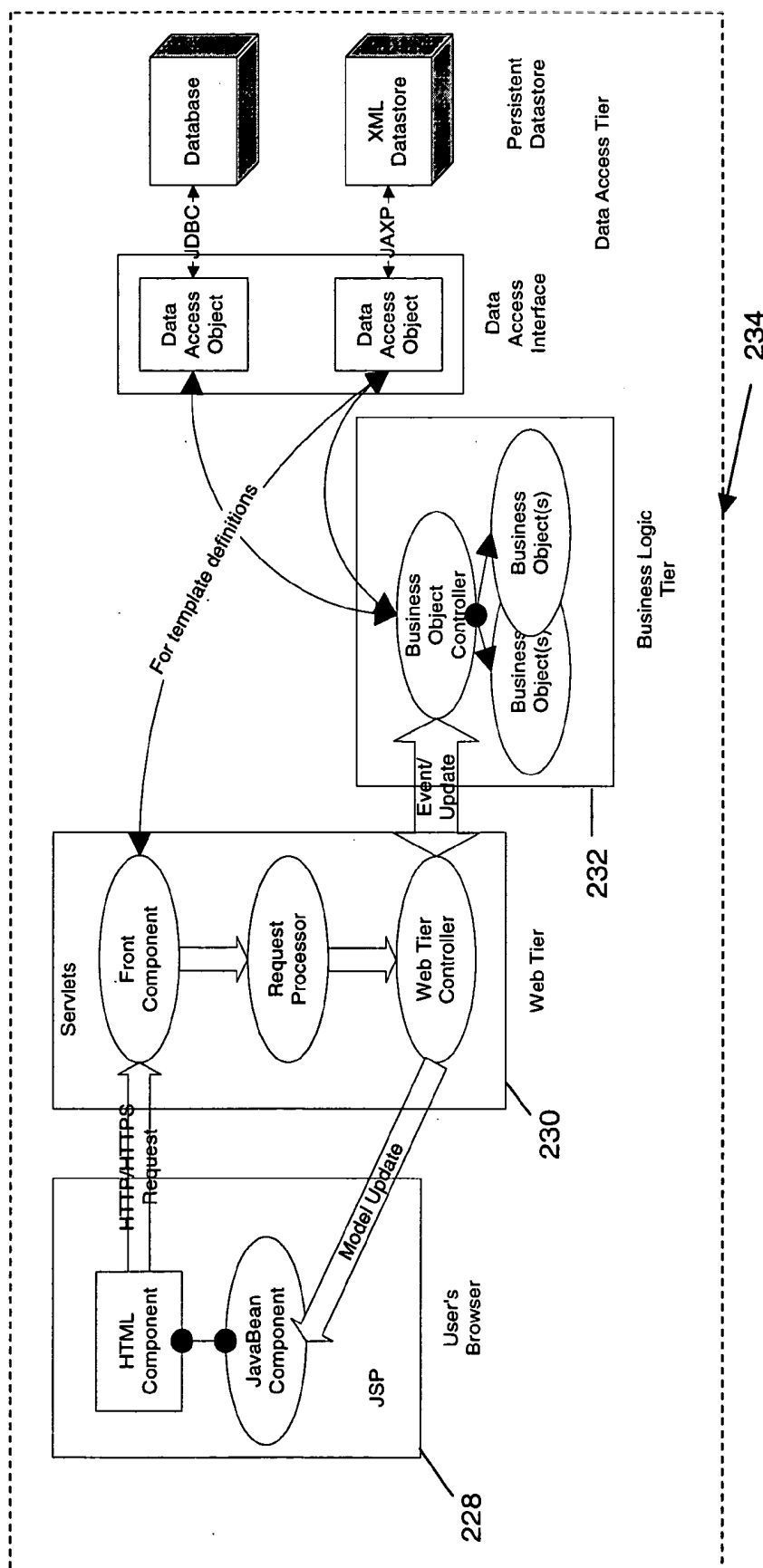


Fig. 11

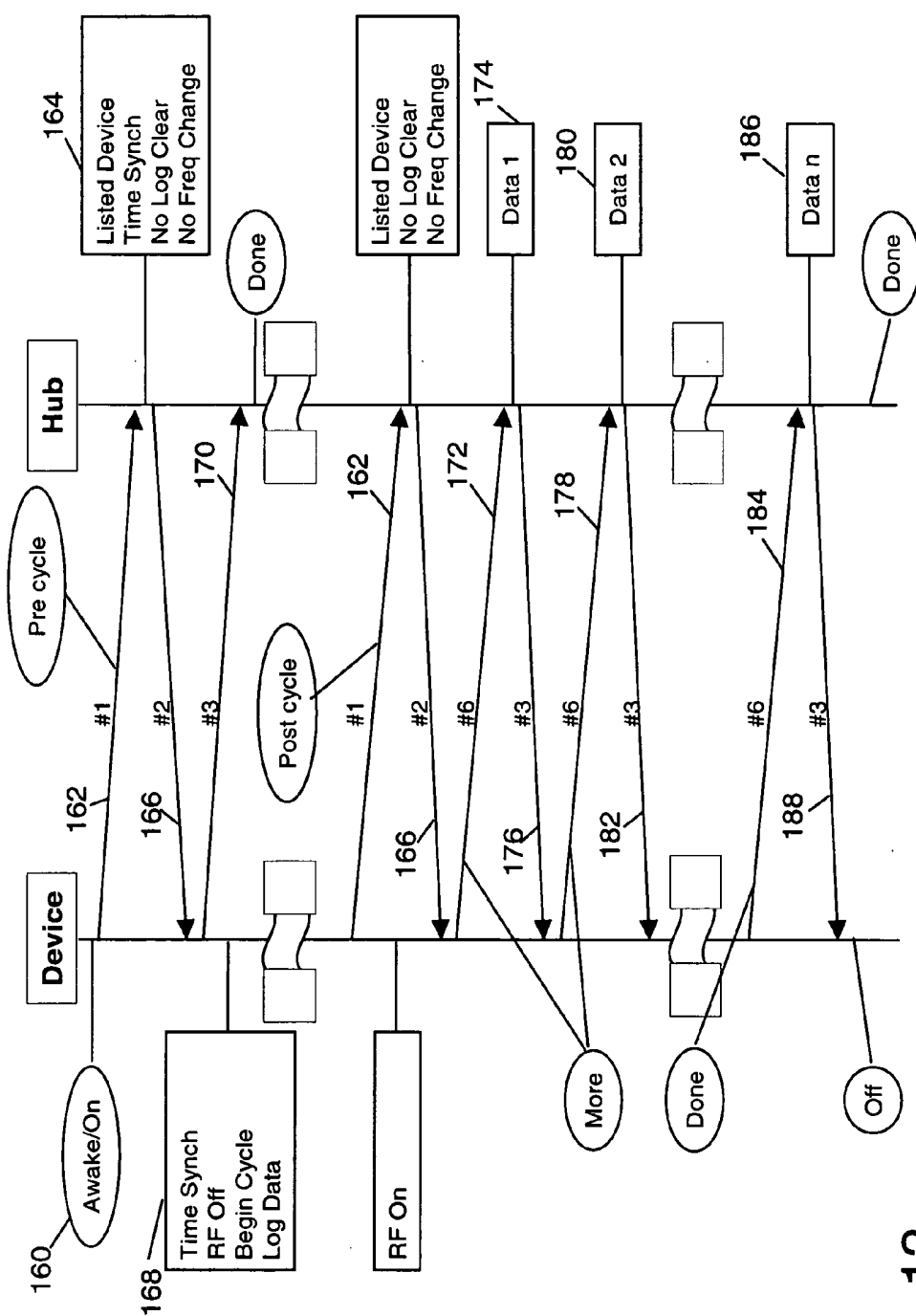


Fig. 12

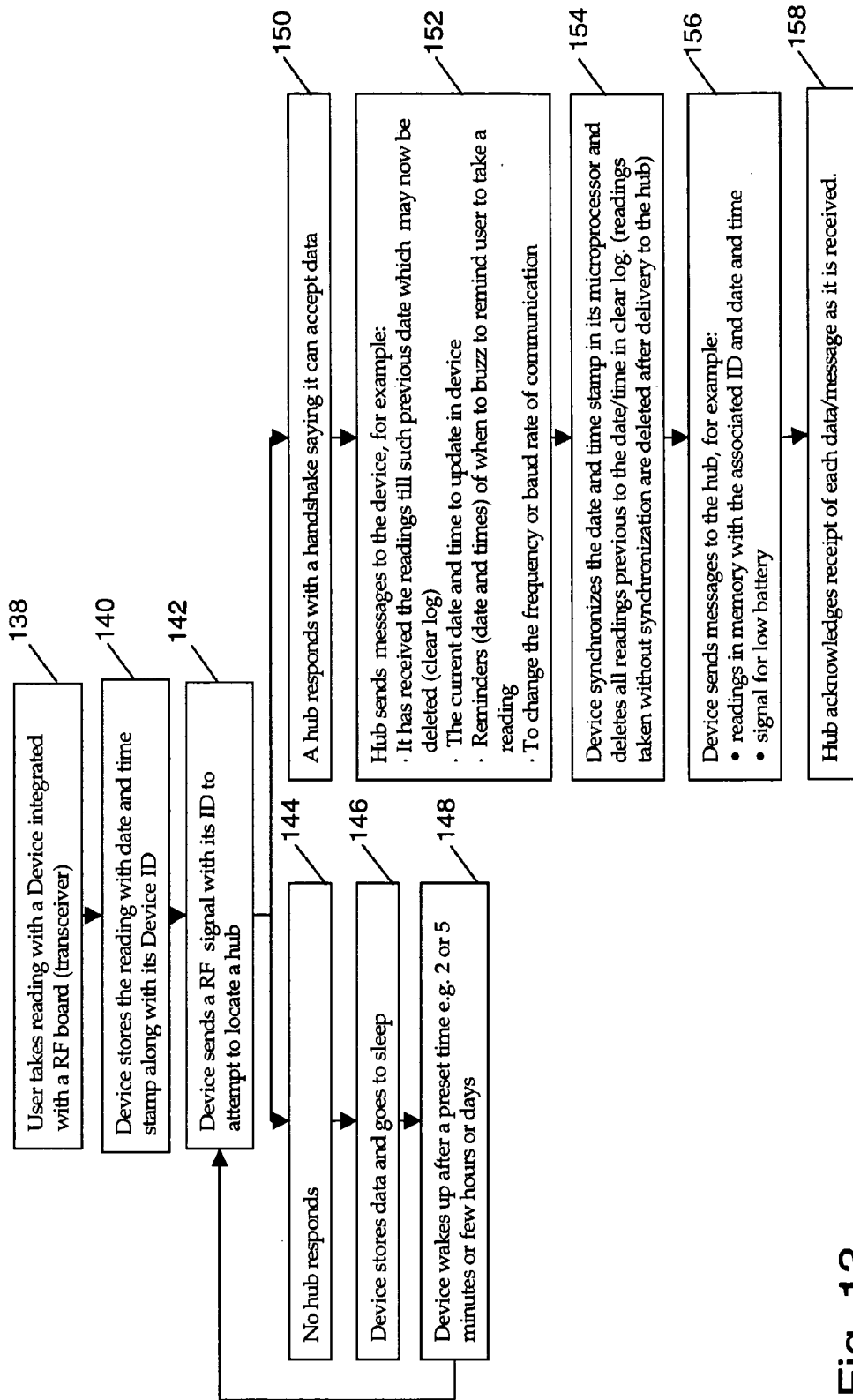


Fig. 13

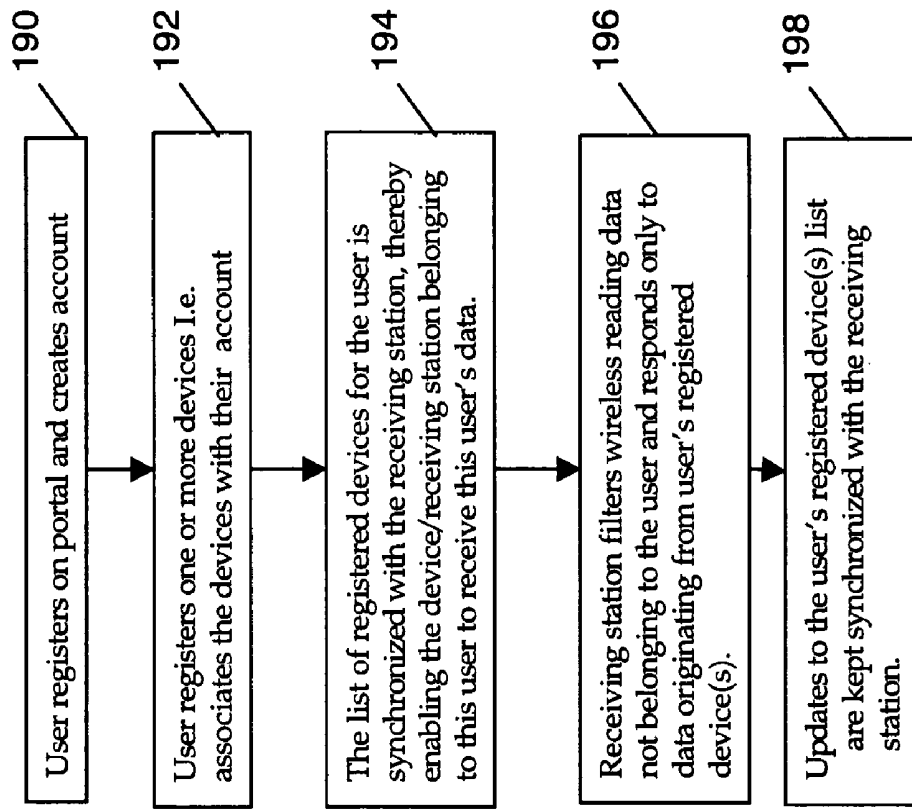


Fig. 14

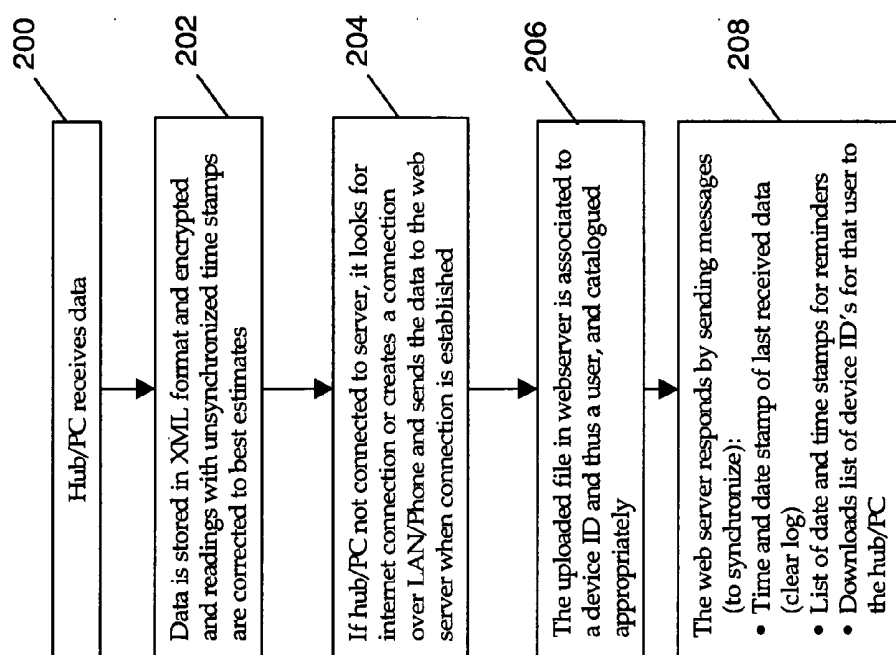


Fig. 15

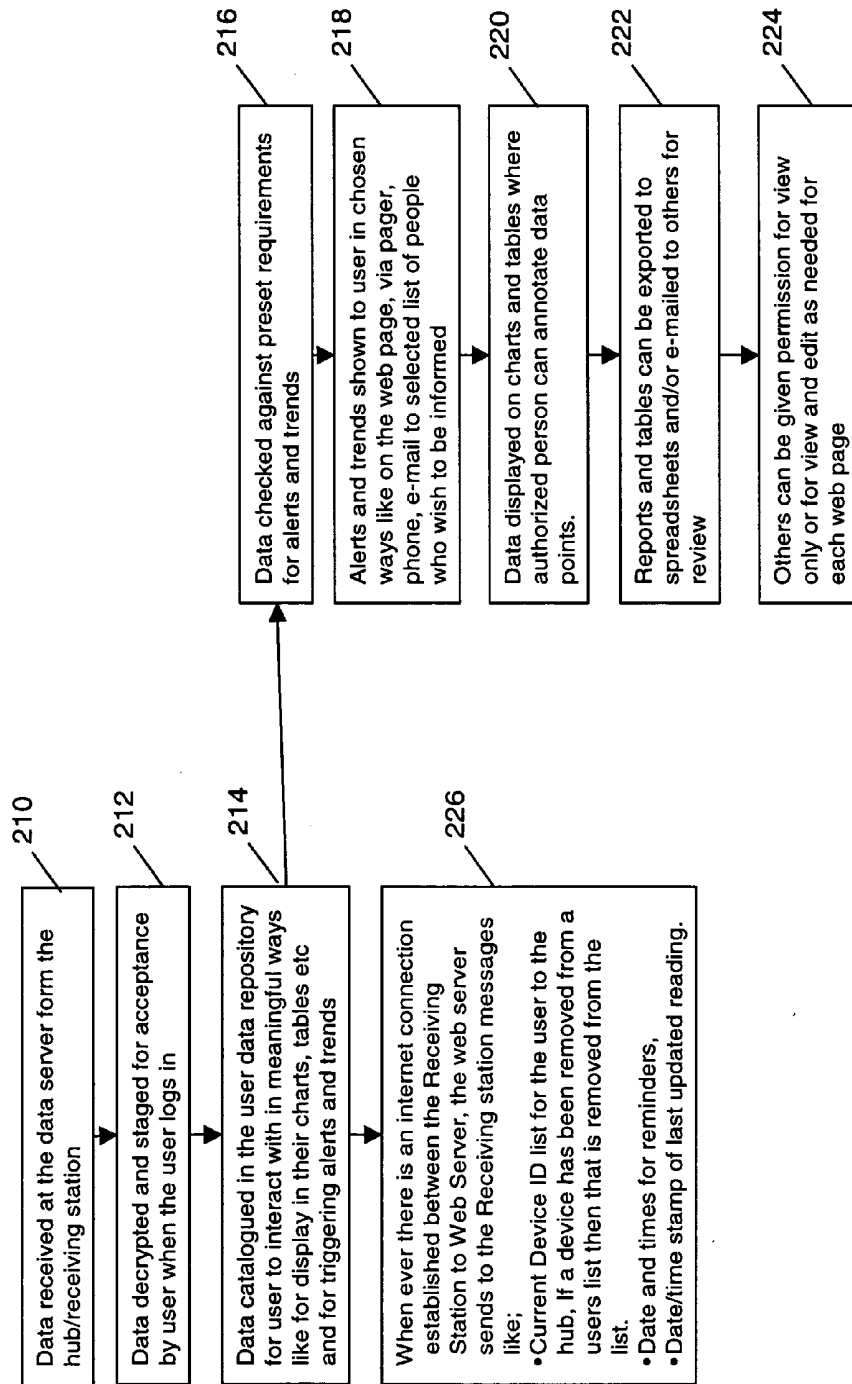


Fig. 16

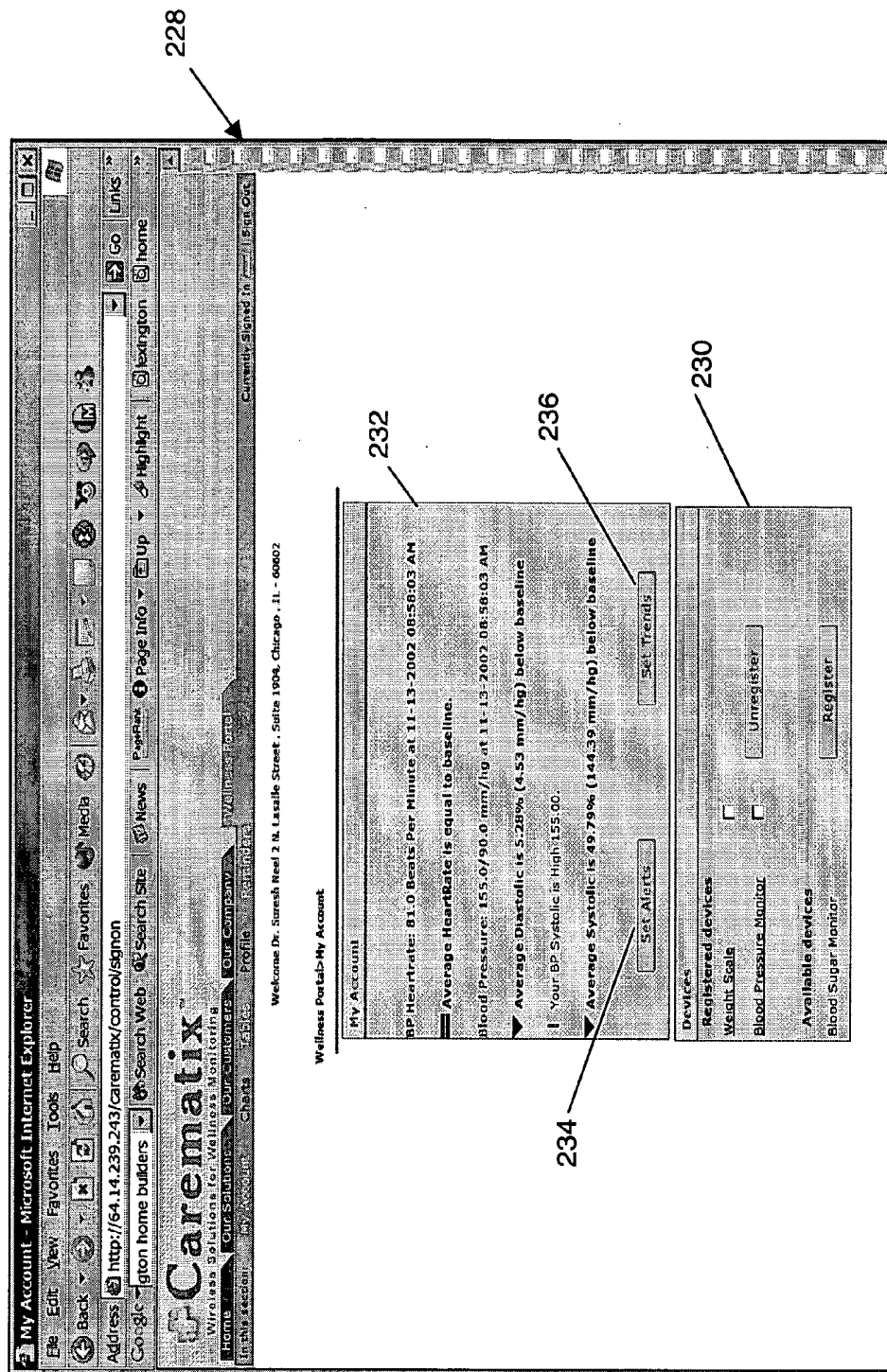


Fig. 17

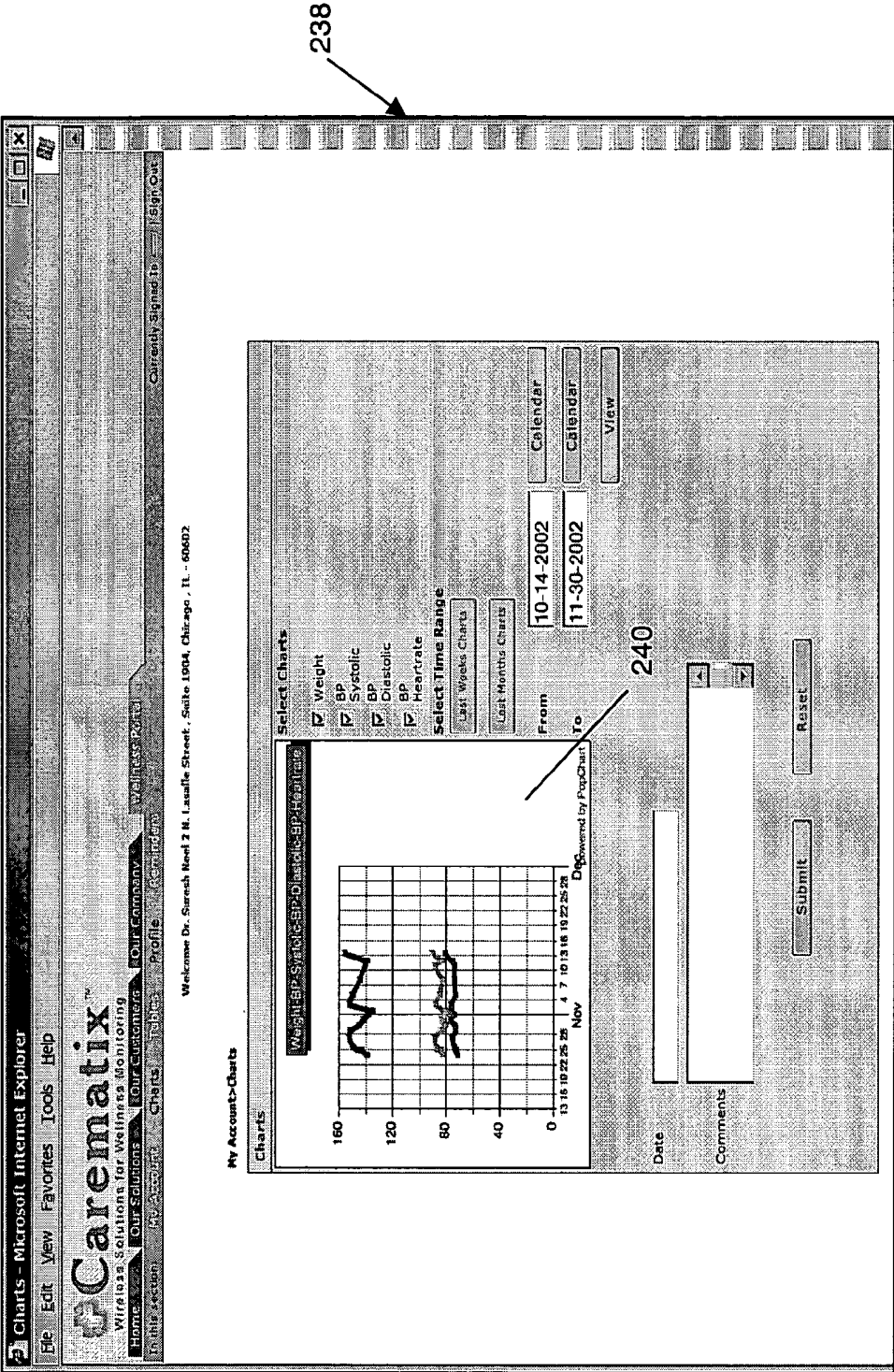


Fig. 18

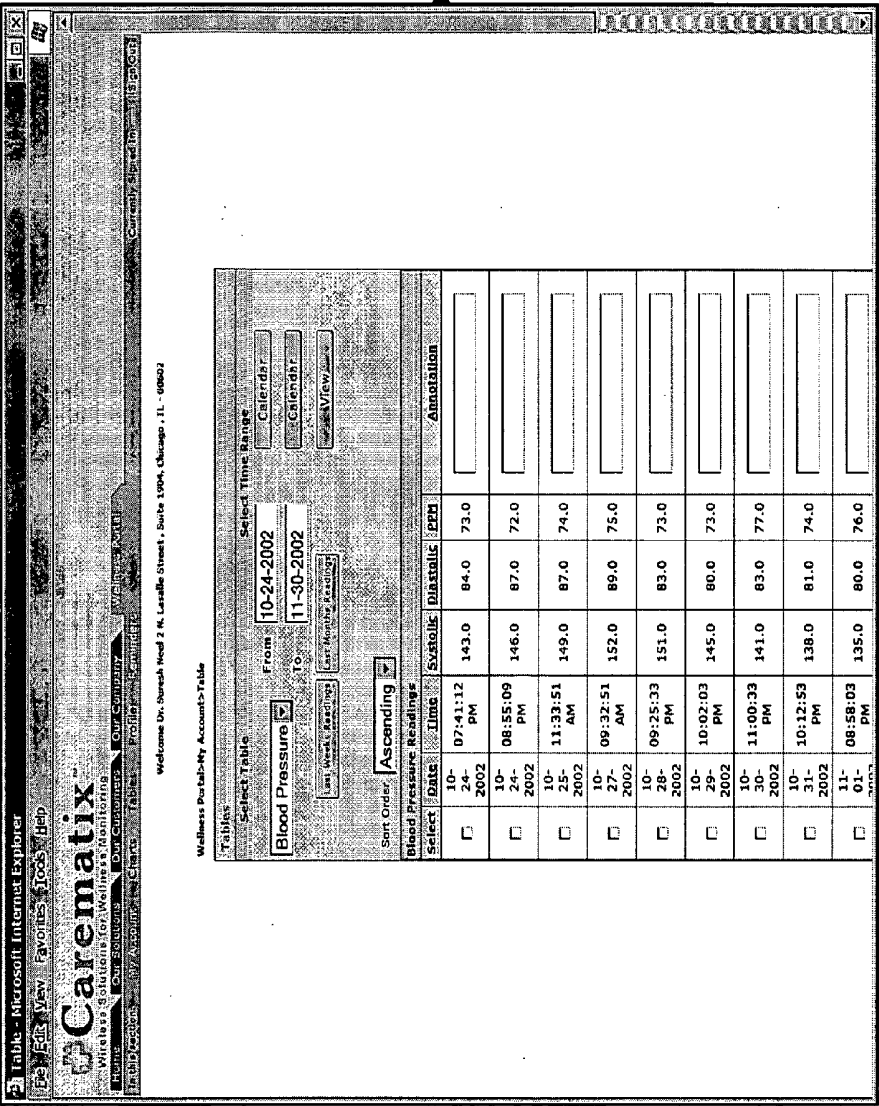


Fig. 19

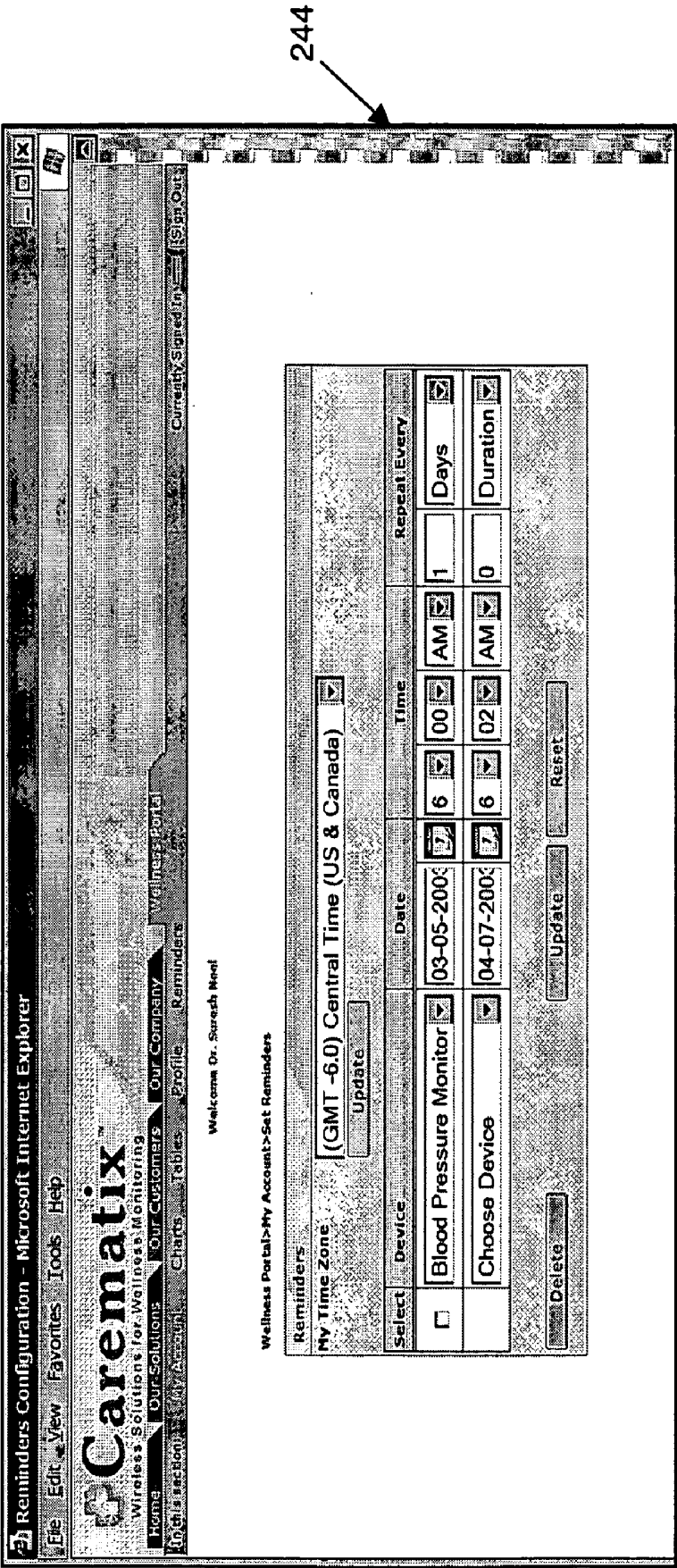


Fig. 20

Alert Configuration - Microsoft Internet Explorer

Home Edit View Favorites Tools Help

Carematix

Wireless Solutions for Wellness Monitoring

Home Our Solutions Our Customers Our Company Wellness Portal

Introduction My Account Charts Tables Profile Reminders

Welcome Dr. Suresh Neel

Wellness Portal > My Account > Set User Alerts

Set User Alerts

Select Alert	Over	Send To
Weight		
<input checked="" type="checkbox"/> if greater than	200.0 Pound for	Readings > My Account >
<input type="checkbox"/> if less than	Pound for	Readings > My Account > Submit > Reset >
BP Systolic		
<input checked="" type="checkbox"/> if greater than	150.0 mm/hg for	Readings > My Account >
<input type="checkbox"/> if less than	mm/hg for	Readings > My Account > Submit > Reset >
BP Diastolic		
<input checked="" type="checkbox"/> if greater than	100.0 mm/hg for	Readings > My Account >
<input type="checkbox"/> if less than	mm/hg for	Readings > My Account > Submit > Reset >
BP Heart rate		
<input checked="" type="checkbox"/> if greater than	100.0 Beats Per Minute for	Readings > My Account >
<input checked="" type="checkbox"/> if less than	45.0 Beats Per Minute for	Readings > My Account > Submit > Reset >

Fig. 21

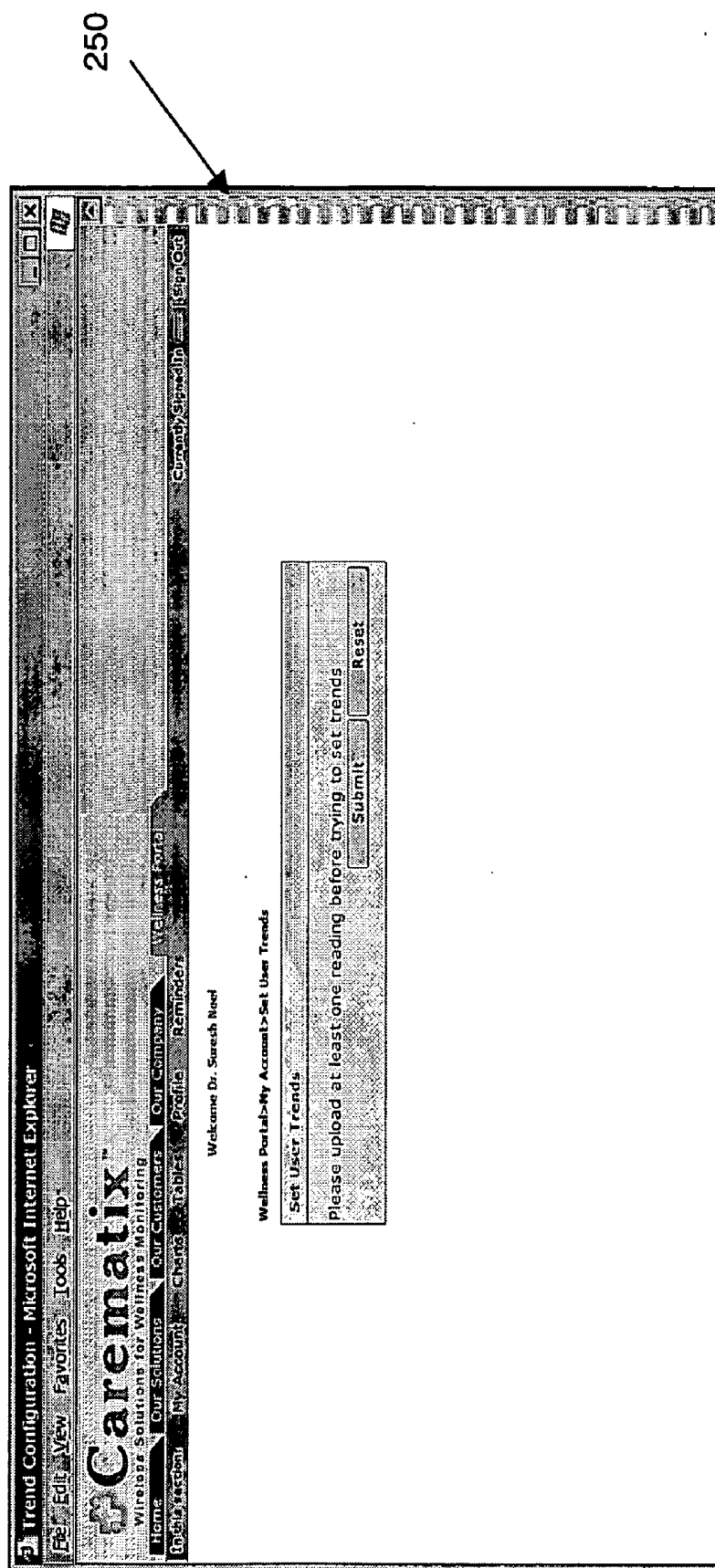


Fig. 22

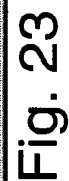


Fig. 23

254

Welcome to Carematix - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Carematix
Wireless Solutions for Wireless Monitoring
Home Our Solutions Our Customers Our Company
Administration/Tools/Reports

User	Registered Devices	User Number	Manufacturer Name	Serial Number	Model Number
tasterone	Blood Pressure Monitor	0	Carematix	010101	1024
taster90	Weight Scale	0	Carematix	1048590	2048
Jennith	Blood Sugar Monitor	0	Carematix	1048576	3192
jdoo	Blood Pressure Monitor	0	Carematix	1048586	1024
	Blood Sugar Monitor	0	Carematix	1048588	3192
sparg	Blood Pressure Monitor	0	Carematix	1048591	1024
neel	Blood Pressure Monitor	0	Carematix	1048658	1024
	Weight Scale	0	Carematix	1048608	2048
vince	Blood Pressure Monitor	0	Carematix	12345878	1024
whatcher	Blood Pressure Monitor	0	Carematix	1048590	1024
taster2	Weight Scale	0	Carematix	1048578	2048
default	Blood Pressure Monitor	0	Carematix	1048576	1024
taster5	Weight Scale	0	Carematix	1048579	2048
taster1	Blood Pressure Monitor	0	Carematix	1048598	1024
taster10	Blood Pressure Monitor	0	Carematix	123456	1024
	Weight Scale	0	Carematix	654321	2048
vince d	Weight Scale	0	Carematix	1	2048

Fig. 24

METHOD AND APPARATUS FOR REMOTELY MONITORING THE CONDITION OF A PATIENT

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit and priority of U.S. provisional patent application No. 60/372,894, filed on Apr. 16, 2002.

BACKGROUND OF THE INVENTION

COMPUTER LISTING APPENDIX

[0002] This application includes a Computer Listing Appendix on compact disc, hereby incorporated by reference.

[0003] 1. Field of the Invention

[0004] The present invention relates to a patient monitoring system and more particularly to a patient monitoring system for monitoring various physiological characteristic data of a patient, such as blood pressure, pulse rate, blood glucose, weight and others, which wirelessly transmits such data to a hub, located near the patient, which, in turn transfers the data automatically to a remote server, for example, over a public or private communications network, which, in one embodiment, the remote server is configured as a web portal which selectively allows access to patient data by selected third party users, such as physicians, clinicians, patients, and/or relatives, and, in addition provides increased functionality relative to known systems by enabling trends of the patient data to be developed as well as automatically generate communications with the patient or other third parties by way of e-mail or pager when predetermined thresholds for selected physiological characteristics are exceeded and/or to remind the patient to take physiological measurements.

[0005] 2. Description of the Prior Art

[0006] Healthcare costs have been increasing at a tremendous rate for the past decade, far exceeding the rate of inflation. The compound average growth rate for healthcare spending over the past decade was 6%, amounting to nearly \$1.3 trillion in the year 2001. Chronic disease patients, whose numbers have doubled during the same decade, account for nearly \$700 billion of this spending. Managed care organizations have begun to seek help from disease management companies to contain the spending on chronic diseases. Disease management companies thus have developed systems to monitor the chronically ill patients and help lower healthcare spending by improving patient compliance to medication schedules thereby lowering the hospitalization rates.

[0007] Various types of patient monitoring systems are known. For example, U.S. Pat. Nos. 5,810,747 and 5,694,940 disclose patient monitoring systems in which the patient monitoring devices are hard wired to a local hub, which, in turn, is connected to a remote station, for example, over a communication network. The fact that the patient monitors are hard wired to the local station significantly reduces the utility of such systems. For example, such systems are obviously not suitable for ambulatory patients and many applications where it may be desired to remotely monitor the physiological characteristics of a patient outside of a non-clinical environment.

[0008] Accordingly, various systems have been developed in which the patient monitoring device, normally worn by a patient, is connected by way of a wireless link to a local hub, which, in turn, is connected to a remote station or server by way of a public communication network, such as the plain old telephone system (POTS). Examples of such systems are disclosed in U.S. Pat. Nos. 3,882,277; 5,522,396; and 6,093,146. Each of these systems include a patient monitoring device, connected to a local hub by way of a wireless link, which, in turn, is connected to a remote server by way of a public communication link, such as POTS. Such systems depend on the use of telephone modems which require patients to place a phone receiver into a modem cradle and dial up the remote server. Such systems have been found to be far too complicated and difficult for elderly and critically ill patients.

