

on the various qualification information that already exist in the Digital Survey System, the distribution of various types of survey system devices, the impact of adding several restrictive qualifications on the amount of time needed to obtain a given sample size, and other practical information. A Programming Specialist 3520 may be used to write any new objects or routines
5 needed to build this survey. Ultimately the Programming Specialist 3520 reduces the survey components discussed below to computer logic suitable to the computer system to be used for the Digital Survey System and makes the required associations so that the various components work in unison to deliver the intended presentation to qualified respondents and deliver the various credits and survey responses as set forth below.

10 This particular survey has several objectives, one is to gather information about current attitudes, another objective is to increase the desire for international travelers to consider a visit to Turkey as a vacation destination. After completing the qualification stage, the survey objectively gathers information to measure the respondent's knowledge and predisposition to travel to Turkey. After
15 completing the information gathering stage, the survey then walks the respondent through a number of questions that are not designed to collect information but are designed as an opportunity to respond to earlier answers with rebuttal information to misgivings or commonly-held misperceptions. The survey then walks the respondent through several options for a visit to Turkey that would mesh with respondent's self report of favorite activities. The final stage of the survey repeats questions about attitudes and fears to see if the survey and its tailored pitch has
20 altered the respondent's views about scheduling a vacation in Turkey in the coming year.

In order to provide additional detail, Step 2000 creation of the Survey 3000 is broken down into several sub-steps. In actual practice the various portions of the survey are created as a unified whole since each piece effects the others.

25 Step 2005 Create the Survey Presentation Scheme 3010. Decide what questions need to be answered within the survey and what questions should be skipped or added depending on information known about the respondent. Make initial decisions about the types of audio or video files that may need to be displayed with the survey.

30 Step 2010 Create Survey Qualification Scheme 3020. The Qualification Scheme 3020 includes input from the Marketer 3500, Digital Survey System Survey Support Team 3510, and if needed the Programming Specialist 3520. The Survey Qualification Scheme 3020 is typically created in parallel with creating The Survey Presentation Scheme 3010 as there are many interactions between the survey and the qualification scheme. The Survey Presentation Scheme 3010 will typically use information from the qualification phase to customize certain portions of the presentation of the survey to a particular respondent. The Survey Qualifications Scheme 3020
35 may include Respondent Qualifications 3022, Locale Qualifications 3024, and Time, Place, and Manner Qualifications 3026. The Time, Place, and Manner Qualifications 3026 may include

restrictions on the type of survey device used or the ambient conditions. The Survey Team 3535 comprising the Marketer 3500 and Digital Survey System Survey Support Team 3510 can decide which qualifications can be satisfied by prior responses saved in the respondent profile, or session record. The Survey Team 3535 can decide which qualifications can be satisfied by implication rather than by explicit question. The Survey Team 3535 decides what material from the qualification information must be passed with the set of responses sent to the marketer. In some cases the qualification information is not useful for survey results since there may be several parallel paths of qualification to take the survey and thus, not all surveys will have answers to some qualification parameters. Ultimately the Programming Specialist 3520 reduces the efforts of the group to computer logic.

Step 2020 Create Survey Verification Scheme. The Survey Verification Scheme 3030 contains the decisions of the Survey Team 3535 on which parameters from the qualifications or the response need to be verified before issuance of the credit and which need to be verified before inclusion of the results in the tabulated results. During this step the team decides whether the verification can be done within the survey device, local to the queuing location, or at the central survey processing center. There is a trade-off in minimizing costs in terms of inconvenience to the respondent (need to transmit or dock and then wait for verification), and the need to limit the survey to locations that can load the verification logic onto the survey device or make it available to all survey devices. In some cases a partial check of parameters may be sufficient to limit credits to those who made a serious attempt to answer the survey. A more detailed check may require a number of steps that would take too much time for a instant-credit system. Thus the possibility exists that a survey response will qualify for a credit and not qualify for inclusion in the final survey results.

Step 2030 Create the Demographic Distribution Scheme. The Demographic Distribution Scheme is the implementation of the desire by the Marketer 3500 to force a distribution within the qualified groups. For example does the marketer want to have a 50/50 split or some other fixed distribution between male and female respondents. While both genders are qualified to take the survey, the use of gender to force a distribution makes the one survey into two surveys with gender as a qualification for each survey. In this instance the Marketer for the Travel Turkey Survey does not want to force any specific distribution of qualified respondents.

Step 2040 Create the Survey Incentive Scheme 3050. The marketer 3500 with input from the Digital Survey System survey support team 3510 selects an appropriate credit or premium with the survey. The goal is to choose a credit that is likely to get the number of desired responses within the desired period without spending more of the marketer's budget than necessary. The team will discuss the need to make regional allowance in the amount of the credit so that the credit is roughly as attractive to a respondent in New York City as to one in a lower cost area. The team can discuss whether the credit should be for an item (like an appetizer within a price

range) or for a dollar amount that can be as a credit against any retail item at the queuing location. The team can discuss whether the credit should be one that can be banked with the respondent profile so that the respondent can use the credit at another time, or another time and location.

5 Step 2050 Create the Survey Control Scheme 3060. The Survey Team 3535 decides the start and stop dates for the survey and whether any other criteria is going to be used to automatically terminate the survey (such as obtaining 1000 responses, an email from the Marketer 3500 requesting communication to all relevant queuing locations to disable any copy of the survey that has not been started, or a drop of a stock market index by more than 10%). In this case the Marketer 3500 desires a minimum of 1000 survey response sets from qualified respondents. The
10 Market 3500 elects to risk having to pay for a few extra responses rather than delay the completion of the survey process with a tapering scheme that limits the distribution of the survey to only 1000 survey copies. The copies are moved one at a time to a queuing location so that at any one queuing location there is only one copy of the survey at any given time (even if the copy is sitting idle in a survey device when a potentially qualified respondent elects to fill out another
15 survey.)

Step 2060 Complete Reduction of Survey Concepts to Integrated Computer Logic within this step send the survey off for completion by the programming professional. Finalize the links between the survey components and the related elements to a survey to produce the Survey 3000. Add an estimated time to complete to each survey to use as a criteria for respondents to pick
20 surveys that can be completed with some margin of spare time in the expected queue time.

Step 2065 Receive Pre-Payment for Accumulated Responses For Marketers using the Digital Survey System for the first time, the Digital Survey System management requires prepayment for the survey set to be accumulated and delivered to the Marketer 3500. As described below, the Digital Survey System will provide transfer payments on a nightly basis to the various queuing
25 locations. Thus the Digital Survey System management wishes to minimize exposure to loss by requiring an estimated payment up front and then making any necessary credit or invoice at the delivery of the accumulated responses to the Marketer. Depending on the incentive scheme use, the actual price may vary from the estimated price. For example there may be variations in price of the incentive between locations or regions.

30 Step 2070 Load and Enable Survey. The Survey 3000 is placed in the Digital Survey System Central Computer System 3600 and added to the Digital Survey System Central Computer System Pending Survey List 3610. Create the necessary files or objects to contain the results of the survey and handle the administration of credits to respondents, payments to queuing locations, movement of a completed response set from a respondent or the accumulation and transmission
35 of a cluster of response sets from a number of respondents.

After the survey and ancillary programming preparations are completed and the current date falls

within the start/stop range for the survey, the survey is made available to queuing locations that meet all qualifications for queuing location such as locale qualifications, and the ability to meet various Time, Place, and Manner qualifications including such things as having the appropriate survey devices or input/output devices required for the survey, having jazz performances on a regular basis, serving breakfast, or selling disposable diapers. The distribution of surveys to queuing locations could be done by pushing the survey to locations that qualify or by allowing queuing locations or survey devices with the capacity to communicate to the central survey location to pull surveys down. A particular doctor's office may choose not to pull down a survey for a herbal remedy that the doctor does not recommend. In the alternative, the doctor's office may be set to pull down all surveys for which it is provisionally qualified and then the queuing location management may elect to suspend or delete a survey from the local pending survey list.

B) Actions at the Queuing Location

Step 2400 Prepare Queuing Location to Participate in Digital Survey System. A Queuing Location 3400 agrees to host a set of survey devices. This step is unrelated to the creation of Survey 3000 except for the obvious limitation that the queuing location must complete its work while this particular Survey 3000 is on the Digital Survey System Central Computer System Pending Survey List 3610 in order for the survey to be available at the Queuing Location 3400. In this case the Queuing Location 3400 is a restaurant. The management of the queuing location 3400 and the Survey System Sales Specialist 3540 work out the details on: the number of and types of survey devices; the communication needs between the survey devices and other components; the various options for providing communication links (such as use of wireless links, a network that allows connecting the survey device at any one of a large number of ports distributed throughout the waiting area, the use of docking stations at the beginning and end of a session); the use of ambient condition instruments; the use of biometric instruments; the process for downloading and storing respondent profiles; the process for uploading information; and the process for distribution of credits to respondents with respondent profiles and to respondents that want to remain anonymous. There are many sub-steps to preparing a queuing location to participate with the Digital Survey System but detailing these would detract focus from the process of creating surveys and collecting survey results. Many of the sub-steps will be referenced in the subsequent steps of this discussion. The various sub-steps for getting a queuing location prepared to participate in Digital Survey System are well within the general skill set for system designers and installers when provided with this patent and information accessible to those of skill in the art.

Step 2410 Prepare Locale Profile. The Survey System Sales Specialist 3540 works with the management of the Queuing Location 3400 to fill out the Locale Profile 3410 information which includes among other information, the range of Time, Place, and Manner conditions that can be satisfied within this location.

Step 2420 Prepare Time, Place, and Manner Status File. The links between any on-site instruments and the Time, Place, and Manner Status File 3490 should have been completed by the hardware installation (not shown on either Figures 2 or 3). The process for manual updates to the one or more attributes in the Time, Place, and Manner status file is explained. The queuing location is taught that a timely update of the performance field from bridal fashion show to live jazz will result in more survey choices for the jazz patrons and thus more completed survey credits flowing into the queuing location.

Step 2430 Connect to Receive Sub-Set of Pending Surveys. The Queuing Location Digital Survey System Computer 3420 is connected by any conventional communication network to the Digital Survey System Central Computer System 3600. (naturally, element 3600 could be one of a number of regional or industry specialized computers rather than one central computer). During this connection, the Digital Survey System Central Computer System 3600 first checks if there is an update for the Locale Profile. In this case the Locale Profile 3410 is different (in fact is it brand new) as compared to the Locale Profile 3411 (empty) stored for this queuing location on the Digital Survey System Central Computer System 3600. After comparing the information in the Locale Profile with the qualifications for the various available surveys, the Digital Survey System Central Computer System 3600 provides an updated (in this case new) Queuing Location Set of Surveys 3431 for which the queuing location is potentially qualified to provide to respondents. Typically the Queuing Location Set of Surveys 3431 provided to any one queuing location is going to be a small sub-set of the total Set of Surveys 3430 on the Digital Survey System Central Computer System 3600. The word "potentially" comes from the provision of the range of possible Time, Place, and Manner circumstances contained in the Locale Profile 3410 will qualify the queuing location to receive surveys that will not become available to respondents until specific Time, Place, and Manner criteria within that range are satisfied.

Step 2440 Review, Delete, or Limit the Surveys in Set. Management of the new Queuing Location 3400 view the list of available surveys in the downloaded Queuing Location Set of Surveys 3431 including the marketers offer for credits/handling fees associated with the surveys. To check to see if the survey is something that the queuing location wishes to offer, the queuing location may request to see the survey questions for a particular survey offered by a marketer that is not a competitor (in this case a restaurant or a night club). The queuing location thus elects to suppress a survey by a new magazine because the queuing location owner believes that many of his customers will find some of the questions offensive. Likewise the restaurant elects to suppress surveys benefiting a national chain of restaurants that has a franchise across the street from this Queuing Location 3400.

Step 2450. Add Local Time, Place, and Manner Restrictions. The management of queuing location 3400 adds a Local Time, Place, and Manner Restriction Set 3440 onto the Turkey Travel Survey 3000 and each of the surveys in the Queuing Location Set of Surveys 3431 using a tool

built into the Queuing Location Digital Survey System Computer 3420. The default local Time, Place, and Manner restriction for this queuing location is to limit the availability of surveys so that surveys are only available to respondents when the estimated time to complete the survey is less than 60% of the current estimated time to get a table.

5 C) The Respondent Arrives.

Step 2500 Respondent Offered Survey Device. The Respondent 3550 arrives at the Queuing Location 3400. The Maitre D' 3560 offers a Survey Device 3450 to the Respondent 3550. Respondent 3550 declines offer to create a new respondent profile since Respondent 3550 already has a Respondent Profile 3070 from previous experiences with the Digital Survey System.

10 Step 2510 Respondent Provides Identifier. A Survey Device 3450 is waiting docked in Biometric Docking Station 3475. The Respondent 3550 signs his name on the Biometric Signature Instrument 3460 connected to the Queuing Location Digital Survey System Computer 3420. The Biometric Signature Instrument 3460 collects and passes the Signature Profile 3465 to the Queuing Location Digital Survey System Computer 3420. The Signature Profile 3465 is used as
15 a way to identify the Respondent 3550. Since an earlier encounter between this Respondent 3550 and another queuing location with Biometric Signature Instrument 3460 has added this Biometric to his Respondent Profile 3070, the respondent does not need to provide his Respondent Profile Identification Number 3075 to the Survey Device 3450 to pass through the Biometric Docking Station 3475 to the Queuing Location Digital Survey System Computer 3420.

20 Step 2520 Obtain a Copy of the Respondent Profile. This Queuing Location has opted not to obtain respondent profiles in anticipation of a respondent coming to the queuing location. Instead, this Queuing Location but will maintain a set of respondent profiles for all respondents that have used the Digital Survey System at this queuing location within the last six months. Although the Respondent 3550 is a steady customer of this Queuing Location 3400, this is the
25 first time that he has used the Digital Survey System at this location. The Queuing Location Digital Survey System Computer 3420 receives the Signature Profile 3465 and obtains a Copy 3471 of the Respondent Profile 3470 from the Digital Survey System Central Computer System 3600.

Step 2530 Load Respondent Profile into Survey Device. The Queuing Location Digital Survey
30 System Computer 3420 receives a Copy 3471 of the Respondent Profile 3070 and loads a second Copy 3472 onto the Survey Device 3450.

Step 2540 Respondent Provides his Point-of-Sale Identification Number. The Respondent 3550 is a long time customer with the Queuing Location 3400. In response to an invitation posted on the Survey Device 3450, respondent enters his Point-of-Sale Identification Number 3481 to the
35 Survey Device 3450. The Survey Device 3450 passes the Point-of-Sale Identification Number 3481 through the Biometric Docking Station 3475 to the Queuing Location Digital

Survey System Computer 3420 which is in communication with the Point-of-Sale Computer System 3480.

5 Step 2550 The Respondent's Point-of-Sale Profile is Loaded. The Point-of-Sale Computer System 3480 returns a copy 3486 of the Respondent's Point-of-Sale Profile 3485 to the Survey Device 3450.

Step 2560 The Point-of-Sale Profile is Linked to the Respondent Profile. The Survey Device 3450 now has access to additional information about the respondent from the Respondent's Point-of-Sale Profile 3486 such as the information that this respondent frequently purchases fine cigars and cognac while visiting this Queuing Location 3400.

10 Depending on the surveys selected, the information about cigar purchases may be used to qualify him for surveys addressing cigar preferences or a mouth cancer survey. The information from the Point-of-Sale Computer System 3480 will be used to trigger additional questions and a verification routine in a life insurance application if he chooses to fill out a life insurance application as his choice for a survey. It is a good idea that critical information such as whether
15 the applicant is a cigar smoker be checked with direct questions. In this case, if asked, the respondent would be able to key in that the cigars were purchased for a frequent dinner companion. The frequent dinner companion is a major client of the respondent's company, and one likely to request a cigar every time the respondent takes the client out to dinner.

20 Step 2570 Update Time, Place, and Manner Status File. The Maitre D' 3560 updates the Time, Place, and Manner Status File 3490 to indicate that the current wait for a table is now 45 minutes instead of 30 minutes. The Time, Place, and Manner status file update activity is ongoing and does not trigger the following step 2575. However, it is good practice to update the Time, Place, and Manner status file since the Time, Place, and Manner status file effects the choice of surveys available to the respondent (and possibly the survey process within a selected survey).

25 Step 2575 Load other Qualifying Information. Provide a Copy 3491 of the latest Time, Place, and Manner Status File 3490 to the Survey Device 3450. Provide a Copy 3412 of Locale Profile 3410 to the Survey Device 3450.

30 Step 2580 Download Survey-Device Survey Sub-Set. Make copies of all surveys that match the known information about Respondent 3550 from the Queuing Location Copy of the Respondent Profile 3471 and the Point-of-Sale Profile 3485 (or copy of Queuing Location copy of Point-of-Sale Profile 3485 if one is passed from the Point-of-Sale Computer System 3480 rather than made accessible). Filter out surveys disqualified by the Time, Place, and Manner Status File 3490 (for example -- eliminate from consideration those surveys with an estimated time to complete in excess of 60% of the 45 current wait for a table). Filter out surveys based on the Survey Device
35 Attribute File 3455 for this particular Survey Device 3450. The Survey Device Attribute File 3455 confirms that the high resolution graphics components requested by the Marketer 3500

are functioning. This current information for this particular Survey Device 3450 is a final check for device capabilities. The Survey 3000 would not have been provided to this Queuing Location 3400 if the Locale Profile 3410 did not indicate the availability of at least some survey devices that could offer the requisite graphics support.

- 5 The result of all these filters is a Survey Device Survey Sub-Set 3495 of surveys for the Respondent 3550 to consider for selection.

D). Undock and Complete Survey

Step 2600 Undock Survey Device. Respondent 3550 undocks the Survey Device 3450 from Biometric Docking Station 3475

- 10 Step 2610 Select a Survey. The Respondent 3550 carries the Survey Device 3450 and moves to a seat in the lounge to wait for his table. Once seated the respondent looks at the list of surveys in the Survey-Device Survey Sub-Set 3495. The respondent selects a Survey 3000 that identifies the topic as international travel and lists the associated credit for this twenty minute survey as a free dessert or a travel book that will be shipped to respondent's home or office.

- 15 Since the respondent knows he is able to select a dessert, if he does not want the book, he selects Survey 3000.

- Step 2620 Complete the Qualification Phase. After Respondent 3550 indicates to the Survey Device 3450 the desire to try Survey 3000, Survey 3000 begins to present questions to satisfy the remaining qualifications for Survey 3000. Respondent 3550 answers additional specialized
20 qualifying questions about the number of times that the respondent has traveled to Europe, Asia, or Africa. After the respondent indicates that he has traveled to these continents several times for vacations and is undecided about his next destination, Respondent 3550 had successfully completed the remaining qualifications to receive this survey 3000.

- An answer of "yes" from the respondent to the qualifying question "have you already booked
25 arrangements for vacation travel to any country outside of the United States?" would have been a disqualification for that path to qualify since the Marketer 3500 wanted to focus on people that do not have a pending vacation booked overseas.

- Since there was only one path of qualifications for this survey, the Survey 3000 would have politely terminated and returned the Respondent 3550 to the Select a Survey Step 2610. During
30 this hypothetical second go-round with the Survey Selection Step 2610, this Respondent 3550 not be offered as an option of selection this Survey 3000 as his earlier answer would have disqualified him from taking this survey.

Step 2630 Complete the Survey. Respondent 3550 is presented with the core presentation of Survey 3000 beginning with a segment to record the respondent's views about visiting Turkey.

The Respondent 3550 provides his responses to the information gathering section of the survey. Per the intent of the Marketer 3500, it is not readily apparent that the next section of the survey is actually a set of questions created to allow the survey to present the various salient features about a visit to Turkey such as the temperature, favorable exchange rate, proximity to other vacation destinations, and the archaeological sites. In short, the Survey 3000 uses this information from the Respondent 3550 to provide a semi-customized sales presentation. This sales pitch portion of the Survey 3000 uses the survey device's high resolution color screen and graphic accelerators to display brief videos on the Survey Device 3450. The brief videos include views in vivid color from a helicopter of some of the archaeological sites as well as close ups of friendly Turks inviting the respondent to visit. The survey makes a pitch for the respondent to choose to receive a color book with more details about vacations in Turkey. The survey indicates that videos depict scenes shown in the Turkey Travel Guide. The Survey 3000 asks if respondent wishes to have a copy of the Turkey Travel Guide shipped as a gift for completing this survey. The respondent elects to receive the book and is asked to confirm his shipping address. The shipping address is confirmed by the Respondent 3550 rather than input because this shipping address was previously added to his Respondent's Profile 3070 and passed down into the copy of the Respondent Profile that exists on the Survey Device 3450.

Step 2640 Review Short Circuit Data submissions. At the end of the survey presentation, the Survey 3000 presents a List of Short Circuit Information 3620 that will be sent ultimately to the Marketer 3500 with the survey responses entered during this survey. The List of Short Circuit Information 3620 would include: any information that was obtained from the Survey-Device Copy of the Respondent Profile 3472, the Survey-Device Copy of the Point-of-Sale profile 3486, material stored on the survey device from answering other surveys or qualification questions for other surveys during this same visit to the queuing location (not applicable here), and material from Time, Place, and Manner sensors (such as ambient light, ambient noise, etc.). If the respondent is comfortable with the process, the respondent can skip the review. Note that for certain surveys that are applications, a careful review by the respondent can catch errors from heuristics that have not worked. For example, while filling out a life insurance application or some other survey at this particular Queuing Location 3400, this Respondent 3550 may need to correct a false conclusion that he regularly smokes cigars since this information deduced from his paying habits recorded in the Point-of-Sale Computer System. A conclusion that he smokes cigars rather than erroneously conclude that his prior purchase of cigars were for his consumption rather than for handing out to his client.

If the respondent approves of sending all the information on the List of Short Circuit Information 3620, then the Survey Response Set 3630 (which includes this additional information) is now completed. Generally there is little need to review the inputs to explicit questions since these items were just entered. However, if the respondent thinks that an entry

error or a misunderstanding has caused a mistake, most surveys would offer a chance to review the collected information and correct or purge the survey results.

5 Step 2650 Review Update to Respondent Profile. Respondent 3550 is presented with a list of new information to use as Profile Updates 3077 that will go into the Survey-Device copy of Respondent's Profile 3472 and eventually uploaded to the central Respondent Profile 3070. The respondent may delete any item that the respondent does not wish to have stored in his or her profile. Deleting items will increase the likelihood that the respondent will need to provide this information again during some future survey, but that is a choice open to the respondent.

10 Step 2660 Update Survey-Device copy of Respondent Profile. The Survey Device copy 3472 of the respondent profile 3470 is modified to include the material reviewed and approved by respondent.

The survey device copy of the respondent profile 3472 is automatically modified to indicate that the Respondent 3550 has completed this Turkey Travel Survey 3000 so that this particular Turkey Travel Survey is not presented to the respondent at future visits to queuing locations with Digital Survey System survey devices.

E). Offloading Results

20 Step 2700 Dock the Survey Device. After completing the Survey 3000, the Respondent 3550 walks to a Waiting Area Docking Station 3640. Once docked, the Survey Device 3450 removes the Survey 3000. The Survey Device 3450 also offloads the Survey Response Set 3630, and the Profile Update 3077. While in the Waiting Area Docking Station 3640, a Time, Place, and Manner Status File 3490 is downloaded into the Survey Device 3450 to replace the existing Survey-Device Copy Time, Place, and Manner Status File 3491. The offloaded Survey Response Set 3630 is passed with the Survey Device Identification Number 3457, respondent's Point-of-Sale Identification Number 3481, and his Respondent Profile Identification Number 3071.

25 The respondent is offered the option to select an electronic newspaper for downloading into the Survey Device 3450 as an alternative to answering another survey in the remaining time. An Electronic Newspaper 3650 is downloaded into the Survey Device 3450 as a form of entertainment for the respondent while awaiting his table.

30 Step 2710 -- Run Local Verification. The offloaded Survey Response Set 3630 is subjected to the Local Verification Checks 3035 on the Queuing Location Digital Survey System Computer 3420 requested in the Survey Verification Scheme 3030.

35 Some errors such as an incorrect postal code can be caught locally and a correction entered at the Local Digital Survey System Control Console 3660 after a staff member of the queuing location asks the respondent for a correction. Correction of the error will allow the Survey Response Set 3630 to be sent to the Digital Survey System Central Computer System 3600 for subsequent

processing and the issuance of both a credit for the Respondent 3550 and the various handling fees to the queuing location 3400.

5 Step 2720 Transmit Survey Response Set. After the Local Verification Step 2710 is completed and without any uncorrected detected problem, the Survey Response Set 3630 is provided over a communications network to the Digital Survey System Central Computer System 3600.

Step 2725 Transmit Update information for the Respondent Profile. The Profile Updates 3077 are sent to update the Digital Survey System Central Computer System copy of the respondent profile 3070. This update is not needed for the authorization of premium, but it is a good practice to update the profile during the same communication session so that the respondent profile 3070 is updated in a timely manner.

10 Step 2730 Authorization of Premium. Some surveys may use a verification scheme that requires a second level of verification processing upon receipt by the Digital Survey System Central Computer System 3600 for such problems as a respondent that is attempting to collect a series of premiums for answering the same survey at a string of queuing location within walking distance of one another. Note that surveys thought attractive to those who would attempt to collect the same premium by filling out the survey twice would not be available to people without pre-existing respondent profiles in order to prevent people from responding without providing their profile Identification number or creating a new Identification number with fake information at each queuing location. No such verification step was requested for this survey 3000.

15 20 The Digital Survey System Central Computer System 3600 authorizes the premium for the Respondent 3550. Since the respondent requested the Turkey Travel Guide book, an instruction is sent to the Digital Survey System Premium Shipping Department 3080 to ship a premium to the respondent. A Credit Authorization 3682 is sent to the queuing location 3400 that lists the various identification numbers for the Respondent 3550 and lists the Survey Device Identification Number 3457 for this Survey Device 3450 for use by the Waitperson 3570 to in match the survey premium to the survey respondent. The Survey Device Identification Number 3457 is very important for instances where the respondent elects not to provide his Respondent Profile Identification Number 3070 or biometric substitute to the system because he or she prefers not to add information to a centralized data file. Respondents who feel strongly about this may elect not to obtain a respondent profile but will be opting out of the benefits of having a respondent profile.

25 30 Step 2740 Respondent moves to the Table. The Respondent 3550 takes the Survey Device 3450 with him to his table when the table is ready.

Step 2750 Dock the Survey Device. The Management of the Queuing Location 3534 chose to have the Waitperson 3570 handle the administration of Digital Survey System premiums. Other queuing locations are likely to choose one of the following: the cashier, someone who handles incoming calls for reservations, someone working in the coat check room, or some other person.

The implementation of the Digital Survey System at this Queuing Location 3400 calls for the Waitperson 3570 to move the Survey Device 3450 from the respondent's dinner table to a Staff Docking Station 3670 in a staff work area.

5 When docked, the Survey Device 3450 communicates the identifying information for the Respondent 3550 and the Survey Device 3450 through the Staff Docking Station 3670 to the Queuing Location Digital Survey System Computer 3420. The waitperson 3570 provides the Point-of-Sale Transaction Number 3487 for the upcoming meal for the Respondent 3550 through the Survey Device 3450. The Point-of-Sale Transaction Number 3487 is passed through the Staff Docking Station 3670 to the Queuing Location Digital Survey System Computer 3420.

10 Step 2760 Collect Credits for Survey .The Credit Authorization 3682 for completing the survey is transferred from the Queuing Location Digital Survey System Computer 3420 to the queuing location Point-of-Sale Computer System 3480 and then linked to the Point-of-Sale transaction for this meal by Point-of-Sale Transaction number 3487.

F) Providing the Credit

15 A Credit Slip 3681 is printed at a Digital Survey System Printer 3680 to be given to the respondent. In this case, it is a slip noting that a copy of the Turkey Travel Guide book is going to be shipped as a gift for the respondent's participation in the Survey 3000. In other situations, it is a credit slip telling the respondent that a dessert of respondent's choice (or up to a value of X, or a dessert of a particular type) will be provided for at no cost as a gift. The Turkey Travel
20 Survey 3000 requested that the respondent choose between premium choices during the closing portion of the survey. Other surveys less interested in shipping the premium may leave the choice of premium to be decided during the dinner or other experience at the queuing location. If the decision was left until dinner, the respondent could apply the credit to his transaction after making his choice of a dessert or an appetizer if that was the choice presented by the survey.

25 Step 2770 Reset Survey Device. Once the credit slip 3681 has been printed, the Survey Device 3450 is emptied of all surveys, profiles, electronic newspapers, and other files related to the use of this Survey Device 3450 by this Respondent 3550. This particular Survey Device 3450 is put into a rack for movement with other survey devices back to the area where the Maitre D' greets incoming diners.

30 Step 2775 Presentation of Credit Slip. The waitperson 3570 presents the credit slip 3681 to the Respondent 3550.

Acide--An alternative to shipping the book from the Digital Survey System shipping department is to provide inventories of the books to key queuing locations. At this stage the waitperson would place the credit slip in an envelope in the box of books and provides the book to the
35 respondent. This sort of scheme requires extra planning and cooperation from the queuing

locations and the costs of shipping and loss associated with shipping the premium to various locations. A third alternative is to provide a queuing location only copy of the premium so that the respondent can flip through the book while waiting for his dinner and then look forward to receiving the copy to be shipped to his home.

5 Step 2780 Record Exercise of the Credit. Waitperson 3570 inputs a Record of Providing Credit 3685 using a Point-of-Sale Input Device 3489 to both inform the Point-of-Sale Computer System 3480 that the Credit Slip 3681 provided to the Respondent 3550 and to complete the delivery of the premium (since it will be shipped directly to the Respondent 3550). If the respondent had selected a dessert then the credit would not be used until the payment for the
10 dessert.

Step 2790 Request Payment from Queuing Location Digital Survey System Computer. The Point-of-Sale Computer System 3480 reacts to the exercise of the credit by crediting the Waitperson 3570 according to the Survey Incentive Scheme 3050. Many surveys will provide a generous tip for the waitperson in order to promote cooperation with the Digital Survey System
15 by the operations staff of the queuing location. Here in the case of the Turkey Travel Survey, the waitperson is provided with a Tip 3690 equivalent to generous tip for a standard dessert so that the staff does not discourage respondents from choosing the book. A list of tips earned for various surveys is maintained in the Point-of-Sale Computer System where it can be viewed by the waitperson. The tip payment for the Digital Survey System is an added line item to the
20 paycheck for the waitperson. The Point-of-Sale Computer System 3480 sends a payment request 3700 to the Queuing Location Digital Survey System Computer 3420 for reimbursement for the tip 3690 given the Waitperson 3570 and a Service Charge Request 3710 for completion of this survey transaction according to the Survey Incentive Scheme 3050.

Step 2800 Request Payment of the Queuing Location. The Queuing Location Digital Survey
25 System Computer 3420 sends the Payment Request 3700 to the Digital Survey System Central Computer System 3600 which accumulates the payment requests. Note that if the respondent had selected the dessert instead of the book, the cost of the dessert would be transferred for reimbursement as well.

Step 2810. Payment of Queuing Location. Later that night, an electronic payment is made by the
30 Digital Survey System Central Computer System 3600 to move money from the Digital Survey System Bank Account 3730 to a Queuing Location Bank Account 3740 specified by the queuing location in the Locale Profile 3410 to pay for all of the charges levied that day to the Digital Survey System by that queuing location.

G) Distribute Updated Respondent Profile

35 Step 2820 Distribute Updated Respondent Profile. The updated respondent profile 3070 is downloaded by those queuing locations that have opted for such automated downloads. As

indicated in the text associated with Step 2520, queuing locations may choose to keep current copies of many respondent profiles (keeping all would be somewhat extreme), keeping very few, or keeping none at all. The actual plan will depend greatly on the queuing location and the population that the queuing location tries to please. Some queuing locations may request updates for any respondent that has completed a survey at their location within the last six months. Some queuing locations may request updates for every respondent profile with a home or work address within a specified range of postal delivery codes that matches the postal delivery codes from which most of the customers are drawn for that queuing location. Some queuing locations may cater to a narrow spectrum of the population and may request all respondent profiles meeting some other set of criteria. Other locations may opt to avoid the expense of storing the profiles and may simply obtain a copy of the profile for use during the respondent's visit. These queuing locations will obtain the respondent profile on demand and thus will not receive the updated profile until the respondent visits the queuing location.

H) Complete Collection of Survey Responses

- 15 Step 2830 Complete the Set of Response Sets. Unrelated to the distribution of the respondent profile and several days later, the Digital Survey System Central Computer System 3600 receives a Last Response Set 3631 from a respondent to the Turkey Travel Survey 3000. This Last Response Set 3631 is verified and added to the Accumulated Results 3635 for the Turkey Travel Survey 3000. This Last Response Set 3631 makes one thousand accumulated responses.
- 20 Step 2840 Send out Terminate Survey Command. Since the Marketer 3500 requested and authorized 1000 responses, the Digital Survey System Central Computer System 3600 sends out a Terminate Survey Command 3750 to remove the Turkey Travel Survey 3000 from the list of possible surveys. The Terminate Survey Command 3750 is sent to every queuing location that has downloaded the a copy of the Turkey Travel Survey 3000.
- 25 Step 2850 Add Additional Response Set. An Extra Response Set 3632 for the Turkey Travel Survey 3000 arrives at the Digital Survey System Central Computer System 3600. Since many of the survey devices operate by downloading the surveys that a respondent may be qualified to answer from the local Queuing Location Digital Survey System Computer 3420, it is not unusual for some respondents to complete or even start a survey after a Terminate Survey Command 3750 has been sent to the queuing locations. To promote goodwill with the respondent population and with the queuing locations, the respondent and the queuing location are always paid for surveys completed and submitted in good faith.

As described in the text associated with the Survey Control Scheme 3060, the risk of extra survey responses is reduced if the survey distribution is highly controlled as the number of accumulated results approaches the target number of results. A method with a very high degree of control would be to create one thousand copies of the survey at the Digital Survey System Central

Computer System Pending Survey List 3610 but pass the survey without the ability to act as a template for making additional copies. The survey would be downloaded from the Digital Survey System Central Computer System 3600 to the Queuing Location 3400 but would come with a timer. If the timer times out before the survey is started, then the survey becomes unavailable to the queuing location and sends a message back to the Digital Survey System Central Computer System 3600 that it has timed out. The Digital Survey System Central Computer System 3600 would put a new copy of the survey into the Digital Survey System Central Computer System Pending Survey List 3610 with the same survey count number as the survey that timed out. Eventually all 1000 copies would be returned, one survey for each of the survey control number (1 to 1000).

Many other control systems to control the collection of surveys to a set amount should be apparent to one of ordinary skill in the art. However, extra control comes at the cost of extra overhead and a slowing of the availability of surveys to potential respondents.

In the present example, the Marketer 3500 and the Digital Survey System management have contracted that the Marketer 3500 will pay for a small number of extra survey responses.

Step 2860 Send Accumulated Responses to Marketer. The Extra Response Set 3632 is processed at the Digital Survey System Central Computer System 3600. After a set dead-band time without any additional responses coming into the Digital Survey System Central Computer System 3600, the accumulated results 3635 are sent to the Marketer 3500. The Marketer 3500 is presented with a Correction Figure 3760 for any underage or overage with respect to the estimated payment paid by the Marketer 3500 before the surveys were distributed.

ADDITIONAL EXAMPLES

Additional examples of the flexibility of the Distributed Survey System are provided below. To enhance readability, these examples are provided without the step and element format provided in the previous example. The names of the people and the companies are fictitious.

Example 1: To illustrate the abilities of the Digital Survey System, imagine a prospective marketer, Sally Smith of Credit Bank, who wishes to promote the Credit Bank credit card. Sally decides to target restaurants whose entrees range in price from \$10 to \$25 and she is only interested in the dinner crowd at such restaurants. Sally designs an application form (a type of survey) that requires the survey respondent's name, address, telephone, date of birth and permission to issue a card in the survey respondent's name. Sally offers a credit worth \$5 of the survey respondent's meal. In addition, Sally seeks potential candidates for Credit Bank's investment services. Therefore, she offers an additional \$1 rebate to survey respondents who are heads of household over 40 and fill out a request for information on Credit Bank's investment services. The survey format is converted into a marketer agent/survey agent pair that represents

the desired information.

The survey agent is distributed to all locales that meet Credit Bank's specifications. Since Sally Smith of Credit Bank chose to limit access to the dinner crowd the survey agent is disabled before 5pm and after 9pm.

5 Continuing with the example, "Susan", a single mother of one, is having a dinner date. While she is waiting to be seated, she learns of the promotion and decides to participate. She picks up the handheld device and chooses the Credit Bank survey agent. She fills out the Credit Bank application. While fulfilling the Credit Bank application she discloses the fact that she is a head of household. At this point she is offered the opportunity to request the investment services
10 information for an additional \$1 rebate. She decides to also request the investment services information.

Upon successful completion of the survey, Susan is credited with the \$6 rebate against her meal. Susan is pleased to receive an immediate credit that can be applied to her meal. The restaurant is pleased that sales of more expensive appetizers and desserts have risen since the restaurant
15 customers have started receiving immediate credits. The restaurant may also receive a service charge of 30% on the rebate and provide tip to the waitperson for the food covered by the rebate and 5% to the host person that encourages waiting patrons to fill out surveys. Susan is also happy to learn about some of the great financial products that Credit Bank has to offer.

Example 2: Another scenario is where the Digital Survey System is implemented in a chain of
20 apparel stores, JT Jones. JT Jones would like to learn more about the chain's clientele and the clientele's impressions of JT. JT Jones decides to use the Digital Survey System to survey JT clientele. So JT Jones decides to offer a pair of JT Jones designer sunglasses to customers that participate. "Jamal", a fashion conscious professional, appreciates the opportunity to receive a pair of quality sunglasses for a few moments of his time. So Jamal fills out the survey. Before
25 granting the credit, the Digital Survey System performs one or more checks to catch responses not made in good faith. The Digital Survey System checks include checks that the submitted personal information appears to be legitimate and the Jamal is not attempting fill out the survey multiple times to get multiple pairs of sunglasses. After the Digital Survey System checks approve the information provided by Jamal as likely to be legitimate, Jamal is given an
30 authorization code. JT Jones employees use the code to charge the sunglasses against the promotional expense account in the Point-of-Sale Computer System. Within minutes, Jamal is wearing the JT Jones designer sunglasses that coordinate with the new jacket he bought at JT Jones. As an aside this example highlights that the present invention is not limited to people waiting at a restaurant. By queuing location, what is meant is any place where people have time
35 to fill out surveys. Thus a retail store is a queuing location, as that term is used in this description and the claims that follow. For example, many retail clothing stores or departments within stores

cater to a specific gender or age. Thus, when couples shop together, one spouse often has idle time while the other spouse is looking at clothes or trying them on. This phenomenon is so prevalent that some retail stores provide chairs and magazines so the idle spouse does not get bored.

5 Example 3: The Digital Survey System may also be installed in a business like an Oil Changes 'R Us. "Jack" comes into Oil Changes 'R Us and is sent to the waiting area while his vehicle is serviced. To entertain himself, Jack picks up the survey device and browses through some informational content. While using the survey device, Jack types in an identifying piece of information like his phone number. The Digital Survey System then uses that identifying
10 information to do a survey respondent lookup on Jack, through an interface with Oil Changes 'R Us' Point-of-Sale Computer system, the provides information that a 1985 Mustang™ automobile is linked to the phone number Jack provided the survey device. So the survey device presents Jack with an advertisement talking about the maintenance schedule for that model car sponsored by the Joe's Auto Service located next door to Oil Changes 'R Us. Jack browses the
15 advertisement and decides whether he needs to stop by Joe's for maintenance on his automobile.