[0009] In order to resolve this problem, patient monitoring systems, for example, as disclosed in U.S. Pat. No. 6,336,900, have been developed, which not only provide a wireless link between the patient monitor and the local hub, but also automatically dial and connect to the remote server to facilitate the transfer of the physiological data to the remote server. Unfortunately, the functional capability of such systems is relatively limited. For example, such systems only provide limited access to the patient data. In addition, such systems can not be used to provide reminders to patients to take readings or provide messages to the patients or third parties when the physiological characteristics of a patient exceed predetermined thresholds. Thus, there is a need for a patient monitoring system for monitoring the physiological characteristics of a patient that provides enhanced functionality and expanded access to the healthcare data while not tethering the patient to a local hub, thereby improving patient compliance, affording the healthcare provider the ability to capture adverse events sooner and avoid costly emergency care and reduce costly home health visits.

SUMMARY OF THE INVENTION

[0010] Briefly, the present invention relates to a patient monitoring system which provides enhanced functional capability relative to known systems and provides a wireless communication link between a patient monitoring device, worn by a patient, and a local hub. The patient monitoring system in accordance with the present invention is adapted to monitor various patient physiological characteristics, such as blood pressure, pulse rate, blood glucose, weight, pulse oximetry and others. The data from the patient monitoring device is wirelessly transmitted to a local hub, which, in turn, is configured to automatically transfer the data to a remote server, for example, over a public or private communications network. In one embodiment of the invention, the server is configured as a web portal to selectively allow access to such patient physiological data by designated third parties, such as physicians, clinicians, relatives and the patient themselves. In accordance with another important aspect of the invention, the system provides for enhanced functionality relative to known systems and allows trends of the physiological data to be selectively generated. In addition, any third party can set thresholds to automatically notify the patient or other third parties by various communication methods including e-mail and/or pager when a particular physiological characteristic exceeds a predetermined threshold. The system is also configured to provide reminders to patients to take readings.

DESCRIPTION OF THE DRAWINGS

[0011] These and other advantages of the present invention will be readily understood from the following specification and attached drawing wherein:

[0012] **FIG. 1** is a block diagram of the patient monitoring system in accordance with the present invention.

[0013] **FIG. 2** is a block diagram of an alternate embodiment of the invention, illustrated in **FIG. 1**, illustrating the use of the invention in a multi-user, multi-hub environment.

[0014] **FIG. 3** is a block diagram of an alternate embodiment of the invention, illustrated in **FIG. 1**, for a multi-user environment, single hub environment.

[0015] **FIG. 4** is a block diagram of another alternate embodiment of the invention illustrated in **FIG. 1** for a multi-user, single hub environment.

[0016] **FIG. 5** is a block diagram of an exemplary physiological transducer for relatively complex transducers such as an electrocardiograph for use with the present invention.

[0017] **FIG. 6** is a block diagram of exemplary self contained hub or base station for use with the present invention.

[0018] **FIG. 7** is an exemplary schematic diagram of a relatively simple physiological transducer such as a blood pressure monitor, for use with the present invention.

[0019] **FIG. 8** is an exemplary schematic diagram of an alternative hub or base station that is adopted to be connected to a middleware device, for use with the present invention.

[0020] **FIG. 9A** is a schematic diagram of the hub illustrated in **FIG. 6**.

[0021] **FIG. 9B** is a schematic diagram of the transceiver incorporated into the hub illustrated in **FIG. 9A**.

[0022] **FIG. 10** is a schematic diagram of the exemplary physiological transducer illustrated in **FIG. 7**.

[0023] **FIG. 11** is an exemplary block diagram of the server application architecture in accordance the present invention.

[0024] **FIG. 12** is a diagram illustrating an exemplary message exchange between a patient monitoring device and a hub for the system in accordance with the present invention.

[0025] **FIG. 13** is an exemplary software flow diagram for synchronizing the patient monitoring device to a hub in accordance with the present invention.

[0026] **FIGS. 14 and 15** are exemplary flow diagrams illustrating the hub to web server synchronization.

[0027] **FIG. 16** is an exemplary flow diagram illustrating the data server/portal to user interaction in accordance with the present invention.

[0028] **FIGS. 17-24** are exemplary screen shots for use with a web-based embodiment of the present invention.

DETAILED DESCRIPTION

[0029] The present invention relates to a patient monitoring system for monitoring various physiological characteristics of a patient, such as blood pressure, pulse rate and others, as discussed below. The system includes a portable physiological transducer or patient monitoring device that can be worn by a patient in both single user and multi-user applications. The patient monitoring device is adapted to monitor physiological characteristic data of the patient and wirelessly transmit the data to a local hub or base station. In order to reduce the complexity of the system, the hub or base station is configured to automatically transfer the data to a remote server by way of a public or private communication network. In accordance with an important aspect of the invention, the system is configured to selectively enable access to a patients' physiological data by various selectable third party users, as well as the patient. In accordance with another aspect of the present invention, the patient monitoring system provides enhanced functionality relative to known systems. For example, the system in accordance with the present invention enables selectable third party users, as well as the patient, to monitor trends in the data for the physiological characteristics, as well as set alarms, which can automatically generate communications with the patient and/or third parties when a physiological characteristic exceeds a predetermined value. These communications can be, for example, by e-mail or pager. In one embodiment of the invention, a bi-directional communication link is provided between the patient monitoring device and the local hub. This bi-directional communication link allows communications to be sent to the patient monitoring device, for example, to remind a patient to take a physiological reading. The bi-directional communication link also allows handshaking between the patient monitoring device and the local hub to insure the transfer of physiological data to the local hub.

[0030] As will be discussed in more detail below, the system in accordance with the present invention can be utilized in a wide variety of applications. For example, **FIG. 1** illustrates a single user/single hub application. **FIG. 2** illustrates a multiple user/multiple hub application. **FIGS. 3 and 4** illustrate a multiple user/single hub environment, for example, a cardiac rehabilitation facility in which each of the patients wear a portable heart monitor during cardiac rehabilitation.

[0031] **FIGS. 5-10** illustrate exemplary hardware diagrams for the portable physiological transducer and the hub for use with the present invention. In particular, **FIG. 5** illustrates an exemplary block diagram of a portable physiological transducer for relatively complex transducers, such as a electrocardiograph. **FIG. 7** illustrates an exemplary block diagram of a relatively simple portable physiological transducer, for example, a blood pressure transducer. Both transducers illustrated in **FIGS. 5 and 7** may include an audio or visual indicating device to enable reminder messages to be sent to the patient monitoring device. **FIG. 6** illustrates a block diagram of an exemplary self contained hub or base station for use with the present invention. **FIG. 8** illustrates a block diagram of an exemplary hub that is adapted to be connected to a middleware device. **FIG. 9A** is a schematic diagram of a self contained hub illustrated in **FIG. 6**. **FIG. 9B** is a schematic diagram of the transceiver incorporated into the hub, illustrated in **FIG. 9A**. **FIG. 10**

illustrates an exemplary schematic diagram for a relatively simple patient monitoring device, such as a blood pressure monitor.

[0032] FIGS. 11-16 relate to the software. FIG. 11 is a block diagram of the server application architecture. FIG. 12 is an exemplary block diagram illustrating an exemplary message exchange between a portable physiological transducer or patient monitoring device and the self contained hub in accordance with the present invention. FIG. 13 is an exemplary software flow diagram illustrating the synchronization of the patient monitoring device with the hub in accordance with the present invention. FIGS. 14 and 15 illustrate the synchronization of the hub to the web server. FIG. 16 illustrates an exemplary flow diagram illustrating the data server to user interaction in accordance with the present invention.

[0033] In one exemplary embodiment of the invention, the system is configured as a web portal. In this embodiment, the server is configured as a web server and selectively allows third party access as well as access by the patient to the physiological data transmitted by the portable patient monitoring device. In addition, the system can be used to provide aural and/or visual signals to the patient monitoring device to remind a patient to take readings. As discussed above and as will be discussed in more detail below, various third parties, such as the physician and/or relatives, as well as the patient, can not only access the physiological data of the patient but also view trends of such data and also set alarms which automatically generate messages to either the patient and/or other third parties when the physiological characteristics of the patient exceed a preset value. In accordance with this embodiment, various screen shots as illustrated in FIGS. 17-23 are provided.

[0034] System

[0035] FIGS. 1-4 illustrate various system applications for the patient monitoring system in accordance with the present invention. In particular, single user/single hub applications, as well as multiple user/single hub and multiple user/multiple hub applications are within the broad scope of the present invention. As such, the patient monitoring system in accordance with the present invention can be used in a relatively wide variety of applications from non-clinical applications, such as a home healthcare and health club applications, as well as various clinical applications.

[0036] Referring first to FIG. 1, an embodiment of a patient monitoring system is illustrated and generally identified with the reference numeral 30. The patient monitoring system 30 illustrates a single user/single hub application, for example, a home healthcare application. In this application, the patient monitoring system 30 includes a portable physiological monitor or patient monitoring device, identified with the reference numeral 32, a local hub 34, a middleware device 36, which may or may not be incorporated into the hub 34, as well as a remote server 38. The patient monitoring device 32 can be one of multiple portable physiological transducer, such as a blood pressure monitor, heart rate monitor, weight scale, thermometer, spirometer, single or multiple lead electrocardiograph (ECG), a pulse oxymeter, a body fat monitor, a cholesterol monitor or a signal from an exercise machine, such as a heart rate. As will be discussed in more detail below, the patient monitoring device 32 is a portable device worn by the patient and includes a single or

bi-directional wireless communication link, generally identified with the reference numeral 40, for transmitting data from the patient monitoring device 32 to the local hub or receiving station 34 by way of a wireless radio frequency (RF) link using a proprietary or non-proprietary protocol.

[0037] In this embodiment of the invention, the system includes a local hub or receiving station 34 which wirelessly receives data from the patient monitoring device 32 and automatically communicates it to a middleware device 36, which may be a personal computer, an Internet gateway, a home gateway, a phone, a video phone, or a phone modem. Alternatively, the middleware device 36 may not be required if the hub 34 includes its own processing equipment, as will be discussed below. In particular, the middleware device 36 may form part of the hub 34.

[0038] The hub or receiving station 34 may be a receiver or transceiver for receiving data from the patient monitoring device 32. Communication, if required, between the hub 34 and the middleware device 36, may be over various communication links, such as a direct connection, such a serial connection, USB connection, Firewire connection or may be optically based, such as infrared or wireless based, for example, home RF, IEEE standard 802.11a/b, Bluetooth or the like.

[0039] The middleware device 36 transfers the data received by the hub or receiving station 34 to a remote server 38. The communication link between the middleware device 36 and the remote server may be by direct connection, such as a serial connection, USB, Firewire or optically based such as infrared or wireless based such as home RF, IEEE standard 802.11b, Bluetooth or others. For longer distances, a communication link between the middleware device 36 and a remote server 38 may be by DSL, T-1 connection over a private communication network or a public information network, such as the Internet.

[0040] In accordance with one aspect of the invention, the remote server 38 may be configured to provide third party access to the patient physiological data forming a portal. The portal, for example, a web portal, allows the patient and/or third party users, such as relatives and physicians, to interact with the patient data in various ways as discussed below.

[0041] FIG. 2 illustrates a multi-user/multi-hub application of the present invention, generally identified with the reference numeral 42. In this embodiment, three (3) exemplary patient monitoring devices 44, 46 and 48 are used to transmit data to multiple hubs 50 and 52. The hubs 50 and 52 transfer the data to middleware devices 54 and 56, which, as discussed above may be a personal computer (PC) 54 or a phone line, web phone or video phone 56 or may be integrated into the hubs 50 and 52. In this embodiment, the middleware devices 54 and 56, in turn, transmit the data to a remote server 58. In accordance with an important aspect of the invention, various third parties, such as a physician or relatives, can selectively access the patient physiological data in the server 58, as indicated by the function block 60.

[0042] FIGS. 3 and 4 illustrate a multi-user/single hub application, for use in multiple user applications, such as a retirement home. Two embodiments of this application are contemplated as illustrated in FIGS. 3 and 4. Referring first to FIG. 3, the patient monitoring system is generally identified with the reference numeral 62. The patient monitoring

system 62 includes a patient monitoring device 64, a hub 66, an ID reader 68, a middleware device 70 and a remote server 72. The patient monitoring device 64, hub 66, middleware device 70 and the server 72, as well as the communication links therebetween, are as described above. In the exemplary application illustrated in FIG. 3, the ID reader 68 is operatively connected to the hub 66 to accommodate different users in a multiple user environment. In such an application, each user is given a unique ID. This ID is input into the hub 66. Various types of identification are contemplated. For example, the ID reader 68 may be a swipe card, RF tag or push button. In this embodiment, the user ID is read from the ID tag of the user at the same time physiological data is being read from the patient monitoring device 64.

[0043] FIG. 4 illustrates an alternative embodiment of the invention illustrated in FIG. 3. This embodiment is generally identified with the reference numeral 74. The patient monitoring system 74 includes a patient monitoring device 76, a hub 78, a server 82 and an ID reader 84. The patient monitoring system 74 is similar to the patient monitoring system 62, illustrated in FIG. 3, except that the patient monitoring system 74 is configured such that the ID reader 84 is coupled to the patient monitoring device 76. In this embodiment, the user ID and physiological data is transmitted together to the hub 78. In lieu of the ID reader, the patient monitoring device 76 can be configured such that the device transmits a unique ID with the physiological data. Thus, different users wearing different devices can easily be correlated by the system.

[0044] Hardware

[0045] The hardware for the patient monitoring device, as well as the hub, is illustrated in FIGS. 5-10. As mentioned above, FIGS. 5 and 7 relate to a patient monitoring device, generally identified with the reference numerals 84 and 85, for use with the present invention. FIGS. 6 and 8 illustrate exemplary hubs for use with the present invention. FIG. 10 is an exemplary schematic diagram of the patient monitoring device 84, illustrated in FIG. 5.

[0046] Exemplary Patient Monitoring Devices

[0047] Various embodiments of the patient monitoring devices 84 and 85 are contemplated. Both embodiments may optionally include a visual or audio indicating device to allow visual and/or audio communications to be sent from the system to the patient monitoring device. The patient monitoring device 85 (FIG. 7) is representative of patient monitoring devices which utilize relatively simple transducers, such as a blood pressure monitor, heart rate monitor and others. The patient monitoring device 85 includes a transducer 88, such as a blood pressure transducer, coupled to a communication module 90. The communication module 90 may include a microprocessor 92, a memory real time clock 94, as well as a transmitter for unidirectional communications or a transceiver 96 which enables bi-directional communication between the patient monitoring device 85 and a local hub, as discussed above. An audio or visual indicating device 93 may also be provided to enable alerts or messages to be sent to the patient monitoring device 85. The transducer 88 may be directly wired to the communication module 90, as indicated by the electrical leads 98. An antenna 100 is provided in order to provide a wireless link to the local hub, as discussed above. A power supply 102, such as a battery may be provided.

[0048] The MCU 92 may be, for example, Motorola model number 68HC908GP32. The radio 96 may be, for example, a Xemics model number XE 1201. The memory 94 may be, for example, an Atmel model number CAT 24AA65. The real time clock may be a Dallas Semiconductor model number DS 1675.

[0049] More sophisticated patient monitoring devices 84, as illustrated in FIG. 5, such as electrocardiograph (ECG) devices, may include one or more transducers which directly communicate with a microprocessor 106. An input/output (I/O) device 108 may be directly coupled to the microprocessor 106, for example, for receiving data from electrodes attached to the patients' body by way of electrical leads (not shown). As is well known in the art, this data is received by the microprocessor 106 and configured by the transducer 104 and communicated to a communication module 110. The communication module 110 includes its own microprocessor 112, a memory 114 such as an electrically erasable programmable read only memory (EEPROM), a radio 114 for bi-directional communication or simply a transmitter for unidirectional communication, as well as an antenna 116.

[0050] Various physiological transducers can be integrated into the system. These transducers are similar to various commercially available transducers such as: a blood pressure transducer, such as A&D blood pressure cuff, model number UA-767PC; a weight scale, such as an A&D scale, model number UC-321; a blood glucose meter, such as a LifeScan ONE TOUCH® Ultra; a pulse oximeter, such as a PalmSat Model 2500; a spirometer, such as a Micro Direct model MICRO DL; a prothromben time test for Coumadin® therapy, such as a PT/NR Protime Microcoagulation System; and a cholesterol monitor, such as a LSP 3101 Personal Cholesterol Monitor Kit. Virtually any physiological transducer which generates an electrical signal can be implemented in the system.

[0051] Hub Hardware

[0052] Exemplary block diagrams for the hub for use with the present invention is illustrated in FIGS. 6 and 8. Referring first to FIG. 8, the hub 87 includes a radio 116, a microprocessor 118 and a memory/real time clock 120. The hub 87 also includes an antenna 120 and a power supply 122 if required. This embodiment of the hub 87 is for an embodiment in which there is a separate communication link as discussed above to a middleware device 122 for embodiments in which the middleware device 122 is separated from the hub 87.

[0053] FIG. 6 illustrates an alternative embodiment of the hub, identified with the reference numeral 86 in which the middleware device is integrally formed with the hub. In this embodiment, the hub 86 includes a radio 126 and an antenna 128 for receiving data, wirelessly from the patient monitoring device. The hub 86 also includes a microprocessor 130, a real time clock 132, a memory device such as a EEPROM 134 and a modem 136. The microprocessor 130, radio 126, real time clock 132 and memory 134 may be as discussed above. The modem may be a ConnectOne embedded iModem, Model Number iM336UC-EM.

[0054] FIG. 9A is an exemplary schematic diagram of the hub 86 illustrated in FIG. 6. FIG. 9B is an exemplary schematic diagram of the transceiver 126 illustrated in FIG. 9A.

[0055] Software

[0056] FIGS. 11-16 illustrate the system software. FIG. 11 illustrates the server application architecture. FIG. 12 illustrates the message exchange between a patient monitoring device and the hub. FIG. 13 illustrates the software for synchronizing data from the patient monitoring device to the hub. FIG. 14 illustrates the synchronization between the hub and the remote server and specifically relates to user registration. FIG. 15 illustrates the software flow diagram for interaction between the hub and the web server. FIG. 16 illustrates the software flow diagram for interaction between the user and data server/portal.

[0057] Referring first to FIGS. 13-16, FIG. 13 illustrates the software for controlling the patient monitoring device and transmitting the data from the patient monitoring device to the hub. Initially, in step 138, on power up, the patient monitoring device, either on its own or in response to patient input, begins taking readings of a patient physiological characteristic. These readings are stored in the patient monitoring device along with a time stamp and device ID in step 140. In step 142, the patient monitoring device sends out RF signals with its ID in an attempt to locate a local hub. If no hub responds as indicated in step 144, the patient monitoring device stores the data and goes to sleep in step 146. Subsequently, after a preset time, the device awakens in step 148 and returns to step 142 and attempts to locate a hub.

[0058] Once a hub is located, the hub responds with a handshake signal indicating that it is available to receive data as indicated in step 150. After the handshake, the hub may communicate various information to the patient monitoring device in step 152. For example, the hub may send a message to the patient monitoring device indicating that previous readings have been received. The hub may also send a message to the patient monitoring device with the current date and time to update that device. In addition, the hub may send reminders at specified dates and times to remind a patient when to take readings for those patient physiological devices which do not constantly take readings. Finally, the hub may also send a message to the patient monitoring device to change the frequency or baud rate of the communication. Subsequently, in step 154, the patient monitoring device synchronizes its date and time stamp and deletes all previous readings to the date and time in order to initialize the patient monitoring device. After the patient monitoring device is initialized in step 154, the patient monitoring device then sends the data to the hub in step 156 with its associated ID and date and time. The patient monitoring device may also send data to the hub indicating that its battery is low. In step 158, the hub acknowledges receipt of each message received.

[0059] The message exchange between the patient monitoring device and the hub is illustrated diagrammatically in FIG. 12. Initially, as indicated in box 160, the patient monitoring device on power up sends a query to the hub, as indicated by the message 162. The hub in response to that query checks whether the ID number for the patient monitoring device is listed, as indicated by the block 164. In addition, the hub may also communicate its local date and time as well as control information with respect to clearing the log and frequency change, as message 164. This information is acknowledged and acted upon by the patient monitoring device, as indicated by the message 168. After

the patient monitoring device is time synchronized with the hub, the patient monitoring device begins sending data to the hub along with its ID and time stamp, as indicated by the message 170.

[0060] The above message exchange may be identified as a pre-cycle message exchange. Such a pre-cycle message exchange is identified as a time period when the patient monitoring transducer is not actively recording data such as during a time period when a blood pressure cuff has not been inflated.

[0061] After the data has been transmitted from the patient monitoring device to the hub, post cycle messages include repeating the messages 162 and 166 after each data cycle has been completed. In addition, a message 172 may be sent from the patient monitoring device to the hub for the first piece of data, indicated by the box 174. After the hub receives the data 174, it sends a message back to the patient monitoring device indicating that the data 174 has been received. Once the patient monitoring device acknowledges from the hub that the first bit of data has been received, additional data is sent by way of a message 178 to the hub, as indicated by the data block 180. The hub responds that the data 180 was received by way of a message 182 back to the patient monitoring device. After all of the data has been transmitted from the patient monitoring device to the hub, the patient monitoring device sends a message 184 indicating that the next bit of data is the last bit of data in the memory to be sent by the patient monitoring device by way of the message 184. The hub then acknowledges receiving the last bit of data 186 by acknowledging by way of message 188 back to the patient monitoring device 188 that it has received that data 186.

[0062] FIGS. 14 and 15 illustrate the synchronization between the hub and the remote server. In accordance with an important aspect of the invention, the server in this embodiment is configured as a portal, for example, a web portal, which enables third party users to selectively access patient physiological data transferred to the remote server. In this application, users register on the portal and create an account in step 190. In step 192, the user registers one or more devices (i.e. device ID's) with their account. Next, in step 194, the list of registered devices for a particular user is synchronized with the hub thereby enabling the hub to receive data for the specified device IDs. In step 196, the hub filters wireless communications from various patient monitoring devices and only responds to data originating from registered devices. In step 198, any updates of the registered devices with respect to a user's account are synchronized with the hub.

[0063] FIG. 15 illustrates a software diagram for transferring data between the hub and the remote server. Initially, in step 200, data is received by the hub as discussed above. In step 202, the data may be stored, for example, in XML format, for example, and optionally encrypted with readings with unsynchronized time stamps, which may be corrected to best estimates. In step 204, if the hub is not connected to the server, the server looks for an Internet connection or creates a connection over a LAN/phone line automatically and sends the data to the web server when connection is established. In step 206, data is uploaded to the web server and associated to a device ID and user. The web server responds in step 208 by sending messages to synchronize the

data including a time and date stamp of last received data (clear log); a list of date and time stamps for reminders; and may include a down load list of device IDs associated with that hub.

[0064] FIG. 16 is a software flow diagram used to control the data server/portal in connection with user interaction. Initially, in step 210, data is received by the data server from the hub. In step 212, the data may be decrypted and staged for acceptance by the user after a user logs in. Subsequently, in step 214, the data is cataloged in a user data repository or database to enable the user to interact with the data in meaningful ways like display charts, tables and trigger alarms and trends. As indicated in step 216, the data may be checked against preset requirements for alerts and trends. These alerts and trends may be shown to the user in step 218 in various ways on a web page, a pager, a telephone, an e-mail to a selected list of people to be informed. The web server may then display the charts and tables in step 220. The reports and tables can be exported to spreadsheets and/or e-mailed to third parties in step 222 for review. In step 224, the server allows additional third party access for the various data in step 224.

[0065] As indicated in step 226, when connection, such as Internet connection, is established between the hub and the server, the server sends the hub various messages. These messages may include the current device ID list for the user to the hub. The messages may also include date and time stamps for reminders and the date/time stamp of the last updated reading.

[0066] Web Pages

[0067] In an exemplary embodiment, as discussed in detail below, the remote server may be configured as a web portal. Exemplary web pages for a patient monitoring system are illustrated in FIGS. 17-24. In particular, FIG. 17 illustrates exemplary home page generally identified with the reference numeral 228. This web page 228 includes a registration box 230 and an account box 232. The registration box 230 allows patients to register and unregister various physiological characteristic measurements. The account box 232 provides a summary of a patients' alert and trends. The account box 232 includes links 234 and 236 to screens where alerts and trends can be set.

[0068] FIG. 18 illustrates a exemplary web page 238 which allows a user, either a patient or a third party user, to track data by way of a display window 240. The exemplary web page 238 allows a user to track data, view it by device type, change the time scale as well as annotate the data. FIG. 19 is an exemplary web page which allows a user to view data in tabular form. On this page, the time range and device data can be selected by the user. FIG. 20 is an exemplary web page 244 which enables a user to set reminders for the device to provide reminders, either aurally or visually at certain times or at periodic intervals. FIG. 21 is an exemplary web page 246 which allows a user to set alert parameters on absolute values and to specify how to receive the alerts via browser, e-mail, voicemail, pager, etc. In particular, "My Account" boxes generally identified with the reference numeral 248, allow the particular message link to be selected. FIG. 22 illustrates an exemplary web page 250. The web page 250 allows a user to set up parameters, baselines and trends. FIG. 23 illustrates an exemplary web page 252 which may be used by a clinical administrator to

monitor a summary status of all assigned patients including the last reading and any alerts generated. FIG. 24 illustrates an exemplary web page 254 for use by an administrator to view the devices registered by various users in the system.

[0069] Server Application Architecture

[0070] FIG. 11 illustrates the server application architecture. The server application architecture may include four levels or tiers 228, 230, 232, 234. The tier 228 may be a user or client tier which allows users to interact with the underlying application. This client tier translates user actions and inputs them as server requests and formats a server response, usually visual. The client tier may be a client, such as a browser, or a custom application running on a hub or specific devices such as wired, wireless handheld Personal Digital Assistants (PDAs). The client tier may also be a non-user action, such as automated services, such as Internet Explorer.

[0071] The web tier 230 makes application functionality available on the worldwide web. This tier 230 accesses data and business functionalities, manages screen flow and often encapsulates some user interaction. It may be used to decouple the client from the business logic tier 232 to provide a uniform service model of client request. The web tier interacts with other tiers using standard protocols. Other applications may take place of the client programs accessing the underlying application through its web tie, such as Apache/Tomcat.

[0072] The business logic tier 232 encapsulates the business logic and comprises the software application functionality. The logic is organized as a set of interacting objects allowing for additions and enhancement of the logic at any point while still preserving the interface of other tiers. This business logic tier 232 may be implemented using J2EE, Microsoft.NET or a custom architecture, such as Tomcat.

[0073] The data access tier 234 may be used to integrate the application with other enterprise information systems. This tier 234 provides data storage and other information services to the application. It may consist of a database, enterprise resource planning systems, mainframe transaction processors, legacy systems and enterprise integration technologies. Other tiers access these databases using industry standard or custom protocols such as Oracle.

[0074] Web Portal Details

[0075] As mentioned above, in one exemplary embodiment of the invention, a web portal (hereafter referred to as the portal) provides meaningful access to a patient's personal health data, as collected by the patient monitoring device(s) and stored in a server-based data repository (hereafter referred to as the repository). Some key mechanisms of the portal are listed below and are explained in further detail in subsequent sections. These key mechanisms include:

[0076] A mechanism for self-registration of devices by the user.

[0077] A mechanism to synchronize data between the Repository and device(s) via the Receiving Station.

[0078] A mechanism to allow the user to preview the readings received from the device(s) and reject erroneous readings.

- [0079] A mechanism to generate alerts against received data, based on user defined parameters.
- [0080] A mechanism to determine trends in received data, based on user defined parameters.
- [0081] A mechanism to match computed trends with visual perception of the trend.
- [0082] A mechanism to set reminders on the device(s). This mechanism can be extended to set up any arbitrary "event" on the device(s) e.g. local alert generation.
- [0083] A mechanism by which a clinician may monitor a group or a number of users via a summary "dashboard" of their readings data, with ability to drill-down into details for each user.
- [0084] A mechanism to enhance user (healthcare) education by providing a healthcare questionnaire dynamically created (and updated) based upon certain rules applied to recent user readings data.
- [0085] A mechanism to provide data feeds (i.e. export, either "raw" or aggregated) to external systems with transformations between appropriate data formats.
- [0086] Device Registration Through Portal
- [0087] Device registration is the process by which a patient monitoring device is associated with one or more users of the system. This mechanism is also used when provisioning devices for a user by a third party, such as a clinician (or their respective delegate). The mechanism is as follows:
- [0088] 1. The user (or delegate) identifies himself/herself by logging into the portal. The user must have previously registered an account created either by themselves or by delegates.
- [0089] 2. User (or delegate) registers a device by:
- [0090] a. Choosing from a list of kinds of devices available for registration.
- [0091] b. Providing identification information for the device such as manufacturer information, device model information, device serial number (available on device packaging) and optionally a hub number (available on hub packaging). The user may register more than one device at this point.
- [0092] c. Optionally setting up a service subscription for device(s) usage. This includes selecting service plans and providing payment information.
- [0093] 3. The device(s) are then associated with this user's account and a Device Control File (discussed below) comprised of device identification information is synchronized between the server and the Receiving Station on initialization (see the section Data Synchronization between Receiving Station and Repository). This enables the Receiving Station to respond to and accept input from the devices that have their identification information in the control file.
- [0094] 4. On each interaction between the hub and the server, the control file entries for the current user (or hub) are synchronized.
- [0095] 5. Similarly, whenever the user registers a device, its entry is added to the control file on the hub if not already there on the next synchronization interaction with the server.
- [0096] 6. No association information between the actual user (person) and the device is kept on the user's hub. This determination is made after the data is uploaded to the server.
- [0097] 7. The hub may be shared between users and data for all users resident on the hub is uploaded to the server on synchronization.
- [0098] Device Control File
- [0099] This file contains records in the format
- [0100] <device id>, <data update timestamp>
- [0101] where <device id> is all the information required to uniquely identify a device to the system (such as manufacturer number, model number, serial number and user number) and the <data update timestamp> is the timestamp (including date) that data for this device was last received by the server. Note that this is not the timestamp for which data was last uploaded from this particular hub (on which the control file resides) but is the timestamp for which the server last received data (from any hub) for the particular (registered) device. Additional information for controlling the device(s) e.g. reminder information is also maintained in this file.
- [0102] This file is internally maintained in XML format for ease of interfacing but is either kept encrypted or in a non-readable format on the hub for security reasons.
- [0103] User Data File
- [0104] This file contains records in the format
- [0105] <device id>, <reading timestamp>, <reading data>
- [0106] where <device id> is as described in the section Device Control File. The <reading timestamp> is the time of the reading and the <reading data> is a data record in a format specific to the type of reading. This file is internally maintained in XML format for ease of interfacing but is kept encrypted or in a non-readable format on the hub for security reasons.
- [0107] Data Synchronization between Receiving Station and Repository
- [0108] The hub and the Repository frequently synchronize data. The hub may use one of various transportation methods to connect to the repository e.g. using a PC as conduit or via a connection established using an embedded modem (connected to a phone line) or via another network-connected device (such as, but not limited to, a web-phone, video-phone, embedded computer, PDA or handheld computer). The mechanism is as follows:
- [0109] 1. When a user logs into the portal, the User Data File, resident on the hub, is uploaded to server. The server filters duplicate readings records.
- [0110] 2. Alternatively, the Data Synchronization Client may upload the data from the hub to the Server without user intervention automatic data synchronization without user intervention also happens in the case of a stand-alone hub.