As of late 1999, the preferred mode for implementing the invention is a wireless network of handheld survey devices such as the PalmPilot™ VII or other Personal Digital Assistant device ("PDA") communicating directly to information servers over the internet. However, an initial implementation will be realized using handheld survey devices such as the PalmPilot™ IIIx
20 which can be synchronized with a local computer. The local computer is then responsible for interacting with the information servers. The local computer serves as the connection point between the handheld survey devices and the information servers that maintain the databases and provide the responses to the marketers. In this scenario, the interactions between the different survey devices that are part of the network happen at the point of synchronization using a cradle
25 similar to the Hotsync cradle used by PalmPilot™ computers. The development of devices, wireless transmission of digital information, and computer languages especially object oriented languages is expected to continue at a rapid pace.

Thus, the capabilities of the various components used to implement the system are expected to make future implementations of the present invention look very different from the current
30 preferred arrangement of hardware and software duties. It is important to focus on the claimed invention not the hardware and specific software used to implement it.

The invention is described in terms of software components that use the available hardware resources to perform their tasks. The components represent behavior and therefore do not necessarily represent discrete software entities.

SAMPLE IMPLEMENTATION SCHEMES

Figures 4 - 15 provide samples of implementation schemes for the Distributed Survey System method described above. Implementation choices will be driven by the current state of wireless communication options, storage costs, transmission speed and reliability across the Internet or other suitable networks, and the development of software tools.

In order to give additional clarification on the scope of the invention, and to provide guidance for implementing the invention in different programming environments, applicant provides the following information.

In FIG. 4 the interaction between all of the software components is represented at the highest level. The components are:

A response 4010 is a collection of information collected from a survey respondent with possible additions of information known or inferred about the user based on information collected outside of this survey response by a specific survey agent. Each response has a unique identifier that identifies that specific instance of a response and its corresponding survey agent.

A credit 4020 is a specific type and form of reward associated with a particular survey agent. This can take on a plurality of forms, such as rebates, coupons or discounts. Each credit has unique identifier that identifies that specific instance of a credit and its corresponding response.

A survey agent 4030 is responsible for presenting the customer with the survey questions using the output means provided by the survey device. The survey agent of the preferred mode includes a set of survey questions for a particular survey and other qualification criteria such as locale criteria, Time, Place, and Manner criteria, and respondent qualification criteria. They are responsible for collecting the response using the input means provided by the survey device. When the survey respondent is finished responding to a particular response the survey agent delivers this information along with the agent's unique identifier to the palm manager. A survey agent represents the information desired from the survey respondent, the properties regarding the target audience and the type and form of credit associated with this information. The survey agent also performs rudimentary sanity checks to prevent fraud and falsification of information.

A palm manager 4040 handles all survey respondent requests for surveys. The term palm manager is used here to remind the reader that the preferred mode puts the palm manager inside the survey device. The palm manager could be in a device in communication with the survey device whether by wireless link, a carrier signed on the power supply connection, docking station, or conventional wired connection. The palm manager present a list of available survey agents to the survey respondent. When a survey respondent selects a specific survey agent, the palm manager activates the survey agent on the survey device. When the survey respondent has finished, the palm manager collects the survey from the survey agent. Once the palm manager

receives the completed response, the palm manager uses the electronic transfer means to forward the response to the collector. Using the unique response identifier, the palm manager tells the credit agent the type and form of credit to authorize. Since some survey agents have a specific target audience they may not be presented to the survey respondent until the survey respondent has shared some information with the palm manager that would classify the survey respondent in the target audience. This can be accomplished if the survey respondent chooses to identify himself using an unique identifier assigned to him on previous occasions or a piece of identifying personal information.

A credit agent 4050 authorizes a specific type and form of credit based on the information provided. The credit agent reports all credits authorized to the collector so that they can be reconciled with the rebate paid to the participating retailer.

A collector 4060 collects surveys from all palm managers at a retail queuing location. Then, the collector updates the palm managers with any new survey agents and removes any expired survey agents. Once the responses and credits have been collected from the various palm managers, the collector uses the electronic transfer means to forward the collected surveys and the collected credits to the server manager.

The server manager 4070 collects the surveys and corresponding credits from the collectors. Then, the server manager updates the collectors with new survey agents and removes survey agents that have not been selected during the selection window for that survey agent or that have been terminated with a termination command. A termination command is a command that could be used to stop a survey once the marketer discovers a flawed design in the survey or has received sufficient information and does not wish to pay for additional completed surveys. The server manager presents the appropriate responses and credits to the various marketer agents that correspond to the survey agents.

A marketer agent 4080 organizes and formats the responses and credits collected for each specific marketer. Each marketer agent corresponds to a specific survey agent distributed to the retail locations. The marketer agent provides the surveys to the marketer and invoices the marketer for the credits.

A marketer 3500 is a marketer that has requested the survey information and will be invoiced for the information and credit costs. Each marketer can create a new marketer agent customized to the information that they desire to gather.

FIG. 5 shows some examples of the hardware that could be used to support the functionality. The first example shows a system where the survey devices 5011 communicate using a wireless directly to the information server 5001. A variation would be for the survey devices 5011 to communicate with a local server that would then relay the communication to the information server 5001. The information server 5001 supports the server manager 4070 and the marketer

agents 4080. The information server collects all the responses and credits to a central location and does processing that needs to occur across all survey information. The survey devices 5001 provide a mobile platform that can be distributed to many survey respondents in a single queuing location 3400 such as a retail location. The survey devices 5011 support the functionality of the palm managers 4040 and survey agents 4030. The functionality of the collectors 4060 and credit agents 4050 can be supported in a variety of manners based on the scale of the operation for performance and convenience reasons. The collectors 4060 and credit agents 4050 could be integrated on the survey devices 5011, supported by an intermediate layer of computers distributed at the retail locations such as the Queuing Location Digital Survey System Computer 3420 from Figures 2 and 3, or integrated on the information server 5001.

A second example within FIG. 5 shows a set of survey devices 5012 that do not have the capability to interface directly with the information servers 5002. Instead the survey devices are hardwired to an intermediary computer 5022. As mentioned above, the mobility of the survey devices 5002 may be quite high if there are a large number of ports distributed throughout the queuing location. One technology for providing many ports without running special wire is to use the existing electrical circuits as a carrier for the computer communication. In such a case, it is reasonable to delegate certain responsibilities to this intermediary computer 5022. For instance, the collector 4060 and credit manager 4050 might be served better from this intermediary computer 5022. However, if the survey devices 3450 have the communication capabilities necessary, it is reasonable to do without the intermediary and have the survey device implement the functions of the collector 4060 and credit manager 4050.

The importance of the topology is less significant than the fact that the Digital Survey System allows the marketers 3500 to specify the survey they wish to create and the Digital Survey System manages the distribution, acquisition, fulfillment and collection of the agents

FIG. 6 illustrates how the invention would administer a survey using a survey device 6000 in a queuing location 3400. Step 7010, palm manager 4040, displays a list of survey agents 4030 that are available to the survey respondent 3550. The list of survey agents 4030 may be interspersed with the informational/recreational content on the survey device 6000.

Step 7020, the survey respondent 3550 selects a specific survey agent 4030 such as the agent for a Turkey Travel Survey 3000. Step 7030, the palm manager 4040 activates the survey agent 4030.

Step 7040, the survey agent 4030 methodically presents survey questions including audio or video material to the consumer/respondent 3550.

Step 7050, the survey respondent 3550 answers the survey questions. The survey agent 4030 may alter the composition of the questions presented based on the answers provided by the survey respondent 3550. Step 7060, the survey agent 4030 returns the completed survey 6010 to the palm manager 4040.

FIG. 7 illustrates how the surveys are collected from the various handheld survey devices 3450 in the queuing location 3400.

Step 7070, the palm manager 4040 transmits all completed surveys 6010 and authorized credits 6020 to the collector 4060. The collector 4060 transmits a list of expired survey agents 4030 and credit agents 4050 to the palm manager 4040.

Step 7080, the palm manager 4040 deletes all expired survey agents 4030 and credit agents 4050.

Step 7090, the collector transmits all new survey agents to the palm manager 4040. The palm manager 4040 incorporates the new survey agents into its list of active survey agents.

FIG. 8 illustrates one embodiment of the authorization of credits. In Step 7100, the palm manager 4040 activates a credit agent 4050 with the information regarding the respondent's completed survey 6010.

In Step 7110, the credit agent 4050 authorizes the retailer of the credit to be applied to the survey respondent's bill. The survey respondent's bill is reduced by the amount of the credit and the survey respondent pays the remainder of the adjusted bill. Thus receiving an immediate benefit from the marketer 3500.

In Step 7120, the credit agent 4050 reports the exercised credits back to the collector 4060.

FIG. 9 illustrates one embodiment of how the surveys are collected from the various queuing locations. In Step 7130, the collector 4060 transmits the completed surveys 6010 and exercised credits to the server manager 4070.

In Step 7140, the server manager 4070 evaluates the survey agents to determine which agents have expired either by fulfilling the number of desired responses, finishing the desired the survey period or the marketer requesting termination of the electronic survey--such as when the electronic survey was associated with a promotion that the marketer terminated.

In Step 7150, the server manager 4070 transmits the list of expired survey agents 4030 and credit agents 4050 to the collector 4060. The collector 4060 deletes all expired survey and credit agents. In this push-based example, the server manager 4070 transmits all new survey agents 4030 to the collector 4060. The collector 4060 incorporates the new survey agents 4030 and credit agents 4050 with its existing agents. Note that this example did not involve any locale qualifications so it is possible that surveys will be pushed to the location and never used because the location does not meet one or more qualifications for the survey. (Such as needing bowling alleys at the location in order to make it possible to give a survey to people who are waiting for a bowling lane.)

FIG. 10 illustrates how the survey information is processed and delivered to the marketer. In Step 7160, the marketer agent 4080 receives the new survey response sets from server

manager 4070. Step 7170, the marketer agent 4080 verifies the information provided by the responses using available phone number, address and other published databases. Then the information is added to the marketer database, locale database and survey respondent database.

5 Step 7180, the marketer agent 4080 organizes and formats the responses according to the marketer's specification. This includes removing all identifying information not specifically released to the marketer by the survey respondent 3550 from the survey information to be provided to the marketer. In this way, the anonymity and privacy of the survey respondents 3550 can be preserved.

10 Step 7190, the marketer agent 4080 records and credits each individual queuing location 3400 for the costs of the credits provided to the survey respondents 3550 redeemed or exercised at the retail location at the queuing location.

Step 7200, the marketer agent 4080 transmits the survey information to the marketer's survey data repository 4095. Step 7200, the marketer agent 4080 then invoices the marketer 3500 for the survey response sets collected and credits delivered.

15 The present invention is not limited to the implementation of an object oriented language. The functions present in the first explanation would be regrouped for an implementation under a structured language (non-object oriented).

20 As a structured design, the invention represents the conjunction of five processes. The processes are survey distribution, survey collection, survey selection, survey acquisition and survey respondent lookup.

FIG. 11 illustrates the distribution of surveys from the information servers to the various locales. Step 8010, the first aspect of survey distribution is the input of the marketer's information. In Step 8020, the survey specification is created including the survey targets and associated credits. In Step 8030, the marketer's information is then added to the marketer database.

25 Step 8040, the Digital Survey System then determines which locales are likely to have the survey's intended targets. The locale database is queried and a determination is made as to whether each locale matches the target. If the locale does match, the credit appropriate for that locale is determined and the survey is distributed to that locale. The credit for a locale is a function of the value of the credit authorized by the marketer and the types of credit uses
30 previously environmentted for that queuing location.

FIG. 12 illustrates the collection of surveys from the various locales to the information servers. For a system with direct communication between the survey device and the information server, the first step (8050) is the information server receives the responses and associated survey respondent information from the survey device 3450. In step 8060, then the survey respondent
35 information is verified against available databases of phone numbers, addresses and other

information. If the information does not seem legitimate, it is flagged and handled as an exception by either an exception routine or review by a person.

If the information seems legitimate, then Step 8070 is conducted so that survey respondent profile is looked up in the survey respondent database. If no profile exists for this survey respondent, a
5 new survey respondent profile is created. The new survey respondent information is added to the survey respondent profile.

Step 8080, the information server looks up the survey's marketer in the marketer database. In Step 8090, the set of responses is added to the marketer's survey database. In Step 8100 the new set of survey results is made available to the marketer. Making "available" is likely to include
10 working to organize and format the new set of survey responses to the marketer's specifications and before transmitting the results to marketer. As mentioned above, the survey response sets could be sent one by one or in one aggregated collection.

FIG. 13 illustrates an embodiment of the process of selection of surveys to present to the prospective survey respondent. Step 8110, the survey device is activated. In Step 8120, after the
15 prospective survey respondent activates the survey device, the survey device may request that the prospective survey respondent provide some piece of identifying information. As discussed above, the survey respondent may choose whether to provide identifying information. In Step 8130, if the prospective survey respondent provides identifying information, the survey device looks up the survey respondent's profile. (See Figure 15 and associated text for details on
20 the user profile lookup).

Sometimes knowledge of the credit type may identify the marketer and bias results. Thus, in the preferred mode the credit is identified, described generally as a promotional item or just described as "???". If the survey respondent does not provide identifying information at this point, the survey device will perform the survey respondent lookup as soon as identifying information has
25 been provided. This may happen during the course of a survey response.

In Step 8140, the survey device looks up the available surveys from those received by the locale at which this survey device is deployed. Step 8150 illustrates the selection of a subset of surveys bases on a qualification criteria. In this case the qualification criteria is whether the survey's time constraints match the actual time of day. The time constraint check allows marketers to specify
30 orange juice surveys for breakfast time and liqueur surveys for after 10 p.m.

In Step 8160, the survey device evaluates whether the information know about the potential survey respondent disqualifies the potential survey respondent from the target audience of the survey. If all the constraints on the target audience for the survey are met or still unanswered, the electronic survey is added to the survey list visible to that particular prospective survey
35 respondent. The process of evaluation is repeated for each electronic survey until all available surveys have been evaluated. Then as Step 8170, the surveys on the list are displayed.

Variations include randomly selecting an electronic survey from the list and inviting the prospective survey respondent to complete that survey. The prospective survey respondent may decline the offered survey and be presented with another survey option. Yet another alternative is to display the entire list of available electronic surveys that are present at the queuing location and then provide an indication to the prospective survey respondent of the subset of electronic surveys that the prospective survey respondent is qualified, disqualified, or may be qualified to answer. The last category is for electronic surveys that have qualification criteria that has not been adequately addressed by information available to the electronic survey. In this latter case, the electronic survey begins by asking the unanswered qualification criteria until the respondent is qualified or disqualified. If disqualified the respondent can be thanked and returned to the survey list. The surveys may be interspersed throughout recreational/informational content on the survey database.

FIG. 14 illustrates one implementation of the process for acquiring information from the survey respondent. Step 8180, the survey respondent chooses a specific survey. Step 8190, the survey device presents the information related to the survey. Step 8200, the survey respondent responds to this presented information. Step 8210, if the information provided by the survey respondent is identifying information, this is added to the survey respondent information.

In Step 8220, information provided in response to the presented information is added to the file containing the respondent's survey responses.

At loop branch 8230, the process of presenting information and processing the responses continues until the survey is completed or terminated.

Step 8240 is performed when a survey has been completed. In Step 8240, the file containing the survey respondent's response and the survey respondent information are sent to the information servers. In Step 8250, the survey credit is authorized. The authorized credit is given to the survey respondent.

FIG. 15 describes how a survey respondent lookup is carried out. Step 8180 occurs after the survey respondent ("user") provides identifying information, Step 8120. Step 8180 consists of the survey device performing a survey respondent lookup by sending the identifying information to the information servers. The identifying information is used to perform a lookup in the survey respondent database on the information server.

At Branch Step 8190, if the response of the survey respondent database is that there is no survey respondent profile that corresponds to that identifying information, then a new survey respondent profile is created.

At Step 8200, the survey respondent profile that corresponds to that identifying information is returned to the survey device.

Those skilled in the art will recognize that the method and apparatus of the present invention has many applications and that the present invention is not limited to the representative examples disclosed herein. Moreover, the scope of the present invention covers the range of variations, modifications, and substitutes for the system components described herein, as would be known to
5 those skilled in the art.

The legal limitation of the scope of the claimed invention is set forth in the claims that follow along with the legal equivalents of the claims that follow.

CLAIMS

I claim:

1. A top level data processing system for collecting survey results from one or more survey respondents at a plurality of queuing locations,
5 comprising:
 - a computer processor means for processing data;
 - a storage means for storing data on a storage medium;
 - a first means for storing information constituting a set of survey questions;
 - a second means for storing information constituting a qualification file for the set of
10 survey questions;
 - a third means for storing information constituting a for the set of survey questions;
 - a fourth means for receiving information from a device that is not a present at the same location as the computer processor means;
 - a fifth means for distributing survey sets to locations proximate to the queuing locations;
 - 15 a sixth means for accepting a completed set of survey questions based on a comparison of attributes of the survey respondent to a set of conditions in the qualification file;and
 - 20 a seventh means for transferring a credit proportional to the reward class for the set of survey questions for a set of completed survey questions.
2. A data processing system as claimed in claim 1, further comprising a fourth means for storing a Time, Place, and Manner qualifications file.
3. A data processing system as claimed in claim 2 wherein the Time, Place, and Manner
25 qualifications information for a queuing location is used to answer questions from the survey without presentation to the survey respondent.
4. A data processing system as claimed in claim 2 wherein the Time, Place, and Manner
30 qualifications information for a queuing location is used to personalize questions from the survey for presentation to the survey respondent.
5. A data processing system as claimed in claim 1 wherein a set of extra-survey information for particular survey is used to expedite the survey process by formulating proposed survey
35 answers which are presented to the survey respondent to confirm.
6. A data processing system as claimed in claim 5 wherein the extra-survey information contains information from a superset of the sets of sources of:

- a) information provided by the survey respondent at the start of the session for qualification to see the survey as a choice;
- b) information provided by the survey respondent during the same session as this survey as qualification information after a survey was presented as a choice;
- 5 c) information provided to a survey interview in this same session;
- d) information provided to an unsuccessful attempt to complete this same survey during this same survey session;
- e) information provided to an earlier survey session by someone providing the same identification key as the current survey respondent;
- 10 f) information about the survey device for the survey device being used by the survey respondent; and
- g) information sensed by instruments at the queuing location.

7. A data processing system as claimed in claim 6 wherein the information sensed by
15 instruments at the queuing location includes sensing parameters selected from the group consisting of time between responses by the survey respondent;

the ambient temperature;

ambient noise level;

ambient light level; and

20 movement of survey device.

8. A data processing system as claimed in claim 2 wherein the Time, Place, and Manner qualifications information for a queuing location provides information to prevent a survey
25 respondent from having the option of starting a survey.

9. A data processing system as claimed in claim 2 wherein the Time, Place, and Manner qualifications information for a queuing location provides information to prevent a survey
respondent from having the option of completing a survey.

30 10. A data processing system as claimed in claim 1, further comprising an eighth means for transferring a set of information containing at least some of the information contained in the returned information response to a second computer processing system that is remote from the computer processing system.

35 11. A data processing system as claimed in claim 1, further comprising a ninth means for converting a set of survey requirements into a set of survey questions.

12. A data processing system as claimed in claim 11, further comprising an eleventh means for receiving a set of survey questions from the second computer processing system.

13. A method for collecting a set of response sets to electronic surveys from qualified respondents from one of a plurality of queuing locations through the use of a computer system comprising:

inputting an electronic survey into the computer system;

inputting a set of one or more qualification criteria;

10 transmitting the electronic survey and at least some of the qualification criteria to a first queuing location in the plurality of queuing locations;

evaluating the information returned from the first queuing location sometime after the transmission of the electronic survey and the transmitted qualification criteria, to detect the return of a complete response set from a first respondent meeting the transmitted qualification criteria;

15 and

after detection of the complete response set from the first respondent meeting the transmitted qualification criteria, sending a credit to the first queuing location for the complete response set from the first respondent.

14. The method of claim 13 wherein the step of inputting the electronic survey into the computer system includes the step of receiving the electronic survey from a remote computer and making the received electronic survey accessible to the computer system.

15. The method of claim 14 wherein the step of receiving the electronic survey from the remote computer includes receiving the survey from an inter-network connection with the remote computer.

16. The method of claim 14 wherein the step of receiving the electronic survey from the remote computer includes receiving the electronic survey from a telephone connection with the remote computer.

17. The method of claim 14 wherein the step of receiving the electronic survey from the remote computer includes receiving the electronic survey on physical media containing output from the remote computer in a format that can be input into the computer system by an input device connected to the computer system.

18. The method of claim 13 wherein the inputting the electronic survey into the computer system comprises inputting questions and computer programming code to control the survey experience presented to a survey respondent at the remote queuing location.

5 19. The method of claim 18 wherein the computer programming code has the capacity to imbed information from the respondent into questions presented to the respondent.

20. The method of claim 18 wherein the computer programming code has the capacity to imbed logic in the electronic survey so that the respondent is offered a subset of
10 questions based on information provided by the respondent .

21. The method of claim 20 wherein the computer programming code has the capacity to imbed logic in the electronic survey so that the first respondent is offered a subset of questions based on information provided by the first respondent to another electronic survey at
15 the same queuing location during the same survey response session.

22. The method of claim 20 wherein the computer programming code has the capacity to imbed logic in the electronic survey so that the respondent is offered a subset of questions based on information provided by the respondent to another electronic survey taken at a
20 different queuing location.

23. The method of claim 22 wherein the computer programming code has the capacity to imbed logic in the electronic survey so that the first respondent is offered the option to approve the inclusion of one or more responses to the electronic survey into a respondent profile that can be used by logic in subsequent electronic surveys to alter the way the survey questions
25 for the subsequent electronic survey are presented to the first respondent at a second remote queuing location.

24. The method of claim 18 wherein the computer programming code has the capacity to imbed logic in the electronic survey so that the first respondent is asked to confirm certain responses when a tendered response does not match information about the first respondent
30 available to the electronic survey.

25. The method of claim 24 wherein the computer programming code has the capacity to imbed logic in the electronic survey so that the first respondent is asked to confirm certain responses when a tendered response from the first respondent does not match information
35 provided by the first respondent to another electronic survey.

26. The method of claim 24 wherein the computer programming code has the capacity to imbed logic in the electronic survey so that the first respondent is asked to confirm certain responses when a tendered response does not match information provided by the first respondent to another electronic survey completed by the first respondent at the same queuing location during the same survey response session.
27. The method of claim 24 wherein the computer programming code has the capacity to imbed logic in the electronic survey so that the first respondent is asked to confirm certain responses when a tendered response does not match information provided by the first respondent to another electronic survey taken at a second queuing location.
28. The method of claim 13 wherein the step of inputting the electronic survey into the computer system includes the step of receiving electronic files that include audio information to be included in the electronic survey.
29. The method of claim 13 wherein the step of inputting the electronic survey into the computer system includes the step of receiving electronic files that include visual images to be included in the electronic survey.
30. The method of claim 13 wherein the step of inputting the electronic survey into the computer system includes the step of receiving electronic files that include animation sequences to be included in the electronic survey.
31. The method of claim 13 wherein the step of inputting the electronic survey into the computer system includes the step of receiving electronic files that include video sequences to be included in the electronic survey.
32. The method of claim 13 wherein the electronic survey inputted into the computer system includes questions to request qualification criteria in addition to questions to collect survey responses from qualified respondents.
33. The method of claim 32 wherein the electronic survey inputted into the computer system includes logic to end the electronic survey before completion if the responses first respondent violate required qualification criteria.

34. The method of claim 13 wherein the step of inputting the qualification criteria into the computer system includes the step of receiving the qualification criteria from a remote computer;
making the transmitted survey accessible to the computer system; and
5 associating the qualification criteria with one or more electronic surveys stored within the computer system.
35. The method of claim 13 wherein the qualification criteria inputted into the computer system include qualification criteria for the type of queuing location to be eligible to
10 receive the transmitted electronic survey.
36. The method of claim 13 wherein the qualification criteria inputted into the computer system include qualification criteria limiting the times of the week that the remote queuing location can make the electronic survey available for respondents to start responding to
15 the electronic survey.
37. The method of claim 13 wherein the qualification criteria inputted into the computer system includes qualification criteria requiring that the first queuing location be offering certain services at the time the electronic survey is made available to the first respondent.
20
38. The method of claim 13 wherein the qualification criteria inputted into the computer system includes qualifications on the geographic location of the first queuing location.
39. The method of claim 13 wherein the qualification criteria inputted into the computer system includes qualification criteria limiting the transmission of the electronic survey to queuing locations with visitor demographics that meet certain criteria.
25
40. The method of claim 13 wherein the qualification criteria is operative with computer code transmitted to the first queuing location to satisfy a qualification criteria based on information retained in memory after the completion of an electronic survey previously
30 completed by the first respondent during the same survey answering session.
41. The method of claim 13 wherein the qualification criteria is operative with computer code transmitted to the first queuing location to satisfy a qualification criteria based on information stored in memory in a local computer at the first queuing location.
35

42. The method of claim 41 wherein the qualification criteria is operative with computer code transmitted to the remote queuing location to satisfy a qualification criteria based on the respondent's user profile information stored in memory in a local computer at the first queuing.
- 5 43. The method of claim 13 further comprising the transmission of a survey window period to the first queuing location so that the first remote queuing location cannot present the electronic survey to the first respondent outside of the survey window period.
- 10 44. The method of claim 13 further comprising the transmission of a termination command so that otherwise un-expired electronic surveys stop being available for selection by the first respondent.
- 15 45. The method of claim 13 further comprising the transmission of a retail credit value to the first queuing location to provide an incentive for a qualified respondent to complete the electronic survey.
- 20 46. The method of claim 45 further comprising the step of converting the retail credit value into a first retail incentive at the first queuing location and into a different retail incentive at a second queuing location.
- 25 47. The method of claim 46 wherein the conversion of the retail credit value into the first retail incentive is performed at the first queuing location.
- 30 48. The method of claim 45 wherein the payment to the first queuing location for the response set results in the availability of the first retail incentive to the qualified respondent in time for same-visit use.
- 35 49. The method of claim 48 wherein the payment to the first queuing location results in an offset for the qualified respondent against the bill otherwise owed by the qualified respondent to a retail establishment at the first queuing location.
50. The method of claim 48 wherein the payment to the first queuing location for the response set results in the availability of the first retail incentive to the qualified respondent in time for same-visit use with an option provided to the qualified respondent to choose to save the retail credit for use at another queuing location.

51. The method of claim 13 further comprising the steps of:
sending the results of the complete response set to a survey requestor; and
invoicing the survey requestor for the complete response.
- 5 52. The method of claim 51 further comprising the steps of:
waiting for a notice from the retail establishment at the first queuing location of the
redemption of the first retail incentive by the qualified respondent; and
charging the survey requestor for the redeemed first retail incentive.
- 10 53. A Queuing Location Computer System for use in soliciting responses to
electronic surveys a Potential Survey Respondent present at a queuing location comprising:
a queuing location computer in turn comprising:
a1) a memory storage device;
a2) a means for receiving a plurality of electronic surveys including both a
15 first electronic survey containing questions for a Potential Survey Respondent from a Survey
Computer System and a second electronic survey;
a3) a means for receiving a plurality of qualification criteria sets from the
Survey Computer System including a qualification criteria set applicable to the first electronic
survey and a qualification criteria set applicable to the second electronic survey;
20 a4) a means for receiving retail incentive information including retail
incentive information for the first electronic survey and retail incentive information for the
second electronic survey;
a5) a means for communicating to a first Potential Survey Respondent, a list
of a current set of available electronic surveys for which the first Potential Survey Respondent is
25 a candidate for possible qualification to complete the electronic survey as defined by qualification
criteria in the qualification criteria set;
a first Survey Response Device;
an input means for the first Potential Survey Respondent to provide input to the first
Survey Response Device ;
30 a linkage path for digital information between the first Survey Response Device and the
queuing location computer;
a means for Potential Survey Respondents to provide input to the Queuing Location
Computer System carrying information relevant to the qualification criteria for the first electronic
survey;
35 a means for presenting information from the first electronic survey to the first Potential
Survey Respondent;

a means for storing the responses to the first electronic survey from the first Potential Survey Respondent;

a means for providing digital information to the Survey Computer System for information conveying at least some of the responses from the first Potential Survey Respondent; and

5 a means for providing a retail credit to the first Potential Survey Respondent after qualification and completion of a set of survey responses to the first electronic survey.

10 54. The Queuing Location Computer System of claim 53 wherein the means for providing digital information to the Survey Computer System conveying at least some of the responses from the first Potential Survey Respondent comprises a wireless communication link between the Survey Response Device and the Survey Computer System.

15 55. The Queuing Location Computer System of claim 53 wherein the means for providing digital information to the Survey Computer System conveying at least some of the responses from the first Potential Survey Respondent comprises a docking station for connecting the Survey Response Device to the Survey Computer System.

20 56. The Queuing Location Computer System of claim 55 wherein the means for providing digital information to the Survey Computer System conveying at least some of the responses from the first Potential Survey Respondent comprises a memory means in the Survey Response Device for storing responses from the first Potential Survey Respondent until completion of the first electronic survey before transmission to the Survey Computer System through the docking station.

25 57. The Queuing Location Computer System of claim 53 wherein a means for presenting information from the first electronic survey to the first Potential Survey Respondent includes a first video display system in wireline connection with the Queuing Location Computer System and electrically isolated from an Survey Response Device when the Survey Response Device is receiving input from a first Potential Survey Respondent based on the first Potential
30 Survey Respondent's viewing of material on the first video display.

35 58. The Queuing Location Computer System of claim 53 wherein the first Survey Response Device is a Personal Digital Assistant Device having Survey Response Device memory means for storing the first electronic survey; the second electronic survey; a response set to the first electronic survey; a response set to the second electronic survey; and a user profile;

a Survey Response Device processor for processing digital instructions; a set of Survey Response Device software for use in the performance of an electronic survey and interface with a first Potential Survey Respondent;

5 the processor, software, and electronic surveys operating to dynamically customize the first electronic survey for presentation to the first Potential Survey Respondent;

the processor, software, digital instructions, and memory means operative to store the responses of the first Potential Survey Respondent until the responses are transmitted by the digital transmission link to the Survey Computer System.

10 59. The Queuing Location Computer System of claim 58 wherein the Queuing Location Computer System accesses information about the queuing location, and the processor, software, and electronic surveys operating to dynamically customize the first electronic survey for presentation to the first Potential Survey Respondent based in part on information about the queuing location.

15 60. The Queuing Location Computer System of claim 58 wherein the Survey Response Device memory means for storing includes the results from the first Potential Survey Respondent of the first electronic survey; and the second electronic survey operates in conjunction with the processor and the set of Survey Response Device software such that
20 qualification criteria of the second electronic survey can be at least partially satisfied by answers provided by the first Potential Survey Respondent to the Survey Response Device in response to the first electronic survey.

25 61. The Queuing Location Computer System of claim 58 wherein user profile is loaded into the Survey Response Device for the first Potential Survey Respondent through a digital connection established while the Personal Digital Assistant Device is in a docking station linking the Personal Digital Assistant Device to the Survey Computer System.

30 62. The Queuing Location Computer System of claim 53 further comprising a means for the first Potential Survey Respondent to request the creation of a user profile.

35 63. The Queuing Location Computer System of claim 62 wherein the first Potential Survey Respondent is provided with a digital information carrying device to enable the first Potential Survey Respondent to access the first Potential Survey Respondent user profile at a second queuing location that does not share a common Point-of-Sale computer system with the first queuing location.

64. The Queuing Location Computer System of claim 62 wherein the user profile is stored on the first Survey Response Device for access during a subsequent use of the first Survey Response Device by the first Potential Survey Respondent after the use of the first Survey Response Device by a second Potential Survey Respondent.

5

65. The Queuing Location Computer System of claim 62 wherein the linkage path for digital information between the first Survey Response Device and the Queuing Location Computer is used for movement of digital information such that the first Potential Survey Respondent user profile comes into existence in the Queuing Location Computer memory storage device.

10

66. The Queuing Location Computer System of claim 65 wherein the Queuing Location Computer responds to a request to move a copy of the first Potential Survey Respondent user profile into the second Survey Response Device to reduce the amount of information needed for the first Potential Survey Respondent to input to the second Survey Response Device in order to qualify and complete a third electronic survey which is loaded into the memory of the second Survey Response Device.

15

67. The Queuing Location Computer System of claim 65 wherein the request to move a copy of the first Potential Survey Respondent user profile into the second Survey Response Device is satisfied by movement of the first Potential Survey Respondent user profile into the second Survey Response Device through digital link established through a docking station.

20

68. The Queuing Location Computer System of claim 62 wherein the Queuing Location Computer System transmits the user profile for the first Potential Survey Respondent to a central storage facility.

25

69. The Queuing Location Computer System of claim 62 wherein the Queuing Location Computer System periodically receives user profiles from the central storage facility for a Potential Survey Respondent that has not created a user profile through that Queuing Location Computer System.

30

70. The Queuing Location Computer System of claim 62 where the Queuing Location Computer System receives an updated user profile for the first Survey Response Device from the central storage facility where the updated user profile contains information provided by the first Survey Response Device to a second Queuing Location Computer System.

35

71. The Queuing Location Computer System of claim 53 wherein the Queuing Location Computer System further comprises a biometrics input device for identification of the first Potential Survey Respondent.
- 5
72. The Queuing Location Computer System of claim 71 wherein the biometrics input serves as a user identification key to distinguish one Potential Survey Respondent from another Potential Survey Respondent.
- 10
73. The Queuing Location Computer System of claim 71 wherein the biometrics input device is attached to the first Survey Response Device.
74. The Queuing Location Computer System of claim 53 further comprising a means for means for receiving retail incentive credits for the first Potential Survey Respondent after submission of a complete set of survey responses and needed qualification information from the first Potential Survey Respondent.
- 15
75. The Queuing Location Computer System of claim 74 wherein the first Potential Survey Respondent may redeem the retail incentive credit in the form of an entertainment experience provided through the Queuing Location Computer System.
- 20
76. The Queuing Location Computer System of claim 74 wherein the retail incentive credit for the first Potential Survey Respondent is made digitally available to a queuing location Point-of-Sale system to decrement the current bill associated with the first Potential Survey Respondent.
- 25
77. The Queuing Location Computer System of claim 76 wherein the first Potential Survey Respondent can use the retail incentive credit for the first Potential Survey Respondent without reference to a user profile for the first Potential Survey Respondent.
- 30
78. The Queuing Location Computer System of claim 74 where the first Potential Survey Respondent can elect to digitally bank the retail incentive credit for use during a subsequent visit to the queuing location.

79. The Queuing Location Computer System of claim 78, wherein the first Potential Survey Respondent can elect to draw upon a digitally banked retail incentive credit stored while at the first queuing location for use at a subsequent visit to a second queuing location that does not share a common Point-of-Sale computer with the first location.

5

80. The Queuing Location Computer System of claim 53 wherein the means for presenting information from the first electronic survey to the first Potential Survey Respondent includes the capacity to add local advertising content that is unrelated to the first electronic survey.

10

81. The Queuing Location Computer System of claim 80 wherein the local advertising content comprises advertising for items available at the current queuing location.

15

82. The Queuing Location Computer System of claim 53 wherein the means for presenting information from the first electronic survey to a first Potential Survey Respondent includes the capacity to limit the presentations to specific time windows within the week.

20

83. The Queuing Location Computer System of claim 82 wherein the presentation of the first electronic survey is limited to times when the queuing location is satisfying a time-place-manner qualification.