[0111] 3. The data may be decrypted on the server, assigned to each user (by determining which user the device is registered with) and put into a staging area. This staging area is so that the user may optionally preview the data before accepting the values. The server also filters for duplicate reading values.

[0112] 4. The device control file is uploaded to the server.

[0113] 5. The server downloads updated device control file with last updated timestamps for each device in list. If device has been removed from the system, then its entry in the file is deleted. The old device list file is backed up.

[0114] 6. The data file is cleared/deleted.

[0115] Data Synchronization Client

[0116] There is often the need for the reading data to be uploaded to the server without explicit user intervention. The Data Synchronization Client runs on the hub to upload the user's data to the server while synchronizing the information contained in the control file.

[0117] Alerts

[0118] Users may set up alerts that are triggered when one or more reading meet a certain set of conditions, depending on parameters defined by the user. The mechanism is as follows:

[0119] 1. The user sets up an alert by choosing the condition that they would like to be alerted to and by providing the parameters (e.g. threshold value for the reading) for alert generation.

[0120] 2. Each alert thus set up has an interval associated with it. This interval may be either the number of data points or a time duration in units such as hours, days, weeks or months, depending on the type of reading for which this alert is being set up.

[0121] 3. The values of readings lying within the interval specified by the user must all either positively or negatively exceed the specified threshold for the alert to be generated i.e. there is an implied "and" across all the readings within the interval.

[0122] 4. The user chooses the destination where the alert may be sent. This destination may include the user's portal, e-mail, pager, voice-mail or any combination of the above.

[0123] 5. Specific, preset alerts are provided based on medically established conditions in the case of common reading types e.g. heart rate, weight, blood sugar etc. The user may modify the parameter values for these alerts if they so desire.

[0124] Trends

[0125] The system computes trends in the user's data values over an interval. The mechanism is as follows:

[0126] 1. Trends are determined by applying mathematical and statistical rules (e.g. moving average and deviation) over a set of reading values. Each rule is configurable by parameters that are either automatically calculated or are set by the user.

[0127] 2. Currently the system tracks deviations from baseline values (see section Baselines) either automatically calculated using intelligent defaults or provided by the user.

[0128] 3. The user also supplies a range over which the trend is calculated. This is specified either as the number of data points or a numerical unit of time (e.g. hours, days, weeks etc, depending on reading type).

[0129] 4. This range is always applied backwards from the most current reading i.e. if the user specifies that the trend needs to be calculated over the past 5 days, then the system will calculate the trend of the past 5 days from current date.

[0130] 5. The system displays the deviation between the trend values computed from received data and the baseline value. This deviation is indicated both as a percentage as well as a numerical difference, along with icons ("+", "-", "=") to visually indicate the trend direction.

[0131] Baselines

[0132] 6. The system may be used to calculate a baseline value for each reading type using algorithms specific to each reading type. The user can override the default baseline average by changing the default parameters used to calculate the baseline e.g. the start date and the number of readings. Additionally, the user may directly provide baseline value(s) into the system, overriding the computed baseline values. If the user provides parameters such that there is insufficient data for calculating the baseline, then the system alerts the user to the fact. Previously set baseline values are not altered.

[0133] During early use of the system, there may be insufficient data for automatically calculating baselines for the user's reading types. In this case too, the system alerts the user to the fact.

[0134] Strength of Trend Correlation

[0135] The system may contain algorithms to compute the strength of the trend and correlate it to the visual perception of the trend as would be observed by a human observer.

[0136] Example Algorithm: In order for a weight trend to be visually perceived as increasing or decreasing, a certain positive or negative standard deviation from a moving average baseline must be observed over a pre-defined time period to be able to inform the user that their weight is either increasing or decreasing.

```

old-baseline = calculate moving average for initial data range
old-standard-deviation = 0
for each subsequent interval
    new-baseline = calculate moving average for updated data range
    new-standard-deviation = standard-deviation(new-baseline, old-baseline)
    if (new-standard-deviation > 0)
        then trend is increasing
        if standard-deviation(new-standard-deviation, old-standard-deviation) > 0
            then strength of trend is increasing
        else strength of trend is decreasing
    else if (new-standard-deviation < 0)
        then trend is decreasing
        if standard-deviation(new-standard-deviation, old-standard-deviation) > 0
            then strength of trend is increasing
        else strength of trend is decreasing
    old-baseline = new-baseline
    old-standard-deviation = new-standard-deviation

```

[0137] Reminders

[0138] The user may set up reminders on the device(s) via the portal. The device then draws the user's attention at the appointed time e.g. by sounding a buzzer. The mechanism is as follows:

[0139] 1. Each device associated with the user's account has its own independent set of reminders.

[0140] 2. Each reminder can be made repeatable. The user can set the reminder to repeat every n <units of time> (where <units of time> may be hours, days etc.) starting from a preset date and time.

[0141] 3. The reminders are associated with a user's account and are synchronized with the device(s) via the hub. The reminder information is maintained in the control file (see section Device Control File).

[0142] 4. The user can specify their time zone so as to account for differences between time on the server vs. the user's local time.

[0143] Buddy or Role Based Access

[0144] The user may give permission to others as needed to read or edit their personal data. This is useful, for example, for allowing a well-wisher to view ones data/charts, or receive alerts. The clinician could be allowed to edit data for example to annotate it, while the patient would have read-only privileges and that too for certain pages. An authorized person could set the reminders and alerts parameters with limited access to others.

[0145] The user or clinician could have a list of people that they want to monitor and have it show on their "My Account" page, which serves as the central monitoring station. The central monitoring concept could be particularly useful in an environment where on person wishes or needs to monitor multiple people.

[0146] Clinician Administrator

[0147] A clinician administrator (also referred to as Administrator in this context) may monitor the data for and otherwise administer a number of users of the system (also referred to as Patients in this context). The mechanism is as follows:

[0148] 1. A summary "dashboard" of readings from all Patients assigned to the Administrator is displayed upon log in to the Portal by the Administrator. At a minimum, the Patient's last reading received from all devices is shown (color coded to visually distinguish normal vs. readings that have generated an alert), along with description of the alert generated.

[0149] 2. The Administrator may drill down into the details for each Patient to further examine the readings data, view charts etc. in a manner similar to the Patient's own use of the system.

[0150] 3. The Administrator may also view a summary of all the devices registered to all assigned Patients, including but not limited to all device identification information.

[0151] 4. An Administrator has access only to information about Patients that have been assigned to the Administrator by a Super Administrator. This allows for segmenting the entire population of monitored Patients amongst multiple Administrators.

[0152] The Super Administrator may assign, remove and/or reassign Patients amongst a number of Administrators.

[0153] Patient Questionnaire

[0154] The Portal provides a mechanism to enhance the healthcare interaction and education of the users of the system (also referred to as Patients in this context). The mechanism is as follows:

[0155] 1. A questionnaire may be made available to the Patient containing questions pertinent to their health condition.

[0156] 2. The Patient's answers to the questions are stored and tracked by the time of each "session" (user interaction). Statistical summaries of these answers for various intervals of time are available for display.

[0157] 3. The Administrator (or delegate) defines the entire set of questions that may be made available to all Patients. Of these, pertinent questions are selected based on rules applied to the Patient's recent data in the Repository.

[0158] Data Export and Reporting

[0159] Data may be extracted, exported and reported from the Repository into a variety of formats, both application specific and industry standard. The mechanism is as follows:

[0160] 1. The data in the Repository is internally extracted into a neutral format represented in XML.

[0161] 2. This data is further processed to either aggregate information or to remove user identification information, as may be the specific requirement.

[0162] 3. The internal, neutral format may be transformed into custom, application specific formats or industry standard formats depending on specific application requirements.

Software Architecture Specification**[0163] Requirements**

[0164] The architecture for the web application meets the requirements below. In particular, the architecture must be relatively independent of the user's client environment. At a minimum, the most common browsers (Netscape 4.0 and higher, IE 4.0 and higher) are supported on Windows (95, 98, NT, ME, 2000 now, XP in the near future). In addition, the architecture must be continuously adaptable to support newly emerging and evolving business requirements. This includes changes in the presentation, the business logic as well as the data tier.

[0165] In addition, the architecture should meet the following characteristics:

[0166] Scalable to meet performance levels as the user base grows, with the cost of scaling contained for at least 300,000 users.

[0167] Support rapid, iterative development. It is based on the J2EE Blueprints reference architecture with its timeline adapted to our schedule.

[0168] Not require expensive development tools. Most of the development tools are available free as either open source or extended evaluation versions.

[0169] Being based on industry-standard interfaces, it must allow for significant functionality to be purchased off-the-shelf (with integration and possible customization).

[0170] By adapting a standard reference architecture, it allows for the time required for developers to gain familiarity with the architecture to be compressed provided they are well versed in J2EE technology and architecture. This allows for the development effort to be outsourced at a later point in the development cycle. Moreover, the standard architecture, supports rapid and frequent changes to the look of the application. Such an architecture also supports the division of labor amongst the development team e.g. the application is partitioned such that the front-end designer requires minimal programming knowledge and vice versa.

[0171] The architecture should also partition the application to allow for the business logic to be accessed via a variety of interfaces e.g. the web (HTML) for now, wireless (WML) and/or voice response (VRU) in the future. It should also accommodate internationalization and localization of the application in the near future without significant changes. In addition, the architecture should leverage emerging technologies to provide a superior user experience e.g. it does not require the user to download, install and configure complicated software by utilizing emerging server-side technology. Moreover, the architecture should have a growth path to allow use of new technology as it becomes available over the application's lifespan (5 years). It should be designed to be resilient for at least the next 1-2 years while accommodating change. Finally, the architecture should not be tied to a particular set of products i.e. any J2EE compatible server can serve as application host. It also is relatively independent of database. This allows for flexibility in choosing and switching between hosting providers.

Technology

[0172] Platform

[0173] The Java 2 Enterprise Edition (J2EE) may be used as the standard technology platform for this application. Of the available J2EE components, the initial development phase may use Java Server Pages (JSPs), servlets, Java Database Connectivity (JDBC), Java API for XML Processing (JAXP). Other J2EE components, such as Enterprise Java Beans (EJBs), transactions APIs etc. may be used as required for subsequent development phases. Using J2EE components has the following advantages:

[0174] Simplified architecture and development

[0175] Scalability to meet demand variations

[0176] Integration with a variety of databases

[0177] Choices of servers, tools, components

[0178] Flexible security model

Relevant J2EE Features and Components

[0179] Scalability

[0180] J2EE provides mechanisms that support simplified scaling of distributed applications, without requiring significant effort on the part of the application development team. Because application services such as transaction support, database connections, life cycle management etc. are provided by the platform, the application can leverage these capabilities by adhering to the rules requiring it to be compatible with these services. For example, by using the appropriate JDBC APIs, connection pooling is automatically available for increased performance. Similarly, J2EE servers now provide many scaling mechanisms that were previously designed by the application developer.

[0181] Declarative Security

[0182] Security (access restriction, authentication) is configured declaratively at deployment time instead of being coded at development time. This allows for continuous refinement of the security mechanism (matching users with roles, with groups of users having specific access permissions) without requiring changes to the code followed by redeployment. For comparison, commercial products providing these kinds of capabilities for web applications (e.g. getAccess from Entrust Technologies) are typically priced at \$15-\$25 per user for quantities less than 5000 users.

[0183] Expandability

[0184] As the functionality provided by the J2EE expands, these capabilities are immediately available for use by the application. For example, the Connector architecture in the upcoming release of the J2EE will provide built-in capabilities of communicating with legacy systems (once commercial implementations are available).

[0185] Supported by Hosting Services

[0186] There are several commercial and open source implementations of J2EE compatible servers with the result that most hosting services offer reasonably priced hosting packages for J2EE applications.

Other Enabling Technologies and Products

Page Layout and Rendering

[0187] Flash

[0188] The pages should be as lightweight as possible while being aesthetically pleasing. This leads to using Flash as the rendering technology for items such as buttons and image maps etc. since this leads to compact graphics. However, there are no large animations.

[0189] CSS

[0190] The pages should not be hard-coded for a particular user screen size i.e. pages may be designed using Cascading Style Sheets (CSS).

[0191] Charting

[0192] Popchart (<http://www.popchart.com>) from Corda may be used to provide charting functions. It runs as an image server and services requests for charts from the web server over a pre-configured TCP port. It supports multiple output formats (including Flash, WML and SVG) and good

chart interaction such as pop-up text and drill-down. The charts can be visually designed by a graphic designer and later rendered programmatically.

[0193] Reporting (Printing)

[0194] The reporting technology/product should support the following requirements:

[0195] It must be able to produce charts similar in nature to those produced on-screen for the user.

[0196] It must be able to export the user's data in Excel format for download.

[0197] It must be able to provide the user with the facility to print a report on his/her local printer.

[0198] It must be able to provide the user report in PDF format for use as an e-mail attachment.

[0199] The final choice of product is still pending. Among the products evaluated include Formula One from Tidestone Technologies and Style Report from Inetsoft Corp.

Development Tools

User Interface

[0200] Visual Design

[0201] Dreamweaver software may be used as a tool for developing the front-end HTML. Flash and (optionally) Fireworks software can be used along with Dreamweaver. MS Frontpage software, although popular, is not preferred since it introduces irrelevant HTML into each document.

[0202] Chart Design

[0203] Popchart software (see section Charting above) includes a graphics chart builder that allows charts to be interactively designed. The designer may use this builder to create aesthetically pleasing charts.

[0204] HTML Validation

[0205] All HTML should pass validation for acceptance. Tidy is one utility that can check HTML for validity.

Web Server

[0206] An Apache web server may be used for development. The deployment web server is provided by the hosting service.

[0207] Servlet Engine

[0208] Tomcat software may be used for development due to it providing integrated debugging with many IDEs (Integrated Development Environments) and its smaller memory footprint when compared with other similar commercial products.

Framework

[0209] J2EE Blueprints (JPS)

[0210] The current application architecture is modeled after the J2EE Blueprints reference application (the Java Pet Store version 1.1.1). Although EJBs are not used at this stage, the tier separation is maintained to allow them to be introduced at a later stage.

[0211] com.oreilly.servlet

[0212] This is a collection of utility classes available at www.servlets.com that may be useful in servlet development. Developed and maintained by Jason Hunter, author of the book *Java Servlet Programming*, 2nd Ed. (must read).

Java Code

[0213] Integrated Development Environment

[0214] Forte for Java, Community Edition may be used because it provides a rich feature set for free. JRun Studio has also been evaluated and may be used if we switch to JRun in a later development phase.

[0215] JSP Tag Libraries

[0216] Tag libraries from the Apache-Jakarta (and the Struts framework) project may be used if needed. Proprietary taglibs will not be used. Taglibs should be used if their use results in significant effort saving.

[0217] Code Documentation

[0218] Javadoc.

[0219] Build Tool

[0220] Any industry standard build tool (such as Ant or make) can be used as the build tool.

[0221] Unit Testing

[0222] The JUnit unit-testing framework (www.junit.org) may be used to unit test all developed code. Extensions to JUnit for server-side testing and J2EE testing (e.g. HttpUnit, Cactus) may also be used.

Database

[0223] Either Oracle or MS SQL Server 2000 may be used as the database. It is important to be careful to not have dependencies on the database to allow for switching if required by hosting providers.

Process Control

[0224] Configuration Management

[0225] Visual SourceSafe or CVS may be used. CVS is accessible over the Internet.

[0226] Bug Tracking

[0227] EbugStomp software may be used for web-based bug tracking.

Application Architecture

Request and Response Flow

[0228] This section describes the architecture of the application: exploring the partitioning of functionality into modules, the assignment of functionality to tiers, and object decomposition within the tiers.

Application Structure

[0229] Modules

[0230] This is a list of the functional modules of the system. Some of the modules are relatively simple but have been broken out to provide an idea of the broad functional areas of the application.

[0231] Control Module

[0232] This module provides the basic framework for the other functional modules. It provides key features such as navigation, security etc.

[0233] User Account Module

[0234] This module maintains information about a person's account (contact information, payment and credit card information) as well as the devices associated with a particular account.

[0235] User Profile Module

[0236] This module maintains information about a person's profile such as demographic information, medical conditions, medications and allergies.

[0237] Device Information Module

[0238] This module maintains information about the various types of devices supported by the system.

[0239] Data Synchronization Module

[0240] This module provides the functionality to upload the newly acquired readings data from a user's PC to the server. It flags potentially erroneous readings and provides a preview of the data so that the user may optionally reject some readings. Please refer to the section Data Synchronization between User's PC and Server for more details.

[0241] Data Management Module

[0242] This module manages the data obtained for each user. It also provides functionality to associate annotations with data points.

[0243] Data View: Charts Module

[0244] This module provides the functionality for the user to view his/her readings data as charts. The user may choose to view either preset ranges of data or choose an arbitrary range.

[0245] Data View: Tables Module

[0246] This module provides the user with a tabular view of their data. There are no complex table-editing functions.

[0247] Trends Module

[0248] This module contains the logic to provide the user indication of certain trends in their data.

[0249] Alerts Module

[0250] This module provides the functionality where the user can select from preset patterns to look for in the data to be alerted about. The user may choose certain parameters of the triggering events e.g. if their blood pressure is above X for Y days, then trigger an alert, where X and Y are chosen by the user.

[0251] Reminders Module

[0252] The reminders module generates preset reminders to a user about certain events that were supposed to have occurred but have not.

[0253] Groups Module

[0254] This module will provide functionality in which the user can choose to invite another registered user to be a part of his/her group. The invited user can then have access to the inviting user's data.

[0255] E-Mail Module

[0256] This module provides the functionality to send e-mail arising from a number of places in the application: when an alert is generated, for sending a user report etc.

[0257] Report Module

[0258] The report module will provide functionality to generate simple reports for the user incorporating charts.

[0259] Print Module

[0260] This module will provide the functionality to print on the client PC.

[0261] Export Module

[0262] This module provides the capability to export the user's data in Excel format.

[0263] Interfaces

[0264] The web application will have the following interfaces with external systems. These interfaces may be either synchronous (i.e. online, in real time) or asynchronous (i.e. as a batch operation, deferred in time).

[0265] Interface with Credit Card System

[0266] The application will transmit the user's credit card number and obtain online verification. It will also make periodic charges to the credit card as the user's account is automatically renewed on expiry.

[0267] Interface with E-Store

[0268] An outsourced e-store will provide the capability for a user to buy a device online. The application should be able to transfer information such as credit card information etc. to this system. Additionally, the front-end for such a system needs to be created and integrated with the rest of the application.

[0269] Interface with Refund Generation System

[0270] This system will generate pro-rated refund checks for users canceling their accounts. The application must transfer the required information to this system.

[0271] Interface with Customer Support System

[0272] A customer support system providing (at least) e-mail based support needs to be integrated with the application/site.

[0273] Interface with Device Control Software on the User's PC

[0274] At this time, the interface with the device control software running on the user's PC is asynchronous and is handled by uploading and downloading files to a known location on the user's disk. This will eventually be enhanced to obtain the file location from the Windows registry.

Key Mechanisms

Business Object Framework

[0275] The Business Object Framework (BOF) is designed as a lightweight stand-in for Enterprise Java Beans (EJBs). The idea is to create structural parallels with the EJB component structure (such as the home and remote interfaces, session and entity beans etc.) in a simple object framework that allows for easy migration to EJBs at a later time. The key differences between EJBs and BOs are:

[0276] There is no container that the BOF objects (and their derived objects/components) run in. Because of that, BOF-derived objects must manage their own lifecycle such as creation and destruction. This is however minimal and is abstracted behind utility (factory) methods.

[0277] BOF-derived objects are generically referred to as BOs (for business objects) and are packaged in the bo package within the application structure. The scope of BOs is the same as for regular Java objects. This is different from EJBs, whose scope is controlled by the container and managed by methods on the home interface. The BO implementation objects must directly implement the BOF equivalent of the remote and home interfaces i.e. the BOInterface and BOHome. This is a key distinction from EJBs, in which the implementation object does not directly implement its home and remote interfaces.

[0278] The methods in the BOHome that are implemented by the BO have the same names as defined in the interfaces. This is different from EJBs which have a naming convention of prefixing ejb to the method name e.g. ejbCreate for a method named create in the home interface. BOs extend either the SessionBO or EntityBO base class. This is different from EJBs, which instead implement either the SessionBean or EntityBean interfaces. BOs are not remotely referenced and execute in the local JVM instead.

[0279] Components

[0280] The BOF has the following main types of components as set forth below:

[0281] Business Object Interface

[0282] This is the equivalent to the EJB remote interface. All business process methods are defined in this interface. Interfaces of this type extend com.carematix.bof.BOInterface and are named per the EJB convention i.e. using the domain class name. The BO must implement this interface directly.

[0283] Business Object Home (Interface)

[0284] This interface is the equivalent of the EJB home interface. At the very least, it must define a create(. . .) method to return a concrete BO object implementing the BOInterface. Interfaces of this type extend com.carematix.bof.BOHome and are named per the EJB convention i.e. by appending "Home" to the domain class name. The BO must implement this interface directly.

[0285] Business Object

[0286] This is the abstract base class (com.carematix.bof.bo) for Session Business Objects (SBOs) and Entity Business Objects (EBOs) described below.

[0287] Session Business Object (SBO)

[0288] An SBO is the equivalent of a session EJB. One key distinction is that SBOs can only be stateless, not stateful beyond the scope of the calling method. It is best to assume them as stateless. Typically, SBOs define most of their methods in the BOInterface and have just a simple create() method in the BOHome interface. A session BO extends the base class com.carematix.bof.SessionBO.

[0289] Entity Business Object (EBO)

[0290] An EBO is the equivalent of an entity EJB and encapsulates some persistent data representation. Typically, EBOs have a simple BOInterface and have most of their methods in the BOHome interface such as create(. . .), load(. . .), store(. . .), remove(. . .), findByPrimaryKey(. . .) etc. Not all of these methods are required in every EBO and only those needed by the application must be defined. An entity BO extends the base class com.carematix.bof.EntityBO.

EXAMPLES

[0291] Source code for the placeholder classes comprising the BOF is available. Accompanying that are two example packages: boexample1 and boexample2. These provide example implementations for a session BO and entity BO respectively. They illustrate the patterns and conventions required to create BOs and should be thoroughly studied.