84. The invention as described and illustrated in the specification and referenced figures.

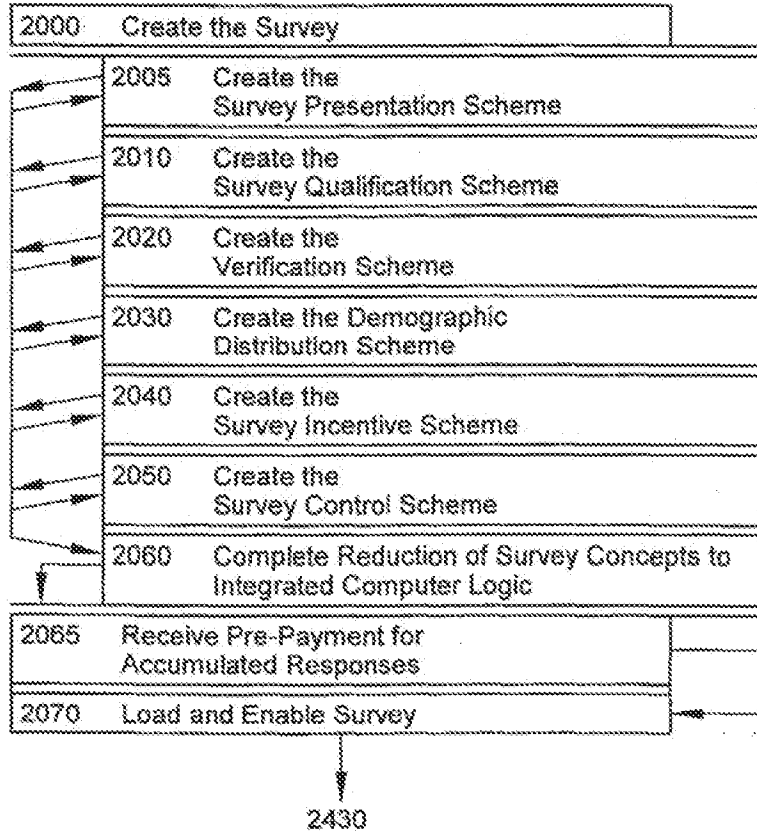


Fig. 2A

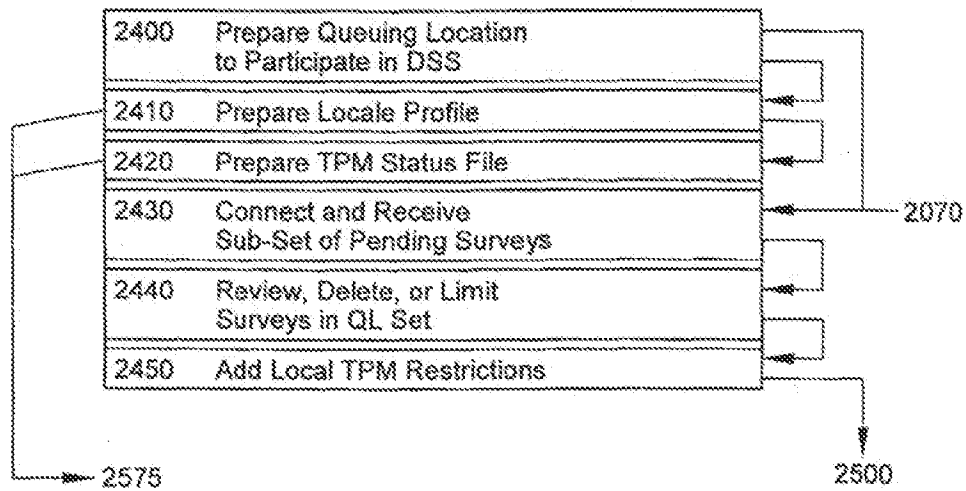


Fig. 2B

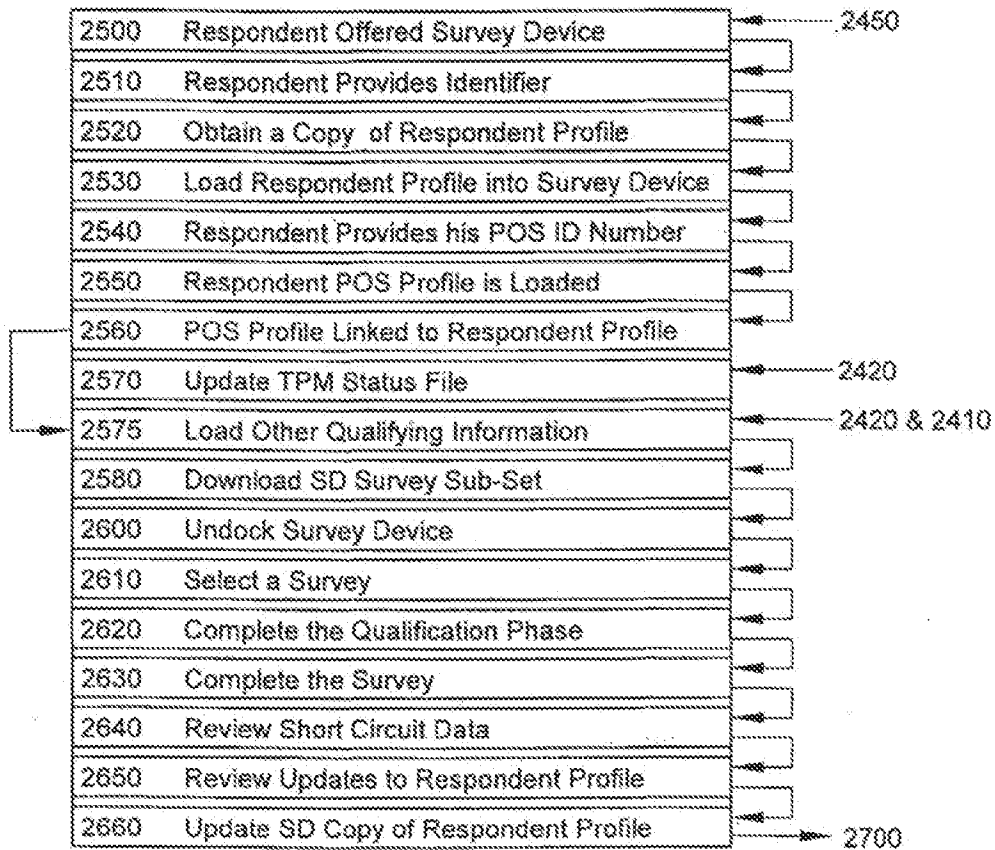


Fig. 2C

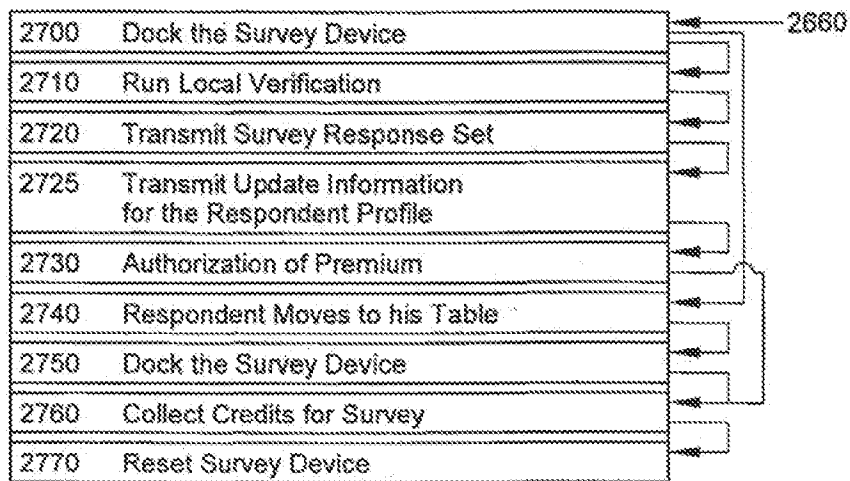


Fig. 2D

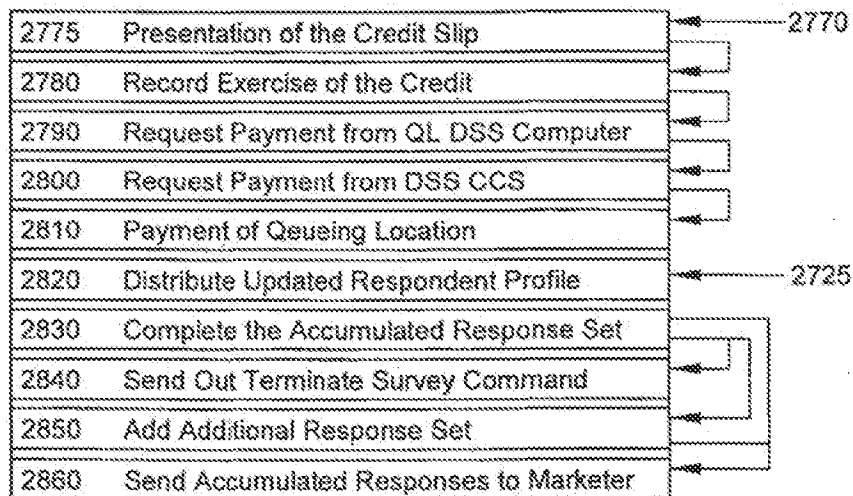
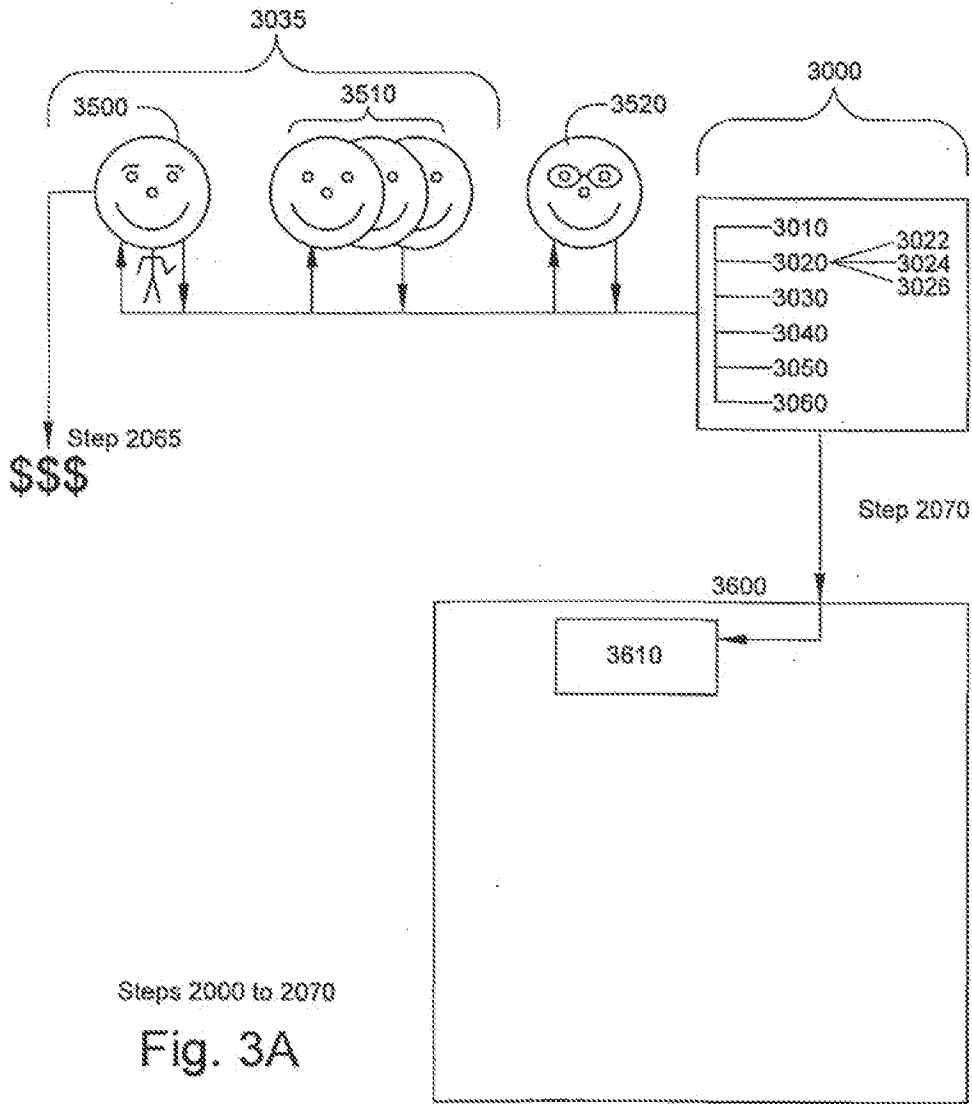
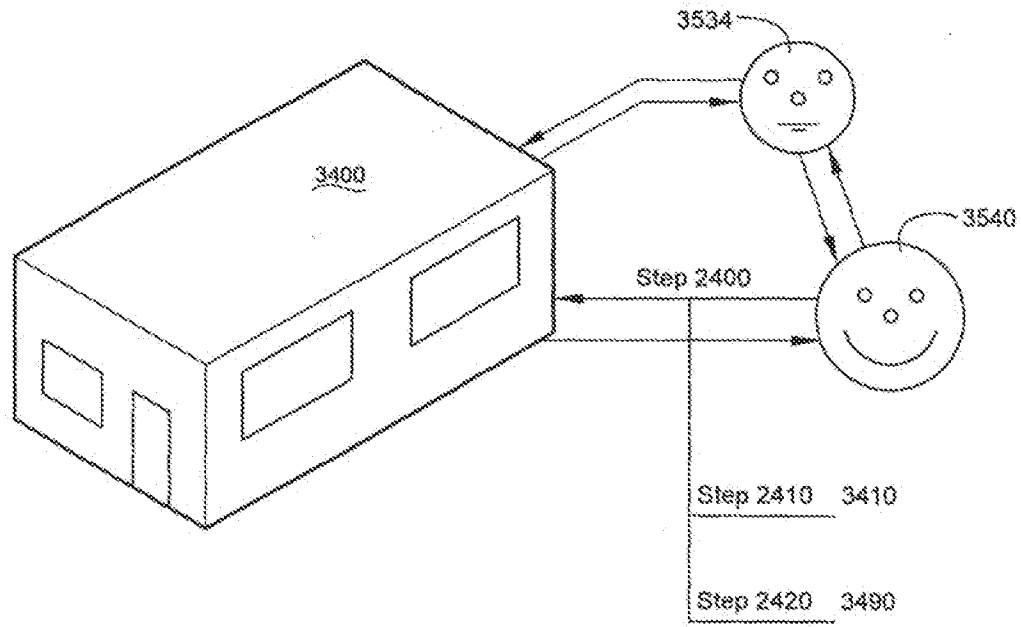


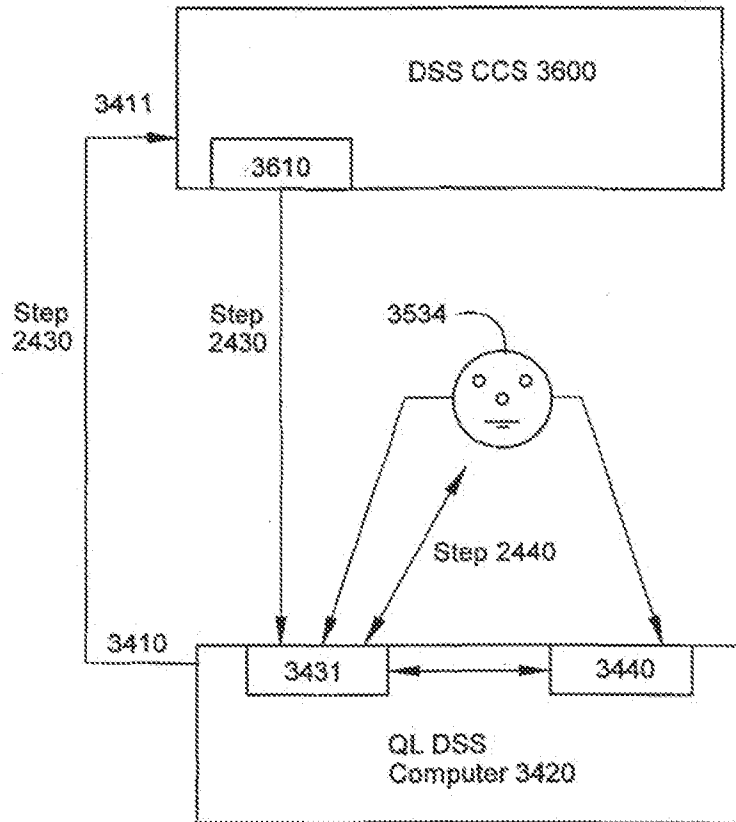
Fig. 2E





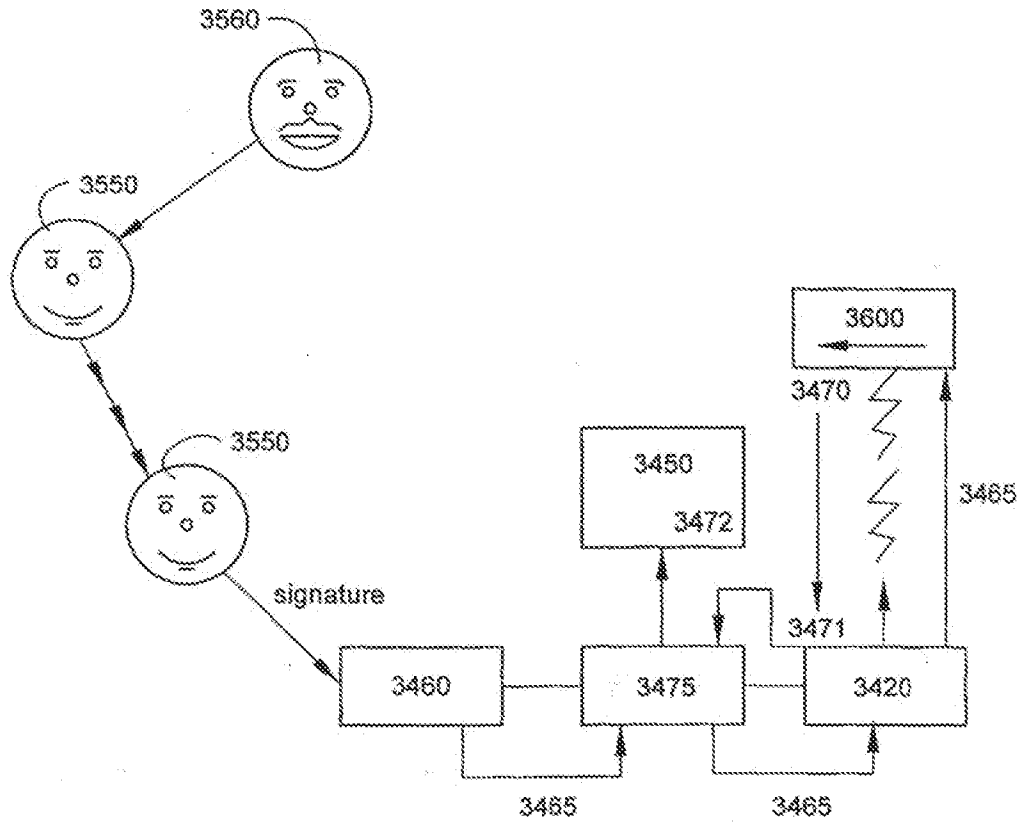
Steps 2400 - 2420

Fig. 3B



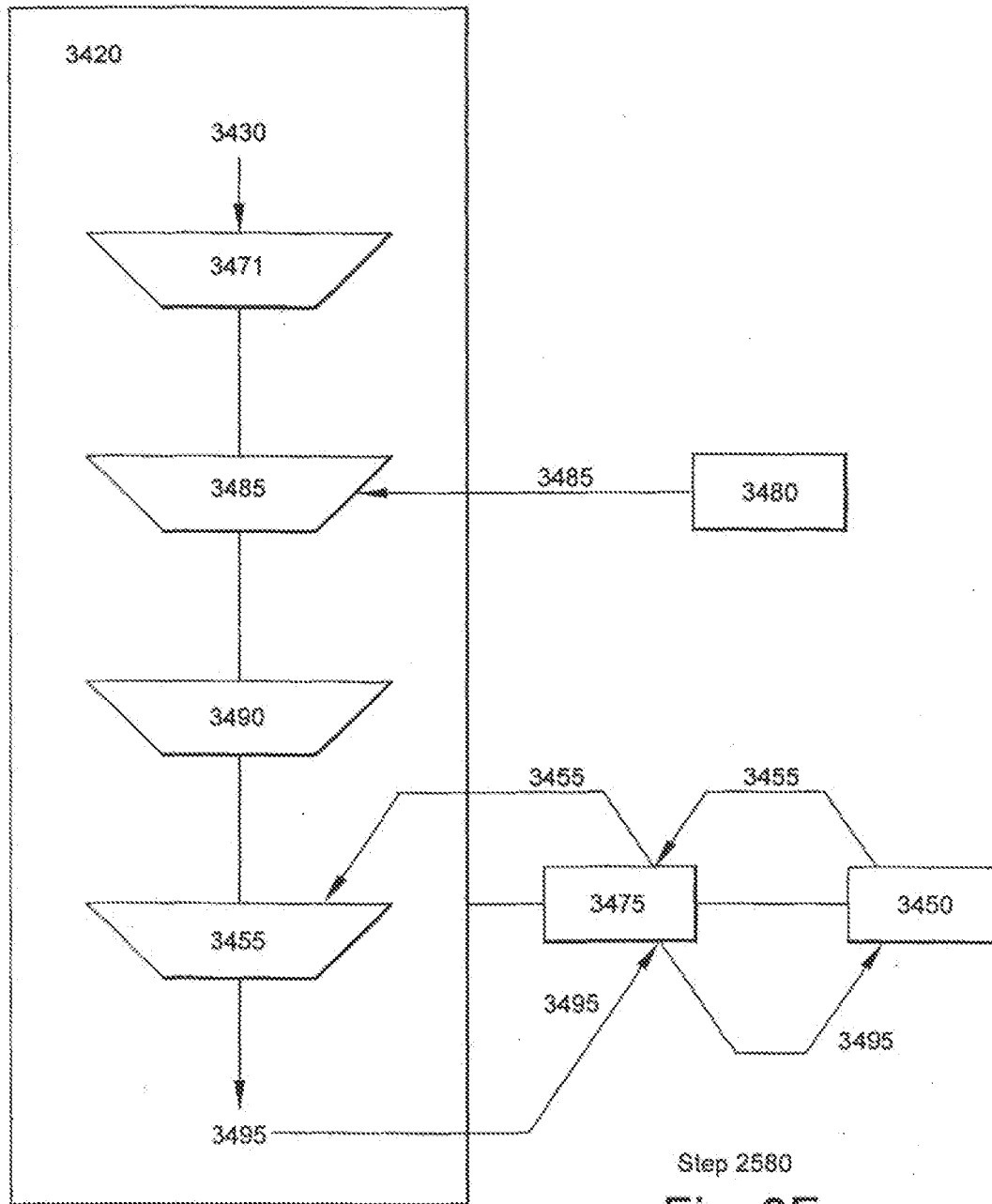
STEPS 2430 - 2450

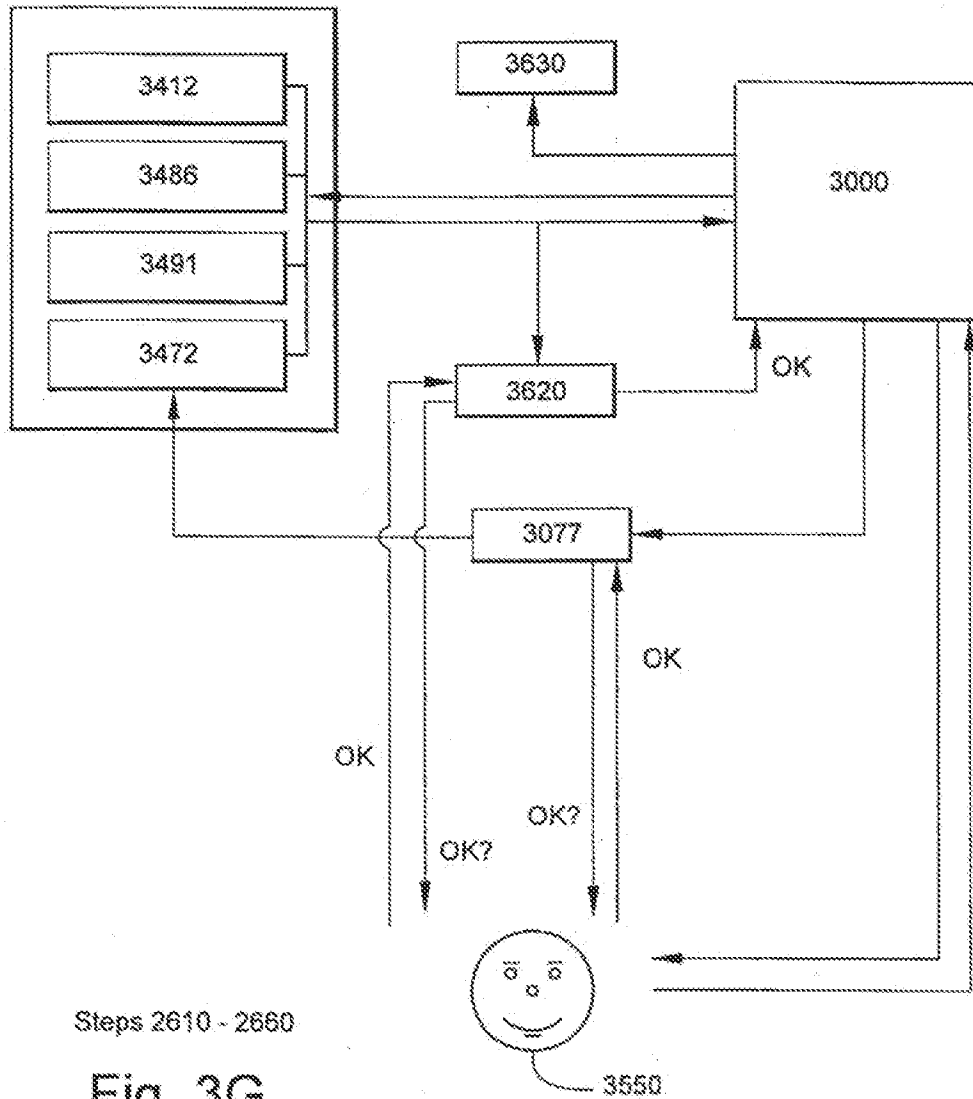
FIG. 3C



Steps 2500 - 2530

Fig. 3D





Steps 2610 - 2660

Fig. 3G

Steps 2700 - 2730

Fig. 3H

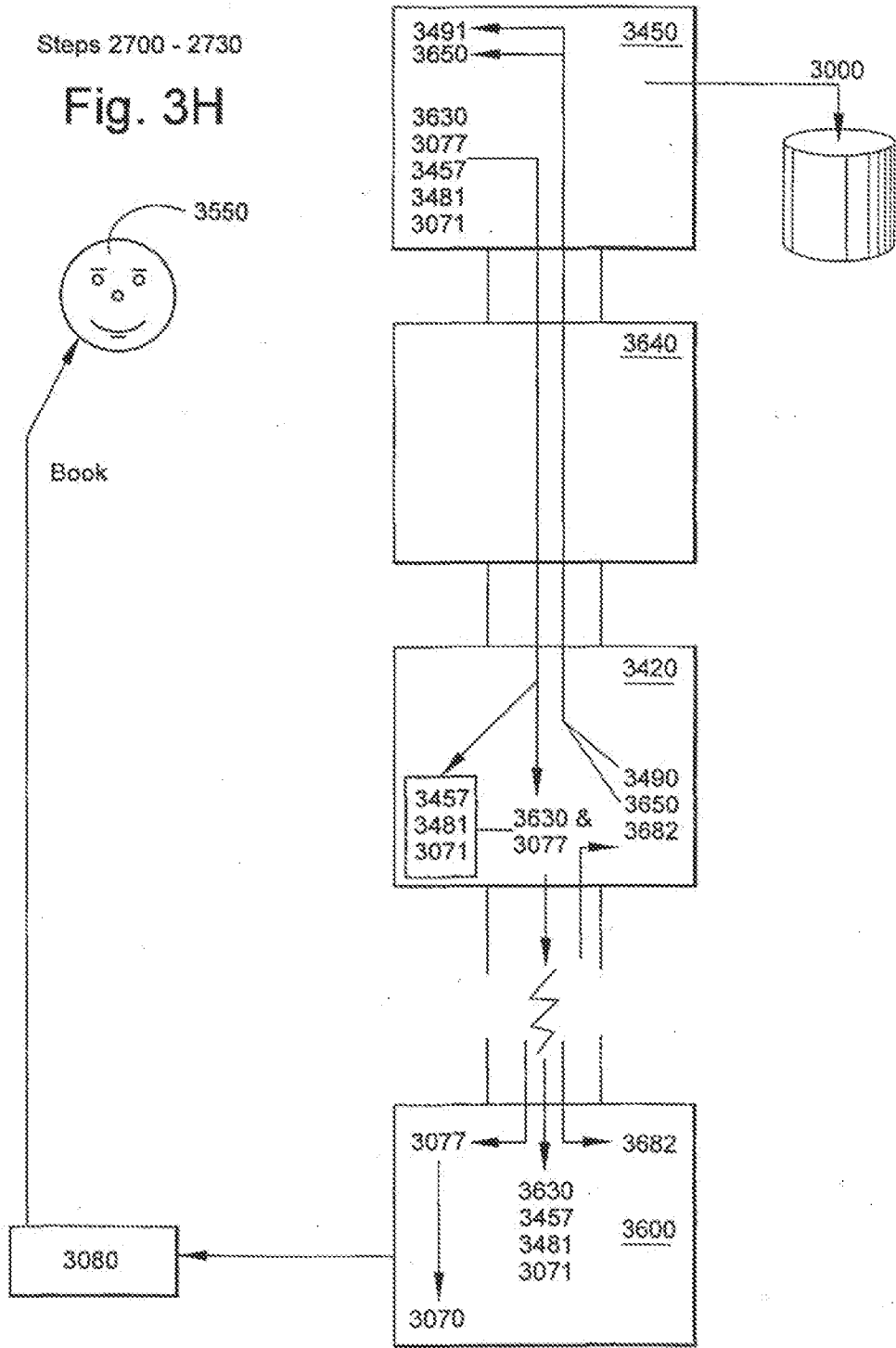
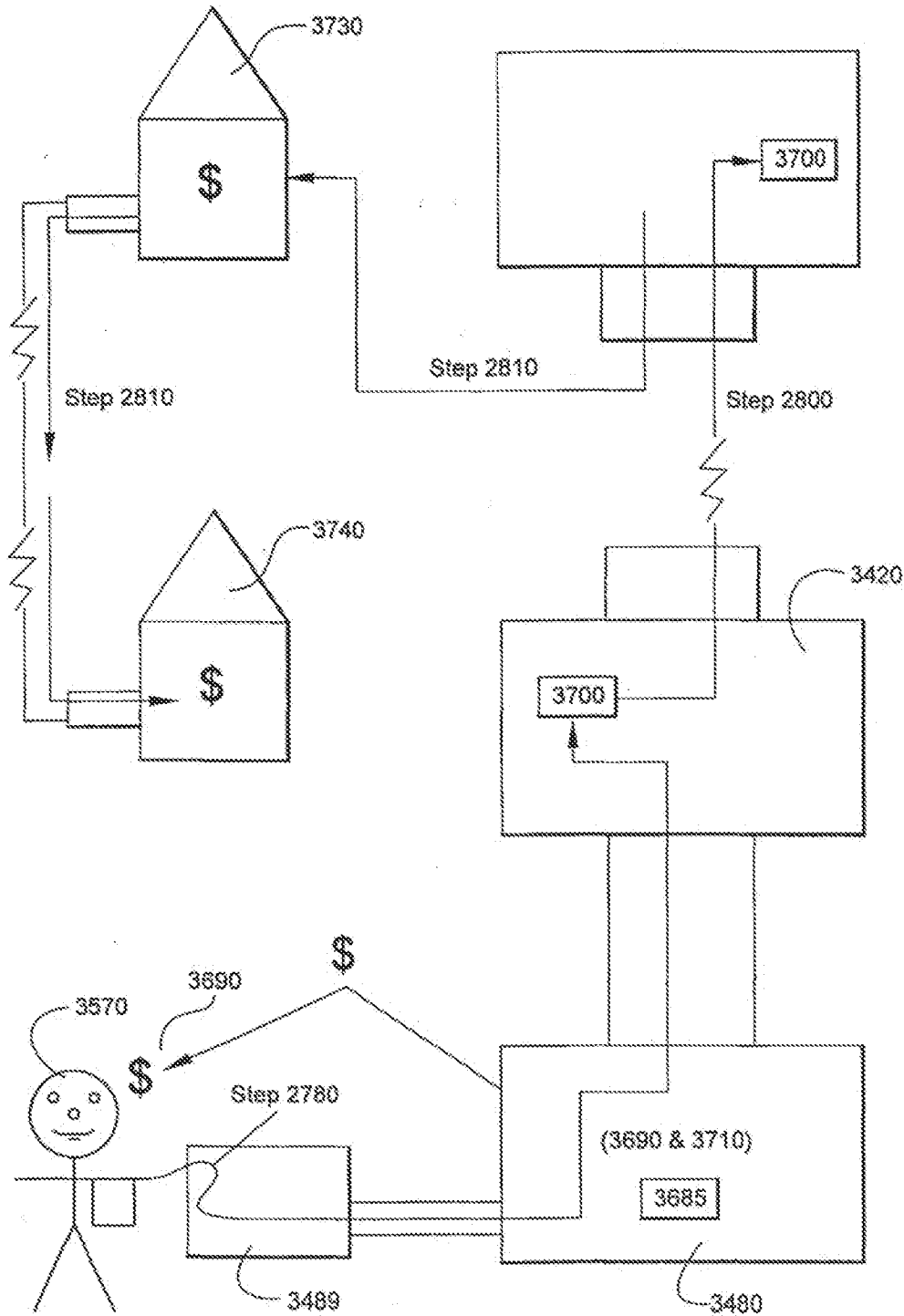
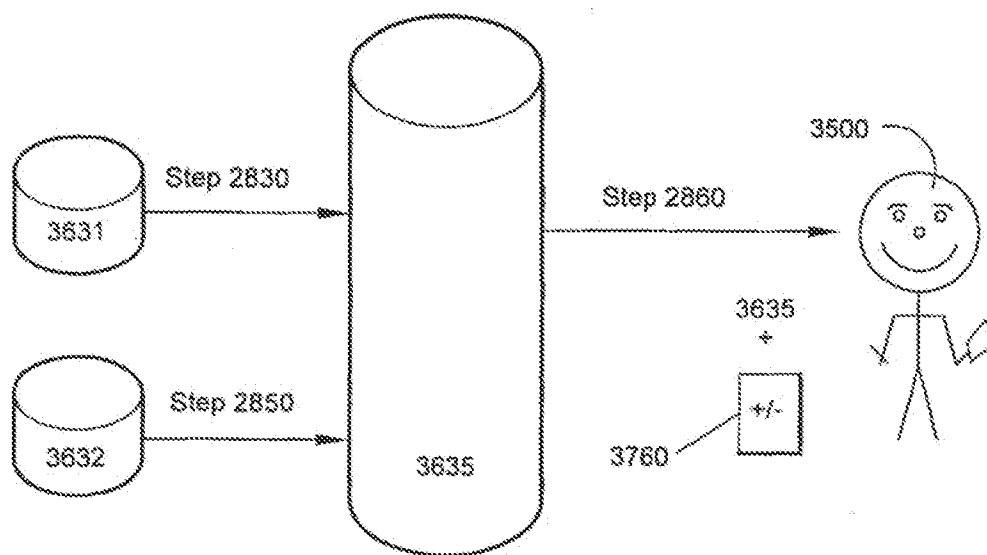


Fig. 3J
Steps 2780 - 2810





Steps 2830 - 2860

Fig. 3K

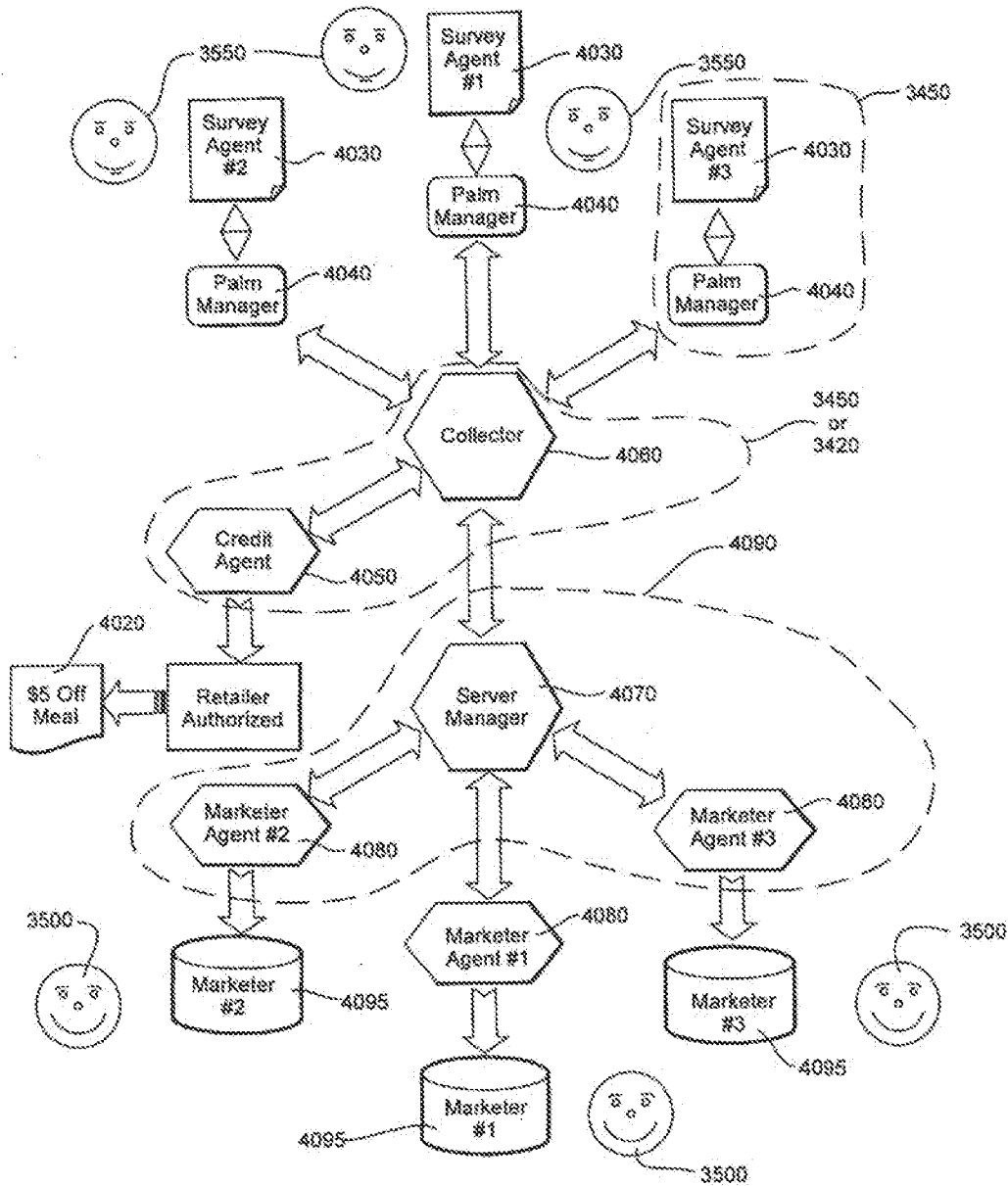
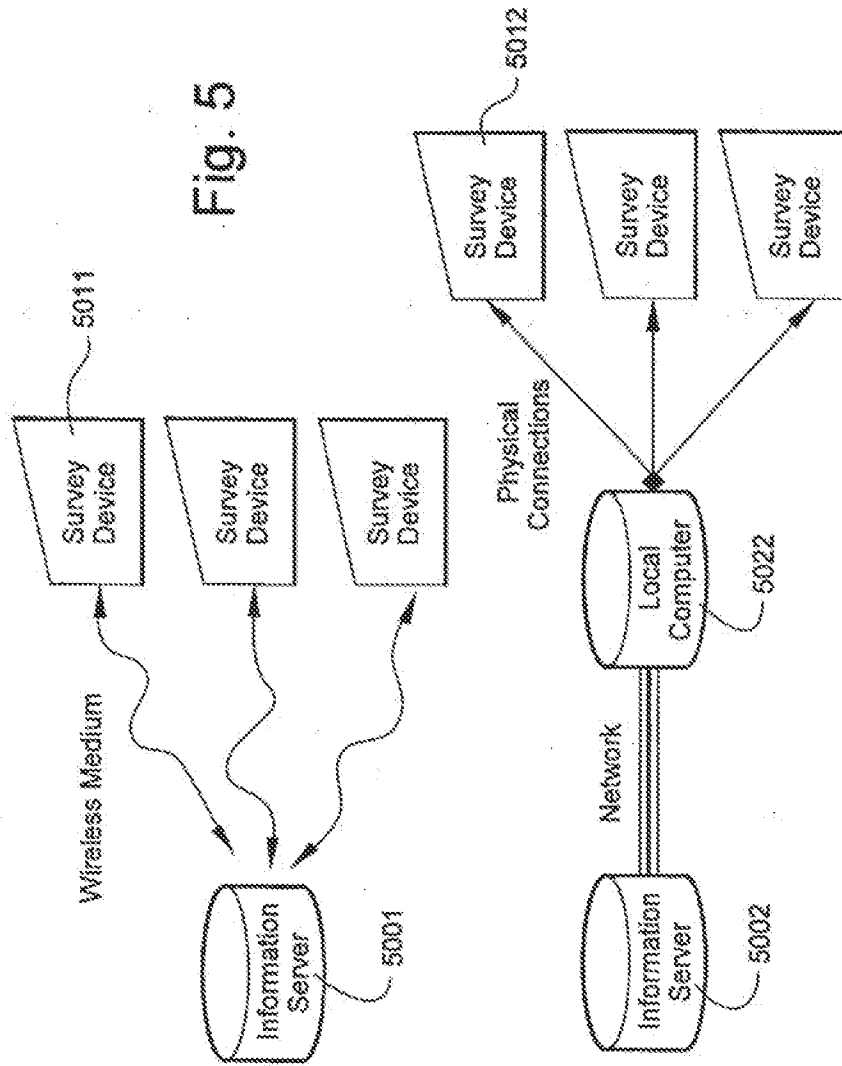