[0292] Usage

[0293] In order to access a BO from calling code, the following example of a handler method for handling account events is illustrative:

[0294] public void perform(PWMAppEvent event)
throws PWMAppEventException

```

{
    AccountEvent ae = (AccountEvent)event;
    ...
    switch (ae.getActionType( ))
    {
        case AccountEvent.CREATE_ACCOUNT:
        {
            try
            {
                UserHome userHome = BOUtil.getUserHome( );
                User user = userHome.create( );
                user.createAccount(ae.getUserId( ), ...);
                ...
            }
            catch (DuplicateKeyException dke)
            {
                ...
            }
            // Catch more exceptions
        }
    }
}

```

[0295] Deployment

[0296] Entries need to be created in the web.xml file for each BO in order to enable JNDI lookup. Entries for the BO examples are as follows:

[0297] <?xml version="1.0" encoding="ISO-8859-1"?>

[0298] <!DOCTYPE web-app PUBLIC "-//Sun Microsystems, Inc./DTD Web Application 2.2/EN" "http://java.sun.com/j2ee/dtds/web-app_2.2.dtd">

[0299] <web-app>

[0300] . . .

[0301] <env-entry>

[0302] <env-entry-name>bo/boexample1/Example1</env-entry-name>

[0303] <env-entry-type>java.lang.String</env-entry-type><env-entry-value>com.carematix.boexample1.bo.Example1SBO</env-entry-value>

[0304] </env-entry>

[0305] <env-entry>

[0306] <env-entry-name>bo/boexample2/Example2</env-entry-name>

[0307] <env-entry-type>java.lang.String</env-entry-type>

[0308] <env-entry-value>com.carematix.boexample2.bo.Example2EBO</env-entry-value>

[0309] </env-entry>

[0310] <env-entry>

[0311] <env-entry-name>bo/boexample2/Example2DAOClass</env-entry-name>

[0312] <env-entry-type>java.lang.String</env-entry-type><env-entry-value>

[0313] com.carematix.boexample2.dao.Example2DAOImpl</env-entry-value>

[0314] </env-entry>

[0315] . . .

[0316] </web-app>

[0317] Template Mechanism

[0318] The XML-based templating solution provides many benefits. XML is a useful, consistent way of representing application parameters. The XML format is easy to use for both presentation specialists and third-party tools. The fact that XML is standardized means we can use off-the-shelf software to handle all the details of parsing and writing our screen definitions, configuration files, and other forms of structured data. Finally, XML can be validated against a DTD (Document Type Definition), so the parser can handle errors relating to ill-formed input, instead of in the application code.

[0319] Three files control the templating system:

[0320] The template file (template.jsp), which defines the basic layout of all of the pages generated from the template. There is one template file per supported language.

[0321] The request mappings file (requestmappings.xml), which maps virtual URLs to screen names (or request handler classes). There is one request mappings file in the application.

[0322] The screen definitions file (screendefinitions.xml), which defines the parameters for each screen. There is one screen definitions file per supported language.

[0323] The main controller servlet MainServlet receives incoming HTTP requests. MainServlet matches the incoming url to the URLs defined in the request mappings file, obtaining a screen name. The template parameters corresponding to the screen name are retrieved from the screen definitions file, and the template is then served using those parameters. The MainServlet does not do all of the work. Most of the work is delegated to other classes that specialize in screen flow management, template instantiation, and the like.

[0324] The XML files in the templating system are parsed by a standard XML parser, created using the JAXP (Java API for XML Parsers) optional package. The template is served as a JSP page, and the inclusion of the dynamic content occurs because the JSP engine calls the custom tag <j2ee:insert> each time it occurs.

[0325] In a Web application, each screen presented to the user can be considered as a different view. However, unlike the classic MVC architecture, all these views share the same controller. There needs to be a mechanism that allows the controller to choose a particular view to render in response to a user request. In the sample application, the controller makes this selection by specifying the screen ID of the screen to present as the response. This screen ID is mapped to a screen definition, then the template is instantiated. Recall that the file template.jsp defines the template for the sample application. This file includes another file, ScreenDefinitions.jsp, which defines all the screens of the sample application. When the controller invokes the template file at request time, it sets the appropriate screen definition in the request scope. The template file passes this information to the screen definitions file which then returns the appropriate screen definition for the request. One goal in structuring template and screen definition files is to facilitate internationalization (discussed in Section 4.5). This is achieved by separating text content from Java code. Since screen definitions that contain direct and indirect parameters are candidates for internationalization, we want to keep ScreenDefinitions.jsp devoid of Java technology code. We achieve this through the use of JSP custom tags.

[0326] Data Access Mechanism

[0327] The application uses a bimodal data access mechanism to increase efficiency of data access. Since it is anticipated that the majority of data access will be read-only (50% or more, as a conservative estimate), the data access mechanism will not utilize a transactional component for reading, relying instead on direct JDBC access to reduce overhead. Conversely, data updates will still continue to rely on a transaction component. When a transactional update does occur, the model will be refreshed from the data store and a model update event propagated to all registered views to update themselves, ensuring data display consistency.

[0328] Data Synchronization Between User's PC and Server

[0329] The functional specifications for the data synchronization mechanism are listed below, followed by details on each of the components that implement the mechanism. In cases where the implementation details of a component is

not yet decided (such as whether to use an applet or not on the client PC), it is referred to generically (e.g. data sync client).

[0330] Functional Specifications

[0331] When a user logs on to the home page (FIG. 17) using the PC the data file is uploaded to server. The server filters duplicate readings records. The records are decrypted on the server, assigned to each user (by determining which user the device is registered with) and put into a staging area. This staging area is so that the user may optionally preview the data before accepting the values. The device control file is uploaded to the server. The server downloads updated device control file with last updated timestamps for each device in list. If device has been removed from the system, then its entry in the file is deleted. The old device list file is backed up. The data file is cleared/deleted. If a user has logged onto the site for the first time from this PC, then a new control file is downloaded. This control file will contain entries for the registered devices of the current user only and will be built up as other users log into the PC. Similarly, when a user registers a device, its entry is added to the control file on the PC if not already there.

[0332] For this phase of implementation, the locations of the control file and the data file on the user's PC is fixed. Note the following:

[0333] No association information between the user and the device is kept on the user's PC. This determination is made after the data is uploaded to the server by matching the device id against the RegisteredDevice table.

[0334] Data for all users (i.e. from all devices in the control file) is uploaded to the server, regardless of the user logged on.

[0335] Device Control File

[0336] This file contains records in the format

[0337] <device id>, <data update timestamp>

[0338] where <device id> is all the information required to uniquely identify a device to the system (such as manufacturer number, model number, serial number and user number) and the <data update timestamp> is the timestamp (including date) that data for this device was last received by the server. Note that this is not the timestamp for which data was last uploaded from this particular PC (on which the control file resides) but is the timestamp for which the server last received data (from any user PC) for the particular (registered) device.

[0339] User Data File

[0340] This file contains records in the format

[0341] <device id>, <reading timestamp>, <reading data>

[0342] where <device id> is as described in the section Device Control File. The <reading timestamp> is the time of the reading and the <reading data> is a data record in a format specific to the type of reading. Each record in the file is individually encrypted (encryption to be implemented in a sub-

sequent phase) and is appended to the end of the file. Note that data for all users is aggregated in the same file.

[0343] Data Synchronization Client

[0344] This client runs on the user's PC to handle the file operations to and from the server such as uploading and downloading, checking for file existence, creating backup files etc. Note that these are all file operations and do not require any business logic in this client. All of the business logic decisions are made at the server and the appropriate files are downloaded. This allows for the client to be implemented as a signed applet.

[0345] A standalone Data Synchronization Client that does not require a PC and is directly able to connect to the server via a network connection is developed by embedding the synchronization logic in the firmware of a microcontroller and using it to connect to the server through a connection established either via a phone line (which requires an embedded modem) or via a connection established by another network terminal such as a web-phone or video-phone to which the Data Synchronization Client is connected.

[0346] Data Synchronization Manager

[0347] The Data Synchronization Manager component runs on the server and is the primary interface between the Data Synchronization Client and the rest of the application. Its primary operations include: receiving upload of the data file from Data Synchronization Client; decrypting the data records; determining which user each record belongs to (from RegisteredDevice mapping); putting each user's respective records in staging area (table) for user preview if so specified. This function is actually delegated to the ReadingsManager; putting each user's respective records in data tables if user does not want preview. This function is actually delegated to the ReadingsManager; signaling Data Synchronization Client to delete data file; receiving upload of control file; if control file does not exist, then creating it initially; updating control file with last uploaded timestamp for each registered device; downloading updated control file to Data Synchronization Client.

[0348] The Data Synchronization Manager has the following invocation points:

[0349] On user login.

[0350] On user registering one or more device(s).

[0351] On user previewing uploaded data prior to accepting it.

[0352] On an embedded Data Synchronization Client connecting to the server.

[0353] The Data Synchronization Manager is implemented as a Session BO as well as a standalone server application known as the iModem server.

[0354] IModem Server and Protocol Specification

[0355] The key attributes and mechanisms of the iModem server (i.e. Data Synchronization Manager for Standalone Data Synchronization Clients) are:

[0356] Accept multiple connections over the same server port at the same time: This will happen (in production) since more than one iModem could be connecting to the server and will require that the server be multi-threaded, using an independent thread for handling each connection. This is

somewhat analogous to how web servers handle HTTP, which is a stateless protocol wherein several clients connect over the same port (80) at the same time. This also implies that the port number to which the iModem connects will remain constant.

[0357] Build in a versioning scheme into the protocol: This allows for upgrades over time without breaking backward compatibility.

[0358] Provide a status value in each message response. Zero indicates OK, non-zero values indicate various error conditions.

[0359] Error handling: There must be enough error handling and recoverability that a user's data is never lost and that each end (iModem or server) can signal to the other that data transfer either occurred or not. There must also be timeouts to handle conditions when no response is achieved. The value of the timeout (in seconds) must be a parameter read from a configuration XML file.

[0360] Break down current messages into multi-step messages: This is to handle two basic limitations: 1) each message to and from the iModem must be less than 256 bytes and 2) the iModem embedded code is more efficient in parsing fixed-length responses rather than variable length responses.

[0361] Encryption and keys: A basic mechanism (using placeholders as necessary) needs to be put in place such that in case the data stream is intercepted by a third party, all information that is private to the patient (such as reading values) and all information that can be correlated to deduce the message format (such as device identification information) is scrambled or encrypted.

[0362] This servlet is a "listener" for an embedded internet-modem (iModem) that has established a dial-up connection with a PPP server and is then communicating over TCP/IP to a specific server and port. The primary purpose of this servlet is to accept TCP connections over the specified port, parse the data coming over the connection and return responses. It also does basic connection management of connections and sockets as necessary. The incoming data stream comprises a series of messages in a simple protocol detailed below. This servlet parses the data stream, decodes the protocol message and responds based on logic detailed below. In summary, this servlet serves as a protocol handler for the iModem hub data transmission, which is done over TCP/IP sockets.

[0363] iModem Protocol.

[0364] The data protocol is a variant of the data exchange (XML) protocol that is used by the main web application for data interchange (defined by the files devicecontrol.xml and readingsdata.xml, hereafter referred to as the Datasync protocol). It differs from the Datasync protocol in the following manner to allow for the limited processing requirements of the iModem hub:

[0365] Numeric values are specified as hexadecimal text with each byte value comprised of 2 ASCII characters i.e. a decimal value 1085, which is 43D hex (or 043D hex, more accurately), will be specified as the characters '0', '4', '3' and 'D'. The most significant byte (MSB) is first when decoding multi-byte values.

[0366] Descriptive tags in the Datasync protocol are replaced with 4-digit hexadecimal numbers referred to as tokens. E.g. a <control-entry> tag may be denoted as <04AD>¹. The mapping of the tags in the Datasync protocol to the iModem protocol must be configurable via another XML file (tagmappings.xml) so that it may be changed as the protocols evolve. End tags are denoted by a 4-digit hex number that is calculated by setting the most-significant-bit (MSB) of the number denoting the start tag. This defines the start tag and end tag relationship of the iModem protocol.

¹ The angle braces <> will still be used to surround tags

[0367] Readings are sent serially as a series of Readings Data messages (see below). This is given the current limitation of the iModem restricting the message length to be less than 256 bytes. The server response is either a success or error indicator.

[0368] All timestamps are in seconds elapsed since Jan. 1, 1970, denoted as a 4-byte number and encoded as the ASCII representation of the hex digits, MSB first. There is an additional command for getting the current server time that is not part of the Datasync protocol. The details of this message are defined below, with a descriptive tag being used in place of a token for illustration purposes.

[0369] Initiate Session

[0370] Command

[0371] <initiate-session><hub-id>H3H2H1H0</hub-id><protocol-version>P1P0</protocol-version></initiate-session>

[0372] Response

[0373] <initiate-session><status>S1S0</status><current-time>T3T2T1T0</current-time><update-flag>F0</updateflag><key>K1K0</key></initiate-session>

[0374] Details

[0375] The HubID is an identifier that is encoded in the iModem hub and is transmitted as a means of identifying itself. It is a long i.e. 4 byte number. This ID will be populated in the database through the device registration UI (which will be enhanced to support this later) and is used to associate a user (and his registered devices) with a specific hub.

[0376] On receiving this message, the server should set up a "session" with this connection and Hub ID and return a status code of 0000 in the response. All subsequent messages coming over this connection are considered in the context of the established session. If an unrecognized Hub ID is received, a status code of 0001 is returned with other message data being same. The <protocol-version> value is two bytes, with the MSB being the major version and the LSB being the minor version. So currently, this value would be 0101 (current version is 1.1=>1<<8+1=257 decimal=0101 hex, formatted as 2 characters for each byte).

[0377] The response returned is the current server time, in seconds elapsed since Jan. 1, 1970. It is a 4-byte number and is encoded as the ASCII representation of the hex digits, MSB first.

[0378] There are two additional fields added for version 1.1. These are the update-flag and key. <update-flag> is a byte value that is used to signal to the iModem that there are

parameter updates (see message Parameter Updates) that it needs to query for. The intent is to provide a way for parameters such as dial out numbers, dialing prefixes etc. to be provisioned via a web interface and indicated to the iModem so that it can update its own memory-based parameters. For version 1.1, the value returned is always 00 (zero).

[0379] The <key> is a placeholder value for now. It is a 16 bit number and its value is always 0000 (zero).

```

Parameter Updates
Command
<param-updates></param-updates>
Response
<param-updates><status>S1S0</status></param-updates>

```

[0380] Details

[0381] This is a placeholder message for now and will be expanded in future protocol revisions. The status code is always 0000.

[0382] Readings Data

[0383] Command

```

<readings-data><reading-record><device-id><manufacturer-no>M2M1M0</manufacturer-no><device-model>D2D1D0</device-model><serial-no>N3N2N1N0</serial-no><user-no>U0</user-no></device-id><timestamp>T3T2T1T0</timestamp><reading><reading-type>BP-HR</reading-type><units>mm Hg</units><values><bp-systolic>V0</bp-systolic><bp-diastolic>V0</bp-diastolic><pulse-rate>V0</pulse-rate></values></reading></reading-record></readings-data>
Response
<readings-data><status>S1S0</status></readings-data>
Details

```

[0384] This message is a fixed-format equivalent to the readingsdata.xml file containing a single reading. The appropriate token values (as set up in tagmappings.xml file) are used for <reading-type> and <units> tag values.

[0385] The <status> value is a 16-bit integer. The value 0000 (zero) indicates no error whereas non-zero values indicate error conditions. Currently, only the value 0001 is assigned as a general error indicator. This list of values will be expanded over time and the code should be written to allow for it without significant change.

[0386] Because the iModem requires a fixed-length, fixed-format data command, the server must do some conversion/transformation to derive the application-specific XML <readings-data> command. The iModem will always send all data (BP, weight or blood sugar) in the format described above (for BP). The only difference for weight and blood sugar will be that the <reading-type> and <units> will correspond to the type of reading, so that for weight, <reading-type> value is WEIGHT and <units> value is POUNDS and for blood sugar, <reading-type> is BLOOD-SUGAR and <units> value is mg/dL. The value of the reading is derived from the byte values contained in the <bp-systolic>, <bp-diastolic> and <pulse-rate> tags. Because of this, the <value> tag is no longer required.

[0387] The server must use the following algorithm/logic to convert these fixed format iModem <readings-data> commands to the Datasync protocol's <readings-data> messages:

```

if (reading-type == BP-HR)
then translate iModem <readings-data> message directly into
Datsync <readings-data> message
else if (reading-type == WEIGHT)
then
value = ((<bp-diastolic> << 8 + <bp-systolic>) >> 6) * 2.2
// this value is in POUNDS, << & >> are left-shift & right-
shift operations
else if (reading-type == BLOODSUGAR)
then
value = <bp-systolic>

```

[0388] Number of Devices

[0389] Command

[0390] <device-control-list></device-control-list>

[0391] Response

[0392] <device-control-list><status>S1S0</status><num-devices>N0</num-devices></device-control-list>

[0393] Details

[0394] This command is used to obtain the number of devices currently registered by the user. The status is generally expected to be 0000, non-zero indicates an unspecified (as of yet) error.

[0395] Device Information

[0396] Command

[0397] <device-info><device-num>N0</device-num></device-info>

[0398] Response

[0399] <device-info><status>S1S0</status><control-entry><device-id><manufacturer-no>M2M1M0</manufacturer-no><device-model>D2D1D0</device-model><serial-no>N3N2N1N0</serial-no><user-no>U0</user-no></device-id><reminders></reminders><first-association-timestamp>T3T2T1T0</first-association-timestamp><last-update-timestamp>T3T2T1T0</last-update-timestamp></control-entry></device-info>

[0400] Details

[0401] The iModem sequentially requests device details for each device registered by the user, passing in a device number (actually, device index) as part of the command. The list is zero-indexed i.e. the first device is 0 (zero) and the index increments sequentially from there. The information in the response is the same as a single-entry device control file. Currently, nothing is returned for reminders (even if there are some set) since the iModem is not set up to handle them. These will be handled in a subsequent protocol revision.

[0402] Terminate Session

[0403] Command

[0404] <terminate-session></terminate-session>

[0405] Response

[0406] This command does not return a response.

[0407] Details

[0408] This command is for the iModem to indicate to the server that it is done with the session prior to terminating the connection so that the server may release resources and perform other cleanup. The server does not return a response to this command since the client's state is undefined after this. It releases resources and terminates the server's socket connection.

[0409] Reminders

[0410] Reminders are in a slightly different format from the Datasync protocol. The structure is as follows:

[0411] `<reminders><reminder><start>T3T2T1T0</start><repeat><nn></repeat><for><nn></for></reminder> . . . </reminders>`

[0412] where . . . denotes more reminder records, each delineated by a `<reminder></reminder>` sequence. Each reminder block is part of the `<control-entry>` block. The `<repeat>` value is the number of seconds to repeat the reminder after the start. It is zero if there is no repetition. The `<for>` value is the number of times this repetition must be carried out. It too is zero if there is no repetition.

[0413] Chart Creation and Display

[0414] The chart creation and display mechanism is comprised of collaborating objects and components across all four application tiers (view, web, business object and data persistence tiers). As previously mentioned, Popchart is the charting server being used to render the data for display in the Flash format. This requires the Flash plug-in to be installed in the client browser, which the majority of the browsers (over 90%) already do.

[0415] The approach to chart creation and display is to isolate dependency on Popchart in the view tier only. This is to allow change over to another charting product if required later. Doing this requires encapsulating all code to create a Popchart chart in presentation components in the view tier. All the information (data) required to produce the chart is kept as generic objects or components and supplied to the view components via the standard application framework. This mechanism is elaborated upon below.

[0416] The View Tier

[0417] This tier is comprised of the JSP page containing the chart as well as an embedded JavaBean that converts the chart model objects into Popchart specific instructions (there may be the need to do specific JavaBeans for each chart type but the approach will remain the same for each). This JavaBean encapsulates the logic to create Popchart instructions from the chart model objects and isolates the dependency on Popchart.

[0418] The Popchart JSPEXample demonstrates such a JSP/JavaBean combination. The key points of difference are:

[0419] The example code has a lot of hard-coded elements such as label information etc. All this information is being supplied in the chart model object(s) instead.

[0420] The example code has a lot of direct database access for retrieving data. All data required to create the chart will be supplied through the chart model object(s) or in objects directly associated with them. Some of these data objects would be the domain model objects themselves e.g. the Range object would contain the range of reading values to be charted.

[0421] The Web Tier

[0422] This tier contains the standard web handler for processing user input to the chart (such as when a user displays the chart, changes the date range, clicks on a point for annotation or drill-down etc.). The handler creates a chart event object (class ChartEvent) encapsulating the user request and forwards it to the Business Object tier for handling.

[0423] Note that there may be requests that can be handled in the web tier itself instead of having to go to the BO tier. For example, the user may choose to view a data range that is a subset of the range already displayed. In this case, there is no need to go to the BO tier to get data for display since the data has already been sent to the web tier in the chart model object(s). The web tier handlers need to be intelligent enough to check for these cases in order to improve application performance.

[0424] The Business Object Tier

[0425] This tier contains the components required to handle chart events generated from the web tier. The main facade component is a session BO named ChartManagerSBO. This BO interprets the chart event and creates/retrieves the chart model objects that are required to generate the requested chart. It invokes the ReadingsManager to get the actual readings model objects. This BO also provides the defaults for chart presentation.

[0426] The chart model objects also belong in this tier. These objects encapsulate the complete information required to render the chart. This includes information about the various chart display parameters as well references to the model objects that hold the actual data for display. The data used to populate the chart model is obtained from the chartconfig.xml file using a DAO implementing the ChartDAO interface.

[0427] The chart model object is mutable i.e. it can be changed when a user adds or changes annotations on data points. The annotation actually changes a particular reading referenced by the range, which is in turn referenced by the chart model object.

[0428] Implementation Notes

[0429] Popchart servlet redirector may be used.

[0430] The implementation DAO object actually is ChartXmlDAOImpl, meaning that it reads the XML configuration file.

[0431] The elements in the chart model object that are specific to the charting product are kept as properties so as to not make the structure of the model object dependent on product-specific attributes.

Patterns

[0432] Following are the design patterns utilized in the architecture. Further details can be found in publicly available J2EE Blueprints documents as well as in a book, entitled "Design Patterns", by Vlissides, Gamma, et al and are thus elaborated upon further here.

[0433] Model-View-Controller (MVC)

[0434] Data Access Object (DAO)

[0435] Bimodal Data Access

[0436] Front Component

[0437] Value Object

File Formats

[0438] Device Control File

[0439] This file is currently in plain text XML format. It may later be stored encrypted if necessary. It does not currently have a DTD and therefore requires a non-validating parser. For now, the URI for this file is file://c:\carematix\config\devicecontrol.xml.

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<device-control-list>
  <control-entry>
    <device-id>
      <manufacturer-no>12345</manufacturer-no>
      <device-model>67890</device-model>
      <serial-no>55555555</serial-no>
      <user-no>1</user-no>
    </device-id>
    <last-update-timestamp>06-05-2001 11:21:33</last-update-
timestamp>
  </control-entry>
  <!-- More <control-entry> records for all other devices that users of
this PC have registered with Carematix.
-->
</device-control-list>
```

[0440] User Data File

[0441] This file is currently in plain text XML format. It may later be converted into a format in which each XML <reading-record> entry is encrypted and stored as a fixed length record, with new records being appended to the end. It does not currently have a DTD and therefore requires a non-validating parser. For now, the URI for this file is file://c:\carematix\data\readingsdata.xml.

[0442] Note: This file may contain duplicate <reading-record> entries. That is allowed.