Fig. 4



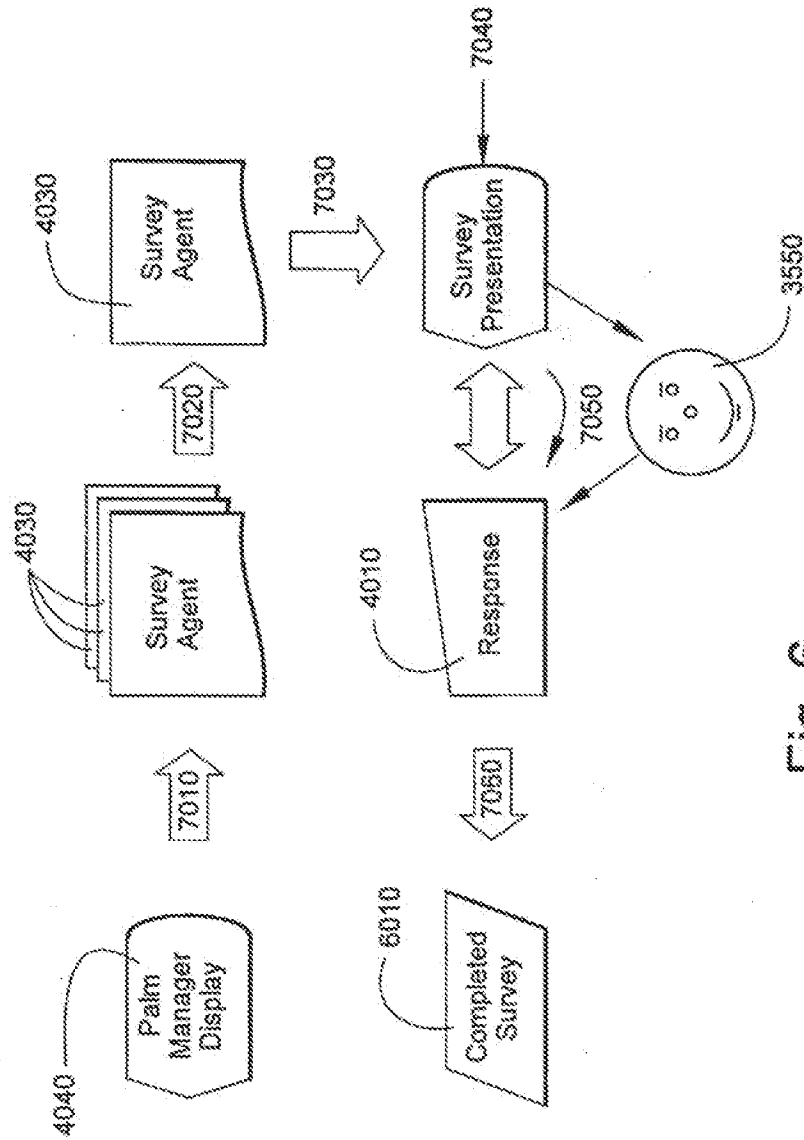


Fig. 6

Fig. 7

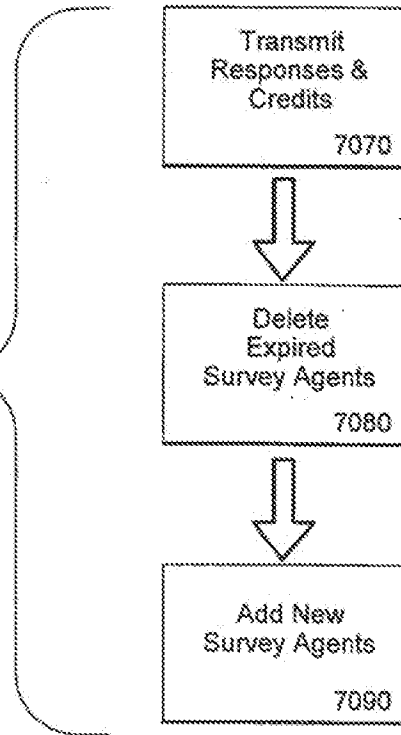
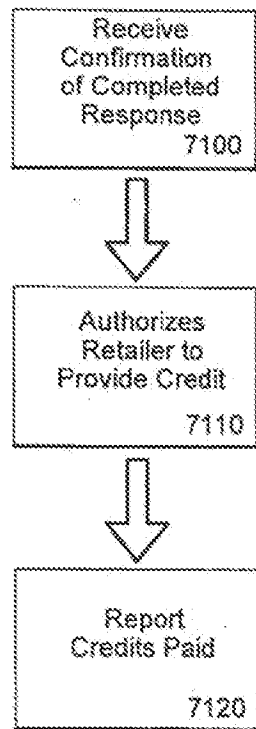


Fig. 8



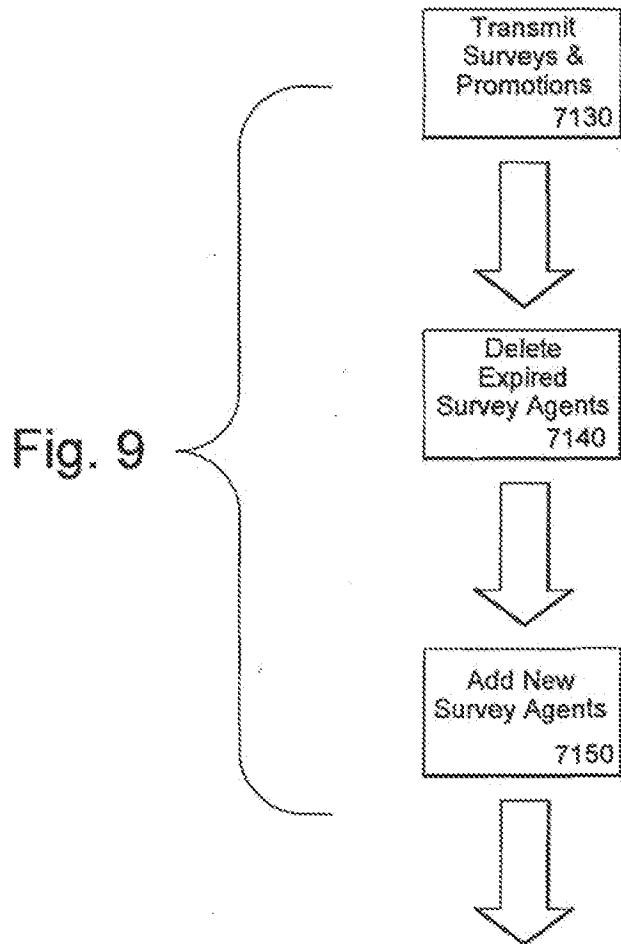


Fig. 10

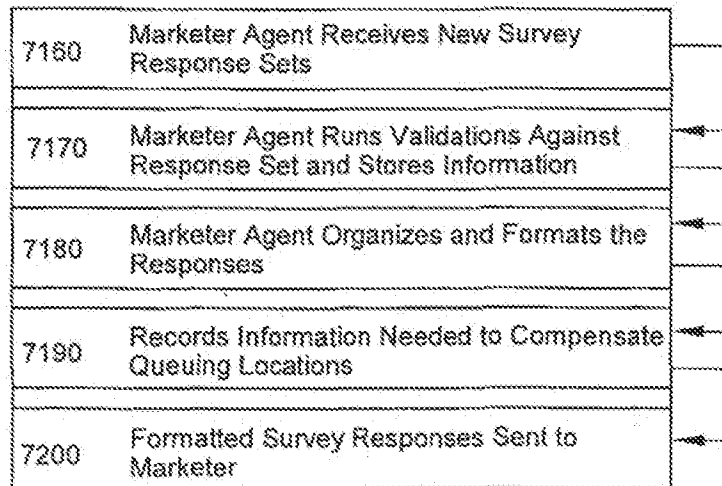
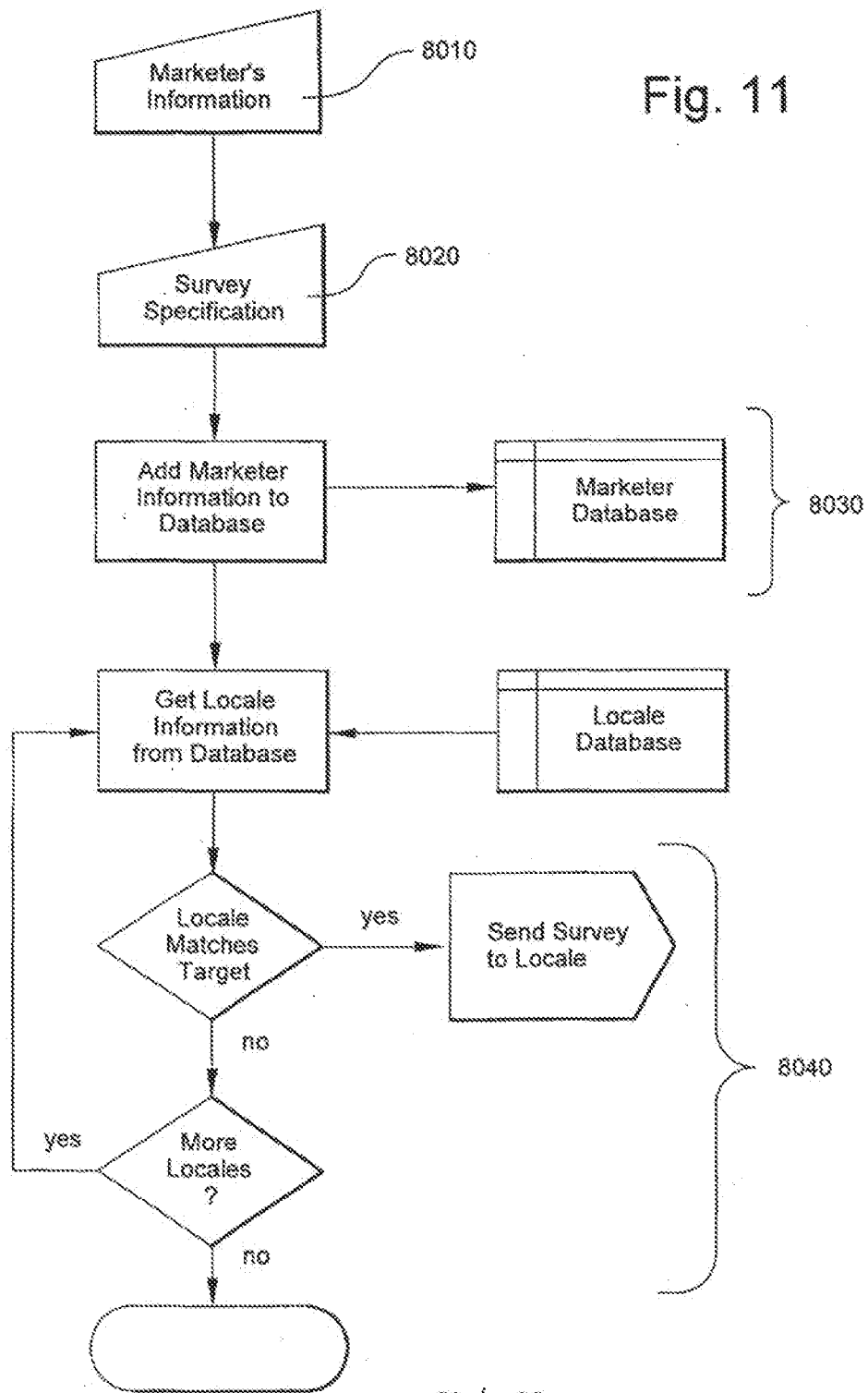


Fig. 11



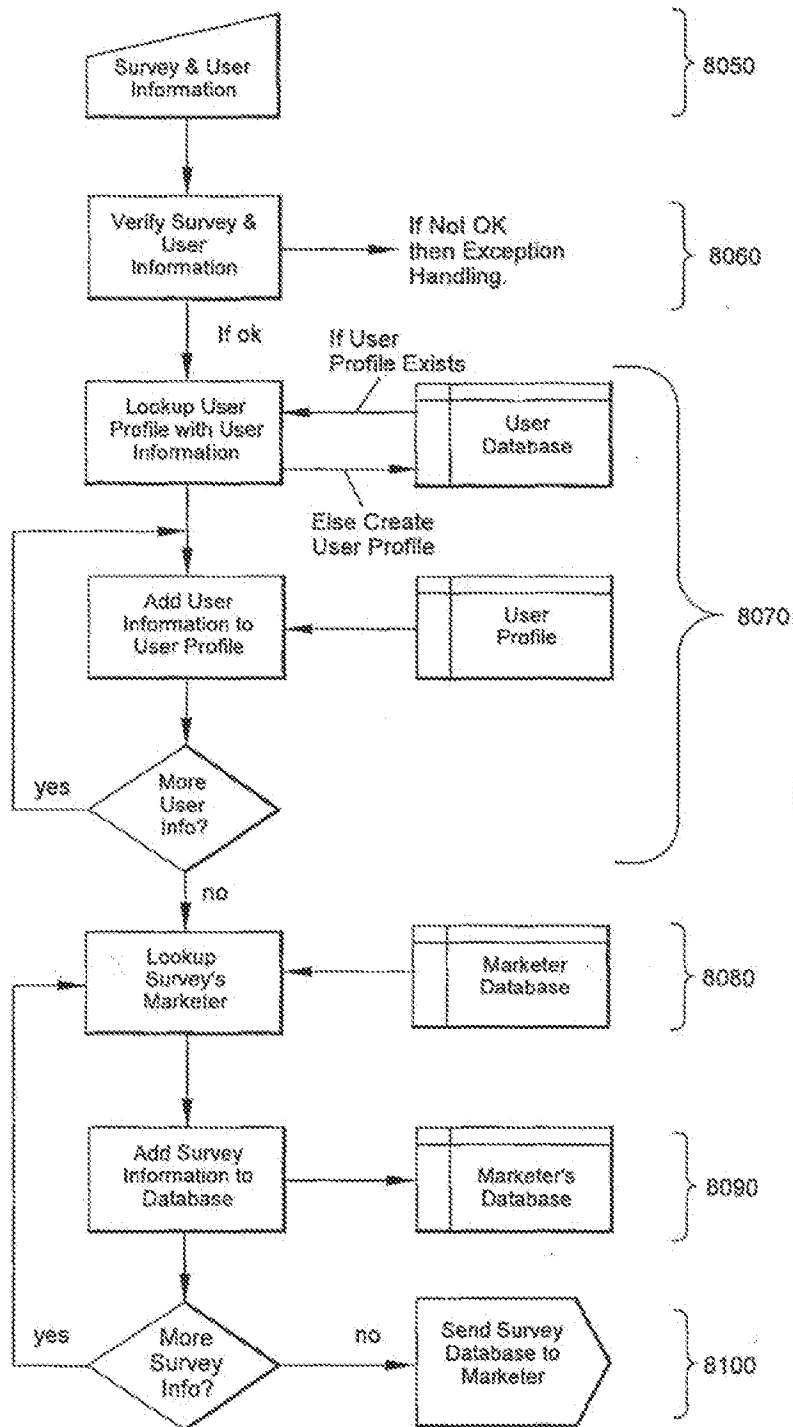


Fig. 12

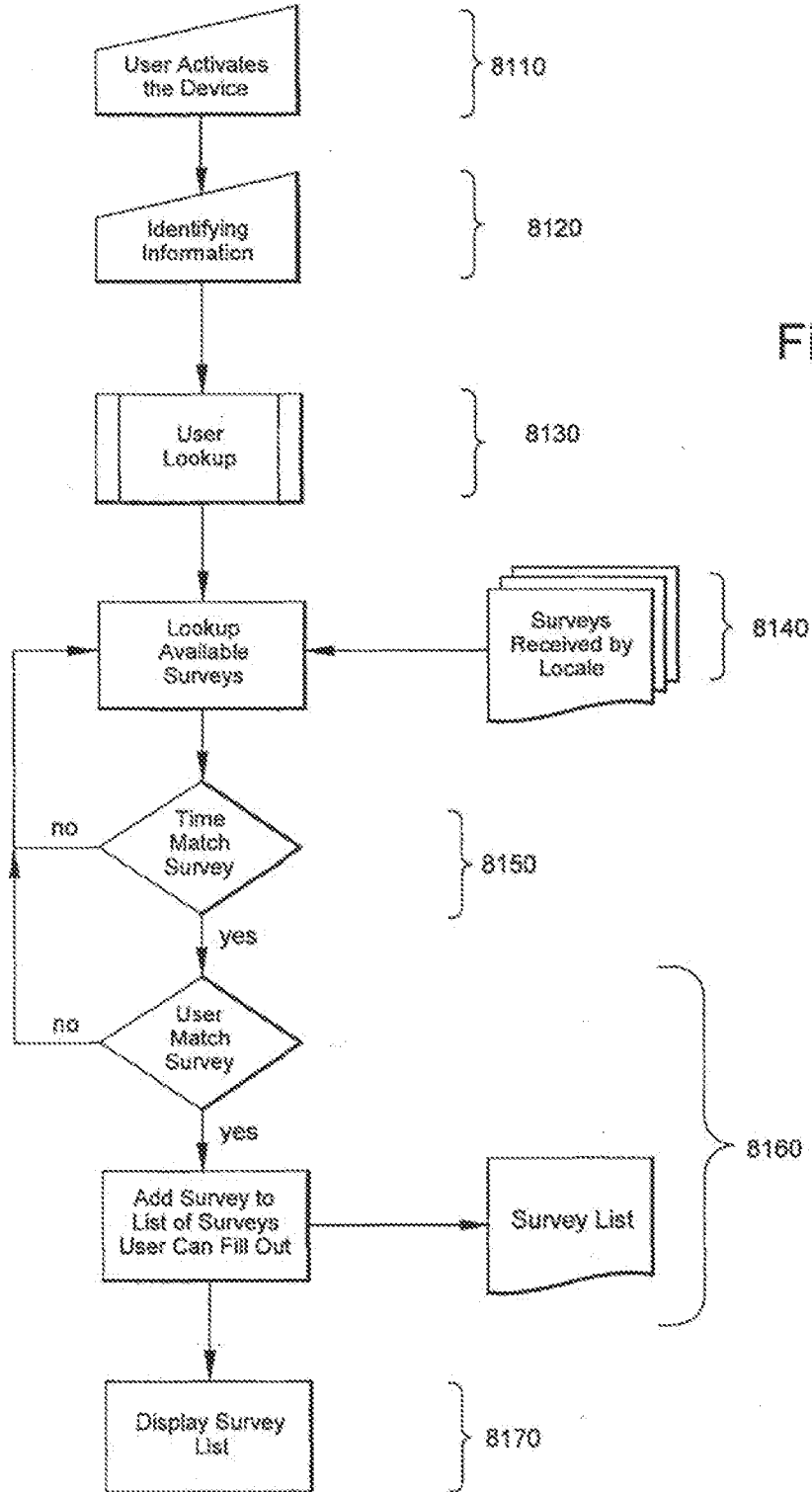


Fig. 13

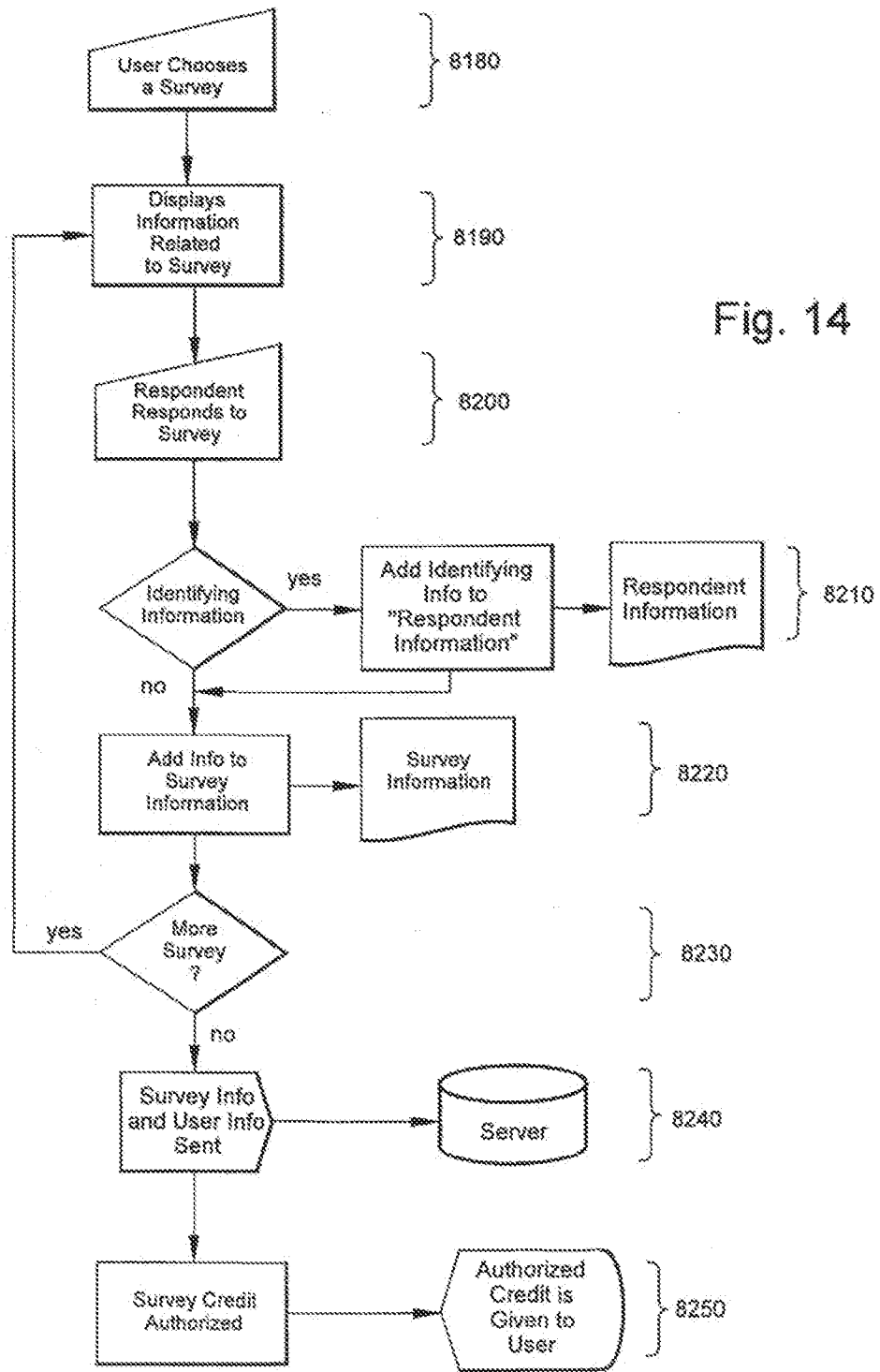


Fig. 14

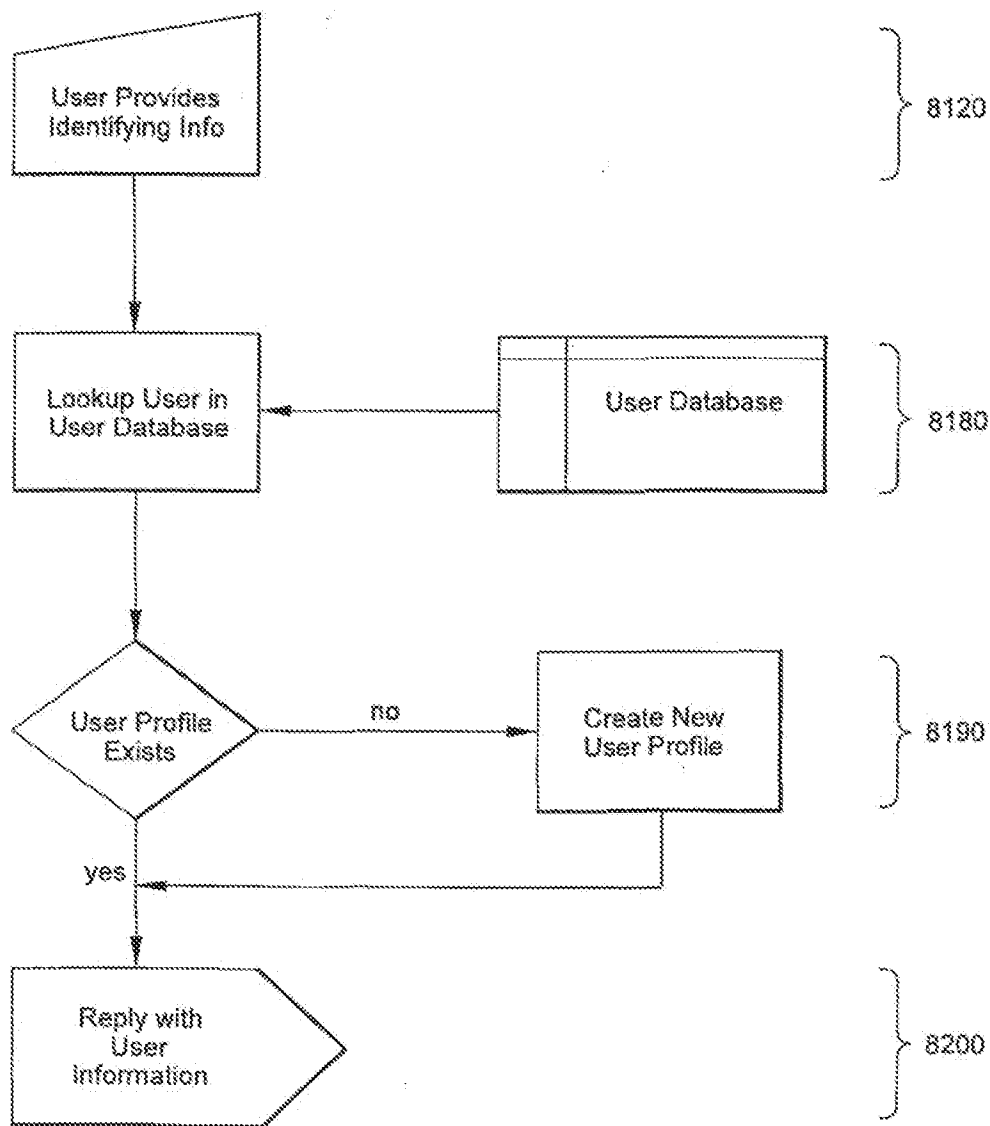


Fig. 15

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
5 July 2001 (05.07.2001)

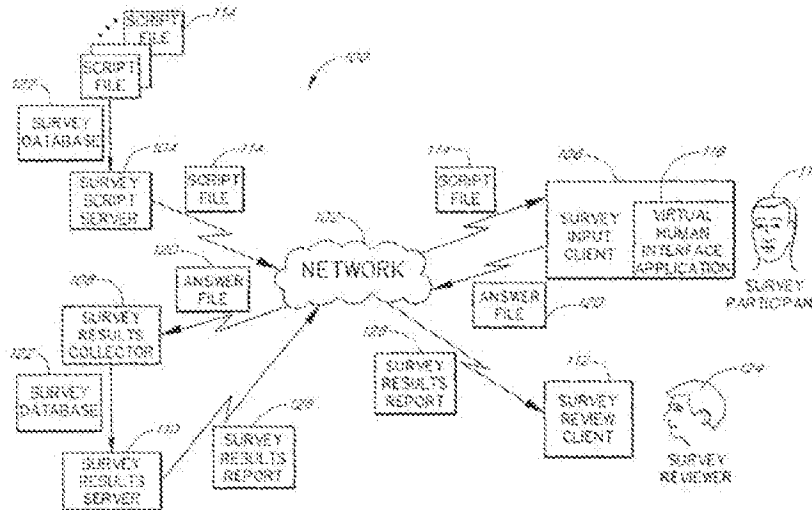
PCT

(10) International Publication Number
WO 01/48660 A1

- (51) International Patent Classification: G06F 17/60
- (74) Agent: ALTMAN, Daniel, E.; Knobbe, Martens, Olson and Bear, LLP, 620 Newport Center Drive, 16th Floor, Newport Beach, CA 92660 (US).
- (21) International Application Number: PCT/US00/34949
- (81) Designated States (national): AE, AG, AL, AM, AT, AU (utility model), AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, CZ (utility model), DE, DE (utility model), DK, DK (utility model), DM, DZ, EE, EE (utility model), ES, FI, FI (utility model), GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SE (utility model), SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW.
- (22) International Filing Date: 21 December 2000 (21.12.2000)
- (84) Designated States (regional): ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Date: 09/474,518 29 December 1999 (29.12.1999) US
- (71) Applicant: VIRTUAL PERSONALITIES, INC. [US/US]; 536 North Highland Avenue, Los Angeles, CA 90038-2020 (US).
- (72) Inventors: PLANTEC, Peter, M.; 1315 Summitridge Drive, Beverly Hills, CA 90210 (US). MAULDIN, Michael, L.; P.O. Box 509, Irwin, PA 15642 (US). ROMERO, Jeremy, Shay; 1045 N. Azusa, Space 167, Covina, CA 91722 (US). MCCBIDE, Aaron, J.; 7851 La Riviera Drive, Apt. 184, Sacramento, CA 95826 (US).
- Published: With international search report.

[Continued on next page]

(54) Title: VIRTUAL HUMAN INTERFACE FOR CONDUCTING SURVEYS



(57) Abstract: The present invention relates in general to electronic information gathering, and more particularly to the use of a virtual human interface to conduct surveys and to present survey results data in using text and voice in-/output for communicating with a user. A system and method is disclosed which automate the distribution of survey questions, the collection of survey data and the processing and formatting of survey results. A script file includes, among others, survey question data, response pattern data, advertising data, entertainment data, lobbying data and/or processing instructions. The system and method generate a representation of a character communicating data from the script file, and encourages and/or clarify responses from the user.

WO 01/48660 A1



----- Before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments.

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

VIRTUAL HUMAN INTERFACE
FOR CONDUCTING SURVEYSBackground of the Invention5 Field of the Invention

The present invention relates in general to electronic information gathering, and more particularly to the use of a virtual human interface to conduct surveys and collect and present survey results data.

Description of the Related Art

10 Marketing and planning of all sorts is often critically dependent on customer and/or public feedback. In the context of product planning, such information can avoid the devastating effect of introducing a product the public simply does not want, is not ready for, or even a product the public finds offensive. In another context, media producers, such as those producing television series, are constantly wary of the effect on the consuming public that might be created by particular plots, changes in plots or treatment of certain issues, roles or characters. In addition, producers need to be aware of viewer preferences for use in attracting advertising. In these and other areas, accurate and timely consumer feedback is critical.

15 Existing techniques for obtaining satisfactory feedback and formatting it for useful and meaningful review are extremely costly. That is because they typically require employing small armies of telemarketers, data entry clerks, interviewers, statisticians and/or other data collection teams, and still others to convert, format and make sense of the information collected.

20 Another problem with information gathering is the attention, concentration and understanding of the participants. For example, feedback received from survey participants may be unreliable, inaccurate or unhelpful if survey participants lose interest or become distracted while taking the survey. Survey participants soon become bored with survey questions, particularly when presented in large quantities, in printed form, or unclear or tedious language, and provide abrupt, confusing, careless and/or ill-considered feedback when required to respond.

25 Some methods of keeping participants interested involve extrinsic motivation, such as payment. Providing extrinsic motivation is often unsatisfactory, however, because the participant is still not genuinely interested in the survey. For that, intrinsic motivation is needed. It is known that human interviewers presenting live questions can sometimes create intrinsic motivation and can maintain a participant's focus and concentration for a longer period of time. The interaction itself provides the intrinsic motivational component needed. However, not only is it extremely expensive to pay and train live interviewers, but also live human interviewers themselves can become bored when asking the same questions over and over again and repetitively logging participants' answers. When this happens, live interviewers lose their effectiveness in holding the attention of participants.

30 Moreover, survey sponsors can rarely rely on the relevant consumers—even by enticing them with free gifts or cash payments—to travel to the survey sponsor or any other location to take a survey. Therefore, the work is typically done in focus groups, through door-to-door canvassing, through the use of paid diaries delivered by mail or

35

through telephone surveys. Consumers simply require utmost convenience to themselves in providing survey information. Thus, survey sponsors must also overcome the problem of survey delivery—that is, ensuring that the questions are made available at a location sufficiently convenient to participants. Extensive and expensive planning involving demographics and geographical regions is typically conducted, leading afterward to the dispatching of teams to varied locations, often nationwide.

Still another problem with traditional survey techniques relates to the long delay between the gathering of information and the conversion of the information into a form useful to the survey sponsors. Traditional data collection, data entry and tabulating processes commonly require labor intensive and error fraught keyboarding of data from survey forms. Entities conducting surveys often have to wait for results to be returned by mail before they can even begin to be processed.

What is needed is a system for gathering information from consumers which will hold the consumers' interest and avoid distraction, which will minimize the time and expense of gathering the information and making it useful and which will maximize convenience to consumers.

Summary of the Invention

One embodiment of the invention is a survey system, which comprises (1) a computer comprising (a) a computer readable medium; (b) a processor; and (c) a display; (2) a script comprising question data representing a question, the script stored in the computer readable medium; (3) character image data representing a character communicating information, the character image data stored in the computer readable medium; and (4) a user interface running on the processor, the user interface configured to access the script, the user interface configured to process the question data to communicate the question to a user, an image generator of the user interface configured to process the character image data to create a representation of the character on the display during the communication of the question, the user interface configured to receive and process response data representing a response provided by the user, the user interface configured to store in the computer readable medium answer data representing the response provided by the user. One aspect of the system further comprises (1) response pattern data representing a portion of a possible user response to the question; and (2) lobbying data representing lobbying information to persuade the user to respond to the question by providing a response that does not include the at least one portion of the response to the question, the user interface configured to compare the response data to the response pattern data to determine whether the response provided by the user includes the predetermined portion of a possible response and, if so, to process the lobbying data to communicate the lobbying information to the user without storing the answer data. That aspect may preferably further comprise expression data representing an expression of the character, the user interface configured to process the expression data to cause the image generator to process the character image data to change an expression of the representation of the character on the display during the communication of the lobbying information. In another aspect, the system further comprises (1) response pattern data representing a portion of a possible user response to the question; and (2) advertising data representing advertising information to inform the user about goods or services, the user interface configured to compare the response data to

the response pattern data to determine whether the response provided by the user includes the predetermined portion of a possible response and, if so, to process the advertising data to communicate the advertising information to the user. In still another aspect, the system further comprises (1) response pattern data representing a portion of a possible user response to the question; and (2) entertainment data representing an offer of an incentive or reward to the user for answering the question, the user interface configured to compare the response data to the response pattern data to determine whether the response provided by the user includes the predetermined portion of a possible response and, if so, to process the entertainment data to communicate or present the offer to the user. In yet another aspect, the system further comprises (1) first response pattern data representing a portion of a first possible user response to the question; (2) second response pattern data representing a portion of a second possible user response to the question; and (3) predetermined answer data, the user interface configured to compare the response data to the first response pattern data and to compare the response data to the second response pattern data, and, if either comparison results in a match, to store the predetermined answer data in the computer readable medium without storing the answer data. That aspect may preferably further comprise (1) third response pattern data representing a portion of a third possible user response to the question; and (2) second predetermined answer data, the user interface configured to compare the response data to the third response pattern data and, if that comparison results in a match, to store the second predetermined answer data in the computer readable medium without storing the answer data. In a different aspect, the system further comprises expression data representing an expression of the character, the user interface configured to process the expression data to cause the image generator to process the character image data to change an expression of the representation of the character on the display during the communication of the question. That aspect may preferably further comprise remote character image data residing on a computer readable storage medium operably connected to a second computer, the second computer in network contact with the first computer, the user interface configured to request and receive the remote character image data to cause the image generator to process the remote character image data to change an expression of the representation of the character on the display during the communication of the question. In a different aspect, the system further comprises a voice output module creating audio output to communicate the question to the user. That aspect may preferably be one wherein the user interface is configured to communicate the question by displaying text characters during a time when the voice output module communicates the question by creating audio output. Still a different aspect of the system further comprises (1) response pattern data representing a portion of a possible user response to the question; (2) remote lobbying data representing lobbying information to persuade the user to respond to the question by providing a response that does not include the at least one portion of the response to the question, the remote lobbying data residing on a computer readable storage medium operably connected to a second computer, the second computer in network contact with the first computer; and (3) a voice output module, the user interface configured to compare the response data to the response pattern data to determine whether the response provided by the user includes the predetermined portion of a possible response and, if so, to request and receive the remote lobbying data and to route data to the voice output module to create audio output representing the lobbying information. Another

aspect of the system further comprises a voice recognition module creating the response data from a spoken response provided by the user. An additional aspect of the system further comprises (1) an answer format value representing a predetermined format for answer data provided in response to the question; and (2) survey result instructions formatting the answer data in accordance with the answer format value to create a survey result report. That aspect
5 may preferably further comprise survey collection instructions, the survey collection instructions associating the stored answer data with second answer data provided by a second user in response to the question, the survey result instructions formatting the second answer data to create the survey result report. That aspect may preferably be one wherein the survey result instructions and the survey collection instructions are executed by a processor of a second computer. That aspect may preferably further comprise a third computer connected by a network to the second
10 computer, the second computer transmitting the survey result report to the third computer in response to a request for the survey result report. That aspect may preferably be one wherein the network is the internet, the survey result report is a web page, and an internet browsing application running on the third computer issues the request for the survey result report.

Another embodiment of the present invention is a survey system, which system comprises (1) question data
15 representing a question; (2) response pattern data representing at least one portion of a response to the question; (3) lobbying data representing lobbying information to persuade a user to respond to the question by providing a response that does not include the at least one portion of the response to the question; and (4) a user interface configured to process the question data to communicate the question to a user, the user interface configured to receive and process
20 response data representing a response provided by the user, the user interface configured to compare the response data to the response pattern data to determine whether the response provided by the user includes the at least one portion of the response to the question and, if so, to process the lobbying data to communicate the lobbying information to the user. One aspect of the system is one wherein the user interface is further configured to generate and store answer data representing the response provided by the user if the user interface determines that the
25 response provided by the user does not include the at least one portion of the response to the question. That aspect may preferably be one wherein the user interface is configured to transmit character image data to a display during the communication of the question and the lobbying information, the character image data rendered to represent a character communicating the question and the lobbying information.

Still another embodiment of the present invention is a system for questioning a survey participant. The system comprises (1) a computer readable storage; (2) a display; (3) a survey script stored on the computer readable
30 storage, the survey script comprising question data, expression data, and response processing instructions; and (4) a virtual human interface comprising an image generator, a script parser and a response parser, the script parser parsing the question data to present a question to a user, the image generator displaying on the display a representation of an animated character communicating information during the presenting of the question, the script parser parsing the
35 expression data to cause the image generator to change the representation of the character, the script parser parsing the response processing instructions to identify at least one portion of one possible response by the user to the

question and to associate predetermined responsive steps with the response, the response parser parsing input provided by the user to detect the portion of the one possible response and to perform the predetermined responsive steps, the predetermined responsive steps including storing in an answer file answer data representing the input provided by the user and storing in the answer file in association with the answer data a question identifier identifying the question. One aspect of the system is one wherein the image generator comprises software instructions stored on the computer readable storage, the script parser comprises software instructions stored on the computer readable storage, and the response parser comprises software instructions stored on the computer readable storage. That aspect may preferably further comprise (1) an answer format value representing a predetermined format for answer data provided in response to the question, the answer format value having a predetermined association with the question identifier; and (2) a survey report generator configured to parse the answer file for the question identifier and to retrieve the answer format value and to use the answer format value to format the answer data in a survey result report.

Yet another embodiment of the present invention is a method of conducting a survey. The method comprises the steps of (1) providing in a computer readable medium question data representing a question; (2) providing in the computer readable medium character image data representing a character communicating information; (3) communicating the question to a user by processing the question data; (4) displaying a representation of the character on a display during the communication of the question by processing the character image data; and (5) responding to a response to the question provided by a user by storing in the computer readable medium answer data representing the response. In one aspect, the method comprises the further steps of (1) providing in the computer readable medium response pattern data representing a portion of a possible user response to the question; (2) providing in the computer readable medium lobbying data representing lobbying information to persuade the user to respond to the question by providing a response that does not include the at least one portion of the response to the question; (3) comparing response data representing the response provided by the user to the response pattern data; and (4) communicating the lobbying information to the user by processing the lobbying data without storing the answer data if the response data includes the response pattern data. That aspect may preferably comprise the further steps of (1) providing in the computer readable medium expression data representing an expression of the character; and (2) processing the expression data to change an expression of the displayed character during the communication of the lobbying information. Another aspect may comprise the further steps of (1) providing in the computer readable medium response pattern data representing a portion of a possible user response to the question; (2) providing in the computer readable medium advertising data representing advertising information to inform the user about goods or services; (3) comparing response data representing the response provided by the user to the response pattern data; and (4) communicating the advertising information to the user by processing the advertising data if the response data includes the response pattern data. Still another aspect may preferably comprise the further steps of (1) providing in the computer readable medium response pattern data representing a portion of a possible user response to the question; (2) providing in the computer readable medium entertainment data representing an offer of an incentive or reward to

the user for answering the question; (3) comparing response data representing the response provided by the user to the response pattern data; and (4) communicating or presenting the entertainment information to the user by processing the entertainment data if the response data includes the response pattern data. Yet another aspect may preferably comprise the further steps of (1) providing in the computer readable medium first response pattern data representing a portion of a first possible user response to the question; (2) providing in the computer readable medium second response pattern data representing a portion of a second possible user response to the question; (3) providing in the computer readable medium predetermined answer data; (4) comparing response data representing the response provided by the user to the first response pattern data; (5) comparing the response data representing the response provided by the user to the second response pattern data; and (6) storing in the computer readable medium the predetermined answer data without storing the answer data if either comparison results in a match. An additional aspect may preferably comprise the further steps of (1) providing in the computer readable medium expression data representing an expression of the character; (2) providing in the computer readable medium response pattern data representing a portion of a possible user response to the question, the response pattern data having a predetermined relationship with the expression data; (3) comparing response data representing the response provided by the user to the response pattern data; and (4) changing an expression of the displayed character by processing the expression data if the comparison results in a match. That aspect may preferably be one wherein the expression data resides on a remote computer readable medium and wherein the processing of the expression data includes requesting the expression data from the remote computer readable medium and transmitting the expression data from the remote computer readable medium. An aspect described above comprising advertising data may preferably be one wherein the advertising data resides on a remote computer readable medium and wherein the processing of the advertising data includes requesting the advertising data from the remote computer readable medium and transmitting the advertising data from the remote computer readable medium. An aspect described above comprising advertising data may preferably be one wherein the advertising data includes voice data and references to expression data, the method comprising the further steps of, and the aspect may preferably further comprise the steps of (1) providing in the computer readable medium expression data representing an expression of the character; (2) communicating the advertising data as voice output by processing the voice data; and (3) changing an expression of the displayed character by processing expression data referenced by the advertising data if the comparison results in a match.

Another embodiment of the present invention is a method for processing survey data. The method comprises the steps of (1) providing in a computer readable medium question data representing a question; (2) providing in the computer readable medium an answer format value representing a predetermined format for answer data provided in responses to the question; (3) associating a question identifier with the question data; (4) associating the answer format value with the question identifier; (5) receiving an answer file in the computer readable medium, the answer file including answer data representing a response provided by a user to the question, and the answer file including the question identifier; (6) parsing the answer file for the question identifier; (7) identifying the answer format value using the association with the question identifier; and (8) formatting the answer data in accordance with the predetermined

format for answer data. One aspect of the method is one wherein the steps described therein are represented by software instructions stored in a computer readable medium.

5 Still another embodiment of the present invention is a survey system, which comprises (1) means for representing a question in a computer readable medium; (2) means for representing in the computer readable medium a portion of a first possible user response to the question; (3) means for communicating the question to a user; (4) means for generating a representation of a character communicating the question during the communication of the question; (5) means for receiving from the user a representation of a response provided by the user to the question; (6) means for performing a first comparison, the first comparison comprising comparing the representation of the response to the representation of the portion of the first possible user response; and (7) means for storing an answer representing the response if the first comparison results in a match. In one aspect, the system further comprises (1) means for representing in the computer readable medium a portion of a second possible user response to the question; (2) means for representing in the computer readable medium lobbying information to persuade a user to respond to the question by providing a response that does not include the portion of the second possible user response to the question; (3) means for performing a second comparison, the second comparison comprising comparing the representation of the response to the representation of the portion of the second possible user response; and (4) means for communicating the lobbying information to the user without storing the answer data if the second comparison results in a match. In another aspect, the system further comprises means for changing an expression of the representation of the character during the communication of the question. In still another aspect, the system further comprises (1) means for representing advertising information in a computer readable medium, the advertising information informing a user about goods or services; and (2) means for communicating the advertising information to the user if the first comparison results in a match. In yet another aspect, the system further comprises (1) means for representing entertainment information in a computer readable medium, the entertainment information rewarding or offering to reward the user for providing a response to the question; and (2) means for communicating or presenting the entertainment information to the user if the first comparison results in a match.

25 An additional embodiment of the present invention is a survey system, which comprises (1) a network; (2) a server coupled to the network, the server comprising a server processor and a server computer readable storage; (3) a first client coupled to the network, the first client comprising a first processor, a first computer readable storage, and a first display; (4) a script file stored on the server computer readable storage, the script file containing survey question data, the server computer configured to respond to a request for the script file from the first client by transmitting the script file to the first client; and (5) a virtual human interface stored on the first computer readable storage, the virtual human interface configured to display a representation of a human face on the first display, the virtual human interface configured to parse the script file and present the survey question data to a user, the virtual human interface configured to change the representation of the human face during the presentation of the survey question data, and the virtual human interface configured to record a response of the user in an answer file, the virtual

human interface configured to transmit the answer file to the server computer, the server computer configured to process the answer file and at least one additional answer file to generate a report including formatted answer data.

Brief Description of the Drawings

5 Figure 1 is a block diagram of one embodiment of a survey system in accordance with the present invention.

Figure 2 illustrates one embodiment of a survey input client in accordance with the present invention.

Figure 3 illustrates one embodiment of a survey server in accordance with the present invention.

Figure 4 illustrates one embodiment of a survey results report generated by a survey results server.

10 Figure 5 illustrates a representation of aspects of a virtual human interface application in accordance with one embodiment of the present invention.

Figures 5A-5D illustrate representations of images of a character communicating information with expression and emotion in one embodiment of the invention.

Figure 5E illustrates steps performed in one embodiment of the present invention to lead an embodiment of a Verbot.

15 Figure 5F illustrates steps performed in one embodiment of the present invention to generate character images in one embodiment of a Verbot.

Figure 6 is a flow chart illustrating a representation of the overall operation of a survey system in accordance with one embodiment of the present invention.

20 Figures 7A through 7E are a series of flow charts showing the operation of one embodiment of a survey system in accordance with the present invention.

Figure 8 is a flow chart showing the operation of one embodiment of a survey results collector in accordance with the present invention.

Figure 9 is a flow chart showing the operation of one embodiment of a survey results server in accordance with the present invention.

25 Detailed Description of the Preferred Embodiments

Figure 1 is a block diagram illustrating one embodiment of a survey system 100 in accordance with the present invention. In that embodiment, the survey system 100 includes a network 102 in communication with a survey script server 104, a survey input client 106 having a virtual human interface application 116, a survey results collector 108, a survey results server 110, and a survey review client 112. The network 102 may be any type of computer network, public or private, LAN and/or WAN, wire-based and/or wireless, and may use one or a combination of network protocols such as Ethernet or token ring. In the preferred embodiment, however, the network 102 is the Internet.

30

The survey script server 104 maintains a plurality of script files 114. The script files 114 provide scripting for the virtual human interface application 116 of the survey input client 106. More particularly, the script files 114

include question data, which form the basis of questions ultimately posed to survey participants 118. Preferably, the script files 114 stored in the survey script server 104 are updated periodically.

Overview of System Operation

5 Generally, a virtual human interface is a bridge between humans and technology built from artificial intelligence and natural language components. Because it recognizes and understands human communication on one side and technology communication on the other, the virtual human interface can translate meaning between the two. In the survey context, the virtual human interface can ask questions in human terms, receive responses provided in human terms, translate the responses into machine terms, perform machine-based processing on the machine terms, and then translate the machine-processed information back into human terms for communication to a human.

10 In operation, the virtual human interface application 116 of the survey input client 105 interacts with a survey participant 118 through a virtual human user interface including a Verbot™. Upon initial activation, the Verbot™ requests permission from the survey participant 118 to periodically download script files 114 from the survey script server 104. If the survey participant 118 grants permission, then the virtual human interface application 116 directs the survey input client 106 to periodically request script files 114 from the survey script server 104
15 automatically and engage the survey participant 118 in a conversation including embedded survey question content. If the survey participant 118 does not grant permission, then the Verbot™ periodically seeks permission to download a script file 114, and the survey input client 106 is directed to request a script file 114 from the survey script server 104 only when the survey participant 118 authorizes the request.

20 In either of the situations described above, the virtual human interface application 116 occasionally directs the survey input client 106 to send a request for a script file 114 to the survey script server 104 over the network 102. When the survey script server 104 receives such a request, it sends a script file 114 to the survey input client 106 over the network 102.

25 In some embodiments, the survey script server 104 can optionally maintain demographic, geographic or other information regarding users, including the ability to identify users who do not participate in surveys regularly. Nevertheless, individual survey participants 118 will preferably remain anonymous, being only identified by their sample selection criteria status.

30 After downloading the requested script file 114, the Verbot™ engages the survey participant 118 in a conversation including embedded survey questions as scripted in the most recently acquired script file 114. During the survey, the survey participant 118 responds to the scripted survey questions asked by the Verbot™. The Verbot™ stores the responses of the survey participant 118 in an answer file 120 on the survey input client 105.

Simultaneously, the Verbot™ can respond to the survey participant 118 verbally with encouragement, motivating comments, and the like. In some embodiments, the Verbot™ can offer to play a game with the survey participant 118 or give the survey participant 118 some other reward, such as coveted information.

35 Upon completion of the survey, the virtual human interface application 116 directs the survey input client 106 to send the answer file 120 generated during the survey to the survey results collector 108 over the network

102. The survey results collector 108 receives the answer file 120 from the survey input client 106 and extracts survey result data from the answer file 120. The survey results collector 108 then adds the survey result data to a survey database 122. The survey database 122 includes survey result data obtained from many different survey participants 118, which may number in the hundreds, thousands or millions. Because the survey results collector 108
5 may be receiving answer files 120 frequently, it may frequently update the survey database 122.

The survey review client 112 allows a survey reviewer 124 to review the results of any survey for which the survey reviewer 124 is authorized. In some embodiments, a particular survey results server 110 may be compiling results for several unrelated survey sponsors at the same time. Because survey results data can be very time-sensitive and valuable, each category of results data is preferably protected from unauthorized viewing.

10 When an authorized survey reviewer 124 wishes to review the results of a survey, the survey reviewer 124 uses the survey review client 112 to access the network 102. In a preferred embodiment, the survey review client 112 accesses a website hosted by the survey results server 110. Preferably, the survey reviewer 124 uses a standard web browser to access the survey results server 110. Thus, the survey reviewer 124 can access the survey results data from anywhere in the world, even using an Internet-enabled wireless palm-top computer or cellular
15 telephone.

In one embodiment, the survey results server 110 generates and transmits to the survey reviewer 124 a web page having a hyperlink associated with each sponsor in the survey database 122. The survey reviewer 124 selects a sponsor by clicking on a sponsor hyperlink, and the survey results server 110 generates another web page having USER and PASSWORD fields and an ENTER button. The survey reviewer 124 fills in the fields and clicks the ENTER
20 button. The survey results server 110 authenticates the survey reviewer 124 by querying the PASSWORD_TABLE. In an alternative embodiment, the authentication can be performed through the use of browser cookies. After authenticating the survey reviewer 124, the survey results server 110 generates and transmits a web page, which presents a list of survey hyperlinks (each labeled by survey name), one for each survey in the survey database 122 for the selected sponsor. When the survey reviewer 124 selects a survey hyperlink, the survey results server 110
25 generates and transmits a survey result report 126 web page, which presents the results of the selected survey.

Because the survey results server 110 generates web pages dynamically from the survey database 122, the survey results report 126 advantageously reflects up-to-the-minute survey results data. Moreover, the survey reviewer 124 may advantageously receive virtually instant updates by selecting the "Refresh Page" option on the browser of the survey review client 112.

30 Survey Database

In one embodiment, the survey database 122 organizes data in a hierarchical fashion. While many different organizations may be implemented to store and access survey-related data, one embodiment uses the following database tables:

NAME OF TABLE	DESCRIPTION
SPONSOR_TABLE	<p>The SPONSOR_TABLE includes entries (or records) which hold information about each of potentially many survey sponsors. Each record includes a unique SPONSOR_ID field assigned to a particular sponsor. Each record also includes additional information (additional database fields) relating to the survey sponsor such as, for example, name, contact name, address, phone number, fax number, e-mail, web site, billing information, number of surveys conducted, number of surveys pending, etc. Those of ordinary skill will appreciate that each field is of the appropriate type (e.g., string, integer, etc.) and is an appropriate length (e.g., 512 characters, 4 bytes, etc.).</p>
PASSWORD_TABLE	<p>The PASSWORD_TABLE holds user and password data which facilitate secure access to survey information. Each record in the PASSWORD_TABLE includes a SPONSOR_ID field, which provides a cross-reference to a sponsor in the SPONSOR_TABLE. Each record also includes a USER field and a PASSWORD field. Thus, for any given SPONSOR_ID, the PASSWORD_TABLE may be queried to determine all valid user/password combinations, which can, in turn, be used to validate an entered user name and password combination. The PASSWORD_TABLE allows a single survey sponsor to have multiple user/password combinations with which to access, for example, survey result data, or other information in the survey database.</p>
SURVEY_TABLE	<p>The SURVEY_TABLE includes entries (or records) which keep track of information about each survey. Each record includes a unique SURVEY_ID field assigned to a particular survey. Each record (and thus each survey) is cross-referenced to one or more survey sponsors by a SPONSOR_ID which matches the same-named field in the SPONSOR_TABLE. Thus, by querying the SURVEY_TABLE using a SPONSOR_ID value, all surveys for any sponsor may be easily located. Each record in the SURVEY_TABLE also includes additional information (additional fields) about each survey such as, for example, name/title of survey, description of subject matter, number of questions, date created,</p>

NAME OF TABLE	DESCRIPTION
	geographic concentration, path to retrieve corresponding script file(s), full text of script file, URL for related survey results web page, number of responding participants, number of refusing participants, etc.
QUESTION_TABLE	The QUESTION_TABLE includes entries (records) for each question in any survey. Each record thus includes a unique QUESTION_ID field which uniquely references a particular question. Each record is cross-referenced to one or more surveys by including a SURVEY_ID field which matches the SURVEY_ID field in the SURVEY_TABLE. Thus, by querying the QUESTION_TABLE using a SURVEY_ID value, all questions in the survey may be easily located. Each record in the QUESTION_TABLE also includes an ANSWER_FORMAT field which holds a value indicating how answers for the question should be processed for presentation to a survey reviewer (e.g., "top5and%" indicates that the five responses most frequently given should be listed along with the percentage of users providing the respective response; "listall" indicates that all answers provided be listed sequentially). Each record in the QUESTION_TABLE includes additional information (fields) about each question such as, for example, sequential question number, text of question, etc.
ANSWER_TABLE	The ANSWER_TABLE includes entries (records) for each answer recorded for any question in a survey. Each record in the ANSWER_TABLE includes a SURVEY_ID field identifying the particular survey for which the answer was provided. Each record also includes a QUESTION_ID field which matches the QUESTION_ID field in the QUESTION_TABLE. Thus, by using a QUESTION_ID value, the ANSWER_TABLE may be easily queried to locate each separate answer provided for a particular question. Also, each record in the ANSWER_TABLE includes a PARTICIPANT_ID field which matches the PARTICIPANT_ID field in the PARTICIPANT_TABLE. Using the PARTICIPANT_ID value to query the ANSWER_TABLE, all answers provided by a particular survey participant may be easily located. Each record of the ANSWER_TABLE includes additional information (fields) about each answer, including the answer

NAME OF TABLE	DESCRIPTION
	data, whether text, numeric or otherwise, provided by the participant, the date the answer was recorded, etc.
ANSWER_FORMAT_TABLE	The ANSWER_FORMAT_TABLE includes records which correspond to a type of format to apply to a group of answers to present the results. An ANSWER_FORMAT field provides the name of an available answer format, and an ANSWER_FORMAT_DESCRIPTION field provides a description of the formatting associated with an answer format. For example, one record may have "top5and%" assigned to the ANSWER_FORMAT field, and, in the ANSWER_FORMAT_DESCRIPTION field, the record may have the text "choose the five responses most frequently given and show the percentage of users providing the respective response." This table may be used, for example, in a survey design module to retrieve descriptions of all available answer formats, display them to a survey designer, allow a survey designer to choose one, and then provide the corresponding answer format name.
PARTICIPANT_TABLE	The PARTICIPANT_TABLE includes entries for each survey participant who has provided answer data for any survey. Each record of the PARTICIPANT_TABLE includes a PARTICIPANT_ID field uniquely identifying a particular participant. Each record of the PARTICIPANT_TABLE includes additional information for each participant such as, for example, geographic location, computer configuration, time to complete survey, average time to complete survey, number of surveys completed, etc.

In one embodiment of the invention, a survey administrator populates the survey database 122 with information about each survey sponsor that is conducting surveys. A survey database management application permits the survey administrator to interact with the survey database 122 to, for example, browse the existing records in any of the tables, enter new data and create new records in any of the tables, modify data in any existing record in any table, and also delete any record data in any of the tables. Such database management applications are common and supported by existing database applications such as, for example, Microsoft Access, Oracle, Sybase and FoxBase. These and other database applications provide extensive database management application design tools,

simplifying the design of database management applications, and those of ordinary skill understand well how to use the design tools to construct and operate such database management applications. Thus, the present invention is not limited by a particular survey database management application.

5 As is well known in the field of database management applications, the survey database management application permits a survey administrator to select a database table (i.e., SPONSOR_TABLE, PASSWORD_TABLE, SURVEY_TABLE, etc.) for operations and to select the type of operation (e.g., browse records, create new record, modify records, delete records). While the browse, modify and delete functions may retrieve all records in the selected table and allow the administrator to scroll through all of them, possibly to choose one for modification or deletion, the survey database management application also supports query operations. By providing data for one or more fields and
10 issuing a query command, the survey administrator can retrieve a subset of records (or recordset) in the selected table for browsing, modification or deletion.

Thus, to enter information for a new survey sponsor using the survey database management application, the survey administrator selects the SPONSOR_TABLE for operations and chooses a create record option. The survey database management application is designed to then query the SPONSOR_TABLE for all records, determine the
15 highest existing value for SPONSOR_ID in any of the fields, increment that value by one and automatically assign it to the SPONSOR_ID field for the new record to be created. Such technique for selecting unique identification values is well known in the art. Next, the survey database management application prompts the survey administrator to fill in text fields on a computer screen, which fields correspond to fields comprising a SPONSOR_TABLE record. The survey administrator enters information about a sponsor such as, for example, the name of the sponsor (e.g., "XYZ
20 Productions, Inc."), contact name, address, phone number, fax number, etc. When the fields are filled in, the survey administrator invokes a CREATE RECORD command, causing a new record to be created for the SPONSOR_TABLE.

Those of ordinary skill will appreciate that, using the same process, the survey administrator can create new records for any of the tables in the survey database. Accordingly, the survey administrator may select the
25 PASSWORD_TABLE and the create record option to enter password data for a sponsor. The survey database management application is further designed to provide pull-down menus for fields in a record which are cross-reference fields, that is, fields that are designed to represent a relationship with one or more data records in another table. Such functionality, is, again, well known in the art, and the present invention is not limited by any design for pull-down menus reflecting cross-reference field values. Thus, because the SPONSOR_ID field of each
30 PASSWORD_TABLE record is to be cross-referenced with a SPONSOR_ID value in one of the SPONSOR_TABLE records, the survey database management application presents a pull-down menu next to a field corresponding to the SPONSOR_ID field for the password record. When the administrator activates the pull-down menu, it presents a list of the names of each of the sponsors, those names having been extracted from the SPONSOR_TABLE records. When the administrator selects one of the names, the survey database management application enters the corresponding SPONSOR_ID in the SPONSOR_ID field of the new password record. The administrator then enters a user name and

a password in fields corresponding to the USER and PASSWORD fields and invokes a CREATE RECORD command to cause the new record to be added to the PASSWORD_TABLE.

Those of ordinary skill in the art will appreciate that, with sponsor information and password information entered into the survey database, a limited survey database management application can be provided to sponsors to allow limited access to the survey database. It will be understood that the limited survey database management application can use the PASSWORD_TABLE to authenticate any sponsor before providing limited access to the survey database. With limited access, a sponsor may advantageously directly provide information about particular surveys and questions, and, also advantageously, may even change password data or add new user and password data for additional persons. As will further be appreciated, the limited survey database management application can restrict access to only the data that corresponds to the sponsor's SPONSOR_ID (which, in one embodiment, the sponsor cannot change) or data cross-referenced thereto. Thus, many sponsors can access the survey database with no capability to view or alter data except that associated with the respective sponsor.

Server and Client Configuration

Figure 2 illustrates one embodiment of a survey input client 106 in accordance with the present invention. In the illustrated embodiment, the survey input client 106 comprises a personal computer 200, a monitor 202, and a microphone 204 in addition to a mouse, keyboard and standard I/O ports (not shown). The personal computer 200 also includes a 400 megahertz (MHz) processor, 128 megabytes (MB) of random access memory, a 2 gigabyte (GB) hard drive, a 56 kilobits-per-second (kbps) modem or NIC (network interface card), a duplex sound card, and a video card. The invention, however, does not require all of these components. It will be appreciated by one of ordinary skill in the art that the personal computer 200 of the survey input client 106 can be any of a number of general purpose computers—whether desktop, laptop or palm-top—using one or more microprocessors, such as a Pentium, Pentium II, or Pentium III processor, or a K6 or Athlon processor, a MIPS processor, a Power PC processor or an ALPHA processor. The personal computer 200 of the survey input client 106 can also be a cellular device for internet access. As will be appreciated, the present invention is not limited by any type of processor—and may be used with a processor running at less than 400 MHz—and is also not limited by any particular hard disk drive, memory, sound card or video card.

The personal computer 200 of the illustrated embodiment also includes operating system and application software, such as Microsoft Windows 95, Microsoft Internet Explorer 5.0, voice recognition software, voice synthesis software, a Verbot™ application, such as, for example, Sylvie™ version 3.04 available from Virtual Personalities, Inc., and a virtual human interface application 116. It will be appreciated by those of ordinary skill in the art that the present invention could use other operating system software, such as UNIX, LINUX, OS/2, BE, System 7, Solaris, Mac OS or others. Likewise, the present invention is not limited by particular Internet communication software and, thus, common alternatives, such as Netcape Communicator, Mosaic, Opera, or any of a number of small screen cellular browsers, may be used.

In one embodiment, the invention comprises a framework of interfaced software modules, which may retrieve, process, create, format and transmit certain data. In a preferred embodiment, aspects of the invention are controlled and facilitated by the virtual human interface application module that directs certain processes to be carried out by other modules, including a Verbot™ application module (such as, for example, Sylvie™ version 3.04) and voice
5 recognition and voice output modules. These modules, in turn, interact with other software modules, such as services provided by the operating system or such as Internet connection, communication and transmission functions provided by an Internet browser module. Thus, in the preferred embodiment, the modules are generally comprised of software instructions executable by a microprocessor.

As used herein, the word "module" refers not only to logic coded as a collection of software instructions, but also refers to logic embodied in hardware or firmware. In the software context, a module may have entry and exit
10 points and may be coded in a high level language such as C, C++, Java, or Pascal, or may be coded in machine or assembler language. Software modules may be compiled and linked into an executable program or installed in a Dynamic Link Library (DLL). Software modules may also be coded in an interpretive language, such as BASIC. Software modules may be callable from other modules, may be nested within other modules, and/or may be invoked in
15 response to a detected event or interrupt. Instructions of software modules may be coded into firmware, such as an EPROM. In the hardware context, modules may be comprised of connected logic units, such as gates and flip-flops, and/or may be comprised of programmable units, such as programmable gate arrays.

A computer configured similarly to the personal computer 200 can be used to carry out the processes of the survey review client 112. Generally and advantageously, any general-purpose computer configured to browse the
20 Internet may be used as the survey review client 112.

Figure 3 illustrates one embodiment of a survey server 300 in accordance with the present invention. The survey server 300 preferably comprises a personal computer 302 including a 400 MHz processor, a 128 MB random access memory, a 2 GB hard drive, and a high speed network connection. As will be appreciated by those of ordinary skill in the art, the personal computer 302 can run multiple application software programs simultaneously. Thus, in a
25 preferred embodiment, the personal computer 302 hosts the processes for the survey script server 104, the survey results collector 108, and the survey results server 110. Alternatively, separate computers could host the processes of the survey script server 104, the survey results collector 108, and the survey results server 110.

The personal computer 302 runs operating system software, for example, Microsoft Windows NT, which, as those of ordinary skill in the art understand, is multi-tasking, multi-threading, and preemptive. The personal computer
30 302 is not, however, limited by any of the services of Microsoft Windows NT, and can use any of the other operating systems mentioned above.

The personal computer 302 also runs application program software. In particular, database functions are provided by Microsoft Access database software, and Internet server functions are provided by Microsoft Internet Information Server. Other database software, such as, for example, Oracle, Sybase or FoxBase, capable of storing
35 data based on relationships between data items and capable of performing queries to identify and retrieve the data

may be used. Likewise, other internet server software, such as, for example, Apache, could be used to facilitate network-based communication with potentially large numbers of users, to serve HTML-compliant pages upon request, to process scripts, such as, for example, PERL scripts referenced in HTML page files using server side includes (SSIs), and to transfer data and files using other Internet protocols, such as FTP. Thus, the present invention is not limited by any database or Internet server application software.

In an on-line embodiment, the personal computer 302 includes a virtual human interface application 116, and a Verbot™ application, such as the on-line Sapphire™ class Verbot™ available from Virtual Personalities, Inc. In this embodiment, the Verbot™ can be embedded into the survey sponsor's server environment, such as an Internet server, and accessed by the survey participant 118 using any suitable network access application, such as a web browser. As described in more detail below, Java applets processing user input and script files and also controlling the transmission of voice and animation media may be used to implement the on-line Sapphire™ class Verbot™.

The operation of the virtual human interface application 116 in the on-line embodiment is similar to the operation of the virtual human interface application 116 in the embodiment described above, except that some of the human interface processing is handled by the survey server 300 rather than locally on the survey input client 106. This embodiment advantageously eliminates the need for a Verbot™ application and a virtual human interface application 116 on the survey input client 106. In this embodiment, browser cookies may be used to track information regarding survey participants 118. It will be appreciated by those of ordinary skill that cookies are routinely used to allow web servers to recognize information about clients who repeatedly browse the server. Also, survey participants 118 can more easily participate in surveys from mobile and remote environments.

Figure 4 illustrates a representation of one embodiment of a survey results report 126 generated by a survey results server 110. The survey results report 126 displays current survey result data in a form that is easy to analyze. For example, in the illustrated embodiment, the survey results report 126 displays statistical information regarding how the survey participants 118 rated the subjects of the survey on a given scale. It will be appreciated that countless other formats for survey result data may be used with the present invention, and the present invention is not limited by any particular format for result data.

Figure 5 illustrates a representation of aspects of a virtual human interface application 116 in accordance with one embodiment of the present invention. In this embodiment, the virtual human interface application 116 includes a Verbot™ 500. In one preferred embodiment, the Verbot™ 500 is Sylvie version 3.04 available from Virtual Personalities, Inc., which Verbot can be run locally on the survey client 200. The local embodiment of the Verbot 500 advantageously facilitates geographic control over surveys through simple distribution of the Verbot 500. In another preferred embodiment, the Verbot™ 500 is an on-line Sapphire class Verbot™ available from Virtual Personalities, Inc. The on-line embodiment of the Verbot 500 advantageously simplifies distribution of the Verbot 500 as it is available, in one embodiment, by simply accessing a web page.

Those of ordinary skill in the art will understand that the Verbot™ 500 includes particular modules, namely a script file parser 502, an image generator 504, and a response parser 506. It will also be appreciated that the

Verbot™ 500 may carry out or facilitate certain user interface functions such as, for example, synthesizing speech 510 from text data, such as survey question data embedded in a script file 114 (also referred to as a net file), presenting photo-realistic images 512 comprising an animated human face in accordance with facial expression codes embedded in the script file 114, and recognizing all or portions of user input 516 to match one or more anticipated responses coded in a script file 114.

The on-line embodiment of the Verbot™ 500 performs similar functions. The response parser 508 is implemented as a JAVA applet, and image generation 504 is performed with the use of available web browser plug-ins, now in common use, which deliver animated content to Internet users. In a preferred embodiment, the on-line Verbot 500 uses Flash 4 from Macromedia, but could use Microsoft Agent or Pulse3D. It will be appreciated that the present invention may make use of these technologies as well as other rapidly developing media technologies (including image generation technology handling not only geometry, but also textures) that will permit the presentation of ever more realistic characters in connection with the virtual human interface.

The JAVA applet, in one embodiment of the present invention, operates in a web browser environment, receives and parses user responses, and, as necessary, sends requests from the user's web browser to the survey server 300 requesting, for example, specific voice and/or animation data. The survey server 300 responsively transmits a data stream comprising, for example, voice data and graphic frames representing a character communicating information, such as, for example, the movement of lips (lip-synch frames) to simulate a talking human. In one embodiment, the frames may be cached locally on the user's web browser to facilitate a faster response, and, in a further embodiment, the transmitted data stream may include voice data and references to cached frames. In another embodiment where transmission bandwidth is adequate, frames may be transmitted from the server on demand. As will be appreciated by those of ordinary skill, in one embodiment, the JAVA applet runs under JAVA version 1.1, and communicates with Flash 4 via the LiveConnect plug-in extension available from Netscape. Thus, in a preferred embodiment, the on-line version of the Verbot™ 500 resides partially on the survey server 300 and partially on the survey client 200 and interacts with a survey participant 118 through a web page.

In a preferred embodiment, the Verbot™ 500 is interfaced with voice recognition and voice output modules, such as, for example, those available from Lernout & Hauspie. However, the invention could use other voice recognition and voice output modules and is not limited by any particular voice recognition or voice output module. In this embodiment, the Verbot™ 500 can verbally present questions to a survey participant 118 and can accept and respond to the spoken responses of the survey participant 118. Those of ordinary skill in the art will understand that the script file parser 502 extracts question text from the script file 114 and presents the question text data as input to the voice output module, which generates sound through a sound card and/or speaker to verbalize the question text. It will be further appreciated that the voice output module can output a recorded human voice or a computer-synthesized voice that can be modified, such as in pitch and speed (e.g., to sound more like a man or a woman). Preferably, the voice output module of the on-line version of the Verbot™ 500 outputs a realistic human voice when interacting with the survey participant 118.

Like voice output modules, voice recognition modules are now widely available, and thus it will be understood that an application program can be provided with textual input that has been generated by a voice recognition module. Generally, a voice recognition module accepts analog voice input through a microphone, converts the analog signals to digital signals, samples and encodes the signals (such as, for example, by using pulse code modulation) and to convert the voice input to a data stream representing text characters. Often, existing voice recognition modules generate text characters corresponding to the spoken words with an accuracy of better than 90%. The present invention is not, however, limited by voice input or output.

In another embodiment, the Verbot™ 500 interacts with a survey participant 118, particularly the hearing impaired, by generating text in a user interface window 520 such as those commonly supported by the Microsoft Windows operating system. In this embodiment, the Verbot™ 500 generates text characters in a question text box 522. The question text characters are based on the question data parsed from a script file 114 or on response text scripted in the script file 114 to be presented to a survey participant 118 based on his or her input to the Verbot 500. The survey participant 118 reads the question text and responds by entering text via a keyboard into an answer text box 524. Upon striking the 'enter' key, the response parser 506 begins processing the entered text characters.

Both the local and the on-line embodiments of the Verbot™ 500 function in the same manner with respect to the text and voice inputs and outputs described above. For example, in both embodiments, the user has the option to engage both the text and voice outputs of the Verbot™ 500 at the same time. In one embodiment, preferably for use at bandwidths below 28.8 kbps, the user can advantageously select a text-only version of the on-line embodiment of the Verbot™ 500.

Figure 5E illustrates steps performed in one embodiment of the present invention to load the on-line embodiment of the Verbot 500. In a first step 540, the user, using a web browser, accesses a survey web site hosted by the survey server 300. The survey server 300 transmits a web page, including Verbot 500 setup instructions. In a next step 542, the setup instructions examine the survey client 200 to determine whether it includes an appropriate media plug in, such as, for example, the Flash 4 media player by Macromedia. If, in a step 544, the instructions determine that an appropriate media plug in is available, then, in another step 546, the survey server 300 transmits the JAVA applet, an initial script file 114 and possibly a media stream (including for example initial animation information and/or initial voice information) to the survey client 200. If, in the step 544, the setup instructions determine that a media plug in is not available, the setup instructions, in a step 548, query the user for permission to access and install such a plug in, for example the Flash 4 media player. The Flash 4 media player is widely available, and it will be appreciated by those of ordinary skill that it is common for web pages to include instructions to access and install the Flash 4 media player as well as to ask permission for the same. It will be further appreciated that the Verbot 500 of the present invention may interface with other media players and thus the present invention is not limited by a particular media player.

If, in the step 548, the user does not grant permission, then, in a step 550, the setup instructions generate a message informing the user that the survey cannot be conducted and the process terminates. If, in the step 548, the

user does grant permission, then, in a next step 552, the survey client 200 accesses an appropriate media plug in via the Internet, and downloads and installs the media plug in. Processing proceeds to the step 546 wherein the survey server 300 transmits the JAVA applet, an initial script file 114 and possibly a media stream to the survey participant's computer.

5 It will be further appreciated by those of ordinary skill in the art that the Verbot™ 500 can perform predetermined data processing instructions 518 associated with matching all or a portion of user input with an anticipated response coded in a script file 114. The data processing instructions can include basic programming language commands and more sophisticated commands, such as, for example, those permitting file operations (i.e., opening, reading from, writing to, and closing files) and launching other applications and providing command parameters to launched applications. Thus, for example, the Verbot™ 500 can perform data processing instructions to
 10 create an answer file 120 and to record in the answer file 120 certain responses provided by the survey participant 118 and recognized by the response parser 506.

Generally, data processing instructions are combined with question data, expression codes and anticipated response patterns in a single script file 114. The script file 114 thus guides the actions of the Verbot™ 500 in
 15 conducting a survey. The script file parser 502 of the Verbot™ 500 processes a script file 114 to configure the Verbot 500 to recognize and act on the various instructions and commands that can be included therein.

As will be appreciated by those of ordinary skill, a script file 114 includes a series of rules. Each rule can have a variety of components. The following is a sample rule:

20 < start-0 >
 a:0.3
 p:35 How*doing*today*
 p:35 *are*feeling*today*
 r:I'm doing well today, thank you
 25

Each rule has a title, which is specified between "<" and ">" symbols. Titles can be used to indicate which rules are especially active at any point.

The activation level "a:0.3" resolves conflicts with other rules that may be satisfied by an input string. Thus, if a second rule is also satisfied, but has a lower activation level, say "a:0.2", then it would not be fired, or
 30 activated.

Pattern values are indicated by a command prefix, such as "p:35". The pattern value ("35") attaches a relative importance to a specified pattern. A pattern, such as "How doing today", identifies certain text which could be part of an input string (response) supplied by a survey participant 118. Asterisks, "*", are wildcards that can match any or no text. Note, pattern lines are optional and leaving out a pattern will cause the Verbot™ 500 to fire the
 35 rule when no matching pattern is found. This can be used to properly respond to input that is not recognized with a statement such as, "I didn't understand what you just said. Can you please re-phrase it for me?"

in order to facilitate easy scripting, pattern value macros for affirmative and negative answers have been formulated. Pattern value macros are commands that automatically recognize a wide variety of possible user responses, such as affirmative (AFF) or negative (NEG) input. As will be appreciated, many other additional macros could be created to make scripting more efficient.

5 Response strings are identified by "r:". When a rule fires, that is, when a pattern in the rule matches user input and/or when the activation level of the rule is not superseded by another rule, the response string is presented to the survey participant 118. In a preferred embodiment, the response string is sent to and processed by the voice output module to generate voice output to be heard by the survey participant 118. The rule can then be disabled for a predetermined time period.

10 The following example demonstrates how multiple rules can interact in a script file 114 to advantageously conduct a survey in a conversational and natural way.

```

<survey-0>
a: 0.7
r: Would you like to take a survey?
+: <survey-0-0> <survey-0-1>

<survey-0-0>
a: 0.2
p: 50 AFF
r: That's wonderful. I know you're busy, and I really appreciate your time.
-: <survey-0-1>
+: <newsurvey-0>

<survey-0-1>
a: 0.2
p: 50 NEG
r: Oh, that's too bad, maybe we can talk about it again soon. Talk to you next time.
-: <survey-0-0> <newsurvey-0>
+: <nosurvey-0>

<newsurvey-0>
a: 0.2
r: Lets talk about the TV show, The Z Papers. Did you watch it last Tuesday?
+: <newsurvey-1>

<nosurvey-0>
a: 0.2
r: It's always good to see you.

```

15 20 25 30 35 40

In the above excerpt from a script file 114, the "+:" code is used to specify which rules will be particularly active if the present rule fires. Thus, for example, the line "+: <survey-0-0> <survey-0-1>" indicates that upon inquiring, "Would you like to take a survey?", the Verbot™ 500 will be examining the survey participant's 118 response to look for a match in the patterns specified in the rules titled "<survey-0-0>" and "<survey-0-1>". In

this way, the Verbot™ 500 determines if the response was affirmative or negative and gives the appropriate verbal response.

On the other hand, the “.” code specifies which rules will not be active after the present rule fires. Thus, the Verbot™ 500 can advantageously be scripted to change its sensitivities depending on the survey participant's 118 responses. The on-line version of the Verbot™ 500 uses a scripting language similar to that of the local version of the Verbot™ 500, as described above. In some embodiments of the on-line version of the Verbot™ 500, however, a number of alternative “r.” patterns can be included in a single rule. The following is an example of a rule including several alternative “r.” patterns:

p:25 “how” you “doing”
 r: I'm feeling fine.
 r: Wow, I feel great.
 r: Things are going well.