```
<readings-data>
<!-- Reading record for blood pressure and heart rate reading -->
<reading-record>
  <device-id>
    <manufacturer-no>12345</manufacturer-no>
    <device-model>67890</device-model>
    <serial-no>55555555</serial-no>
    <user-no>1</user-no>
  </device-id>
  <timestamp 06-05-2001 11:21:33 PM/>
  <reading>
```

-continued

```
<!-- The reading-type tag may be eliminated later -->
<reading-type>BP-HR</reading-type>
<values>
  <bp-systolic 120 />
  <bp-diastolic 80 />
  <pulse-rate 72 />
</values>
</reading>
</reading-record>
<!-- Reading record for temperature reading -->
<reading-record>
  <device-id>
    <manufacturer-no>29933</manufacturer-no>
    <device-model>68909</device-model>
    <serial-no>123456789</serial-no>
  </device-id>
  <timestamp 06-05-2001 11:21:42 PM/>
  <reading>
    <!-- The reading-type tag may be eliminated later -->
    <reading-type>TEMP</reading-type>
    <!-- The units for temperature are FARENHEIGHT
or CELCIUS
<units FARENHEIGHT />
<value 98.7 />
  </reading>
</reading-record>
<!-- Reading record for weight reading -->
<reading-record>
  <device-id>
    <manufacturer-no>29890</manufacturer-no>
    <device-model>68309</device-model>
    <serial-no>123456789</serial-no>
  </device-id>
  <timestamp 06-05-2001 11:25:42 PM/>
  <reading>
    <!-- The reading-type tag may be eliminated later -->
    <reading-type>WEIGHT</reading-type>
    <!-- The units for weight are KILOGRAMS or POUNDS -->
    <units KILOGRAMS />
    <value 65 />
  </reading>
</reading-record>
... more reading records from all devices (and all users)
</readings-data>
```

XML to iModem Tag Mappings

```
<tag-mapping type="request" xml-tag="readings-data" imodem-
tag="0000"/>
<tag-mapping type="request" xml-tag="reading-
record" imodem-tag="0001"/>
<tag-mapping type="request" xml-tag="reading" imodem-
tag="0002"/>
<tag-mapping type="request" xml-tag="reading-type" imodem-
tag="0003" byte-length="1"/>
  <tag-token xml-token="BP-HR" imodem-token="04"/>
  <tag-token xml-token="TEMP" imodem-token="05"/>
  <tag-token xml-token="WEIGHT" imodem-token="06"/>
  <tag-token xml-token="BLOODSUGAR" imodem-token="07"/>
</tag-mapping>
<tag-mapping type="request" xml-tag="values"
imodem-tag="0004"/>
  <tag-mapping type="request" xml-tag="bp-systolic" imodem-
tag="0005" byte-length="1"/>
  <tag-mapping type="request" xml-tag="bp-diastolic" imodem-
tag="0006" byte-length="1"/>
  <tag-mapping type="request" xml-tag="pulse-rate" imodem-
tag="0007" byte-length="1"/>
  <tag-mapping type="request" xml-tag="units" imodem-
tag="0008" byte-length="1"/>
    <tag-token xml-token="mm Hg" imodem-token="00"/>
    <tag-token xml-token="Kelvin" imodem-token="01"/>
    <tag-token xml-token="POUNDS" imodem-token="02"/>
    <tag-token xml-token="mg/dL" imodem-token="03"/>
  </tag-mapping>
<tag-mapping type="response" xml-tag="current-
time" imodem-tag="0009" byte-length="4"/>
```

-continued

```

<tag-mapping type="common" xml-tag="device-id" imodem-
tag="000A"/>
<tag-mapping type="common" xml-tag="manufacturer-
no" imodem-tag="000B" byte-length="3"/>
<tag-mapping type="common" xml-tag="device-
model" imodem-tag="000C" byte-length="3"/>
<tag-mapping type="common" xml-tag="serial-no" imodem-
tag="000D" byte-length="4"/>
<tag-mapping type="common" xml-tag="user-no" imodem-
tag="000E" byte-length="1"/>
<tag-mapping type="common" xml-tag="device-control-
list" imodem-tag="000F"/>
<tag-mapping type="response" xml-tag="control-
entry" imodem-tag="0091"/>
<tag-mapping type="response" xml-tag="reminders" imodem-
tag="0092"/>
<tag-mapping type="response" xml-tag="reminder" imodem-
tag="0093"/>
<tag-mapping type="response" xml-tag="start" imodem-
tag="0094" byte-length="4"/>
<tag-mapping type="response" xml-tag="repeat" imodem-
tag="0095" byte-length="4"/>
<tag-mapping type="response" xml-tag="for" imodem-
tag="0096" byte-length="1"/>
<tag-mapping type="request" xml-tag="timestamp" imodem-
tag="0097" byte-length="4"/>
<tag-mapping type="response" xml-tag="first-association-timestamp"
imodem-tag="0098" byte-length="4"/>
<tag-mapping type="response" xml-tag="last-update-timestamp"
imodem-tag="0099" byte-length="4"/>
<tag-mapping type="response" xml-tag="status" imodem-
tag="009A" byte-length="2"/>
<tag-mapping type="common" xml-tag="device-
info" imodem-tag="009B">
<tag-mapping type="response" xml-tag="num-
devices" imodem-tag="009C" byte-length="1"/>
<tag-mapping type="request" xml-tag="device-
num" imodem-tag="009D" byte-length="1"/>
<tag-mapping type="common" xml-tag="initiate-
session" imodem-tag="0030"/>
<tag-mapping type="request" xml-tag="hub-id" imodem-
tag="0031" byte-length="4"/>
<tag-mapping type="request" xml-tag="protocol-
version" imodem-tag="0032" byte-length="2"/>
<tag-mapping type="response" xml-tag="update-
flag" imodem-tag="0033" byte-length="1"/>
<tag-mapping type="response" xml-tag="key" imodem-
tag="0034" byte-length="2"/>
<tag-mapping type="common" xml-tag="terminate-
session" imodem-tag="0035"/>
<tag-mapping type="common" xml-tag="param-
updates" imodem-tag="0036"/>

```

[0443] Chart Configuration File

[0444] This file contains the necessary information for formatting and displaying data charts. This information is kept in XML format neutral of the specific charting product so as to allow changeover to another charting application if necessary later. It does not currently have a DTD and therefore requires a non-validating parser. This file is stored on the server, not the client and is called chartconfig.xml

[0445] Please note the following: this format is an initial version and will likely be enhanced over time. The method for altering and enhancing chart display is to add tags and/or properties in the sections for each chart and use the values assigned to each to drive the behavior of the code. This is preferred over hard-coding the way a chart is displayed in the code itself.

[0446] In subsequent phases, this file may have an XSL transform applied to generate the product specific charting instructions e.g. if using Popchart, an XSLT would be provided to generate PCScript from this XML file.

```

<chart-config>
  <bp-chart>
    <type>TIMELINE</type>
    <title>Blood Pressure</title>
    <height>xxx</height>
    <width>yyy</width>
    <property name="appearance-file" value=
    "URL of appearance file" />
    <property name="date-format" value="%m/%d/%y %H:%M" />
  </bp-chart>
  <hr-chart>
    <type>TIMESCATTER</type>
    <title>Heart Rate</title>
    <height>xxx</height>
    <width>yyy</width>
    <property name="appearance-file" value=
    "URL of appearance file" />
    <property name="date-format" value="%m/%d/%y %H:%M" />
  </hr-chart>
  <temperature-chart>
    <type>TIMELINE</type>
    <title>Temperature</title>
    <height>xxx</height>
    <width>yyy</width>
    <property name="appearance-file" value=
    "URL of appearance file" />
    <property name="date-format" value="%m/%d/%y %H:%M" />
  </temperature-chart>
  <weight-chart>
    <type>TIMELINE</type>
    <title>Blood Pressure</title>
    <height>xxx</height>
    <width>yyy</width>
    <property name="appearance-file" value=
    "URL of appearance file" />
    <property name="date-format" value="%m/%d/%y %H:%M" />
  </weight-chart>
</chart-config>

```

[0447] Obviously, many modifications and variations of the present invention are possible in light of the above teachings. Thus, it is to be understood that, within the scope of the appended claims, the invention may be practiced otherwise than as specifically described above.

[0448] What is claimed and desired to be covered by a Letters Patent is as follows:

Volume in drive F is CAREMATIX SRC
Volume Serial Number is EB59-B6BE

Directory of F:\

04/14/2003	06:08a	<DIR>	Device
04/14/2003	06:08a	<DIR>	Server
		0 File(s)	0 bytes

Directory of F:\Device

04/14/2003	06:08a	<DIR>	.
04/14/2003	06:08a	<DIR>	..
04/13/2003	05:02a		239,652 HUBCC.LST
04/13/2003	05:02a		354,662 LOPWRBP.LST
04/13/2003	05:02a		404,552 MODEMHUB.LST
		3 File(s)	998,866 bytes

Directory of F:\Server

04/14/2003	06:08a	<DIR>	.
04/14/2003	06:08a	<DIR>	..
04/14/2003	06:08a	<DIR>	iModem
04/14/2003	06:08a	<DIR>	UI
04/14/2003	06:08a	<DIR>	Webapp
		0 File(s)	0 bytes

Directory of F:\Server\iModem

04/14/2003	06:08a	<DIR>	.
04/14/2003	06:08a	<DIR>	..
04/14/2003	06:03a		7,178 v.java
04/14/2003	06:03a		1,334 vw.java
04/14/2003	06:03a		745 vx.java
04/14/2003	06:03a		436 vxw.java
04/14/2003	06:03a		433 vy.java
04/14/2003	06:03a		4,151 wv.java
04/14/2003	06:03a		2,017 www.java
04/14/2003	06:03a		715 wwy.java
04/14/2003	06:03a		427 wx.java
04/14/2003	06:03a		1,449 wxw.java
04/14/2003	06:03a		1,258 wzv.java
04/14/2003	06:03a		1,339 xvw.java
04/14/2003	06:03a		1,469 xww.java
04/14/2003	06:03a		1,879 xx.java
04/14/2003	06:03a		953 yyw.java
04/14/2003	06:03a		431 zv.java
04/14/2003	06:03a		367 zw.java
04/14/2003	06:03a		1,958 zyw.java
04/14/2003	06:03a		298 zzw.java
04/14/2003	06:03a		17,751 zzy.java
		20 File(s)	46,588 bytes

Directory of F:\Server\UI

04/14/2003	06:08a	<DIR>	.
04/14/2003	06:08a	<DIR>	..

11/13/2002	06:39p	16,248	About_us.htm
11/20/2002	04:30p	12,440	about_us.html
01/18/2003	01:11p	709	accountcreationsuccess.jsp
03/12/2003	05:04p	4,856	admin_list.html
01/18/2003	12:42p	9,697	admin_monitor.jsp
03/12/2003	05:10p	5,292	admin_user_assign_unassign.htm
01/15/2003	10:17a	4,186	admin_navigation.jsp
01/22/2003	02:01p	10,219	admin_user_devices.jsp
01/16/2003	11:32a	10,775	admin_user_devices_alt.jsp
02/10/2003	05:01p	13,740	alert.jsp
12/04/2002	12:52p	740	alerts.html
02/10/2003	04:28p	4,851	alreadylogin.jsp
12/04/2002	12:43p	714	arm.html
11/13/2002	06:39p	22,835	Asthama.htm
11/13/2002	06:39p	5,154	banner.jsp
01/20/2003	01:15p	2,198	bpfilter.jsp
03/15/2003	04:58p	3,315	bptable.jsp
11/13/2002	06:39p	2,429	carematix.css
11/13/2002	06:39p	1,979	Carematix.html
11/15/2002	04:40p	1,134	carematix.js
01/15/2003	02:27p	6,921	cart.jsp
01/14/2003	02:35p	1,795	catalogue.jsp
12/04/2002	12:52p	747	central.html
11/15/2002	04:40p	694	changetononsecure.jsp
11/15/2002	04:40p	697	changetosecure.jsp
01/22/2003	02:57p	14,491	chart.jsp
12/04/2002	12:52p	753	charts.html
01/14/2003	02:38p	7,938	checkout.jsp
04/14/2003	06:08a		cm
03/12/2003	09:17a	5,913	cm.css
11/13/2002	06:39p	16,729	CompanyOverview.htm
11/13/2002	06:39p	16,275	Company_Overview.htm
11/19/2002	01:29p	13,509	conditions.html
11/13/2002	06:39p	34,636	Conditions_of_use.htm
01/20/2003	01:20p	4,989	confirmations.jsp
11/19/2002	01:27p	1,590	contact.html
11/13/2002	06:39p	11,509	contactus.htm
11/13/2002	06:39p	14,015	contact_us.htm
11/13/2002	06:39p	26,098	COPD.htm
01/20/2003	01:44p	15,622	createnewaccount.jsp
12/04/2002	01:26p	20,693	customer.html
11/13/2002	06:39p	21,497	Customer_support.htm
12/04/2002	12:51p	749	dashboard.html
11/19/2002	01:04p	2,865	dataupload.jsp
11/13/2002	06:39p	19,428	Development_team.htm
01/20/2003	01:45p	3,947	deviceconfirmations.jsp
11/15/2002	04:40p	653	devicecontrol.jsp
11/13/2002	06:39p	21,765	Diabetes.htm
12/04/2002	12:48p	715	diagram.html
11/19/2002	01:28p	1,123	employment.html
01/21/2003	11:18a	2,079	endDateCalendar.jsp
02/10/2003	04:11p	737	errorpage.html
02/10/2003	04:03p	661	errorpage.jsp
11/13/2002	06:39p	5,900	error_msg.htm
01/20/2003	01:52p	3,083	filterreadings.jsp
12/30/2002	03:14p	353	footer.jsp
11/15/2002	04:40p	177	frame.jsp

<DIR>

11/13/2002	06:39p	1,983	GE.html
12/04/2002	12:48p	726	glucometer.html
11/13/2002	06:39p	1,985	Healthcare.html
12/04/2002	10:46a	5,323	home.html
11/13/2002	06:39p	17,741	Hypertension.htm
04/14/2003	06:08a		images
01/31/2003	12:38p	378	index.jsp
12/27/2002	11:56a	551	index_comp.html
12/27/2002	11:55a	550	index_cust.html
12/27/2002	11:53a	552	index_solu.html
11/15/2002	05:03p	547	index.html
11/13/2002	06:39p	11,656	Investors.htm
11/15/2002	04:40p	2,028	isdataupload.jsp
11/13/2002	06:39p	1,488	leftregistration.jsp
02/10/2003	04:28p	5,722	leftalert.jsp
01/14/2003	03:35p	4,065	leftcheckout.jsp
01/14/2003	03:36p	4,061	leftconfirmations.jsp
01/14/2003	03:36p	4,061	leftcreatenewaccount.jsp
02/10/2003	04:15p	6,938	leftindex.jsp
11/15/2002	04:40p	2,524	leftindexredirect.jsp
01/14/2003	03:37p	4,066	leftorderconfirmations.jsp
01/14/2003	03:38p	4,066	leftorderpayments.jsp
01/14/2003	03:39p	4,078	leftordersummary.jsp
01/14/2003	03:39p	4,060	leftpayments.jsp
01/14/2003	03:40p	4,072	leftpickservice.jsp
01/15/2003	02:25p	1,590	leftproducts.jsp
01/14/2003	03:42p	5,126	leftregisterdevice.jsp
01/14/2003	03:43p	4,065	leftregisteredconfirmations.jsp
01/14/2003	03:44p	4,066	leftregisteredpayments.jsp
01/14/2003	03:45p	4,065	leftregisteredpickservice.jsp
01/15/2003	11:34a	4,062	leftsecurelogin.jsp
02/10/2003	04:29p	6,593	leftwelcome.jsp
03/13/2003	12:01p	1,223	left_admin_users_navigation.jsp
02/10/2003	04:15p	5,689	login.jsp
11/13/2002	06:39p	21,604	Management_Team.htm
12/04/2002	11:23a	8,839	markets.html
01/09/2003	12:24p	500	navigation.jsp
11/13/2002	06:39p	16,017	news.htm
11/19/2002	01:23p	1,906	news.html
01/22/2003	03:39p	314	newwindow.js
11/13/2002	06:39p	14,844	Obesity.htm
11/15/2002	04:40p	1,823	openecho.jsp
01/20/2003	01:58p	6,152	orderconfirmations.jsp
01/20/2003	02:05p	35,654	orderpayments.jsp
01/20/2003	02:05p	1,585	ordersummary.jsp
12/04/2002	12:17p	4,111	overview.html
01/20/2003	02:09p	33,212	payments.jsp
01/20/2003	02:17p	10,448	pickservice.jsp
11/19/2002	01:29p	19,401	potential.html
11/13/2002	06:39p	52,582	Potential_Users.htm
11/13/2002	06:39p	18,462	Preeclampsia.htm
11/15/2002	04:46p	24,965	privacy.html
11/13/2002	06:39p	51,197	privacy_policy.htm
11/13/2002	06:39p	14,656	products.htm
01/16/2003	01:43p	5,949	products.jsp
12/04/2002	11:05a	20,235	products.html
01/15/2003	02:13p	3,663	productdetails.jsp

01/20/2003	02:23p	8,194	registeredpickservice.jsp
01/18/2003	01:11p	16,246	registerdevice.jsp
01/20/2003	02:20p	5,478	registeredconfirmations.jsp
01/16/2003	01:55p	4,980	registeredDeviceDetails.jsp
01/15/2003	02:15p	32,405	registeredpayments.jsp
01/21/2003	02:01p	20,779	reminders.jsp
11/13/2002	06:39p	16,597	Secure_login.htm
02/10/2003	04:15p	6,025	securelogin.jsp
12/04/2002	05:53p	8,686	services.html
11/13/2002	06:39p	14,591	service_plans.htm
01/18/2003	01:11p	1,302	showAlert.jsp
11/15/2002	04:40p	1,545	signonredirector.jsp
01/18/2003	01:11p	605	signonsucessfull.jsp
12/04/2002	10:51a	4,268	solutions.html
02/10/2003	06:18p	2,080	startDateCalendar.jsp
11/08/2002	11:08a	559	sub_frame_potential.html
11/08/2002	10:11a	558	sub_frame_overview.html
11/08/2002	11:08a	556	sub_frame_privacy.html
12/09/2002	01:23p	3,655	subnav_home.html
12/09/2002	01:23p	3,966	subnav_overview.html
12/09/2002	01:22p	3,881	subnav_services.html
12/09/2002	01:23p	3,856	subnav_solutions.html
11/08/2002	11:07a	559	sub_frame_conditions.html
11/08/2002	11:07a	557	sub_frame_contact.html
11/08/2002	10:09a	550	sub_frame_home.html
11/08/2002	11:02a	554	sub_frame_news.html
11/08/2002	10:11a	558	sub_frame_services.html
11/14/2002	02:53p	560	sub_frame_solutions.html
12/09/2002	01:23p	3,587	subnav_support.html
11/15/2002	08:26a	557	sub_frame_support.html
11/13/2002	06:39p	26,233	summary.htm
12/04/2002	11:12a	8,568	support.html
01/22/2003	03:05p	14,265	table.jsp
02/10/2003	04:52p	4,892	template.jsp
11/15/2002	04:40p	442	templatexml.jsp
01/20/2003	01:14p	1,906	tempraturefilter.jsp
03/15/2003	04:59p	2,723	tempraturetable.jsp
12/30/2002	12:09p	67	topindex.jsp
01/31/2003	12:28p	1,126	top.html
01/22/2003	02:23p	44,577	trends.jsp
03/12/2003	05:06p	3,896	unassigned_user_list.htm
11/13/2002	06:39p	11,648	Untitled-1.html
01/18/2003	01:11p	3,916	usernotification.jsp
03/12/2003	05:06p	4,715	user_list.htm
11/13/2002	06:39p	15,501	web_services.htm
12/04/2002	12:43p	717	weight.html
01/20/2003	02:26p	1,957	weightfilter.jsp
03/15/2003	04:58p	2,705	weighttable.jsp
03/14/2003	04:52p	23,336	welcome.jsp
11/13/2002	06:39p	21,571	Welcome to Carematix.htm
11/13/2002	06:39p	1,986	Wipro.html
12/04/2002	12:48p	730	wrist.html
164 File(s)		1,298,447	bytes

Directory of F:\Server\UI\cm

04/14/2003 06:08a <DIR>

```

04/14/2003  06:08a      <DIR>      ..
01/16/2003  05:38p                10,813 cm.css
11/27/2002  10:36a                3,043 floater.js
                2 File(s)           13,856 bytes

```

Directory of F:\Server\UI\images

```

04/14/2003  06:08a      <DIR>      .
04/14/2003  06:08a      <DIR>      ..
03/12/2003  08:22a                564 admin_list.gif
03/12/2003  08:23a                565 admin_list_over.gif
12/04/2002  12:33p           15,474 admin_preview.gif
11/15/2002  04:39p           3,053 alert.gif
11/11/2002  01:57p           13,048 alerts.gif
12/04/2002  12:35p           17,669 alerts_preview.gif
03/12/2003  08:23a           546 all_users.gif
03/12/2003  08:23a           549 all_users_over.gif
12/04/2002  01:00p           5,605 arm.gif
12/04/2002  12:37p           16,653 arm_preview.gif
01/21/2003  10:17a           104 arrow_up.gif
01/21/2003  10:19a           106 arrow_down.gif
03/12/2003  08:25a           524 assign.gif
03/12/2003  08:26a           524 assign_over.gif
11/11/2002  01:56p           637 back.gif
11/15/2002  04:39p           4,048 Banner.GIF
11/15/2002  04:39p           907 bm.gif
11/15/2002  04:39p          14,744 bp_big1.gif
11/15/2002  04:39p          13,310 bp_big2.gif
11/15/2002  04:39p          17,724 bp_big3.gif
11/15/2002  04:39p          15,550 bp_small1.gif
11/15/2002  04:39p           5,228 bp_small2.gif
11/15/2002  04:39p           5,902 bp_small3.gif
11/15/2002  04:39p           3,543 Buddy.gif
11/15/2002  04:39p           571 buddy_blue.gif
11/15/2002  04:39p           572 buddy_green.gif
11/15/2002  04:39p           2,579 Buddy_list.gif
04/14/2003  06:08a      <DIR>      Button
11/15/2002  04:40p           923 by.gif
01/21/2003  11:13a           440 cal.gif
11/15/2002  04:40p            90 cart_deactivate.gif
11/15/2002  04:40p            90 cart_trial.gif
11/15/2002  04:40p            90 cart_trial_over.gif
12/04/2002  01:03p           7,511 central.gif
12/04/2002  01:05p           8,639 charts.gif
12/04/2002  12:39p          18,540 charts_preview.gif
11/15/2002  04:40p           1,751 Checkout.gif
12/26/2002  11:40a           2,720 checkout_step1.gif
12/26/2002  11:40a           2,697 checkout_step2.gif
12/26/2002  11:41a           2,680 checkout_step3.gif
12/26/2002  11:41a           2,669 checkout_step4.gif
11/15/2002  04:40p            292 code.gif
11/15/2002  04:40p          40,216 collage.gif
12/26/2002  11:37a           4,031 combo_timeline.gif
11/15/2002  04:40p           2,015 confirmation.gif
12/04/2002  12:57p           6,717 dashboard.gif
12/04/2002  12:40p          14,152 dashboard_preview.gif
03/12/2003  08:25a           709 demote.gif

```

03/12/2003	08:25a	712	demote_over.gif
11/15/2002	04:40p	2,878	devices.gif
12/04/2002	12:58p	7,730	diagram.gif
12/04/2002	12:30p	7,228	diagram_home.gif
12/04/2002	12:32p	15,646	diagram_preview.gif
11/11/2002	01:53p	10,276	doctor.gif
11/12/2002	03:10p	6,787	doctor.jpg
01/21/2003	10:17a	108	dot.gif
11/11/2002	01:47p	62	end_cap.gif
11/11/2002	01:06p	131	end_slant.gif
11/11/2002	01:03p	82	end_slant_2.gif
01/21/2003	10:16a	99	equal.gif
01/21/2003	10:16a	331	ERROR.GIF
11/15/2002	04:40p	6,343	EX11.JPG
01/21/2003	10:14a	146	exclaim.gif
01/21/2003	10:13a	124	exclamation.gif
11/15/2002	04:40p	1,242	execute.gif
12/04/2002	12:35p	17,783	glucometer_preview.gif
11/11/2002	02:01p	6,793	glucose.gif
12/04/2002	12:58p	16,607	irwin.gif
01/15/2003	10:04a	148	logged_in.gif
01/15/2003	10:05a	149	logged_out.gif
10/11/2002	10:38a	3,255	logo_top.gif
03/12/2003	08:26a	619	make_admin.gif
03/12/2003	08:26a	622	make_admin_over.gif
11/15/2002	04:40p	2,356	myhealth.gif
12/04/2002	01:01p	14,671	navneet.gif
11/15/2002	04:40p	908	nm.gif
11/15/2002	04:40p	925	ny.gif
11/15/2002	04:40p	1,948	order_summary.gif
11/15/2002	04:40p	2,018	payments.gif
12/30/2002	02:29p	3,418	per_step1.gif
12/26/2002	11:38a	3,350	per_step3.gif
12/26/2002	11:39a	3,319	per_step4.gif
12/26/2002	11:39a	3,311	per_step5.gif
01/02/2003	02:39p	43	pixel.gif
12/04/2002	12:57p	6,576	praveen.gif
01/21/2003	10:13a	227	print.gif
04/14/2003	06:08a		<DIR> Products
11/15/2002	04:40p	2,075	registeredstep1.gif
11/15/2002	04:40p	2,124	registeredstep2.gif
11/15/2002	04:40p	2,092	registeredstep3.gif
11/15/2002	04:40p	2,010	registeredstep4.gif
12/26/2002	11:42a	2,922	reg_add_step1.gif
12/26/2002	11:42a	2,912	reg_add_step2.gif
12/26/2002	11:43a	2,900	reg_add_step3.gif
12/26/2002	11:43a	2,873	reg_add_step4.gif
01/21/2003	10:28a	532	reminders.gif
01/21/2003	10:29a	526	reminders_over.gif
11/15/2002	04:40p	1,231	return.gif
12/04/2002	01:01p	14,012	sandeep.gif
11/15/2002	04:40p	124	secure_lock.gif
11/15/2002	04:40p	908	select.gif
01/13/2003	03:55p	85	separator.gif
11/11/2002	01:00p	101	slant_1.gif
11/11/2002	12:59p	137	slant_1a.gif
11/11/2002	01:05p	110	slant_2.gif

11/11/2002	01:01p	131	slant_3.gif
04/19/2002	10:32a	79	spacer.GIF
11/11/2002	01:46p	76	start_cap.gif
11/15/2002	04:40p	2,184	Step1.gif
11/15/2002	04:40p	2,444	Step2.gif
11/15/2002	04:40p	2,466	Step3.gif
11/15/2002	04:40p	2,456	Step4.gif
11/15/2002	04:40p	2,373	Step5.gif
12/04/2002	01:00p	15,619	stu.gif
12/04/2002	12:59p	12,543	sukwant.gif
01/13/2003	01:35p	437	swoosh.gif
11/11/2002	01:03p	51	tab_1.gif
11/11/2002	01:04p	50	tab_2.gif
11/15/2002	04:40p	4,291	thermo_small2.gif
11/15/2002	04:40p	2,359	thermo_small3.gif
11/15/2002	04:40p	14,154	thermo_big1.gif
11/15/2002	04:40p	12,308	thermo_big2.gif
11/15/2002	04:40p	5,640	thermo_big3.gif
11/15/2002	04:40p	4,318	thermo_small1.gif
12/04/2002	01:03p	4,988	top_header.gif
10/11/2002	10:40a	49	top_tile.gif
11/15/2002	04:40p	3,250	trends.gif
03/12/2003	09:15a	568	unassign.gif
03/12/2003	08:24a	751	unassigned_users.gif
03/12/2003	08:24a	754	unassigned_users_over.gif
03/12/2003	09:16a	568	unassign_over.gif
01/17/2003	02:26p	560	update_baseline.gif
01/17/2003	02:26p	552	update_baseline_over.gif
01/17/2003	02:25p	608	Update_my_account.gif
01/17/2003	02:25p	604	Update_my_account_over.gif
03/15/2003	04:36p	1,152	vssver.scc
11/11/2002	02:01p	8,273	weight.gif
12/04/2002	12:36p	17,879	weight_preview.gif
11/11/2002	02:03p	9,514	wrist.gif
12/04/2002	12:34p	18,599	wrist_preview.gif
11/15/2002	04:40p	5,396	wt_scale_small2.gif
11/15/2002	04:40p	5,681	wt_scale_small3.gif
11/15/2002	04:40p	18,491	wt_scale_big1.gif
11/15/2002	04:40p	13,630	wt_scale_big2.gif
11/15/2002	04:40p	15,480	wt_scale_big3.gif
11/15/2002	04:40p	6,737	wt_scale_small1.gif
144 File(s)		671,979	bytes

Directory of F:\Server\UI\images\Button

04/14/2003	06:08a	<DIR>	.
04/14/2003	06:08a	<DIR>	..
11/15/2002	04:39p		1,409 About_Us.gif
11/15/2002	04:39p		1,409 About_Us_over.gif
01/17/2003	01:46p		744 accept_displayed_readings.gif
01/17/2003	01:40p		910 accept_displayed_readings_o.gif
01/17/2003	01:40p		910 accept_displayed_readings_over.gif
01/17/2003	01:45p		568 add_to_cart.gif
01/17/2003	01:46p		550 add_to_cart_over.gif
01/17/2003	01:44p		625 admin_monitor.gif
01/17/2003	01:44p		595 admin_monitor_over.gif
01/17/2003	01:43p		513 calendar.gif

01/17/2003	01:43p	505	calendar_over.gif
01/17/2003	01:42p	459	cancel.gif
01/17/2003	01:42p	455	cancel_over.gif
11/15/2002	04:39p	761	cart.gif
11/15/2002	04:39p	761	cart_over.gif
11/15/2002	04:39p	2,318	change_per_alerts.gif
11/15/2002	04:39p	2,320	change_per_alerts_over.gif
01/17/2003	01:41p	430	chart.gif
01/17/2003	01:41p	422	chart_over.gif
01/17/2003	01:54p	536	checkout.gif
01/17/2003	01:55p	526	checkout_over.gif
01/17/2003	01:54p	589	continue_shopping_over.gif
01/17/2003	01:53p	517	continue.gif
11/15/2002	04:39p	1,513	continue-shopping.gif
11/15/2002	04:39p	1,513	continue-shopping_over.gif
01/17/2003	01:54p	507	continue_over.gif
01/17/2003	01:53p	609	continue_shopping.gif
01/17/2003	01:52p	592	create_my_acct.gif
01/17/2003	01:52p	581	create_my_acct_over.gif
11/15/2002	04:39p	1,728	Customer_support.gif
11/15/2002	04:39p	1,728	Customer_support_over.gif
01/17/2003	01:50p	467	Delete.gif
01/17/2003	01:51p	454	Delete_over.gif
01/17/2003	01:50p	735	delete_selected_readings.gif
01/17/2003	01:51p	708	delete_selected_readings_ov.gif
01/17/2003	01:51p	708	delete_selected_readings_over.gif
01/22/2003	03:37p	15,364	IDS_Store
01/20/2003	03:07p	356	Enter.gif
01/20/2003	03:10p	359	Enter_over.gif
11/15/2002	04:39p	946	Enter_small.gif
11/15/2002	04:39p	946	Enter_small_over.gif
11/15/2002	04:39p	1,377	FAQs.gif
11/15/2002	04:39p	1,377	FAQs_over.gif
01/17/2003	01:48p	424	finish.gif
01/17/2003	01:48p	428	finish_over.gif
11/15/2002	04:39p	999	Go_For_Demo.gif
11/15/2002	04:39p	999	Go_For_Demo_over.gif
11/15/2002	04:39p	1,375	Home.gif
11/15/2002	04:39p	1,375	Home_over.gif
11/15/2002	04:39p	2,284	Invite_buddy.gif
11/15/2002	04:39p	2,285	Invite_buddy_over.gif
01/22/2003	02:48p	628	last_months.gif
01/22/2003	02:49p	610	last_months_charts.gif
01/22/2003	02:49p	589	last_months_charts_over.gif
01/22/2003	02:49p	624	last_months_over.gif
01/22/2003	02:48p	648	last_weeks.gif
01/22/2003	02:47p	613	last_weeks_charts.gif
01/22/2003	02:47p	605	last_weeks_charts_over.gif
01/22/2003	02:46p	636	last_weeks_over.gif
11/15/2002	04:39p	1,425	Learn_more.gif
11/15/2002	04:39p	1,425	Learn_more_over.gif
11/15/2002	04:39p	953	my_acct.gif
11/15/2002	04:39p	953	my_acct_over.gif
11/15/2002	04:39p	1,345	News.gif
11/15/2002	04:39p	949	news_disable.gif
11/15/2002	04:39p	1,345	News_over.gif
01/17/2003	02:09p	518	payments.gif

01/17/2003	01:47p	531	payments_over.gif
01/17/2003	02:08p	577	Pick_service.gif
01/17/2003	02:08p	570	Pick_service_over.gif
11/15/2002	04:39p	1,408	Products.gif
11/15/2002	04:39p	1,408	Products_over.gif
11/15/2002	04:39p	917	profile.gif
11/15/2002	04:39p	987	profile_disable.gif
11/15/2002	04:39p	917	profile_over.gif
01/17/2003	02:07p	590	recalculate.gif
01/17/2003	02:08p	578	recalculate_over.gif
01/20/2003	03:08p	497	register_over.gif
01/20/2003	03:07p	507	register.gif
01/17/2003	02:06p	585	register_a_device.gif
01/17/2003	02:06p	570	register_a_device_over.gif
01/17/2003	02:05p	645	register_more_devices.gif
01/17/2003	02:05p	638	register_more_devices_over.gif
11/15/2002	04:39p	997	reminders.gif
11/15/2002	04:39p	997	reminders_over.gif
01/17/2003	02:04p	439	Reset.gif
01/17/2003	02:03p	439	Reset1.gif
01/17/2003	02:03p	437	Reset1_over.gif
01/17/2003	02:04p	437	Reset_over.gif
11/15/2002	04:40p	1,400	Services.gif
11/15/2002	04:40p	1,400	Services_over.gif
01/17/2003	03:54p	550	set_alerts.gif
01/17/2003	03:52p	527	set_alerts_over.gif
01/17/2003	03:53p	560	set_trends.gif
01/17/2003	03:53p	550	set_trends_over.gif
01/17/2003	02:02p	479	submit.gif
01/17/2003	02:03p	476	submit_over.gif
11/15/2002	04:40p	916	tables.gif
11/15/2002	04:40p	983	tables_disable.gif
11/15/2002	04:40p	916	tables_over.gif
11/15/2002	04:40p	1,423	Take_a_Tour.gif
11/15/2002	04:40p	1,016	Take_a_Tour_disable.gif
11/15/2002	04:40p	1,423	Take_a_Tour_over.gif
01/17/2003	02:01p	599	Track_orders.gif
01/17/2003	02:02p	584	Track_orders_over.gif
01/20/2003	03:08p	559	unregister.gif
01/20/2003	03:09p	551	unregister_over.gif
11/15/2002	04:40p	1,533	Update_my_account.gif
01/17/2003	02:25p	476	update_over.gif
01/17/2003	02:25p	608	Update_my_acct.gif
01/17/2003	02:25p	604	Update_my_acct_over.gif
01/17/2003	02:24p	492	update_small.gif
01/17/2003	02:23p	490	update.gif
01/17/2003	02:26p	560	update_baseline.gif
01/17/2003	02:26p	552	update_baseline_over.gif
11/15/2002	04:40p	1,410	update_cart.gif
11/15/2002	04:40p	1,410	update_cart_over.gif
11/15/2002	04:40p	1,538	Update_my_account_over.gif
01/17/2003	02:24p	425	update_small_over.gif
01/17/2003	02:23p	590	user_devices.gif
01/17/2003	02:23p	598	user_devices_over.gif
01/17/2003	02:22p	413	view.gif
01/17/2003	02:22p	410	view_over.gif
03/15/2003	04:36p	1,696	vssver.scc

124 File(s) 119,823 bytes

Directory of F:\Server\UI\images\Products

04/14/2003	06:08a	<DIR>	.
04/14/2003	06:08a	<DIR>	..
11/15/2002	04:40p		898 bp01.jpg
11/15/2002	04:40p		13,310 bp02.gif
11/15/2002	04:40p		13,310 bpmeter.gif
11/15/2002	04:40p		898 bpmeter.jpg
11/15/2002	04:40p		14,744 bp_big1.gif
11/15/2002	04:40p		5,228 bp_small12.gif
11/15/2002	04:40p		1,170 glucomter.jpg
11/15/2002	04:40p		824 HeartRate monitor.jpg
11/15/2002	04:40p		12,653 thermo02.jpg
11/15/2002	04:40p		5,640 thermo1.gif
11/15/2002	04:40p		7,443 thermo1.jpg
11/15/2002	04:40p		5,640 thermometer.gif
11/15/2002	04:40p		536 thermometer.jpg
03/15/2003	04:36p		304 vssver.scc
11/15/2002	04:40p		15,480 weigh01.gif
11/15/2002	04:40p		1,101 weigh02.jpg
11/15/2002	04:40p		15,480 weightscale.gif
11/15/2002	04:40p		1,101 weightscale.jpg
			18 File(s) 115,760 bytes

Directory of F:\Server\Webapp

04/14/2003	06:08a	<DIR>	.
04/14/2003	06:08a	<DIR>	..
04/14/2003	05:58a		836 v.java
04/14/2003	05:58a		406 vv.java
04/14/2003	05:58a		539 vvvv.java
04/14/2003	05:58a		23,271 vvvw.java
04/14/2003	05:58a		407 vvvx.java
04/14/2003	05:58a		3,533 vvvv.java
04/14/2003	05:58a		1,431 vvv.java
04/14/2003	05:58a		1,287 vvvw.java
04/14/2003	05:58a		2,833 vvvw.java
04/14/2003	05:58a		1,618 vvvx.java
04/14/2003	05:58a		824 vvvv.java
04/14/2003	05:58a		1,399 vvv.java
04/14/2003	05:58a		1,382 vvvv.java
04/14/2003	05:58a		325 vvvw.java
04/14/2003	05:58a		9,959 vvvx.java
04/14/2003	05:58a		589 vvv.java
04/14/2003	05:58a		1,706 vvvv.java
04/14/2003	05:58a		273 vvvw.java
04/14/2003	05:58a		457 vvvx.java
04/14/2003	05:58a		4,389 vvvv.java
04/14/2003	05:58a		1,237 vvv.java
04/14/2003	05:58a		582 vvvv.java
04/14/2003	05:58a		3,053 vvvw.java
04/14/2003	05:58a		2,339 vvvx.java
04/14/2003	05:58a		918 vvvv.java
04/14/2003	05:58a		265 vv.java
04/14/2003	05:58a		3,600 vvvv.java

04/14/2003	05:58a	1,013	vwww.java
04/14/2003	05:58a	907	vwx.x.java
04/14/2003	05:58a	3,728	vwy.y.java
04/14/2003	05:58a	298	vww.java
04/14/2003	05:58a	440	vwww.java
04/14/2003	05:58a	3,434	vwww.java
04/14/2003	05:58a	629	vwx.x.java
04/14/2003	05:58a	1,587	vwy.y.java
04/14/2003	05:58a	362	vww.java
04/14/2003	05:58a	951	vwxv.java
04/14/2003	05:58a	498	vwxw.java
04/14/2003	05:58a	1,403	vwx.x.java
04/14/2003	05:58a	454	vwx.y.java
04/14/2003	05:58a	1,836	vwx.java
04/14/2003	05:58a	1,172	vwyv.java
04/14/2003	05:58a	1,188	vwyw.java
04/14/2003	05:58a	431	vwyx.java
04/14/2003	05:58a	422	vwy.y.java
04/14/2003	05:58a	237	vwy.java
04/14/2003	05:58a	457	vwzv.java
04/14/2003	05:58a	873	vwzw.java
04/14/2003	05:58a	857	vwzx.java
04/14/2003	05:58a	359	vwzy.java
04/14/2003	05:58a	745	vx.java
04/14/2003	05:58a	1,031	vxxv.java
04/14/2003	05:58a	708	vxxw.java
04/14/2003	05:58a	392	vxx.x.java
04/14/2003	05:58a	543	vxx.y.java
04/14/2003	05:58a	7,436	vxx.java
04/14/2003	05:58a	8,275	vxxv.java
04/14/2003	05:58a	1,268	vxxw.java
04/14/2003	05:58a	190	vxx.x.java
04/14/2003	05:58a	1,993	vxx.y.java
04/14/2003	05:58a	5,845	vxx.java
04/14/2003	05:58a	7,364	vxxv.java
04/14/2003	05:58a	6,707	vxxw.java
04/14/2003	05:58a	2,652	vxx.x.java
04/14/2003	05:58a	276	vxx.y.java
04/14/2003	05:58a	834	vxyv.java
04/14/2003	05:58a	1,552	vxyw.java
04/14/2003	05:58a	8,525	vxyx.java
04/14/2003	05:58a	382	vxy.y.java
04/14/2003	05:58a	8,544	vxy.java
04/14/2003	05:58a	2,196	vxzv.java
04/14/2003	05:58a	490	vxzw.java
04/14/2003	05:58a	924	vxzx.java
04/14/2003	05:58a	1,135	vxzy.java
04/14/2003	05:58a	455	vy.java
04/14/2003	05:58a	1,769	vyv.v.java
04/14/2003	05:58a	10,043	vyvw.java
04/14/2003	05:58a	432	vyvx.java
04/14/2003	05:58a	1,075	vyvy.y.java
04/14/2003	05:58a	474	vyv.java
04/14/2003	05:58a	1,057	vywv.java
04/14/2003	05:58a	537	vyww.java
04/14/2003	05:58a	457	vywx.java
04/14/2003	05:58a	529	vywy.y.java

04/14/2003	05:58a	944	vyw.java
04/14/2003	05:58a	5,718	vyxv.java
04/14/2003	05:58a	441	vyxw.java
04/14/2003	05:58a	3,226	vyxy.java
04/14/2003	05:58a	527	vyx.java
04/14/2003	05:58a	402	vyyv.java
04/14/2003	05:58a	17,503	vyyw.java
04/14/2003	05:58a	382	vyyx.java
04/14/2003	05:58a	15,837	vyyy.java
04/14/2003	05:58a	1,262	vyy.java
04/14/2003	05:58a	455	vyzv.java
04/14/2003	05:58a	433	vyzw.java
04/14/2003	05:58a	385	vyzx.java
04/14/2003	05:58a	301	vzyy.java
04/14/2003	05:58a	7,534	vzvz.java
04/14/2003	05:58a	389	vzvw.java
04/14/2003	05:58a	2,827	vzvx.java
04/14/2003	05:58a	454	vzvy.java
04/14/2003	05:58a	284	vzv.java
04/14/2003	05:59a	8,073	vzww.java
04/14/2003	05:59a	296	vzww.java
04/14/2003	05:59a	12,249	vzwx.java
04/14/2003	05:59a	1,080	vzwy.java
04/14/2003	05:58a	502	vzw.java
04/14/2003	05:59a	1,062	vzxv.java
04/14/2003	05:59a	2,204	vzxw.java
04/14/2003	05:59a	1,125	vzxx.java
04/14/2003	05:59a	2,183	vzxy.java
04/14/2003	05:59a	914	vzx.java
04/14/2003	05:59a	1,643	vzyv.java
04/14/2003	05:59a	301	vzyw.java
04/14/2003	05:59a	23,947	vzyx.java
04/14/2003	05:59a	631	vzyy.java
04/14/2003	05:59a	32,802	vzy.java
04/14/2003	05:59a	257	vzzv.java
04/14/2003	05:59a	2,673	vzzw.java
04/14/2003	05:59a	3,322	vzzx.java
04/14/2003	05:59a	262	vzzy.java
04/14/2003	05:59a	530	w.java
04/14/2003	05:59a	966	wv.java
04/14/2003	05:59a	1,539	wvvv.java
04/14/2003	05:59a	2,998	wvvw.java
04/14/2003	05:59a	462	wvwx.java
04/14/2003	05:59a	384	wvvy.java
04/14/2003	05:59a	3,952	wvv.java
04/14/2003	05:59a	504	wvww.java
04/14/2003	05:59a	6,145	wvww.java
04/14/2003	05:59a	393	wvwx.java
04/14/2003	05:59a	221	wvwy.java
04/14/2003	05:59a	1,042	wvw.java
04/14/2003	05:59a	2,308	wvxv.java
04/14/2003	05:59a	2,341	wvxw.java
04/14/2003	05:59a	945	wvxy.java
04/14/2003	05:59a	467	wvx.java
04/14/2003	05:59a	3,620	wvyv.java
04/14/2003	05:59a	438	wvyw.java
04/14/2003	05:59a	493	wvyx.java

04/14/2003	05:59a	390	wvyy.java
04/14/2003	05:59a	713	wvy.java
04/14/2003	05:59a	405	wvzv.java
04/14/2003	05:59a	1,332	wvzw.java
04/14/2003	05:59a	394	wvzx.java
04/14/2003	05:59a	997	wvzy.java
04/14/2003	05:59a	271	ww.java
04/14/2003	05:59a	1,835	wwvv.java
04/14/2003	05:59a	1,423	wwvw.java
04/14/2003	05:59a	1,103	wwvx.java
04/14/2003	05:59a	9,640	wwvy.java
04/14/2003	05:59a	1,354	wwv.java
04/14/2003	05:59a	467	wwwv.java
04/14/2003	05:59a	1,152	wwwv.java
04/14/2003	05:59a	511	wwwx.java