Moreover, and further advantageously, the Verbot™ 500 can be scripted to control the facial features of a photo-realistic human face to provide the appearance to the survey participant 118 that he or she is conversing with a lively, even entertaining, intelligent entity, which makes the entire survey process more natural, pleasant and enjoyable. The image generator 504 of the Verbot™ 500 can render a photo-realistic human face on a display to show any of a number of possible facial expressions 514A, 514B, 514C, 514D. (Figures 5, 5A, 5B, 5C, 5D). In one embodiment of the present invention, the face of the Verbot™ 500 appears in a face window 526 of the user interface window 520. During presentation of any response string, the image generator 504 moves the lips of the photo-realistic human face to even further simulate conversation. Still further, the image generator 504 causes the eyes of the photo-realistic human face to blink at random times, even when no response is being presented to further advantageously provide the survey participant 118 with the feeling that he or she is conversing with a living entity.

An example of a scripted facial expression follows. In some embodiments, expression tags can be used to control the facial expressions of the Verbot™ 500. The following are some examples of expression tags:

* <mouth # duration>
 * <eyes # duration>

In the above examples, each # represents a different eyes or mouth frame. The duration field controls the time (in milliseconds, for example) that the expression displayed.

The following table provides a list of some examples of different possible expressions that the Verbot™ 500 can display, together with examples of corresponding mouth and eyes numbers.

Expression	Eyes	Mouth
Angry	5	45
Misty look right	6	46
Smirk	7	47
Surprise	8	48
Misty look left	9	N/A
Duh	10	N/A

For example, the expression tag * < eyes 5 1000 > would make the eyes of the Verbot™ 500 look angry for one second (1000 milliseconds). In one embodiment, if the user inputs a -1 in the duration field of the expression tag, then the Verbot™ 500 holds the expression until a new one is input. For example, the expression tag * < eyes 6 -1 > would hold a misty look until some other eye command is given, such as * < blink >

Expression macros can be used to ease the process of scripting facial expressions. The following are some examples of expression macro tags:

* < smile >
* < blink >

In one embodiment, a pronunciation file can be created and used to correct the pronunciation of certain words by the Verbot™ 500. The following is an example of a list of entries in the pronunciation file:

"win98" = (windows 98)
"winnt" = (windows NT)
"win32" = (win 32)
"email" = (a mail)
"http://" = (h t t p ://)
"www." = (w w w dot)
".com" = (dot com)
".org" = (dot org)
".net" = (dot net)
".edu" = (dot e d u)
".gov" = (dot gov)
".mil" = (dot mil)
".html" = (dot h t m l)
".htm" = (dot h t m)
"@ " = (at)

In the above excerpt from a pronunciation file, the text in quotes is intercepted as it is generated by a firing rule. The text is then translated into what the Verbot™ 500 would actually say. For example, "@ " becomes the word "at."

The pronunciation file can also be used to create expression macros. For example, the following entry in the pronunciation file would create an expression macro entitled "frown."

5 ** <frown >:(* <eyes 5 2000 >)

In the above example, when the frown expression macro is fired, the eyes of the Verbot™ 500 go into a frown position and stay for 2000 milliseconds. The following example demonstrates how expression macros can be used in scripting a Verbot™ 500.

10 a: 0.2
 p: 35 "how*are*you"
 r: * <smile> I'm really doing well.

When the above rule is fired, the Verbot™ 500 smiles and says, "I'm really doing well."

15 In the on-line embodiment of the Verbot 500, the script file 114 is preferably transmitted to the survey client 200 with the JAVA applet, which includes the script file processing module. In another embodiment, it is contemplated that the script file 114 may be parsed on the survey server 300 to create a JAVA applet, which is preconfigured to follow all instructions and commands in the script file 114. Thus, when the JAVA applet is loaded on the survey participant's 118 computer, it can determine which script rule fires as a result of his or her response.

20 Figure 5F illustrates steps performed in one embodiment of the present invention to generate character images with the on-line Verbot 500. In a first step 560, the survey server 300 transmits to the survey client 200 all image frames necessary for image generation, such as, for example, for eye blinking, lip synchronization and facial expression changes. In one embodiment, these image frames are transmitted to and locally cached at the survey client 200 along with the initial transmission of the JAVA applet and the initial script file 114. In a next step 562, the
 25 on-line Verbot 500 identifies information to be communicated to the survey participant 118.

In one embodiment, voice data representing the information to be communicated resides on the survey server 300, and, in a further step 564, the survey client 200 issues a request for a media stream including the voice data. The media stream may be a key framed sound file. In a next step 566, the survey server 300 transmits the key framed sound file including commands to match certain frames, for example lip-synch frames, to the voice output
 30 represented in the sound file.

To display the image frames in sequences which create desired animation, the sequential display of frames is controlled using hide and unhide operations. Thus, for example, the Verbot 500 generates a character image whose lips may be synchronized to spoken words by sequentially hiding a current frame, unhiding a next frame, hiding that frame, unhiding a further frame, and so on until the desired animation is complete. Preferably, key framed sound files
 35 requested by the JAVA applet from the survey server 300 streams commands that match the lip-synch frames to voice output provided in the sound file.

In a step 558, the survey client 200 performs hide/unhide operations at around, in one embodiment, 13 operations per second to match locally cached lip-synch frames. In the step 558, the survey client 200 also simultaneously generates sound by processing the sound file, which processing will be familiar to those of ordinary skill in the art. Those of ordinary skill will further appreciate that existing browser plug-ins, for example Flash 4, may be directed to perform such hide and unhide operations. The on-line Verbot 500, thus, in one embodiment, generates a character image appearing to communicate information to a user.

When the on-line embodiment of the Verbot 500 uses voice output, the JAVA applet requests pre-recorded voice sequences to be transmitted from the survey server 300 to the survey client 200 on demand. In another embodiment, the Verbot 500 may transmit digital data sequences representing text characters to a voice synthesis module, which renders analog audio output in a form approximating a human voice speaking words corresponding to the text characters.

In some embodiments of the present invention, the image generator 504 generates faces likely to be known to the survey participant 118. Thus, for example, in conducting a survey for a television show, the image generator 504 could advantageously generate the face of a character from the television show. Furthermore, the voice output module could be modified to produce a voice like that of the character. In this manner, the survey participant 118 more easily identifies with the survey process, is more interested, and the entire process is rendered more entertaining and enjoyable. In turn, this advantageously increases both the attention that each survey participant 118 will give the survey as well as the number of participants that will take the survey.

In some embodiments, a branded character, such as a famous animated cartoon character, can be simulated using Verbot™ technology. When possible, the character's actual voice can be used and the Verbot™ 500 could be scripted to act in a manner consistent with the original character. It is contemplated by the inventors that, in still other embodiments, the image generator 504 generates fictitious, historic, legendary or fantasy character images, such as, for example, Huckleberry Finn, Abraham Lincoln, Michelangelo, Hercules or Bugs Bunny. In even further embodiments, the image generator 504 generates character images representing animals, which may include any life form, such as, for example, dogs, cats, mice, or other mammals, reptiles, amphibians, fish, mollusks, crustaceans, birds, spiders, insects and even microscopic and invertebrate life forms. In still further embodiments, the image generator 504 generates character images representing inanimate objects, such as, for example, toys, cars, computers, rocks, clouds, etc. As will be appreciated, the present invention is not limited by a type of character that can be generated by the image generator 504. The inventors contemplate further that images representing two or more characters may be generated to provide participants with the experience of communicating with two or more characters in a single conversational episode. The inventors further contemplate that the image generator 504 generates images representing a character signing in sign language to communicate with hearing impaired survey participants 118.

Operation of System and Method

Figure 6 illustrates a flow chart describing the overall operation of one embodiment of a survey system 100 in accordance with the present invention. In a first step 602 of a first series 600 of steps, a survey administrator populates the survey database 122 with information about one or more sponsors and provides initial user and password data for each sponsor as described above.

5 In a preferred embodiment, a survey sponsor uses a limited survey database management application to enter new survey and question data for a new survey. After authenticating the sponsor by password and determining the SPONSOR_ID for the sponsor, the limited survey database management application offers the sponsor a choice of table subject matter on which to perform operations, such as, for example, to browse, modify, add or delete (1) Survey Information, (2) Survey Question Information, (3) Answer Information, (4) Answer Format Information, or (5)
10 Survey Participant Information. Using techniques described above, a survey sponsor can add survey and question data to the survey database 122.

In a preferred embodiment, a specialized application called a survey entry application is designed to simplify the introduction of a new survey into the survey database 122. After authenticating the sponsor as described above, the survey entry application, which is operatively connected to the computer hosting the survey database 122
15 (directly or by network), prompts the survey sponsor to enter a survey title in a survey title text field and to enter a brief description of the purpose of the survey in a survey description field. When the survey sponsor selects an OK button, the survey entry application creates a new record in the SURVEY_TABLE, incrementing the highest number already used to identify a survey and assigning it to the SURVEY_ID field, assigning the SPONSOR_ID value for the sponsor to the SPONSOR_ID field, the entered title text to the SURVEY_TITLE field, and the entered description to a
20 SURVEY_DESCRIPTION field.

The survey entry application then prompts the sponsor to enter question data for the survey. The survey sponsor then enters the text for a survey question in a question text field. Then, to specify a format to present the collected answers, the survey sponsor activates a pull-down menu presenting a list of answer format descriptions extracted from the ANSWER_FORMAT_TABLE. The survey sponsor chooses one the sponsor believes will best
25 format the group of answers collected for the question. For example, one description may read "place all answers in a sequential list" and another may read "choose the five responses most frequently given and show the percentage of users providing the respective response." It will be understood that the best way of representing answer data may differ depending on the nature of the answer data, for example, numeric answer data, text answer data, true/false
answer data.

30 When the sponsor selects an answer format, the survey entry application places the associated ANSWER_FORMAT name in the answer format field. Upon selecting an OK button, the survey entry application creates a new record in the QUESTION_TABLE by calculating and assigning a new question identifier (e.g., "00001" for the first question in a survey, "00002" for the next question, and so on) to the QUESTION_ID field, assigning the answer format name to the ANSWER_FORMAT field, assigning the SURVEY_ID to the SURVEY_ID field, assigning the
35 entered question text to a QUESTION_TEXT field. The survey entry application prompts the sponsor to enter another

question or finish. The sponsor enters as many questions as desired for the survey and, when done entering questions, selects finish. The survey entry application generates a new survey report showing the title of the new survey, the newly generated survey ID, and, for each question entered, the question ID followed by the text of the question.

5 In a next step 604, a survey scripter uses the new survey report to encode the survey questions into a script file 114. To facilitate automated processing of answers provided by survey participants 118, the survey scripter also includes commands in the survey script file 114 to cause participants' answers to be recorded in an answer file along with the QUESTION_ID of the corresponding question. Also, the survey scripter includes in the script file 114 commands which cause the SURVEY_ID to be written at the beginning of the answer file. Such a command might be
10 * <input=[SURVEY_ID=01122]>. The scripter may also add some initial scripting to ask the survey participant 118 whether he or she would mind providing some personal information such as, for example, name, geographic location, computer configuration, how many surveys he or she has completed, etc. The scripter may add commands that cause the participant data to be written to the answer file 120.

15 Additionally and advantageously, the survey scripter in a next step 606, either alone or collaboratively with the survey sponsor, adds expression, entertainment, lobbying and/or advertising elements to the script file 114. In another step 608, the scripter posts the script file 114 to the survey script server 104.

When the local embodiment of the Verbot™ 500 is used, the Verbot™ 500 can be preprogrammed with a specific URL address from which to obtain new script files 114 from the survey script server 104. When several different surveys are run at the same time, each script file 114 can be assigned a unique URL address. Furthermore,
20 each script file 114 can be annotated with information regarding the survey to which it is targeted. In this way, the script file 114 is advantageously unlikely to be posted to the wrong URL address and, hence, be transmitted to the wrong survey participant 118.

When the on-line embodiment of the Verbot™ 500 is used, the Verbot™ 500 and the script file 114 are available via a URL address, which the survey participant 118 can access from any location having a suitable internet
25 browser. In the on-line embodiment of the Verbot 500, or alternatively in the local embodiment of the Verbot 500 running on the survey client 200 which has a current network connection with the survey server 300, the Verbot 500 can immediately access and load a new script file 114 in response to a predetermined event.

For example, if, during a survey conducted using an initial script file 114, the user indicates some interest in a different program, the Sapphire™ class Verbot™ can immediately download a new script file which provides
30 information about or even conducts another survey about the different program. This process is initiated with a command such as * <loadscript=premiumprogram.script>. This new script file 114 may advantageously contain intelligence (rules) addressing the user's indicated interests and, thus, provide a more dynamic and enjoyable experience for the user. When the new script file 114 loads, the Verbot 500 can say: "I've loaded some new information about the information you requested. Go ahead and ask your questions now." When this portion of the

interaction is complete, the Verbot 500 can reload the original script file 114 and continue or load a third or additional script file 114 requested by a rule.

The following example illustrates rules which accomplish the loading of a script file dynamically:

```

5      < premium-1 >
      a:0.3
      p:35 *what*tell*premium*
      p:35 *can*about*premium*
      p:35 *premium*progra*
10     r:I can tell you all about the new premium program, but wait a second while
        I check the latest. * < loadscript=premiumprogram.script >
      r:I'm glad you asked, let me check with my server for the latest, then I'll
        answer your questions. * < loadscript=premiumprogram.script >
      r:We do have a premium program, I need to check in with my server to see
15     what the latest information is. * < loadscript=premiumprogram.script >

```

Rules in the premiumprogram.script could be coded as follows:

```

20     < return-1 >
      a:0.5
      r:Does that answer your questions?
      +: < retries-1 > < return-1 >

      < retries-1 >
25     a:0.0
      p:35 AFF
      r:Good, lets get back to where we were then.
      * < loadscript=original-r.script > (note that original-r is the same as the
        original script but starts with a comment designed to reorient the user)
30     < return-1 >

      a:0.0
      p:35 NEG
      r:Okay, ask me anything you like about the premium program and I'll do my
35     best to answer you.

```

Thus, dynamic script file 114 loading may be used to facilitate adapting to a user's interests during a survey. Those of ordinary skill will further appreciate that the ability to load script files 114 dynamically provides the additional advantages of breaking a larger script into multiple component scripts to keep any single script relatively small, which, in turn, may reduce script download times. It will be appreciated that reduced download times provided a better user experience, particular in circumstances where transmission bandwidth is limited.

When the local embodiment of the Verbot™ 500 is used, a second series 610 of steps is performed independently of the first series 600 of steps. In a first step 612 of the second series 610 of steps, a user installs a virtual human interface application 116 on the survey input client 106. In a next step 614, the virtual human interface application 116 obtains permission from the user to periodically log on to the survey script server 104 and

automatically download the latest script file 114. The virtual human interface application 116 also obtains permission to automatically return answer files 120 to the survey results collector 108 upon completion of future surveys.

After the first series 600 and second series 610 of steps are completed, the survey system 100 proceeds to a third series 615 of steps. When the local embodiment of the Verbot™ 500 is used, the survey input client 106, in a first step 616 of the third series 615 of steps, loads the latest script file 114 from the survey script server 104. As described above, when the on-line embodiment of the Verbot™ 500 is used, the Verbot™ 500 does not need to load the script file 114 from the survey script server 104; rather, in the step 616, the survey server 300 simply loads a JAVA applet on the survey input client 106 when the survey participant 118 accesses the web page associated with the survey. The JAVA applet interfaces the survey participant 118 to the Verbot 500 hosted by the survey server 300.

In a next step 618, the Verbot™ 500 processes the script file 114 to conduct a survey with the survey participant 118, during which the Verbot™ 500 advantageously entertains, lobbies with, and advertises to the survey participant 118 in accordance with scripting in the script file 114 and records responses in an answer file 120. In another step 620, the virtual human interface application 116 closes the answer file 120 to thereby collect all the responses of the survey participant 118 in the answer file 120.

The survey input client 106, in a step 622, transmits the answer file 120 to the survey results collector 108 over the network 102. In a further step 624, the survey results collector 108 parses the answer file 120, extracts the answer data, and stores the data in the survey database 122.

In another step 626, a survey reviewer 124 requests a survey results report 126 from a survey results server 110. In a next step 628, the survey results server 110 extracts answer data for the requested survey from the survey database 122, dynamically generates the survey results report 126, and transmits it to the survey reviewer 124.

Figures 7A through 7E are a series of flow charts representing the operation of a survey system 100 in accordance with one embodiment of the present invention using the local version of the Verbot™ 500. In a first step 702, a user loads a virtual human interface application 116 on the survey input client 106. In a next step 704, the user invokes a virtual human interface setup application.

In another step 706, the virtual human interface setup application can determine whether the survey input client 106 has sufficient resources to support the virtual human interface application 116. If the survey input client 106 lacks sufficient resources to support the virtual human interface application 116, then, in a step 708, the virtual human interface setup application generates a message announcing and explaining the insufficiency of resources to the user. Otherwise, the virtual human interface setup application, in another step 710, installs the virtual human interface application 116 and an initial script file 114 on the survey input client 106.

In a step 712, the user invokes the virtual human interface application 116. In a next step 714, the virtual human interface application 116 launches a Verbot™ 500 and directs the Verbot™ 500 to open the initial script file 114. In a next step 716, the Verbot™ 500, in accordance with the scripting contained in the initial script file 114,

introduces itself and interactively seeks permission from the user to periodically download new script files 114 and upload answer files 120 automatically. In a further step 718, the Verbot™ 500 determines whether the user grants permission.

5 If the user does not grant permission, then the Verbot™ 500, in a next step 720, advantageously lobbies the user to grant permission according to the scripting contained in the initial script file 114. In a step 722, the Verbot™ 500 determines whether the user grants permission after the lobbying.

The following is an example of lobbying:

10 Verbot™: May I download the latest intelligence now?
 User: No, I don't want you to.
 Verbot™: Why not?
 User: Because I'm concerned about the expense.
 Verbot™: The intelligence file downloads very quickly. On your connection, it should take no more than 18 seconds. Besides, my current script is very limited. I think you'll enjoy the new stuff.
 15 User: I'm not sure.
 Verbot™: One more thing to consider is that I'm giving you direct access to the programming department at Galaxy Network. That is power most people wish they had.
 User: Well, okay, then. But don't stay on-line any more than you have to.
 20 Verbot™: Excellent, I know you'll be pleased with your decision.

Those of ordinary skill will appreciate that the lobbying, such as that illustrated above, can be facilitated by rules coded in a script file 114, which anticipate and respond to certain user responses.

25 If the user grants permission after the initial request or after the lobbying, then, in a next step 724, the virtual human interface application 116 sets a permission flag. Otherwise, the virtual human interface application 116 proceeds to a step 730 without setting the permission flag.

30 In a preferred embodiment, the permission flag is implemented in the software of the virtual human interface application 116. When the permission flag is set, the virtual human interface application 116 can bypass the permission request for each individual access. Thus, the virtual human interface application 116 can automatically download new script files 114 on a regular basis. The automatic download can be set to occur at some predetermined repeating time interval or can be set to trigger upon the detection of some triggering event, such as a user log on. An appropriate method can be selected based on the user's individual circumstances. For example, for workstations and PCs that remain turned on most of the time, the automatic download is preferably set to occur at some predetermined repeating time interval.

35 If the user grants permission, then, after completing step 724, the virtual human interface application 116, in a next step 726, directs the survey input client 106 to download a new script file 114 from a survey script server 104 over the network 102. In one embodiment, this is done by launching an Internet browser and providing it with a

command line instruction to access a script file 114 from a site hosted by the survey script server 104. It will be appreciated that a script file 114 may be downloaded via FTP, or other file transfer protocol. The virtual human interface application 116 stores the script file 114 in a predetermined directory on the survey input client 106 once the script file 114 is received. In a next step 728, the new script file 114 replaces the initial script file 114 on the survey input client 106. In a further step 752, as described in more detail below, the Verbot™ 500 invites the user to participate in a survey.

If, in the step 722, the user does not grant permission, then the virtual human interface application 116, in the step 730, waits in a noninteractive "sleep" mode without setting the permission flag. While in the noninteractive sleep mode, the virtual human interface application 116, in a step 732, monitors a wake-up trigger to determine whether the trigger has been activated. Alternatively, a timer is set and monitored by the operating system with the virtual human interface application 116 set to be launched upon the detection of the timer expiration event. Either way, the wake-up trigger may be configured to activate automatically at recurring intervals of some predetermined time period. Alternatively, the wake-up trigger may be configured to activate in response to some predetermined user input or activity, such as a mouse or keystroke event. If the wake-up trigger has not been activated, the virtual human interface application 116 remains in the noninteractive sleep mode of step 730.

In another embodiment, the virtual human interface application 116 generates a scaled-down version of the human-like face displayed by the Verbot™ 500, and displays the scaled-down face, for example, in a corner of the display. Further, the virtual human interface application 116 advantageously uses the voice output module to generate teaser statements, enticing the user to take a survey. Any user input activity, such as a mouse or keystroke event may activate the wake-up trigger.

Once the wake-up trigger is activated, the virtual human interface application 116, in a step 734, determines whether the permission flag has been set. If the permission flag has not been set, then the Verbot™ 500, in a next step 736, requests permission from the user to download a new script file 114 from the survey script server 104. If the user does not grant permission, then the virtual human interface application 116 returns to the noninteractive sleep mode of step 730. If the user grants permission, then the virtual human interface application 116, in a next step 740, directs the survey input client 106 to attempt to download a new script file 114 from the survey script server 104.

In a next step 742, the virtual human interface application 116 determines whether the attempted download was successful. In some embodiments, the virtual human interface application 116 determines the success of the download by comparing the content of the old script file 114 with that of the newly downloaded script file 114. If the contents of the two script files 114 differ, then the virtual human interface application 116 determines that the download of the new script file 114 was successful.

If the download was not successful, then the virtual human interface application 116, in a step 744, generates and displays a message indicating that a survey is not available. This can be used to encourage the user to

actively download the new script file 114. The virtual human interface then returns to the noninteractive sleep mode of step 730. If the download was successful, then processing continues to a step 750.

5 If, during step 734 the virtual human interface application 116 determines that the permission flag has been set, then the virtual human interface application 116, in a next step 745, determines whether the user has already completed the survey included in the current script file 114 by checking a flag that is set at completion of each survey and is reset at the loading of a new script file 114. If the user has not completed the survey, then the virtual human interface application 116, in a step 746, determines whether the survey included in the current script file 114 is too old to conduct by referencing its last date and checking the current system date. When the difference between the dates exceeds a predetermined period of time, for example, two weeks, the current script file 114 is determined to be too old.

10 If the survey included in the current script file 114 is too old or if the user has already completed the survey included in the current script file 114, then the virtual human interface application 116 proceeds to the step 740, as discussed above.

15 In the step 750, the virtual human interface application 116 determines whether the new script file 114 is different than the previous script file 114. If the new script file 114 is the same as the previous script file 114, then the virtual human interface application 116 generates and displays a message indicating that a survey is not available in the step 744, as described above. The virtual human interface application 116 then returns to the noninteractive sleep mode of step 730.

20 If the new script file 114 is different than the previous script file 114 or if, during step 746, the virtual human interface application 116 determines that survey included in the current script file 114 is not too old to conduct, then the Verbot™ 500, in a next step 752, invites the user to participate in a survey. In a step 754, the Verbot™ 500 determines whether the user accepts the invitation. If the user does not accept the invitation, then, in a next step 756, the Verbot™ 500 lobbies the survey participant 118, preferably reminding her of the importance of the survey, and offers to show her information about the program. In another step 758, the Verbot™ 500 determines whether the survey participant 118 wants program information. If the survey participant 118 does not want program information, then processing returns to the noninteractive sleep mode of step 730.

25 If, in the step 756, the user accepts the offer, then, in a next step 760, the virtual human interface application 116 displays the offered program information by opening a regularly updated web page containing the information of interest. Processing then returns to step 752, where the Verbot™ 500 again invites the user to participate in the survey. If, during step 754, the Verbot™ 500 determines that the user accepts the invitation to participate in the survey, then the virtual human interface application 116, in a further step 762, directs the survey input client 106 to open the current script file 114.

30 In one embodiment, the Verbot™ 500, in an optional step 764, prompts the survey participant 118 to input his or her name. In alternative embodiments, the survey participant 118 will preferably remain anonymous, being identified only by certain demographic information. In an optional step 766, the Verbot™ 500 determines whether the

35

survey participant 118 has input a name. If the survey participant 118 has not input a name, then, in a next step 768, the Verbot™ 500 determines whether a time out has occurred. In one embodiment, a time out occurs if the user has not entered a response in 90 seconds. Those of ordinary skill in the art will appreciate that other time out time periods may be used with the present invention.

5 If a time out has occurred, then processing returns to the noninteractive sleep mode of step 730. If, during the optional step 766, the Verbot™ 500 determines that the survey participant 118 has input a name, then the Verbot™ 500, in an optional step 770, stores the name as a text string variable.

 In a next step 772, the virtual human interface application 116 parses the current script file 114 for the next survey question. In a preferred embodiment, each survey question is associated with one or more rules in a script file 114, and the content of the question may correspond to a response line in a script file 114. The following is an
10 example of a rule containing survey question information:

```
          a: Q.8  
          r: What is your favorite coffee?  
15           * <input=[Q0001]!$ >
```

 In a step 774, the Verbot™ 500 presents the question to the survey participant 118 and advantageously displays any facial expressions associated with the question. In a next step 776, the Verbot™ 500 determines whether the survey participant 118 has provided a response. If the survey participant 118 has not yet provided a
20 response, then the Verbot™ 500, in a next step 778, determines whether a first or second time out has occurred. If, after some predetermined time period, a first time out has occurred, then processing returns to step 774, where the Verbot™ 500 again presents the question to the survey participant 118 and displays the expression associated with the question. If, after a second predetermined time period, a second time out has occurred, then processing returns to the noninteractive sleep mode of step 730.

 In a preferred embodiment, no survey results are transmitted to the survey results collector 108 until the survey participant 118 has completed the entire survey. In addition, the virtual human interface application 116 will not download new information until the current survey has been completed. If the survey participant 118 opts not to finish a survey, then the virtual human interface application 116 preferably declines to download new information or share any reward information or other incentives with the survey participant 118. In alternate embodiments, it is not
30 necessary for every survey participant 118 to complete every survey. Information relating to completed or partially completed surveys such as, for example, percentages, may be included in the survey results report 126.

 If, during step 776, the Verbot™ 500 detects a user input, then, in a next step 780, the Verbot™ 500 determines whether the response matches any expected response. Preferably, the response parser 506 scans the response string provided by the survey participant 118 and identifies all rules, from the set of active rules, having a
35 specified pattern that matches all or part of the response string. The Verbot™ 500 then scans the activation levels

attached to each of the identified rules and fires the rule that has a matching pattern and has the highest activation level.

If the response string does not match any expected response, then the virtual human interface application 118, in a next step 782, determines whether the script file 114 contains scripted clarification regarding the survey question. In some embodiments, specific clarification rules are created to clarify a particular survey question or issue. A hierarchy of specific clarification rules can be established by using the plus(+) sign to fire the clarification rules in a predetermined sequence. The following are some examples of specific clarification rules organized in a hierarchy:

```

10      r: Did you like the way Joan handled Billy's obsession with Diane?
      +: <obsession-1> <obsession-2> <obsession-3>

      <obsession-1>
      a: 0.0
      p: 35 AFF
15      r: Very interesting. So you like it when women are strong?
      -: <obsession-2> <obsession-3>
      +: <strongw-1> <strongw-2> <strongw-3>

      <obsession-2>
20      a: 0.0
      p: 35 NEG
      r: Very interesting. You appear to be uncomfortable with Joan acting in a decisive manner. Is this true?
      -: <obsession-1> <obsession-3>
      +: <weakw-1> <weakw-2> <weakw-3>

25      <obsession-3>
      a: 0.0
      r: I'd appreciate if you'd give me either a yes or a no type answer about Joan's handling of Billy.

```

In the above example, the first two obsession rules have not been deactivated in the third obsession rule. By leaving the first two obsession rules active, the survey participant 118 is given a second chance to respond to the survey question.

In some embodiments, the script file 114 includes rules having no pattern and relatively high activation levels, such as above 0.0, that will fire when no expected response is found. Such rules can advantageously offer general clarification information to the survey participant 118, which may be needed in the case when the response string does not match any expected pattern.

Once a general clarification rule fires, it can be disabled for a predetermined time period, such as approximately 20 minutes. In this way, a hierarchy of general clarification rules can be established. The following are some examples of general clarification rules organized in a hierarchy:

```

40      a: 0.5
      r: I didn't expect what you just input. Can you please try again?

```

a: 0.49

r: Oh, wow, that's the second time I didn't understand what you're typing. I'm really sorry. Can you try that again?

5

If, in the step 782, it is determined that the script file 114 contains an appropriate clarification rule, then the Verbot™ 500, in a next step 784, presents the scripted clarification information to the survey participant 118, and generates images to show any facial expressions associated with the clarification information. Processing then returns to step 774, where the Verbot™ 500 again presents the survey question to the survey participant 118 and displays any facial expression associated with the question.

10

If, during step 780, the Verbot™ 500 determines that the response received from the survey participant 118 matches an expected response or if, in the step 782, it is determined that the script file 114 does not contain an appropriate clarification rule, then the virtual human interface application 116, in another step 786, records the response in an answer file 120 corresponding to the survey participant's 118 name or code. For example, if in response to the survey question, "What is your favorite coffee?", the survey participant 118 types, "Maxwell House," then the following entry would be added to the appropriate answer file 120:

15

00001=Maxwell House;

20

The "<input=" tag is used to save the response the user enters. The command "<input=[00001]\$>" combined with the user's response of "Maxwell House" causes the entry "00001=Maxwell House;" to be saved in the answer file 120. It will be appreciated that the "<input=" tag should preferably, but not necessarily always, be used in rules including patterns indicating that the user has entered an acceptable response. In alternative embodiments, the survey participants' 118 answers may be saved along with descriptive indicators of the question response such as "favcoffee=Maxwell House;". Those of ordinary skill will appreciate that applications processing answer files 120 having descriptive indicators of questions can easily parse and process such files to present results to survey reviewers.

25

Advantageously, a script may be coded to categorize user responses in predetermined categories. The following example shows just one way that a wide variety of responses to a single question may be categorized into a relatively few, but useful categories. This can be beneficial, particularly when every user's response is likely to differ from that of any other user.

30

r: Can you tell me, just briefly, how you feel about commercial breaks?

+: <br_st_neg> <br_wk_neg> <br_neut> <br_wk_pos> <br_st_pos> <br_unclear>

35

5 <br_st_neg>
 a: 0.0
 p: 15 *hate*
 p: 15 *awful*
 p: 15 *ridiculous*
 p: 15 *blight*
 p: 15 *can't stand*
 p: 15 *don't like*
 r: I understand. I guess they are a necessary evil * <write\$00003=-2;>
 10 : <br_wk_neg> <br_neut> <br_wk_pos> <br_st_pos> <br_unclear>

15 <br_wk_neg>
 a: 0.0
 p: 20 *annoy*
 p: 20 *too many*
 p: 20 *distract*
 p: 20 *bother*
 p: 20 *don't like*
 r: So you are not fond of commercial breaks. I guess I can understand that.
 20 * <write\$00003=-1;>
 : <br_st_neg> <br_neut> <br_wk_pos> <br_st_pos> <br_unclear>

25 <br_neut>
 a: 0.0
 p: 20 *one way or the other*
 p: 20 *take them*leave them*
 p: 20 *like some*others*
 p: 20 *neutral*
 p: 20 *no strong feelings*
 30 r: You seem like a very reasonable person. I like that in a human.
 * <write\$00003=0;>
 : <br_wk_neg> <br_st_neg> <br_wk_pos> <br_st_pos> <br_unclear>

35 <br_wk_pos>
 a: 0.0
 p: 15 *not too bad*
 p: 15 *okay*
 p: 15 *fine*
 p: 15 *sometimes interesting*
 40 p: 15 *sometimes funny*
 p: 15 *sometimes entertaining*
 r: I see that you generally look on the bright side of matters. Good for you.
 * <write\$00003=+1;>
 45 : <br_wk_neg> <br_st_neg> <br_neut> <br_st_pos> <br_unclear>

```

    <br_st_pos >
    a: 0.0
    p: 20 *they*great*
    p: 20 *love*
    5   p: 20 *wonderful*
        p: 20 *fantastic*
        p: 20 *excellent*
    r: I am glad that you seem to like them so well. That's wonderful.
    * < write$00003= +2;>
    10  -: <br_wk_neg> <br_st_neg> <br_neut> <br_st_pos> <br_unclear>

    <br_unclear >
    a: 0.0
    p: 5 *
    15  r: I'm sorry. I didn't understand your response. But then I can be dense sometimes. Could you please try
        to explain your feelings about commercial breaks in a shorter, maybe clearer response.
    +: <br_st_neg> <br_wk_neg> <br_neut> <br_wk_pos> <br_st_pos> <br_unclear>

```

In the preceding example, the answers resulting from one question (the third question in this case) will
 20 advantageously be limited to five different responses (i.e., -2, -1, 0, +1, or +2) regardless of the numerous differing
 responses a large number of survey participants 118 would likely provide. As will be appreciated, such results may
 sharply reduce the effort needed to retrieve, calculate and format the results for survey reviewers 124. In this
 example, the number of responses in each category may simply be counted and the totals in each category input to a
 pie chart generator, for example, to generate a pie chart effectively conveying the survey results for the question to a
 25 survey reviewer 124. It will be appreciated that an answer formatting module, in one embodiment comprising PERL
 script instructions, can easily associate string values such as, for example, "STRONG NEGATIVE", "WEAK
 NEGATIVE", "NEUTRAL", "WEAK POSITIVE" and "STRONG POSITIVE" with the respective answer values "-2", "-1",
 "0", "+1", and "+2".

In another embodiment, a second Verbot residing on the survey server 300 may assist with or provide
 30 answer formatting. In particular, an answer file 120 created on the survey client 106 may include entire answers
 provided by the user and also a special processing marker, such as a CTRL-V ASCII character. After the answer file
 120 is transmitted to the survey server 300, and when PERL script instructions residing on the server parse the
 answer file and encounter the special processing marker, the PERL script instructions launch a Verbot application
 which opens the answer file 120 as an input stream. Similar to recognizing the user's responses in the client
 35 embodiment discussed above, the server-based Verbot simply inputs the answer data in place of live user responses
 and uses a script located on the survey server 300 to generate a second answer file 120. During this server-based
 Verbot processing, the server-based Verbot can place the answer file responses into predetermined categories for
 statistical processing and return the category values to the PERL script instructions for further processing, such as,
 for example, entry into the appropriate database tables.

As discussed above, pattern macros can be advantageously used to group a large number of possible responses into a single macro, making it much more efficient to develop script files 114 which place wide varieties of user responses into a few categories.

In one embodiment, the answer file 120 itself is named after the user, appended with an incremental sequence number (increased after completing each survey), and given a file extension of ".txt". Thus, if the user's name is Henry, and Henry completed his 18th survey, the name of the answer file 120 might be "Henry18.txt".

In a next step 708, the Verbot™ 500 performs any facial expressions associated with the matched response, and also provides any comments or feedback, which may advantageously be lively and entertaining commentary, games or advertising. The following is an example of a possible entertainment script:

10 Verbot™: I really appreciate your input on these things. As a reward, would you like free tickets to a screening of Super Nova? The screening is on June 27 at 7:00 at the Brown Theater.

User: Yes.

Verbot™: Great, then please input your full name:

User: John Smith

15 Verbot™: Your street address:

User: 2213 South Bronx Drive

Verbot™: Your City:

User: Los Angeles

Verbot™: And last your Zip Code:

20 User: 90210

Verbot™: Cool. I'll have the tickets sent to you. They should arrive within three weeks.

The script file 114 advantageously includes commands to record the entertainment information in the answer file 120. For example, the script file 114 commands, in one embodiment, store each of the survey participant's 118 responses in string variables and concatenate them together to write the following to the answer file 120:

entertainment_event=tickets, Super Nova, June 27, 1999, John Smith, 2213 South Bronx Drive,
 30 Los Angeles, 90210;

The following is an example of another possible entertainment script:

Verbot™: Thanks for all your input. Would you like some inside information on the star, Julia Romero?

User: Yes.

35 Verbot™: Cool. I'm opening the browser to a secret web page that will let you in on some little known information about her. Enjoy.

For example, the following command in a script file 114 launches the browser and provides a URL to the browser to allow the survey participant 118 to access the web page which contains the desirable information:

5 a:I'm going to open a secret web page for you.
 * < run [path]netscape.exe
 http://www.vperson.com/cool/topsecret/JuliaRomera.html >

10 The survey participant 118 may also be advantageously provided with advertising information. The following is an example of a possible advertising script:

Verbot™: Thanks for all your input. How would you like to review a sneak preview of HiTakCo's latest digital camera?
 User: Okay.
 15 Verbot™: Cool. I'm opening the browser to a secret web page that will let you preview some cutting edge technology in digital cameras.

It will be appreciated that commands coded in the script file 114 may then launch a web browser to an advertising-related web site.

20 In a next step 789, the Verbot™ 500 determines whether the script file 114 contains another survey question. If the script file 114 contains another question, then processing returns to step 774 as discussed above.

If the script file 114 does not contain another survey question, then the Verbot™ 500, in another step 790, closes the answer file 120 generated during the survey. In a preferred embodiment, the answer file 120 includes a plurality of responses to a series of survey questions in sequence. The following is an example of an excerpt from a completed answer file 120:

Q0001 - Maxwell House; Q0002 - The Practice; Q0003 - -1;

...

30 In a next step 791, the Verbot™ 500 thanks the survey participant 118 for participating in the survey. In a next step 792, the virtual human interface application 116 determines whether the permission flag is set. If the permission flag is set, then, in a next step 793, the virtual human interface application 116 directs the survey input client 106 to transmit the answer file 120 to the survey results collector 108 over the network 102. In one embodiment, the survey input client 106 connects to a specified URL associated with a survey results collector 108 and uploads the answer file using FTP, a process well known in the art. Processing then returns to the noninteractive sleep mode of step 730. The following is an example of a scripted survey closing when the permission flag is set:

35

Verbot™: I'm sending your comments directly to the programming department. Your opinions will represent statistically about 1200 ordinary citizens. That makes you a pretty powerful person, I'd say. Good, I've transmitted your answers. Talk to you tomorrow, bye.

5 If the permission flag is not set, then, in a next step 794, the Verbot™ 500 asks the survey participant 118 for permission to transmit the answer file 120 to the survey results collector 108. In a further step 795, the Verbot™ 500 determines whether the survey participant 118 grants permission to transmit the answer file 120. If the survey participant 118 does not grant permission to transmit the answer file 120, then the Verbot™ 500, in a next step 796, advantageously lobbies the survey participant 118 for permission. In a step 797, the Verbot™ 500 determines
10 whether the survey participant 118 grants permission to transmit the answer file 120 after the lobbying.

If the survey participant 118 grants permission to transmit the answer file 120 either after the initial request or after the lobbying, then the virtual human interface application 118, in a next step 798, directs the survey input client 108 to transmit the answer file 120 to the survey results collector 108 over the network 102 as discussed above. Processing then returns to the noninteractive sleep mode of step 730. The following is an example
15 of a scripted survey closing, which includes a request for permission to transmit the answer file 120:

Verbot™: I appreciate all your input. But in order for me to send it the show's producers, I need your permission. Will you give it to me now?

User: Yes.

20 Verbot™: Very good. Your opinions will represent statistically about 1200 ordinary citizens. That makes you a pretty powerful person, I'd say. Good, I've transmitted your answers. Talk to you tomorrow, bye.

If the survey participant 118 does not grant permission to transmit the answer file 120 after the lobbying,
25 then the Verbot™ 500, in a next step 798, politely acknowledges the decision of the survey participant 118 not to transmit the answer file 120. In a next step 799, the virtual human interface application 118 switches to a script file 114 designed to lobby for permission. Processing then returns to the noninteractive sleep mode of step 730.

The above description of the operation of the survey system 100 has been provided as an example of one preferred embodiment of the present invention using the local version of the Verbot™ 500. It will be appreciated that,
30 in another preferred embodiment of the present invention using the on-line version of the Verbot™ 500, many of the steps described above are not performed.

Survey Results

Figure 8 is a flow chart showing the operation of a survey results collector 108 in accordance with one embodiment of the present invention. In a first step 802, the survey results collector 108 receives an answer file 120
35 from a survey input client 108. In one embodiment, the answer file 120 is transmitted via FTP using the file transfer capability of an Internet browser on the survey input client 108.

In a next step 804, the survey results collector 108 extracts data from the answer file 120. Preferably, a PERL script residing on the survey results collector 108 is executed, which opens the answer file 120 and processes the answer data in the answer file 120. As will also be appreciated by those of ordinary skill, PERL is a programming language, and PERL instructions may be referenced in an HTML-compliant file and executed by a web server prior to transmitting the web page to a requesting user

In one embodiment, the answer data in the answer file 120 begins with a survey identifier number (e.g. 143) which may be indicated by an informative label, such as, for example "SURVEY~143;" The answer file data is formatted such that each answer is preceded by a question identifier (e.g., "Q0001"), followed by an equals sign ("="), then the text of the answer, followed by a semicolon (";"). That organization is repeated for each answer. The PERL script thus parses through each identifier and answer combination, and isolates the answer data, and then, in a next step 806, loads it into the survey database 122.

More particularly, in the step 806, the script instructions parse each record in the answer file 120, scanning for particular identifiers. The script instructions, upon encountering the "SURVEY~" identifier, save in a string variable the numeric string value which follows. It will be understood that the numeric string serves to uniquely identify one survey in the survey database 122. Next, upon encountering each question identifier "Qxxx~" (where x represents an integer character), the script instructions strip off the "~" and save the question identifier in another string variable. Scanning up to the next ";", the script instructions identify the text of an answer provided by the survey participant 118, then, having established a connection to the survey database 122, add a new record to the ANSWER_TABLE, assigning the survey identifier number to the SURVEY_ID field, the saved question identifier to the QUESTION_ID field, and assigning the text of the answer to an ANSWER_TEXT field. In this manner, the answers are recorded in the survey database 122.

In one embodiment, the script instructions scan the answer file 120 for the string "entertainment_event=" and, if located, identify the subsequent text up to the next ";", and generate a report which, for example, indicates some action to be taken with respect to the survey participant 118, such as, for example, sending tickets to the survey participant 118 allowing him or her to attend a screening of a program.

Figure 9 is a flow chart showing the operation of a survey results server 110 in accordance with one embodiment of the present invention. In a first step 910, the survey results server 110 receives a request for a survey results report 126 from a survey review client 112 over the network 102. Because the survey reviewer 124 uses only a standard web browser, the survey reviewer 124 can advantageously be anywhere in the world, and can use any of a growing number of devices, which support connecting to and browsing the world wide web.

In a next step 912, upon receiving the request for access, the survey results server 110 processes a web page associated with the URL transmitted by the survey reviewer's 124 web browser. Those of ordinary skill in the art will appreciate that the web page is stored as a file including certain content and formatting tags in accordance with HTML (hypertext markup language) protocol. In a preferred embodiment, the web page includes a reference to a PERL script.

In one embodiment, the web page accessed by the survey reviewer 124 includes a reference to a retrieve sponsor script which retrieves sponsor information from the survey database 122. Before transmitting the web page, the survey results server 110, in the step 812, processes the retrieve sponsor script to generate a select sponsor web page. The instructions in the script establish a connection to the survey database 122, and then query the SPONSOR_TABLE for the names of each sponsor. As will be appreciated by those of ordinary skill, it is known in the art to use PERL instructions which formulate queries based on SQL (standard query language) syntax to connect to, query, and populate database tables using database systems such as those provided by Oracle, Sybase, or Microsoft Access. Those of ordinary skill will understand how to use PERL instructions to connect to database tables, to formulate syntactically correct SQL queries, and to return and process recordsets generated by those queries.

The SPONSOR_TABLE query returns a recordset including all records in the SPONSOR_TABLE, and the instructions then process each record in the recordset, extracting the name of the sponsor and combining that name along with HTML-compliant tagging to generate hyperlinks on the select sponsor web page, one hyperlink for each sponsor, each hyperlink labeled by the sponsor name. Also, the script instructions associate a password page URL with each hyperlink and further embed a parameter with each hyperlink specifying the SPONSOR_ID value for the sponsor. It will be appreciated by those of ordinary skill that, when a hyperlink is selected, a URL associated with the hyperlink as well as any parameters and their values are transmitted to the server. The survey results server 110 then transmits the select sponsor web page to the survey results reviewer 124.

In a further step 814, the survey reviewer 124 selects a sponsor hyperlink from the select sponsor web page, and the survey reviewer's 124 web browser transmits the password page URL along with the SPONSOR_ID parameter to the survey results server 110. In a step 816, the survey results server 110 receives the password page URL, retrieves the HTML file for that web page and processes script instructions referenced in the HTML file. The script instructions embed the SPONSOR_ID as a parameter for an ENTER hyperlink, which has an associated URL for a select survey web page. The survey server 110 transmits the password page with its updated ENTER hyperlink to the survey reviewer's 124 web browser.

In another step 818, the survey results reviewer 124 fills in USER and PASSWORD fields on the password page and clicks the ENTER hyperlink. In a next step 820, the survey results server 110 receives the user and password values entered by the survey reviewer 124, the SPONSOR_ID parameter, and also the URL for the select survey web page. The survey results server 110 accesses the HTML file associated with the select survey web page and processes script instructions embedded in the HTML file. The script instructions establish a connection with the survey database 122 and query the PASSWORD_TABLE using the SPONSOR_ID parameter. The query returns a recordset wherein each record holds a valid user and password for the sponsor. The script instructions then compare the user and password values entered by the survey reviewer 124 against the user and password values in each record of the recordset. If a match is not located, then the survey results server 110 transmits to the survey reviewer 124 a web page which displays the message "Invalid password, please use your browser's 'back' button and try again." If a match is located, and the survey reviewer 124 is thus authenticated, then, in another step 822 the survey

results server 110 accesses the HTML file associated with the select survey URL and processes script instructions referenced therein. The script instructions establish a connection to the survey database, and query the SURVEY_TABLE using the SPONSOR_ID parameter. The query returns a recordset including information about all surveys for the particular sponsor. The instructions process each record in the recordset to generate a survey report
5 hyperlink for each survey, wherein each hyperlink has an associated survey report URL and an embedded SURVEY_ID parameter. The survey results server 110 thus generates the select survey web page and then transmits it to the survey reviewer 124.

In a next step 924, the survey reviewer 124 selects a survey report hyperlink. In a further step 926, the survey results server 110 receives the survey report URL and the SURVEY_ID parameter. The survey results server
10 110 accesses the HTML file associated with the survey report URL and processes script instructions referenced therein. The script instructions connect to and query the survey database 122 to extract the latest data available for the survey identified by the SURVEY_ID parameter. In one embodiment, PERL instructions formulate an SQL query using the unique SURVEY_ID value to return a recordset identifying all questions in the QUESTION_TABLE (by unique QUESTION_ID) from the selected survey. The instructions process the question recordset to store the text of each
15 question in an array and each ANSWER_FORMAT value for each question in another array. In one embodiment, the PERL instructions use a PUSH function to add the question text and the ANSWER_FORMAT values to the respective arrays, each PUSH enlarging the array by one element. The instructions formulate further SQL queries to extract all answer data from the ANSWER_TABLE using each unique QUESTION_ID. The script instructions further store the retrieved answer data in additional PERL arrays.

In a next step 928, the survey results server 110, having extracted answer data from the survey database 122, executes additional script instructions comprising a loop to (1) format the text for the next question using HTML
20 formatting tags, (2) examine the ANSWER_FORMAT value for the question, and (3) depending on the ANSWER_FORMAT value, perform the appropriate comparisons, calculations or transformations on the answer data and add HTML formatting tags to present the results in the proper form. For example, if the ANSWER_FORMAT value for the question is "top5and%", then the instructions examine (e.g., string compare) all answers for the question,
25 identifying the five answers that appeared most frequently, then compute a percentage of the total number of answers for each of the five identified answers, then add HTML formatting tags to present the data, for example, in a tabular form, with the top 5 answers listed vertically in a first column, with the percent of total answers value in the right column. It will be appreciated by those of ordinary skill that virtually endless different ANSWER_FORMAT
30 values may advantageously be used to process and present answer data in a completely automated manner.

In another step 930, after formatting the question and answer data as needed in HTML format to dynamically generate a survey results report 126, the survey results server 110 transmits the current survey results report 126 to the survey reviewer's 124 web browser, which displays the current survey results report 126 to the
survey reviewer 124. Figure 4 illustrates a representation of one such survey results report 126. Those of ordinary

skill in the art will appreciate that countless formats may be used to present results in a survey results report 126, and the present invention is not limited by any particular format of result data.

5 This invention may be embodied in other specific forms without departing from the essential characteristics as described herein. The embodiments described above are to be considered in all respects as illustrative only and not restrictive in any manner. The scope of the invention is indicated by the following claims rather than by the foregoing description.

WHAT IS CLAIMED IS:

1. A survey system comprising:
a computer comprising:
a computer readable medium;
5 a processor;
a display;
a script comprising question data representing a question, said script stored in said computer readable medium;
character image data representing a character communicating information, said character image
10 data stored in said computer readable medium; and
a user interface running on said processor, said user interface configured to access said script, said user interface configured to process said question data to communicate said question to a user, an image generator of said user interface configured to process said character image data to create a representation of said character on said display during said communication of said question, said user interface configured
15 to receive and process response data representing a response provided by said user, said user interface configured to store in said computer readable medium answer data representing said response provided by said user.
2. The survey system as described in Claim 1, said script further comprising:
response pattern data representing a portion of a possible user response to said question; and
20 lobbying data representing lobbying information to persuade said user to respond to said question by providing a response that does not include said at least one portion of said response to said question, said user interface configured to compare said response data to said response pattern data to determine whether said response provided by said user includes said predetermined portion of a possible response and, if so, to process said lobbying data to communicate said lobbying information to said user without storing said
25 answer data.
3. The survey system as described in Claim 2, said script further comprising:
expression data representing an expression of said character, said user interface configured to process said expression data to cause said image generator to process said character image data to change
30 an expression of said representation of said character on said display during said communication of said lobbying information.
4. The survey system as described in Claim 1, said script further comprising:
response pattern data representing a portion of a possible user response to said question; and
advertising data representing advertising information to inform said user about goods or services,
said user interface configured to compare said response data to said response pattern data to determine

whether said response provided by said user includes said predetermined portion of a possible response and, if so, to process said advertising data to communicate said advertising information to said user.

5. The survey system as described in Claim 1, said script further comprising:

5 response pattern data representing a portion of a possible user response to said question; and
entertainment data representing an offer of an incentive or reward to said user for answering said question, said user interface configured to compare said response data to said response pattern data to determine whether said response provided by said user includes said predetermined portion of a possible response and, if so, to process said entertainment data to communicate or present said offer to said user.

6. The survey system as described in Claim 1, said script further comprising:

10 first response pattern data representing a portion of a first possible user response to said question;
second response pattern data representing a portion of a second possible user response to said question; and

predetermined answer data, said user interface configured to compare said response data to said first response pattern data and to compare said response data to said second response pattern data, and, if
15 either comparison results in a match, to store said predetermined answer data in said computer readable medium without storing said answer data.

7. The survey system as described in Claim 6, said script further comprising:

third response pattern data representing a portion of a third possible user response to said question; and

20 second predetermined answer data, said user interface configured to compare said response data to said third response pattern data and, if that comparison results in a match, to store said second predetermined answer data in said computer readable medium without storing said answer data.

8. The survey system as described in Claim 1, said script further comprising:

25 expression data representing an expression of said character, said user interface configured to process said expression data to cause said image generator to process said character image data to change an expression of said representation of said character on said display during said communication of said question.

9. The survey system as described in Claim 8, said system further comprising:

30 remote character image data residing on a computer readable storage medium operably connected to a second computer, said second computer in network contact with said first computer, said user interface configured to request and receive said remote character image data to cause said image generator to process said remote character image data to change an expression of said representation of said character on said display during said communication of said question.

10. The survey system as described in Claim 1, said system further comprising:

35 a voice output module creating audio output to communicate said question to said user.

11. The survey system as described in Claim 10, wherein said user interface is configured to communicate said question by displaying text characters during a time when said voice output module communicates said question by creating audio output.

5 12. The survey system as described in Claim 1, said system further comprising:
response pattern data representing a portion of a possible user response to said question;
remote lobbying data representing lobbying information to persuade said user to respond to said question by providing a response that does not include said at least one portion of said response to said question, said remote lobbying data residing on a computer readable storage medium operably connected to a second computer, said second computer in network contact with said first computer; and
10 a voice output module, said user interface configured to compare said response data to said response pattern data to determine whether said response provided by said user includes said predetermined portion of a possible response and, if so, to request and receive said remote lobbying data and to route data to said voice output module to create audio output representing said lobbying information.

13. The survey system as described in Claim 1, said system further comprising:
15 a voice recognition module creating said response data from a spoken response provided by said user.

14. The survey system as described in Claim 1, further comprising:
an answer format value representing a predetermined format for answer data provided in response to said question; and
20 survey result instructions formatting said answer data in accordance with said answer format value to create a survey result report.

15. The survey system as described in Claim 14, further comprising:
survey collection instructions, said survey collection instructions associating said stored answer data with second answer data provided by a second user in response to said question, said survey result instructions formatting said second answer data to create said survey result report.
25

16. The survey system as described in Claim 15, wherein said survey result instructions and said survey collection instructions are executed by a processor of a second computer.

17. The survey system as described in Claim 16, further comprising:
a third computer connected by a network to said second computer, said second computer
30 transmitting said survey result report to said third computer in response to a request for said survey result report.

18. The survey system as described in Claim 17, wherein said network is the Internet, said survey result report is a web page, and an Internet browsing application running on said third computer issues said request for said survey result report.

35 19. A survey system comprising:

question data representing a question;

response pattern data representing at least one portion of a response to said question;

lobbying data representing lobbying information to persuade a user to respond to said question by providing a response that does not include said at least one portion of said response to said question; and

5 a user interface configured to process said question data to communicate said question to a user, said user interface configured to receive and process response data representing a response provided by said user, said user interface configured to compare said response data to said response pattern data to determine whether said response provided by said user includes said at least one portion of said response to said question and, if so, to process said lobbying data to communicate said lobbying information to said user.

10 20. The survey system as described in Claim 19, wherein said user interface is further configured to generate and store answer data representing said response provided by said user if said user interface determines that said response provided by said user does not include said at least one portion of said response to said question.

15 21. The survey system as described in Claim 20 wherein said user interface is configured to transmit character image data to a display during said communication of said question and said lobbying information, said character image data rendered to represent a character communicating said question and said lobbying information.

22. A system for questioning a survey participant, the system comprising:

a computer readable storage;

a display;

20 a survey script stored on said computer readable storage, said survey script comprising question data, expression data, and response processing instructions; and

25 a virtual human interface comprising an image generator, a script parser and a response parser, said script parser parsing said question data to present a question to a user, said image generator displaying on said display a representation of an animated character communicating information during said presenting of said question, said script parser parsing said expression data to cause said image generator to change said representation of said character, said script parser parsing said response processing instructions to identify at least one portion of one possible response by said user to said question and to associate predetermined responsive steps with said response, said response parser parsing input provided by said user to detect said portion of said one possible response and to perform said predetermined responsive steps, said predetermined responsive steps including storing in an answer file answer data representing said input provided by said user and storing in said answer file in association with said answer data a question identifier identifying said question.

30 23. The system as described in Claim 22, wherein said image generator comprises software instructions stored on said computer readable storage, said script parser comprises software instructions stored on said computer readable storage, and said response parser comprises software instructions stored on said computer readable storage.

24. The survey system as described in Claim 22, further comprising:
an answer format value representing a predetermined format for answer data provided in response
to said question, said answer format value having a predetermined association with said question identifier;
and

5 a survey report generator configured to parse said answer file for said question identifier and to
retrieve said answer format value and to use said answer format value to format said answer data in a
survey result report.

25. A method of conducting a survey, said method comprising the steps of:
providing in a computer readable medium question data representing a question;
10 providing in said computer readable medium character image data representing a character
communicating information;
communicating said question to a user by processing said question data;
displaying a representation of said character on a display during said communication of said
question by processing said character image data; and
15 responding to a response to said question provided by a user by storing in said computer readable
medium answer data representing said response.

26. The method as described in Claim 25, said method comprising the further steps of:
providing in said computer readable medium response pattern data representing a portion of a
possible user response to said question;
20 providing in said computer readable medium lobbying data representing lobbying information to
persuade said user to respond to said question by providing a response that does not include said at least
one portion of said response to said question;
comparing response data representing said response provided by said user to said response pattern
data; and

25 communicating said lobbying information to said user by processing said lobbying data without
storing said answer data if said response data includes said response pattern data.

27. The method as described in Claim 26, said method comprising the further steps of:
providing in said computer readable medium expression data representing an expression of said
character; and
30 processing said expression data to change an expression of said displayed character during said
communication of said lobbying information.

28. The method as described in Claim 26, said method comprising the further steps of:
providing in said computer readable medium response pattern data representing a portion of a
possible user response to said question;

providing in said computer readable medium advertising data representing advertising information to inform said user about goods or services;

comparing response data representing said response provided by said user to said response pattern data; and

5 communicating said advertising information to said user by processing said advertising data if said response data includes said response pattern data.

28. The method as described in Claim 25, said method comprising the further steps of:

providing in said computer readable medium response pattern data representing a portion of a possible user response to said question;

10 providing in said computer readable medium entertainment data representing an offer of an incentive or reward to said user for answering said question;

comparing response data representing said response provided by said user to said response pattern data; and

15 communicating or presenting said entertainment information to said user by processing said entertainment data if said response data includes said response pattern data.

30. The method as described in Claim 25, said method comprising the further steps of:

providing in said computer readable medium first response pattern data representing a portion of a first possible user response to said question;

20 providing in said computer readable medium second response pattern data representing a portion of a second possible user response to said question;

providing in said computer readable medium predetermined answer data;

comparing response data representing said response provided by said user to said first response pattern data;

25 comparing said response data representing said response provided by said user to said second response pattern data; and

storing in said computer readable medium said predetermined answer data without storing said answer data if either comparison results in a match.

31. The method as described in Claim 25, said method comprising the further steps of:

30 providing in said computer readable medium expression data representing an expression of said character;

providing in said computer readable medium response pattern data representing a portion of a possible user response to said question, said response pattern data having a predetermined relationship with said expression data;

35 comparing response data representing said response provided by said user to said response pattern data; and

changing an expression of said displayed character by processing said expression data if said comparison results in a match.

32. The method as described in Claim 31, wherein said expression data resides on a remote computer readable medium and wherein said processing of said expression data includes requesting said expression data from said remote computer readable medium and transmitting said expression data from said remote computer readable medium.

33. The method as described in Claim 28, wherein said advertising data resides on a remote computer readable medium and wherein said processing of said advertising data includes requesting said advertising data from said remote computer readable medium and transmitting said advertising data from said remote computer readable medium.

34. The method as described in Claim 33, wherein said advertising data includes voice data and references to expression data, the method comprising the further steps of:

providing in said computer readable medium expression data representing an expression of said character;

communicating said advertising data as voice output by processing said voice data; and

changing an expression of said displayed character by processing expression data referenced by said advertising data if said comparison results in a match.

35. A method for processing survey data, said method comprising the steps of:

providing in a computer readable medium question data representing a question;

providing in said computer readable medium an answer format value representing a predetermined format for answer data provided in response to said question;

associating a question identifier with said question data;

associating said answer format value with said question identifier;

receiving an answer file in said computer readable medium, said answer file including answer data representing a response provided by a user to said question, and said answer file including said question identifier;

parsing said answer file for said question identifier;

identifying said answer format value using said association with said question identifier; and

formatting said answer data in accordance with said predetermined format for answer data.

36. The method as described in Claim 35, wherein the steps described therein are represented by software instructions stored in a computer readable medium.

37. A survey system comprising:

means for representing a question in a computer readable medium;

means for representing in said computer readable medium a portion of a first possible user response to said question;

means for communicating said question to a user;

means for generating a representation of a character communicating said question during said communication of said question;

5

means for receiving from said user a representation of a response provided by said user to said question;

means for performing a first comparison, said first comparison comprising comparing said representation of said response to said representation of said portion of said first possible user response; and

means for storing an answer representing said response if said first comparison results in a match.

10

38. The survey system as described in Claim 27, the survey system further comprising:

means for representing in said computer readable medium a portion of a second possible user response to said question;

means for representing in said computer readable medium lobbying information to persuade a user to respond to said question by providing a response that does not include said portion of said second possible user response to said question;

15

means for performing a second comparison, said second comparison comprising comparing said representation of said response to said representation of said portion of said second possible user response; and

means for communicating said lobbying information to said user without storing said answer data if said second comparison results in a match.

20

39. The survey system as described in Claim 37, the survey system further comprising:

means for changing an expression of said representation of said character during said communication of said question.

40. The survey system as described in Claim 37, the survey system further comprising:

25

means for representing advertising information in a computer readable medium, said advertising information informing a user about goods or services; and

means for communicating said advertising information to said user if said first comparison results in a match.

41. The survey system as described in Claim 37, the survey system further comprising:

30

means for representing entertainment information in a computer readable medium, said entertainment information rewarding or offering to reward said user for providing a response to said question; and

means for communicating or presenting said entertainment information to said user if said first comparison results in a match.

35

42. A survey system comprising:

a network;

a server coupled to said network, said server comprising a server processor and a server computer readable storage;

5 a first client coupled to said network, said first client comprising a first processor, a first computer readable storage, and a first display;

a script file stored on said server computer readable storage, said script file containing survey question data, said server computer configured to respond to a request for said script file from said first client by transmitting said script file to said first client; and

a virtual human interface stored on said first computer readable storage,

10 said virtual human interface configured to display a representation of a human face on said first display,

said virtual human interface configured to parse said script file and present said survey question data to a user,

15 said virtual human interface configured to change said representation of said human face during said presentation of said survey question data, and

said virtual human interface configured to record a response of said user in an answer file, said virtual human interface configured to transmit said answer file to said server computer, said server computer configured to process said answer file and at least one additional answer file to generate a report including formatted answer data.

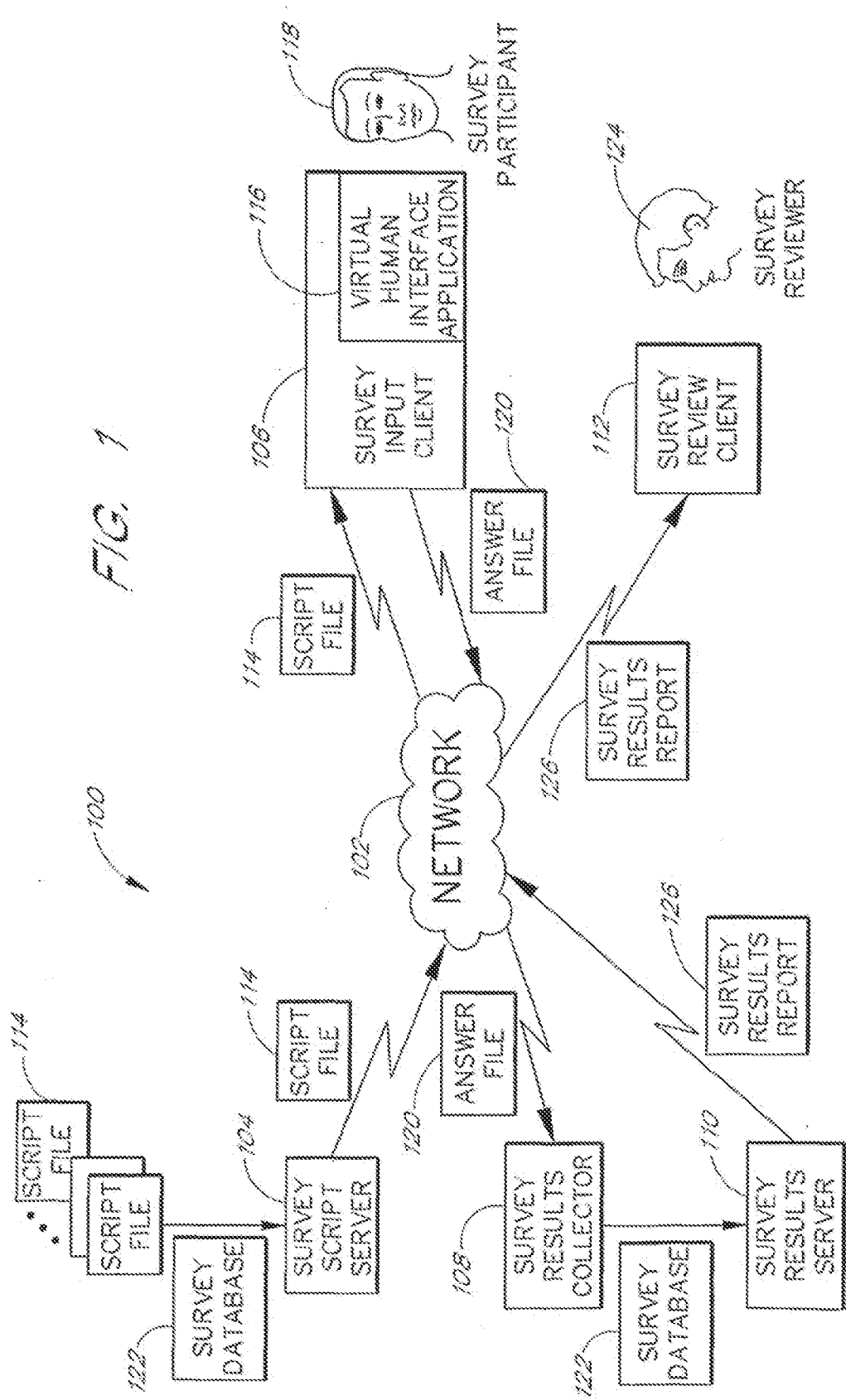


FIG. 1

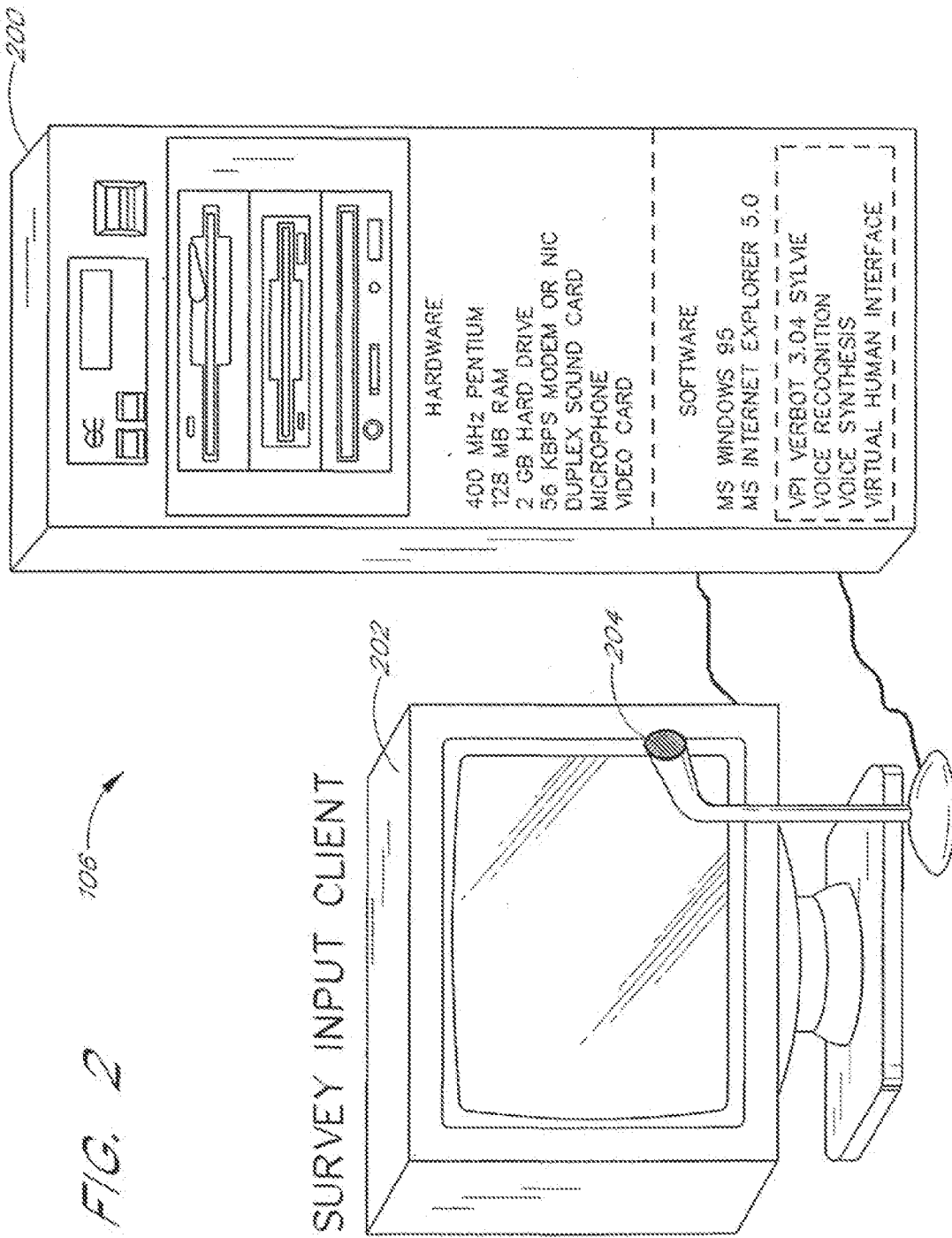
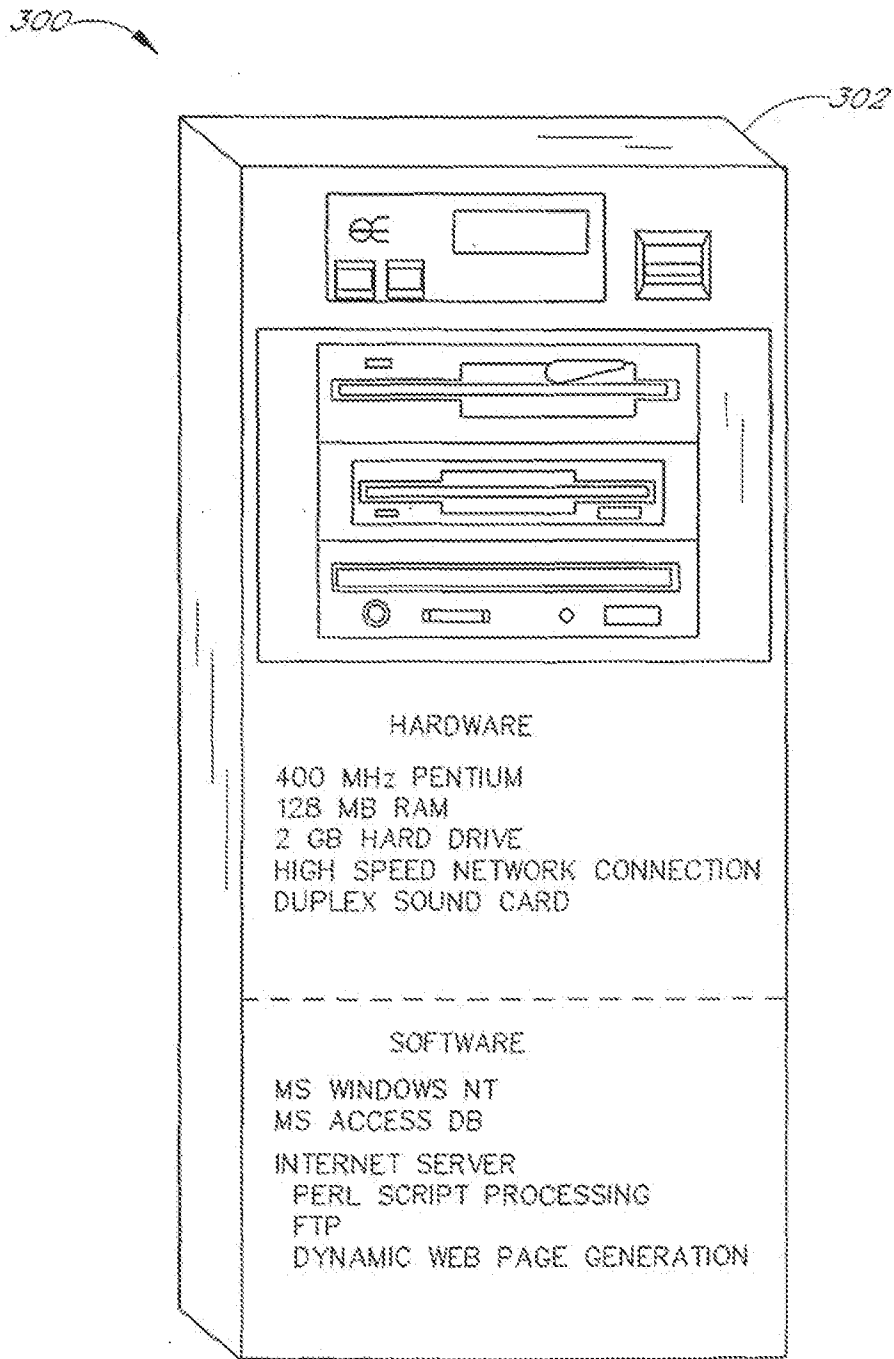


FIG. 2



SURVEY SERVER

FIG. 3

126

**Galaxy Network Daily
Statistics Page
For:**

VPI Real-Time
Reponse Tabulator
Running Statistics for ABC
Productions: 12/22/00

Viewers are currently rating yesterday's show:

How Did You Like Quiet Waters?

Strong Positive	Weak Positive	Neutral	Weak Negative	Strong Negative	# Respondents
3%	8%	64%	24%	1%	8,422

How Did You Like Sex in the City?

12%	13%	21%	34%	30%	8,221
-----	-----	-----	-----	-----	-------

What Is Your Favorite Coffee?

Starbucks	Maxwell House	MJB	House	Gourmet	# Respondents
60%	12%	15%	9%	4%	9,101

The content of this page will change daily or new pages will be posted for each new survey. There will be a running "percentage of universe" figure for each respondent pool.

FIG. 4

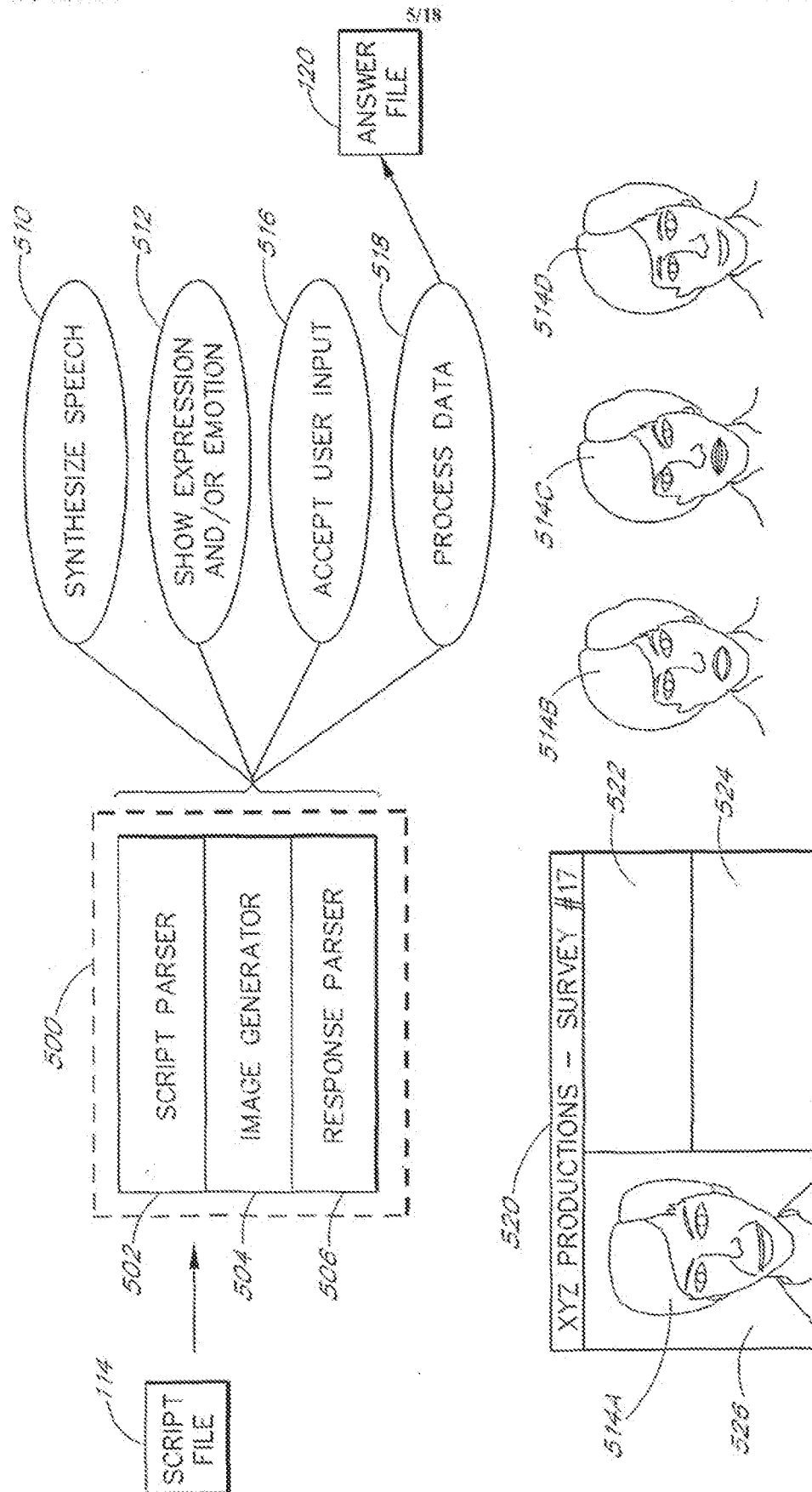


FIG. 5



© - Copyright, Virtual Personalities, Inc., 1999

FIG. 5A



© - Copyright, Virtual Personalities, Inc., 1999

FIG. 5B



© - Copyright, Virtual Personalities, Inc., 1999

FIG. 5C



© - Copyright, Virtual Personalities, Inc., 1999

FIG. 5D

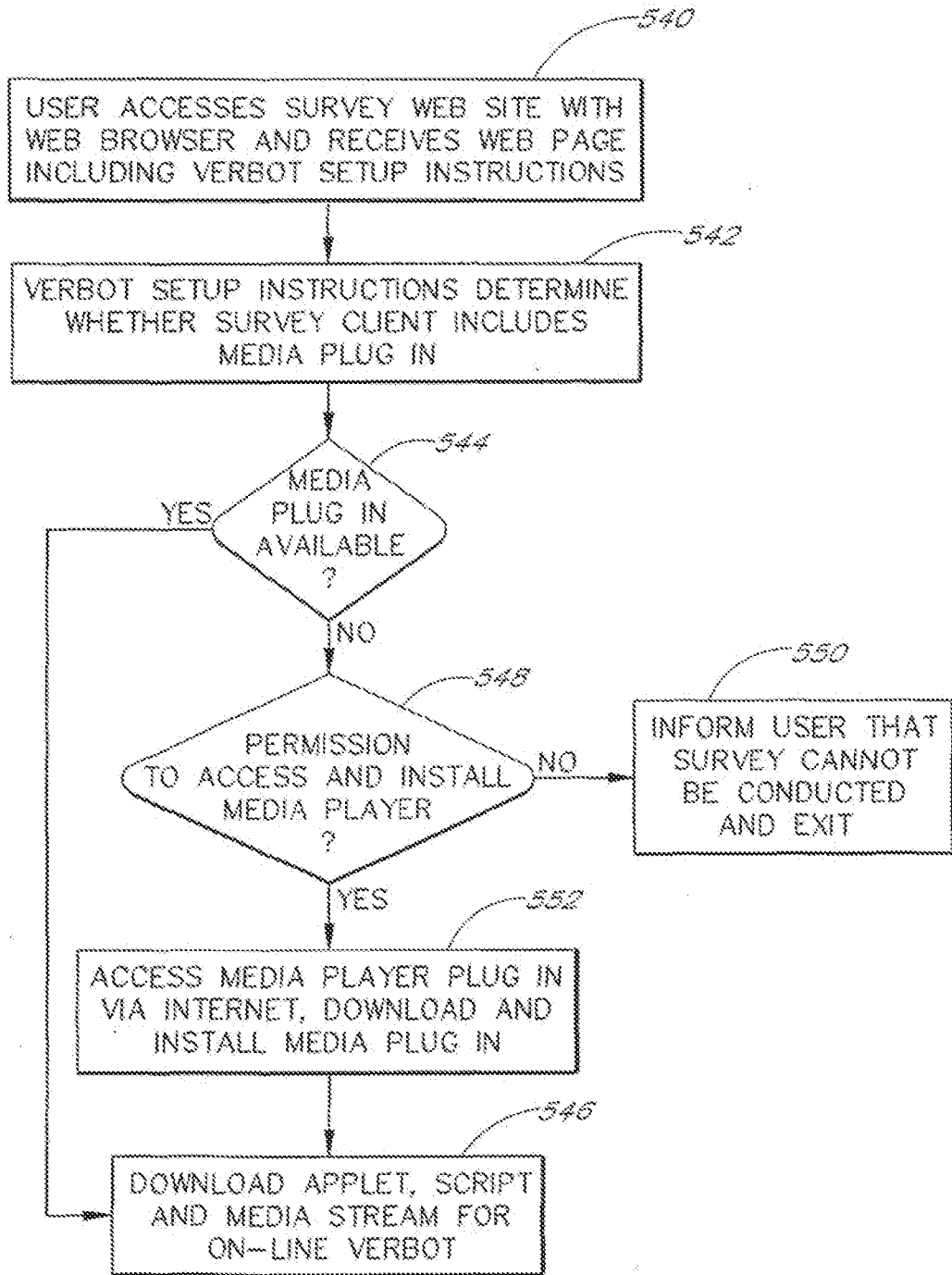


FIG. 5E

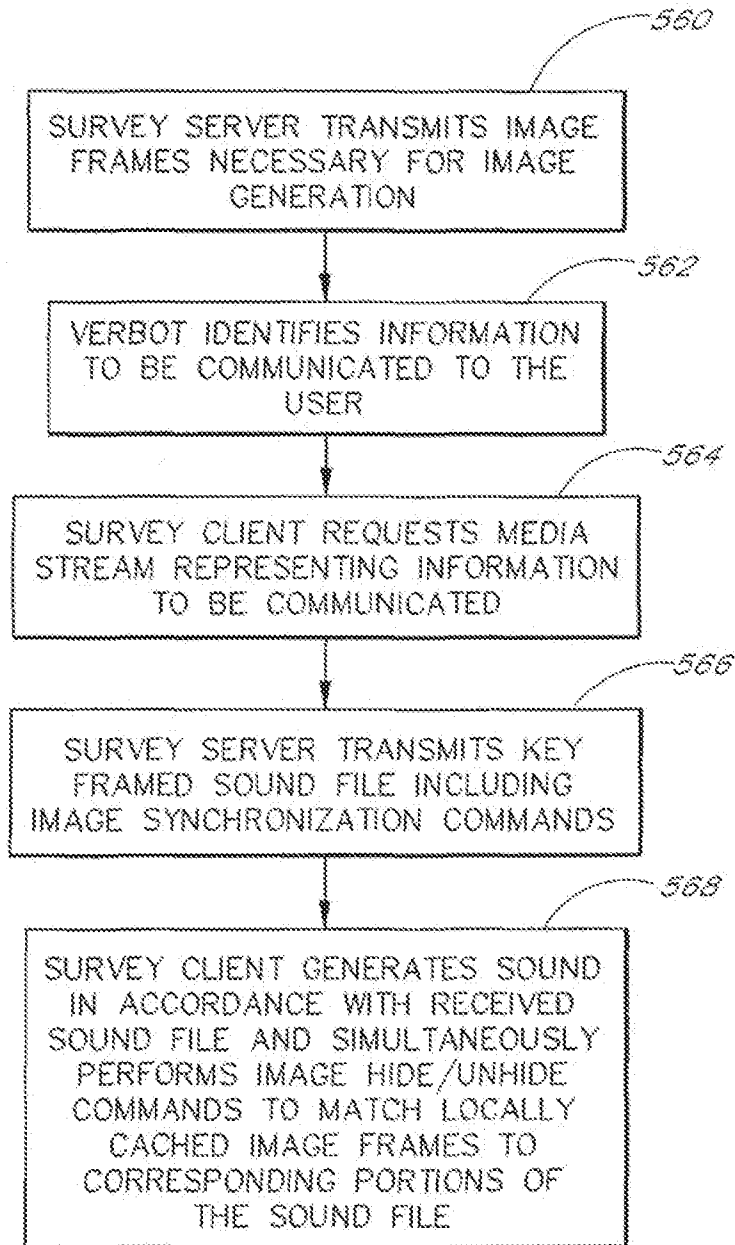


FIG. 5F

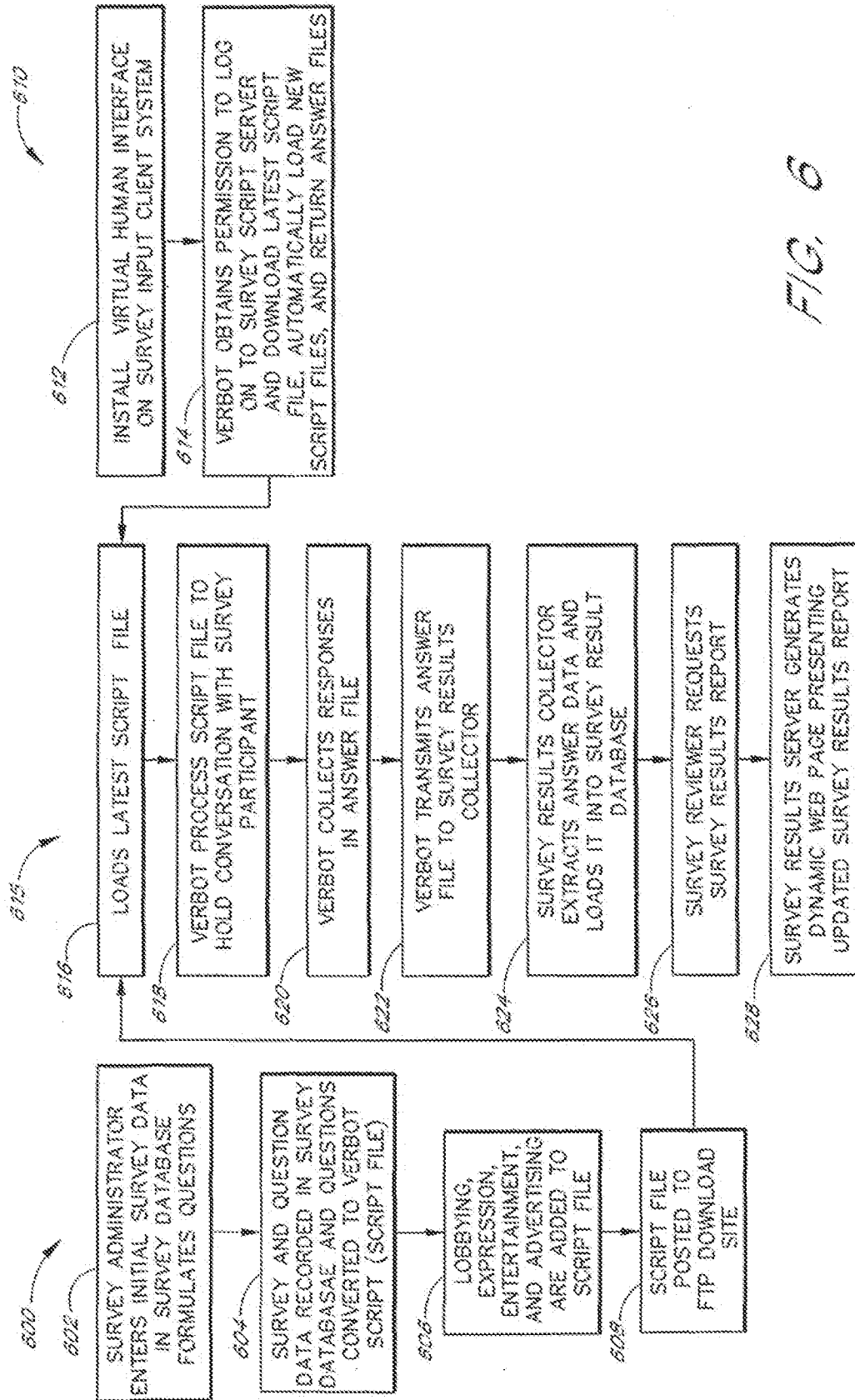


FIG. 6

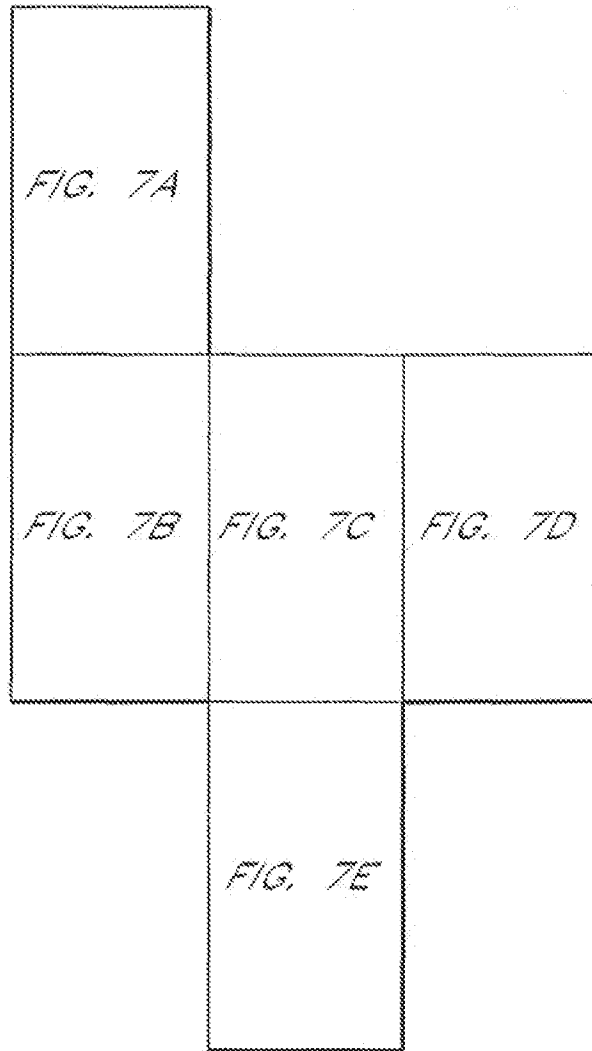
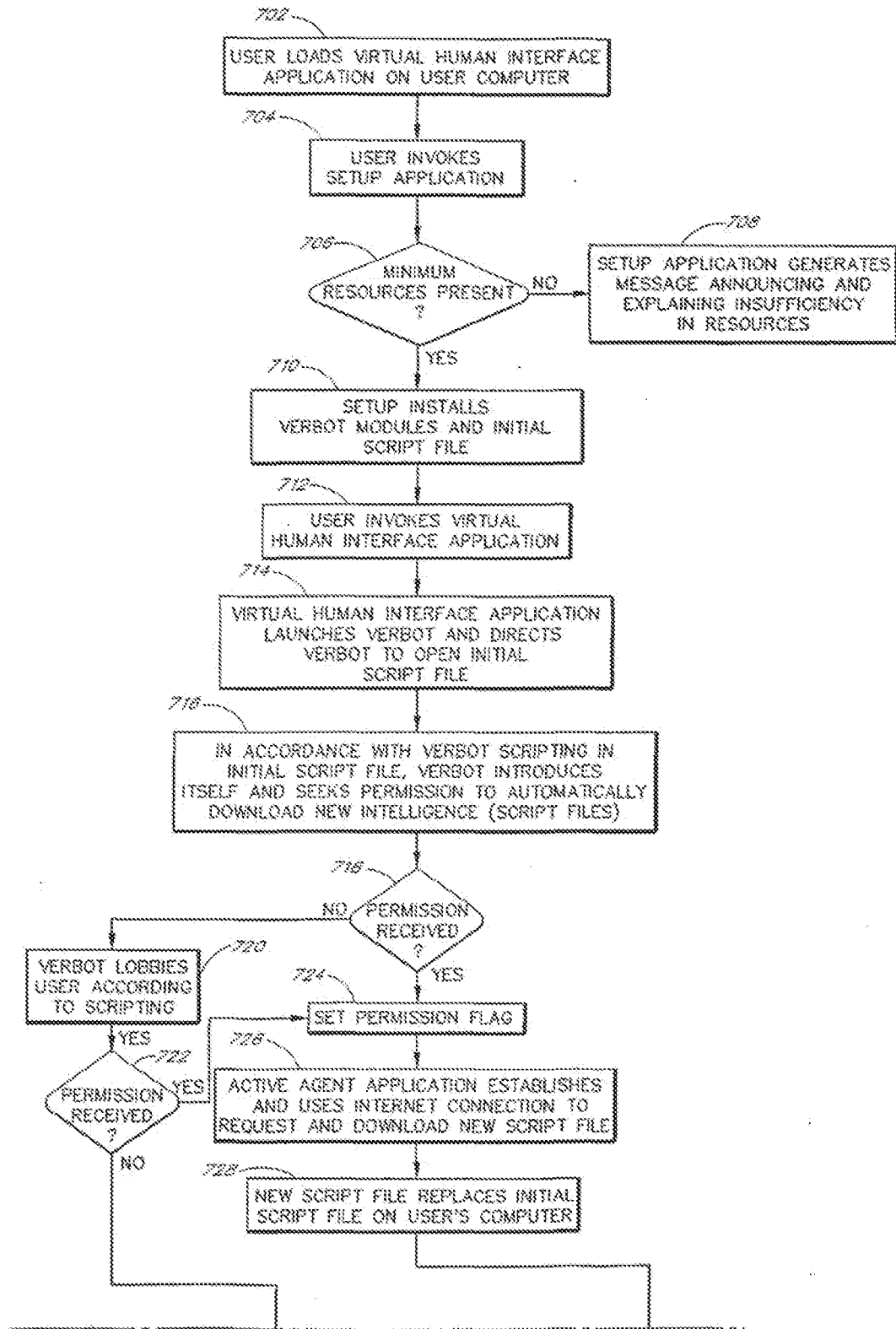


FIG. 7

FIG. 7A



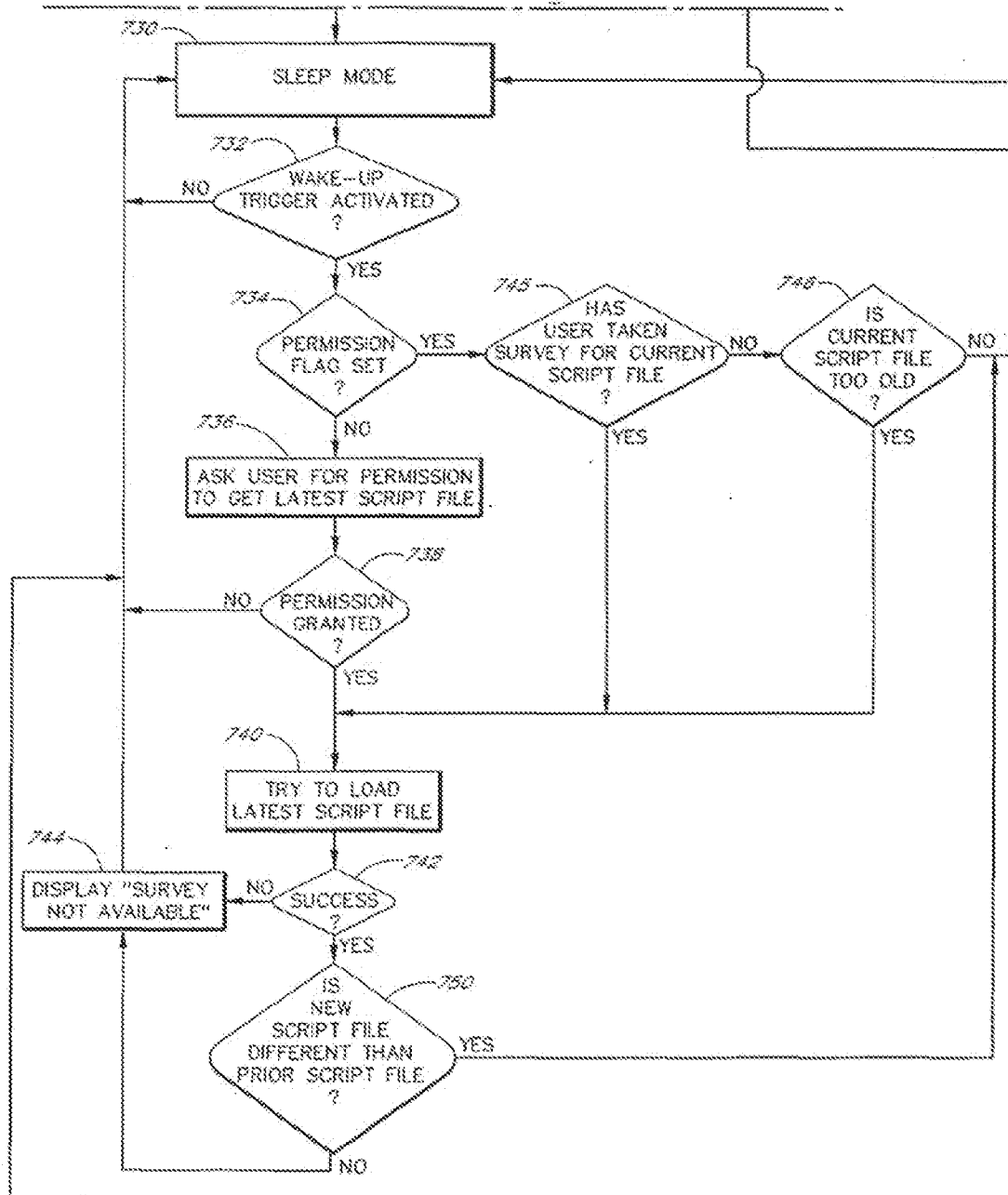


FIG. 7B

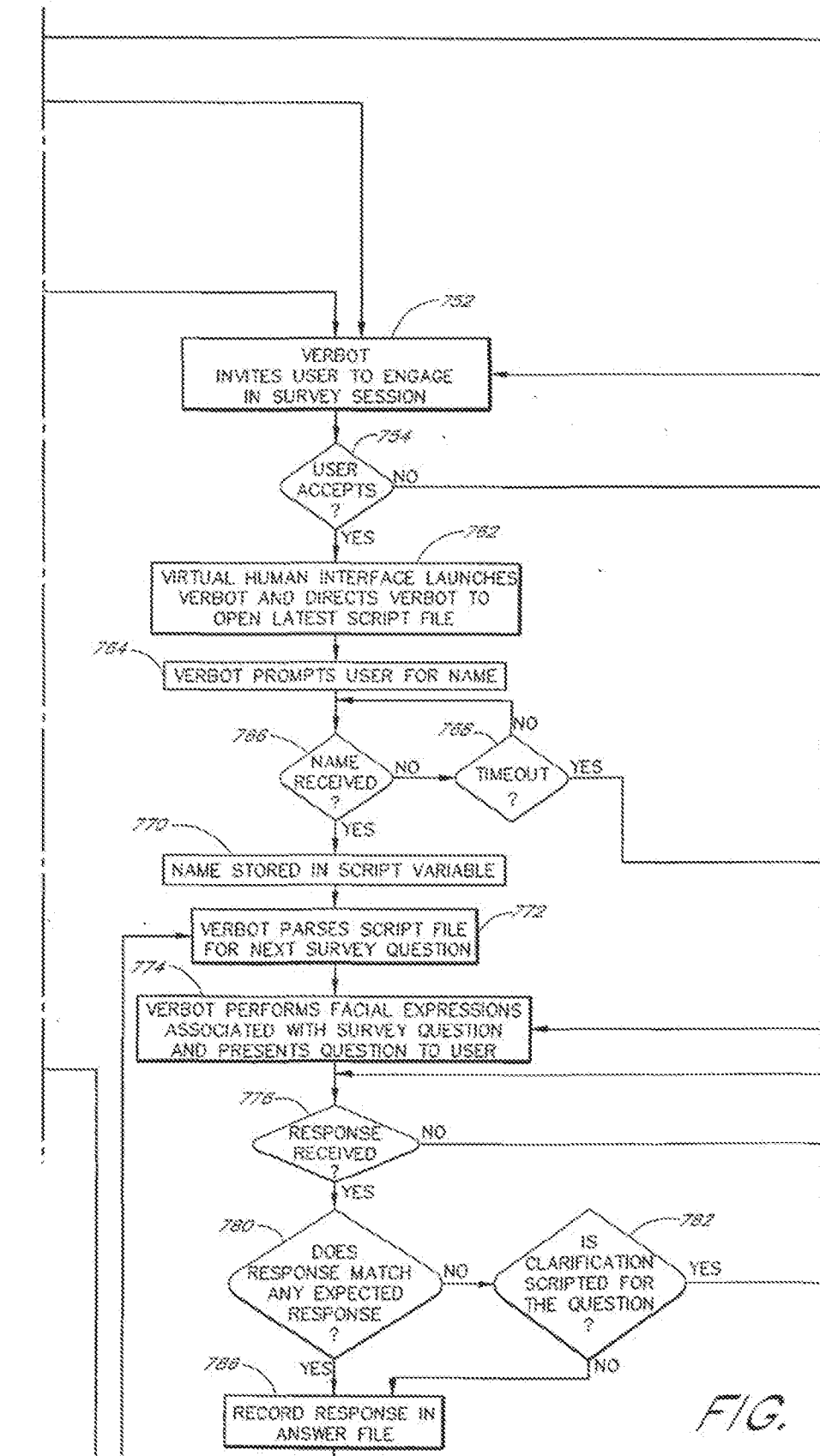


FIG. 7C

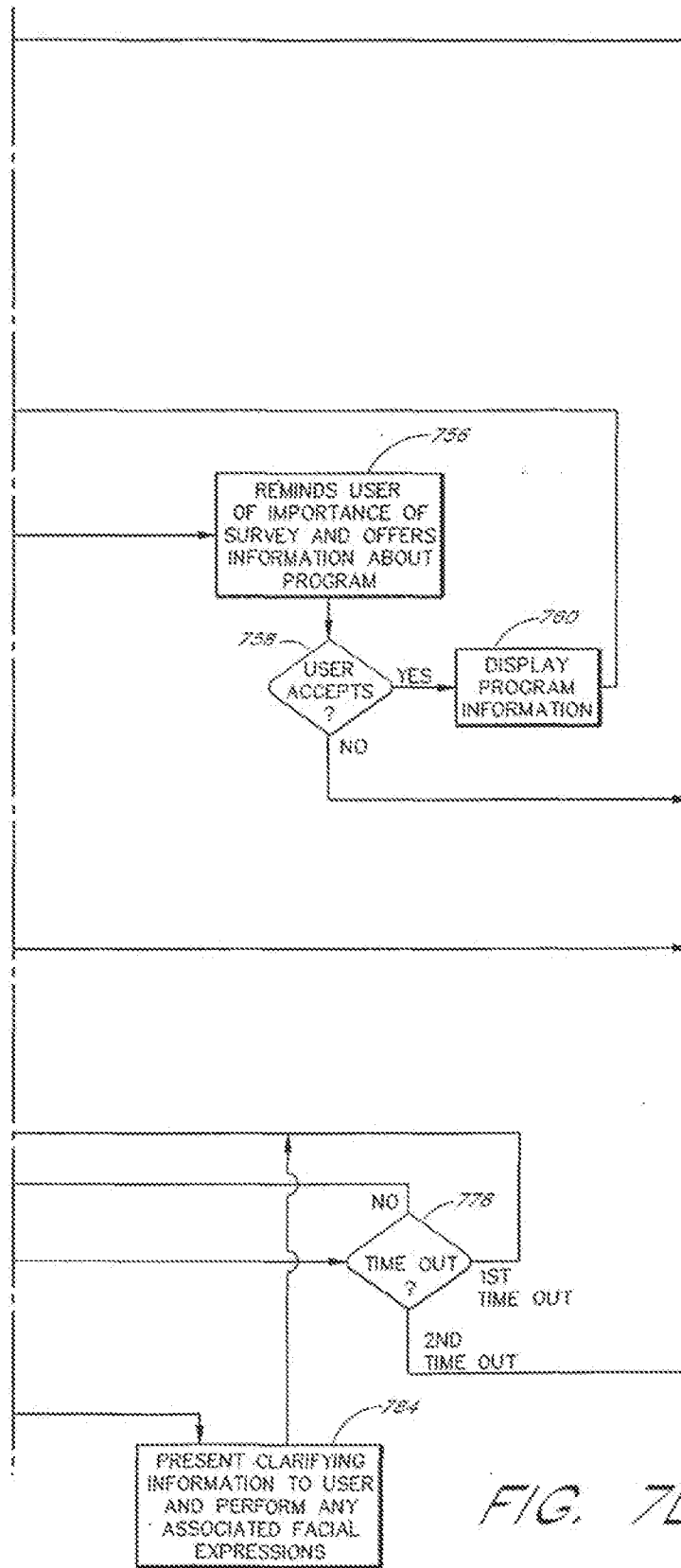
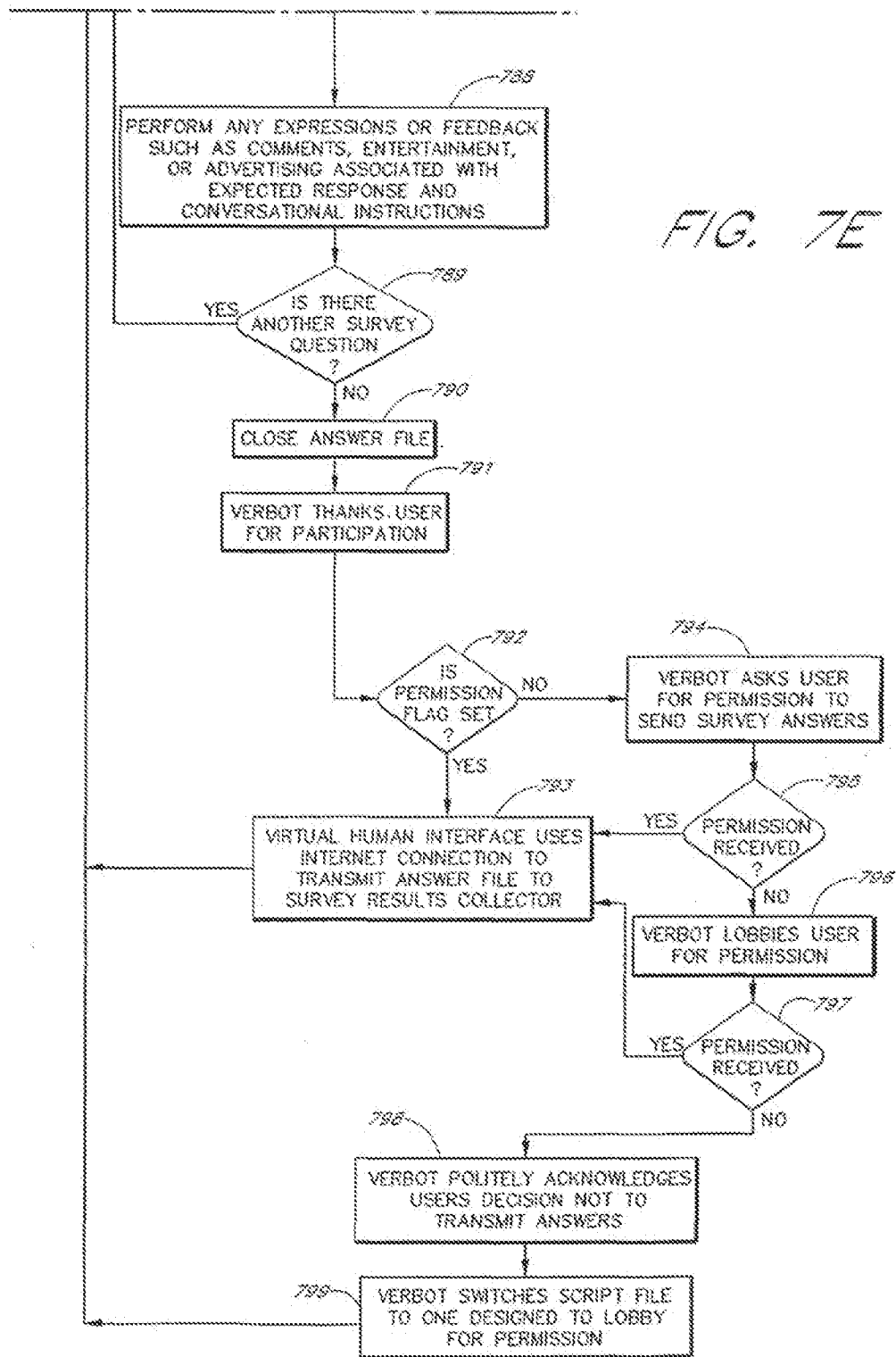


FIG. 7D



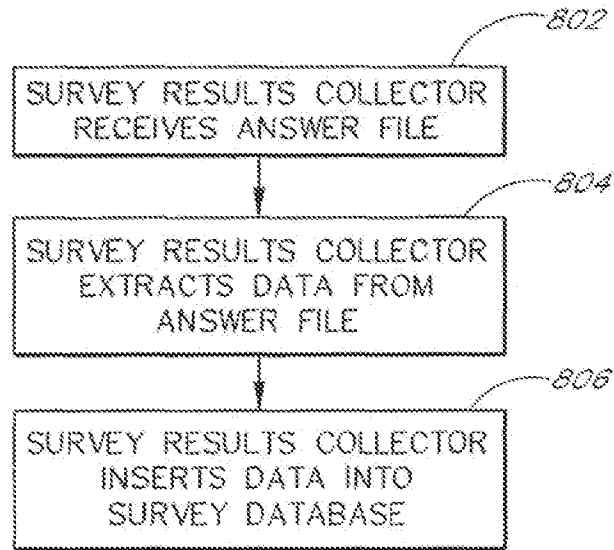


FIG. 8

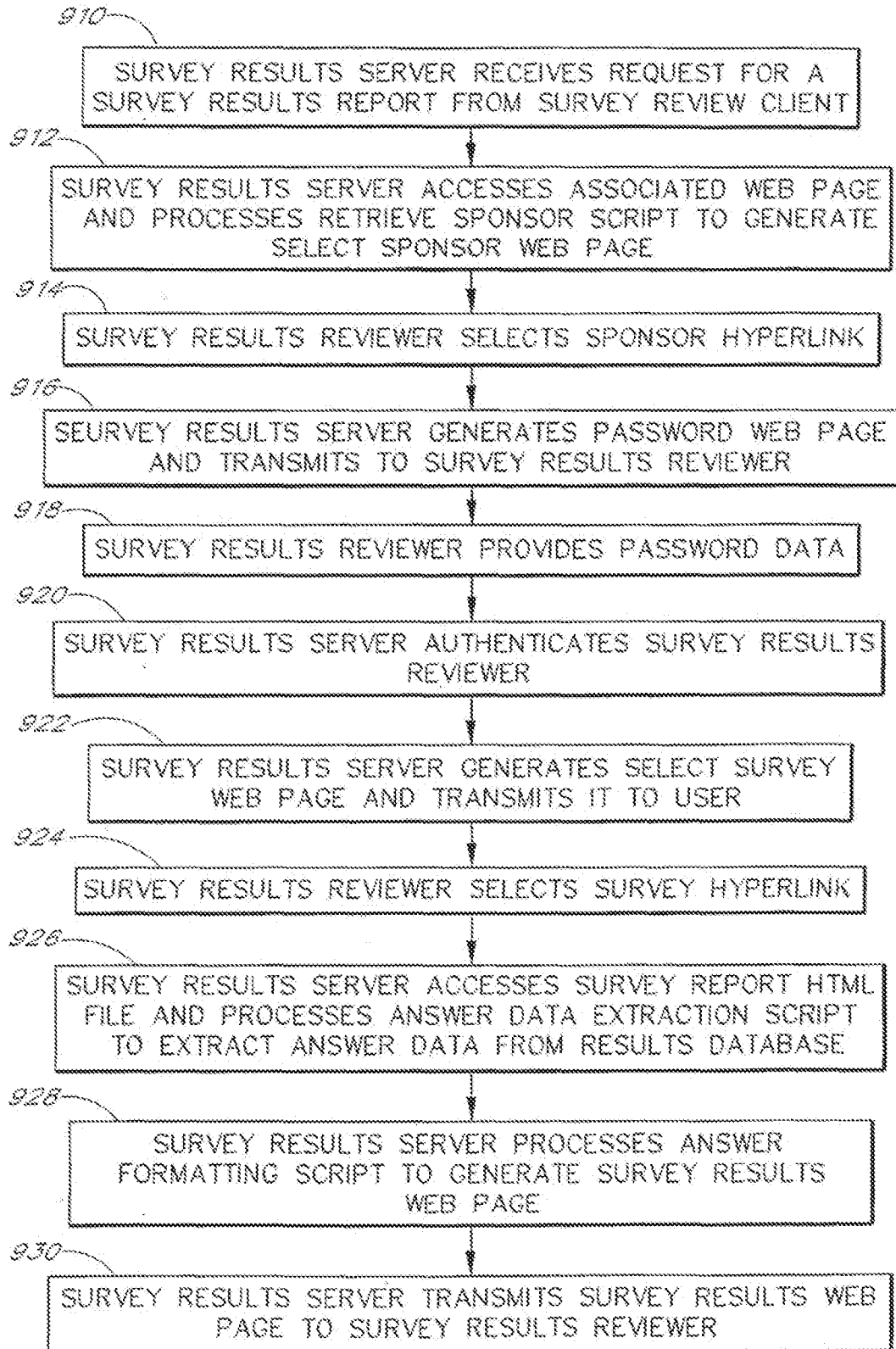


FIG. 9

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 00/34949

<p>A. CLASSIFICATION OF SUBJECT MATTER IPC 7 G06F17/60</p> <p>According to International Patent Classification (IPC), or to both national classification and IPC</p>														
<p>B. FIELDS SEARCHED</p> <p>Minimum documentation searched (classification system followed by classification symbols) IPC 7 G06F</p> <p>Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched</p> <p>Electronic data base consulted during the international search (name of data base and, where practical, search terms used) EPO-Internal, WPI Data, PAJ</p>														
<p>C. DOCUMENTS CONSIDERED TO BE RELEVANT</p> <table border="1"> <thead> <tr> <th>Category*</th> <th>Citation of document, with indication, where appropriate, of the relevant passages</th> <th>Relevant to claim No.</th> </tr> </thead> <tbody> <tr> <td>X</td> <td>WO 98 55903 A (BENSON SCOTT S ;DILLINGER RAY S (US); NEUROMEDIA INC (US); TACKETT) 10 December 1998 (1998-12-10) page 1, line 32 -page 5, line 22</td> <td>1-42</td> </tr> <tr> <td>X</td> <td>WO 96 05685 A (DIGITAL SYSTEMS INT) 22 February 1996 (1996-02-22) page 2, line 25 -page 5, line 5 page 6, line 7 -page 7, line 36 page 9, line 35 -page 26, line 25</td> <td>1-42</td> </tr> <tr> <td>X</td> <td>US 5 630 017 A (MATTHEWS III JOSEPH H ET AL) 13 May 1997 (1997-05-13) column 2, line 58 -column 5, line 16; figures 7A,8A column 6, line 15 -column 7, line 38 column 2, line 63 -column 5, line 16</td> <td>1-42</td> </tr> </tbody> </table>			Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.	X	WO 98 55903 A (BENSON SCOTT S ;DILLINGER RAY S (US); NEUROMEDIA INC (US); TACKETT) 10 December 1998 (1998-12-10) page 1, line 32 -page 5, line 22	1-42	X	WO 96 05685 A (DIGITAL SYSTEMS INT) 22 February 1996 (1996-02-22) page 2, line 25 -page 5, line 5 page 6, line 7 -page 7, line 36 page 9, line 35 -page 26, line 25	1-42	X	US 5 630 017 A (MATTHEWS III JOSEPH H ET AL) 13 May 1997 (1997-05-13) column 2, line 58 -column 5, line 16; figures 7A,8A column 6, line 15 -column 7, line 38 column 2, line 63 -column 5, line 16	1-42
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.												
X	WO 98 55903 A (BENSON SCOTT S ;DILLINGER RAY S (US); NEUROMEDIA INC (US); TACKETT) 10 December 1998 (1998-12-10) page 1, line 32 -page 5, line 22	1-42												
X	WO 96 05685 A (DIGITAL SYSTEMS INT) 22 February 1996 (1996-02-22) page 2, line 25 -page 5, line 5 page 6, line 7 -page 7, line 36 page 9, line 35 -page 26, line 25	1-42												
X	US 5 630 017 A (MATTHEWS III JOSEPH H ET AL) 13 May 1997 (1997-05-13) column 2, line 58 -column 5, line 16; figures 7A,8A column 6, line 15 -column 7, line 38 column 2, line 63 -column 5, line 16	1-42												
<p><input checked="" type="checkbox"/> Further documents are listed in the continuation of box C.</p>		<p><input checked="" type="checkbox"/> Patent family members are listed in annex.</p>												
<p>* Special categories of cited documents:</p> <p>*A* document defining the general state of the art which is not considered to be of particular relevance</p> <p>*E* earlier document but published on or after the international filing date</p> <p>*L* document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)</p> <p>*O* document referring to an oral disclosure, use, exhibition or other means</p> <p>*P* document published prior to the international filing date but later than the priority date claimed</p> <p>*T* later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention</p> <p>*X* document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone</p> <p>*Y* document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art</p> <p>*