04/14/2003	05:59a	12,867	wwwy.java
04/14/2003	05:59a	293	www.java
04/14/2003	05:59a	618	wxxv.java
04/14/2003	05:59a	1,898	wxxw.java
04/14/2003	05:59a	308	wxxx.java
04/14/2003	05:59a	1,100	wxxy.java
04/14/2003	05:59a	386	wxx.java
04/14/2003	05:59a	11,514	wxyv.java
04/14/2003	05:59a	2,433	wxyw.java
04/14/2003	05:59a	687	wxyx.java
04/14/2003	05:59a	398	wxyy.java
04/14/2003	05:59a	602	wyy.java
04/14/2003	05:59a	801	wyzv.java
04/14/2003	05:59a	297	wyzw.java
04/14/2003	05:59a	4,454	wyzx.java
04/14/2003	05:59a	1,232	wyzy.java
04/14/2003	05:59a	243	wx.java
04/14/2003	05:59a	387	wxvv.java
04/14/2003	05:59a	374	wxvw.java
04/14/2003	05:59a	671	wxvx.java
04/14/2003	05:59a	5,515	wxvy.java
04/14/2003	05:59a	831	wxv.java
04/14/2003	05:59a	289	wxwv.java
04/14/2003	05:59a	1,088	wxww.java
04/14/2003	05:59a	430	wxwx.java
04/14/2003	05:59a	1,137	wxwy.java
04/14/2003	05:59a	382	wxw.java
04/14/2003	05:59a	614	wxxv.java
04/14/2003	05:59a	1,085	wxxw.java
04/14/2003	05:59a	1,074	wxxy.java
04/14/2003	05:59a	1,638	wxx.java
04/14/2003	05:59a	296	wxyv.java
04/14/2003	05:59a	314	wxyw.java
04/14/2003	05:59a	3,742	wxyx.java
04/14/2003	05:59a	612	wxyy.java
04/14/2003	05:59a	11,027	wxy.java
04/14/2003	05:59a	16,219	wxzv.java
04/14/2003	05:59a	8,761	wxzw.java
04/14/2003	05:59a	7,471	wzxz.java
04/14/2003	05:59a	727	wxzy.java
04/14/2003	05:59a	1,913	wy.java
04/14/2003	05:59a	3,024	wyvv.java

04/14/2003	05:59a	301	wyvw.java
04/14/2003	05:59a	1,209	wyvx.java
04/14/2003	05:59a	7,854	wyvy.java
04/14/2003	05:59a	1,194	wyv.java
04/14/2003	05:59a	2,382	wyvw.java
04/14/2003	05:59a	1,652	wyww.java
04/14/2003	05:59a	382	wywx.java
04/14/2003	05:59a	934	wywy.java
04/14/2003	05:59a	7,726	wyw.java
04/14/2003	05:59a	1,730	wy xv.java
04/14/2003	05:59a	2,149	wy xw.java
04/14/2003	05:59a	400	wy xx.java
04/14/2003	05:59a	30,517	wy xy.java
04/14/2003	05:59a	486	wy x.java
04/14/2003	05:59a	243	wy yv.java
04/14/2003	05:59a	3,113	wy yw.java
04/14/2003	05:59a	8,101	wy yx.java
04/14/2003	05:59a	5,041	wy yy.java
04/14/2003	05:59a	293	wy y.java
04/14/2003	05:59a	653	wy zv.java
04/14/2003	05:59a	832	wy zw.java
04/14/2003	05:59a	3,121	wy zx.java
04/14/2003	05:59a	4,159	wy zy.java
04/14/2003	05:59a	261	wz vv.java
04/14/2003	05:59a	1,867	wz vw.java
04/14/2003	05:59a	723	wz vx.java
04/14/2003	05:59a	1,969	wz vy.java
04/14/2003	05:59a	379	wz v.java
04/14/2003	05:59a	296	wz wv.java
04/14/2003	05:59a	1,066	wz ww.java
04/14/2003	05:59a	806	wz wx.java
04/14/2003	05:59a	374	wz wy.java
04/14/2003	05:59a	464	wz w.java
04/14/2003	05:59a	469	wz xv.java
04/14/2003	05:59a	340	wz xw.java
04/14/2003	05:59a	901	wz xx.java
04/14/2003	05:59a	524	wz xy.java
04/14/2003	05:59a	219	wz x.java
04/14/2003	05:59a	996	wz yv.java
04/14/2003	05:59a	451	wz yw.java
04/14/2003	05:59a	1,136	wz yx.java
04/14/2003	05:59a	445	wz yy.java
04/14/2003	05:59a	688	wz y.java
04/14/2003	05:59a	797	wz zv.java
04/14/2003	05:59a	654	wz zw.java
04/14/2003	05:59a	296	wz zx.java
04/14/2003	05:59a	3,315	wz zy.java
04/14/2003	05:59a	262	x.java
04/14/2003	05:59a	380	xv.java
04/14/2003	05:59a	1,481	xvvv.java
04/14/2003	05:59a	1,568	xvvw.java
04/14/2003	05:59a	1,226	xv vx.java
04/14/2003	05:59a	436	xv vy.java
04/14/2003	05:59a	4,468	xv v.java
04/14/2003	05:59a	693	xv vw.java
04/14/2003	05:59a	9,394	xv ww.java
04/14/2003	05:59a	409	xv wx.java

04/14/2003	05:59a	1,489	xvwy.java
04/14/2003	05:59a	393	xvw.java
04/14/2003	05:59a	2,752	xvxv.java
04/14/2003	05:59a	1,406	xvxw.java
04/14/2003	05:59a	1,108	xvxy.java
04/14/2003	05:59a	293	xvx.java
04/14/2003	05:59a	190	xvyv.java
04/14/2003	05:59a	603	xvyw.java
04/14/2003	05:59a	459	xvyx.java
04/14/2003	05:59a	340	xvyy.java
04/14/2003	05:59a	1,166	xvy.java
04/14/2003	05:59a	1,287	xvzv.java
04/14/2003	05:59a	479	xvzw.java
04/14/2003	05:59a	450	xvzx.java
04/14/2003	05:59a	474	xvzy.java
04/14/2003	05:59a	576	xw.java
04/14/2003	05:59a	1,441	xwvv.java
04/14/2003	05:59a	591	xwvw.java
04/14/2003	05:59a	5,412	xwvx.java
04/14/2003	05:59a	3,844	xwvy.java
04/14/2003	05:59a	305	xwv.java
04/14/2003	05:59a	11,344	xwwv.java
04/14/2003	05:59a	4,363	xwww.java
04/14/2003	05:59a	429	xwwx.java
04/14/2003	05:59a	2,365	xwwy.java
04/14/2003	05:59a	12,440	xww.java
04/14/2003	05:59a	296	xwxv.java
04/14/2003	05:59a	1,440	xwxw.java
04/14/2003	05:59a	455	xwxx.java
04/14/2003	05:59a	1,074	xwxy.java
04/14/2003	05:59a	899	xwx.java
04/14/2003	05:59a	2,855	xwyv.java
04/14/2003	05:59a	4,471	xwyw.java
04/14/2003	05:59a	36,426	xwyx.java
04/14/2003	05:59a	327	xwyy.java
04/14/2003	05:59a	298	xwy.java
04/14/2003	05:59a	608	xwzv.java
04/14/2003	05:59a	1,719	xwzw.java
04/14/2003	05:59a	515	xwzx.java
04/14/2003	05:59a	377	xwzy.java
04/14/2003	05:59a	874	xx.java
04/14/2003	05:59a	451	xxvv.java
04/14/2003	05:59a	296	xxvw.java
04/14/2003	05:59a	221	xxvx.java
04/14/2003	05:59a	45,924	xxvy.java
04/14/2003	05:59a	10,925	xxv.java
04/14/2003	05:59a	9,190	xxwv.java
04/14/2003	05:59a	10,754	xxww.java
04/14/2003	05:59a	17,015	xxwx.java
04/14/2003	05:59a	392	xxwy.java
04/14/2003	05:59a	393	xxw.java
04/14/2003	05:59a	267	xxxv.java
04/14/2003	05:59a	277	xxxw.java
04/14/2003	05:59a	710	xxxxy.java
04/14/2003	05:59a	448	xxx.java
04/14/2003	05:59a	411	xxyv.java
04/14/2003	05:59a	2,831	xxyw.java

04/14/2003	05:59a	9,413	xxyx.java
04/14/2003	05:59a	796	xxyy.java
04/14/2003	05:59a	474	xyx.java
04/14/2003	05:59a	239	xxzv.java
04/14/2003	05:59a	380	xxzw.java
04/14/2003	05:59a	1,650	xxzx.java
04/14/2003	05:59a	1,477	xxzy.java
04/14/2003	05:59a	1,655	xy.java
04/14/2003	05:59a	4,539	xyvv.java
04/14/2003	05:59a	108,998	xyvw.java
04/14/2003	05:59a	262	xyvx.java
04/14/2003	05:59a	308	xyvy.java
04/14/2003	05:59a	71,040	xyv.java
04/14/2003	05:59a	631	xywv.java
04/14/2003	05:59a	1,094	xyww.java
04/14/2003	05:59a	29,550	xywx.java
04/14/2003	05:59a	943	xywy.java
04/14/2003	05:59a	685	xyw.java
04/14/2003	05:59a	308	xyxv.java
04/14/2003	05:59a	1,205	xyxw.java
04/14/2003	05:59a	296	xyxy.java
04/14/2003	05:59a	527	xyx.java
04/14/2003	05:59a	5,527	xyyv.java
04/14/2003	05:59a	542	xyyw.java
04/14/2003	05:59a	838	xyyx.java
04/14/2003	05:59a	755	xyyy.java
04/14/2003	05:59a	267	xyy.java
04/14/2003	05:59a	991	xyzv.java
04/14/2003	05:59a	264	xyzw.java
04/14/2003	05:59a	1,319	xyzx.java
04/14/2003	05:59a	575	xyzy.java
04/14/2003	05:59a	15,657	xzvv.java
04/14/2003	05:59a	1,978	xzvw.java
04/14/2003	05:59a	336	xzvx.java
04/14/2003	05:59a	301	xzvy.java
04/14/2003	05:59a	5,796	xzv.java
04/14/2003	05:59a	400	xzww.java
04/14/2003	05:59a	415	xzww.java
04/14/2003	05:59a	1,060	xzwx.java
04/14/2003	05:59a	4,149	xzwy.java
04/14/2003	05:59a	804	xzw.java
04/14/2003	05:59a	381	xzxv.java
04/14/2003	05:59a	2,667	xzxw.java
04/14/2003	05:59a	966	xzxx.java
04/14/2003	05:59a	249	xzxy.java
04/14/2003	05:59a	3,868	xzx.java
04/14/2003	05:59a	2,385	xzyv.java
04/14/2003	05:59a	1,190	xzyw.java
04/14/2003	05:59a	397	xzyx.java
04/14/2003	05:59a	1,580	xzyy.java
04/14/2003	05:59a	1,916	xzy.java
04/14/2003	05:59a	2,949	xzzv.java
04/14/2003	05:59a	17,343	xzzw.java
04/14/2003	05:59a	419	xzzx.java
04/14/2003	05:59a	12,135	xzzy.java
04/14/2003	05:59a	1,521	y.java
04/14/2003	05:59a	1,077	yv.java

04/14/2003	05:59a	56,689	yvvv.java
04/14/2003	05:59a	598	yvvw.java
04/14/2003	05:59a	301	yvvx.java
04/14/2003	05:59a	308	yvvy.java
04/14/2003	05:59a	464	yvv.java
04/14/2003	05:59a	536	yvwv.java
04/14/2003	05:59a	296	yvww.java
04/14/2003	05:59a	296	yvwx.java
04/14/2003	05:59a	614	yvwy.java
04/14/2003	05:59a	901	yvw.java
04/14/2003	05:59a	262	yvxv.java
04/14/2003	05:59a	6,608	yvxw.java
04/14/2003	05:59a	400	yvxy.java
04/14/2003	05:59a	2,638	yvx.java
04/14/2003	05:59a	234	yvyv.java
04/14/2003	05:59a	3,301	yvyw.java
04/14/2003	05:59a	6,718	yvyx.java
04/14/2003	05:59a	384	yvyv.java
04/14/2003	05:59a	7,618	yvy.java
04/14/2003	05:59a	670	yvzv.java
04/14/2003	05:59a	529	yvzw.java
04/14/2003	05:59a	4,969	yvzx.java
04/14/2003	05:59a	383	yvzy.java
04/14/2003	05:59a	468	yw.java
04/14/2003	05:59a	2,774	ywvv.java
04/14/2003	05:59a	2,468	ywvw.java
04/14/2003	05:59a	580	ywvx.java
04/14/2003	05:59a	674	ywvy.java
04/14/2003	05:59a	267	ywv.java
04/14/2003	05:59a	296	ywvv.java
04/14/2003	05:59a	9,978	ywww.java
04/14/2003	05:59a	916	ywwx.java
04/14/2003	05:59a	5,878	ywwy.java
04/14/2003	05:59a	385	yww.java
04/14/2003	05:59a	414	ywxv.java
04/14/2003	05:59a	797	ywxw.java
04/14/2003	05:59a	1,031	ywxx.java
04/14/2003	05:59a	1,390	ywxy.java
04/14/2003	05:59a	3,188	ywx.java
04/14/2003	05:59a	13,329	ywyv.java
04/14/2003	05:59a	885	ywyw.java
04/14/2003	05:59a	475	ywyx.java
04/14/2003	05:59a	848	ywyy.java
04/14/2003	05:59a	847	ywy.java
04/14/2003	05:59a	461	ywzv.java
04/14/2003	05:59a	3,589	ywzw.java
04/14/2003	05:59a	39,889	ywzx.java
04/14/2003	05:59a	1,161	ywzy.java
04/14/2003	05:59a	386	yx.java
04/14/2003	05:59a	757	yxvv.java
04/14/2003	05:59a	10,146	yxvw.java
04/14/2003	05:59a	386	yxvx.java
04/14/2003	05:59a	4,370	yxvy.java
04/14/2003	05:59a	4,321	yxv.java
04/14/2003	05:59a	2,185	yxwv.java
04/14/2003	05:59a	451	yxww.java
04/14/2003	05:59a	19,770	yxwx.java

04/14/2003	05:59a	1,935	yxwy.java
04/14/2003	05:59a	366	yxw.java
04/14/2003	05:59a	273	yx xv.java
04/14/2003	05:59a	479	yx xw.java
04/14/2003	05:59a	550	yx xy.java
04/14/2003	05:59a	1,092	yx x.java
04/14/2003	05:59a	1,066	yx yv.java
04/14/2003	05:59a	243	yx yw.java
04/14/2003	05:59a	1,083	yx yx.java
04/14/2003	05:59a	627	yx yy.java
04/14/2003	05:59a	2,024	yx y.java
04/14/2003	05:59a	529	yx zv.java
04/14/2003	05:59a	377	yx zw.java
04/14/2003	05:59a	783	yx zx.java
04/14/2003	05:59a	623	yx zy.java
04/14/2003	05:59a	379	yy.java
04/14/2003	05:59a	3,432	yy vv.java
04/14/2003	05:59a	766	yy vw.java
04/14/2003	05:59a	6,857	yy vx.java
04/14/2003	05:59a	6,133	yy vy.java
04/14/2003	05:59a	1,683	yy v.java
04/14/2003	05:59a	1,732	yy wv.java
04/14/2003	05:59a	2,081	yy ww.java
04/14/2003	05:59a	468	yy wx.java
04/14/2003	05:59a	5,652	yy wy.java
04/14/2003	05:59a	365	yy w.java
04/14/2003	05:59a	473	yy xv.java
04/14/2003	05:59a	1,652	yy xw.java
04/14/2003	05:59a	2,103	yy xx.java
04/14/2003	05:59a	3,030	yy xy.java
04/14/2003	05:59a	1,077	yy x.java
04/14/2003	05:59a	1,853	yy yv.java
04/14/2003	05:59a	453	yy yw.java
04/14/2003	05:59a	357	yy yx.java
04/14/2003	05:59a	433	yy yy.java
04/14/2003	05:59a	2,434	yy y.java
04/14/2003	05:59a	11,052	yy zv.java
04/14/2003	05:59a	1,073	yy zw.java
04/14/2003	05:59a	5,040	yy zx.java
04/14/2003	05:59a	1,999	yy zy.java
04/14/2003	05:59a	409	yz vv.java
04/14/2003	05:59a	392	yz vw.java
04/14/2003	05:59a	452	yz vx.java
04/14/2003	05:59a	282	yz vy.java
04/14/2003	05:59a	5,272	yz v.java
04/14/2003	05:59a	8,694	yz wv.java
04/14/2003	05:59a	1,545	yz ww.java
04/14/2003	05:59a	3,165	yz wx.java
04/14/2003	05:59a	1,120	yz wy.java
04/14/2003	05:59a	293	yz w.java
04/14/2003	05:59a	777	yz xv.java
04/14/2003	05:59a	296	yz xw.java
04/14/2003	05:59a	8,454	yz xx.java
04/14/2003	05:59a	301	yz xy.java
04/14/2003	05:59a	4,993	yz x.java
04/14/2003	05:59a	469	yz yv.java
04/14/2003	05:59a	387	yz yw.java

04/14/2003	05:59a	394	zyyx.java
04/14/2003	05:59a	602	zyyy.java
04/14/2003	05:59a	573	zyy.java
04/14/2003	05:59a	468	zzzv.java
04/14/2003	05:59a	1,426	zzzw.java
04/14/2003	05:59a	453	zzzx.java
04/14/2003	05:59a	3,093	zzzy.java
04/14/2003	05:59a	244	z.java
04/14/2003	05:59a	217	zv.java
04/14/2003	05:59a	469	zvzv.java
04/14/2003	05:59a	682	zvzw.java
04/14/2003	05:59a	388	zvzx.java
04/14/2003	05:59a	341	zvzy.java
04/14/2003	05:59a	1,056	zvv.java
04/14/2003	05:59a	574	zvzv.java
04/14/2003	05:59a	390	zvzw.java
04/14/2003	05:59a	357	zvzx.java
04/14/2003	05:59a	3,529	zvzy.java
04/14/2003	05:59a	1,430	zvw.java
04/14/2003	05:59a	392	zvzv.java
04/14/2003	05:59a	1,248	zvzw.java
04/14/2003	05:59a	16,653	zvzx.java
04/14/2003	05:59a	789	zvzy.java
04/14/2003	05:59a	469	zvyv.java
04/14/2003	05:59a	1,425	zvyw.java
04/14/2003	05:59a	262	zvyx.java
04/14/2003	05:59a	262	zyyy.java
04/14/2003	05:59a	389	zvy.java
04/14/2003	05:59a	451	zvzv.java
04/14/2003	05:59a	467	zvzw.java
04/14/2003	05:59a	2,137	zvzx.java
04/14/2003	05:59a	8,481	zvzy.java
04/14/2003	05:59a	5,379	zw.java
04/14/2003	05:59a	442	zwvv.java
04/14/2003	05:59a	389	zwvw.java
04/14/2003	05:59a	457	zwvx.java
04/14/2003	05:59a	473	zwvy.java
04/14/2003	05:59a	380	zww.java
04/14/2003	05:59a	3,390	zwww.java
04/14/2003	05:59a	1,119	zwww.java
04/14/2003	05:59a	5,763	zwwx.java
04/14/2003	05:59a	384	zwwy.java
04/14/2003	05:59a	1,360	zww.java
04/14/2003	05:59a	301	zwxv.java
04/14/2003	05:59a	326	zwxw.java
04/14/2003	05:59a	1,412	zwx.x.java
04/14/2003	05:59a	2,661	zwx.y.java
04/14/2003	05:59a	995	zwx.java
04/14/2003	05:59a	674	zwyv.java
04/14/2003	05:59a	394	zwyw.java
04/14/2003	05:59a	627	zwyx.java
04/14/2003	05:59a	24,456	zwy.y.java
04/14/2003	05:59a	533	zwy.java
04/14/2003	05:59a	488	zwzv.java
04/14/2003	05:59a	387	zwzw.java
04/14/2003	05:59a	1,115	zwzx.java
04/14/2003	05:59a	1,470	zwzy.java

04/14/2003	05:59a	422	zx.java
04/14/2003	05:59a	500	zxvv.java
04/14/2003	05:59a	401	zxvw.java
04/14/2003	05:59a	2,898	zxvx.java
04/14/2003	05:59a	467	zxvy.java
04/14/2003	05:59a	350	zxv.java
04/14/2003	05:59a	815	zxwv.java
04/14/2003	05:59a	1,332	zxww.java
04/14/2003	05:59a	299	zxwx.java
04/14/2003	05:59a	239	zxwy.java
04/14/2003	05:59a	394	zxw.java
04/14/2003	05:59a	1,881	xxxv.java
04/14/2003	05:59a	312	xxxw.java
04/14/2003	05:59a	376	xxxv.java
04/14/2003	05:59a	5,969	zxx.java
04/14/2003	05:59a	919	zxyv.java
04/14/2003	05:59a	296	zxyw.java
04/14/2003	05:59a	7,539	zxyx.java
04/14/2003	05:59a	649	zxyy.java
04/14/2003	05:59a	1,318	zxy.java
04/14/2003	05:59a	999	zxzv.java
04/14/2003	05:59a	5,013	zxzw.java
04/14/2003	05:59a	551	zxzx.java
04/14/2003	05:59a	1,860	zxzy.java
04/14/2003	05:59a	290	zy.java
04/14/2003	05:59a	2,739	zyvv.java
04/14/2003	05:59a	762	zyvw.java
04/14/2003	05:59a	2,995	zyvx.java
04/14/2003	05:59a	5,224	zyvy.java
04/14/2003	05:59a	1,065	zyv.java
04/14/2003	05:59a	830	zywv.java
04/14/2003	05:59a	1,031	zyww.java
04/14/2003	05:59a	11,192	zywx.java
04/14/2003	05:59a	6,218	zywy.java
04/14/2003	05:59a	432	zyw.java
04/14/2003	05:59a	611	zyxv.java
04/14/2003	05:59a	608	zyxw.java
04/14/2003	05:59a	660	zyxx.java
04/14/2003	05:59a	396	zyxy.java
04/14/2003	05:59a	3,180	zyx.java
04/14/2003	05:59a	773	zyyv.java
04/14/2003	05:59a	451	zyyw.java
04/14/2003	05:59a	3,069	zyyx.java
04/14/2003	05:59a	461	zyyy.java
04/14/2003	05:59a	232	zyy.java
04/14/2003	05:59a	296	zyzv.java
04/14/2003	05:59a	579	zyzw.java
04/14/2003	05:59a	221	zyzx.java
04/14/2003	05:59a	266	zyzy.java
04/14/2003	05:59a	990	zzvv.java
04/14/2003	05:59a	696	zzvw.java
04/14/2003	05:59a	227	zzvx.java
04/14/2003	05:59a	301	zzvy.java
04/14/2003	05:59a	502	zzv.java
04/14/2003	05:59a	308	zzwv.java
04/14/2003	05:59a	1,436	zzww.java
04/14/2003	05:59a	221	zzwx.java

04/14/2003	05:59a	400	zzwy.java
04/14/2003	05:59a	436	zzw.java
04/14/2003	05:59a	301	zzxv.java
04/14/2003	05:59a	447	zzxw.java
04/14/2003	05:59a	2,829	zzxx.java
04/14/2003	05:59a	1,152	zzxy.java
04/14/2003	05:59a	828	zzx.java
04/14/2003	05:59a	1,303	zzyv.java
04/14/2003	05:59a	1,545	zzyw.java
04/14/2003	05:59a	452	zzyx.java
04/14/2003	05:59a	1,533	zzyy.java
04/14/2003	05:59a	406	zzy.java
04/14/2003	05:59a	801	zzzv.java
04/14/2003	05:59a	448	zzzw.java
04/14/2003	05:59a	4,683	zzzx.java
04/14/2003	05:59a	3,601	zzzy.java
	613 File(s)	1,714,117	bytes

Total Files Listed:

1088 File(s)

27 Dir(s)

4,979,436 bytes

0 bytes free

What is claimed:

1. A patient monitoring system comprising:

a patient monitoring device which includes a physiological transducer for monitoring physiological characteristic data of a patient and a wireless communication system for transmitting said physiological characteristic data to a remote hub over a first wireless communication link;

the remote hub which includes a second communication system for receiving said physiological characteristic data over said first wireless communication link from

said patient monitoring device and transmitting said physiological characteristic data to a remote server over a second communication link; and

the remote server, including a third communication system for receiving said physiological characteristic data over said second communication link, said remote server configured as a portal to selectively enable third party access to said physiological characteristic data.

* * * * *

专利名称(译)	用于远程监控患者状况的方法和设备		
公开(公告)号	US20040102683A1	公开(公告)日	2004-05-27
申请号	US10/414326	申请日	2003-04-15
[标]申请(专利权)人(译)	KHANUJA SUKHWANT SINGH GARG SANDEEP SINGH欧文PREET		
申请(专利权)人(译)	KHANUJA SUKHWANT SINGH GARG SANDEEP SINGH欧文PREET		
当前申请(专利权)人(译)	KHANUJA SUKHWANT SINGH GARG SANDEEP SINGH欧文PREET		
[标]发明人	KHANUJA SUKHWANT SINGH GARG SANDEEP SINGH IRWIN PREET		
发明人	KHANUJA, SUKHWANT SINGH GARG, SANDEEP SINGH, IRWIN PREET		
IPC分类号	A61B5/00 A61B5/0205 A61B5/021 A61B5/087 A61B5/117 G06F19/00 H04M11/00		
CPC分类号	A61B5/0002 H04M11/007 A61B5/021 A61B5/087 A61B5/117 A61B5/1455 A61B5/411 G06F19/3406 G06F19/3418 G06F19/3425 G06F19/3487 G06F19/366 G06Q40/08 G06Q50/22 G06Q50/24 H04L67/025 H04L67/28 Y10S128/92 Y10S128/903 Y10S128/905 Y10S128/904 H04L67/2838 A61B5/02055 G16H10/40 G16H15/00 G16H40/63 G16H80/00 H04Q9/00 H04Q2209/40 H04Q2209/823		
优先权	60/372894 2002-04-16 US		
其他公开文献	US7448996		
外部链接	Espacenet USPTO		

摘要(译)

患者监测系统技术领域本发明涉及一种患者监测系统，其相对于已知系统提供增强的功能能力，并且在患者佩戴的患者监测装置和本地集线器之间提供无线通信链路。根据本发明的患者监测系统适于监测各种患者的生理特征，例如血压，脉搏率，血糖，体重，脉搏血氧饱和度等。来自患者监测设备的数据被无线传输到本地集线器，本地集线器又被配置为自动地将数据传输到远程服务器，例如，通过公共或专用通信网络。在本发明的一个实施例中，服务器被配置为门户网站，以选择性地允许指定的第三方（例如医生，临床医生，亲属和患者自己）访问这样的患者生理数据。根据本发明的另一个重要方面，该系统提供相对于已知系统的增强功能，并允许选择性地生成生理数据的趋势。另外，当特定生理特征超过预定阈值时，任何第三方都可以设置阈值以通过包括电子邮件和/或寻呼机在内的各种通信方法自动通知患者或其他第三方。该系统还配置为向患者提供提醒读数。

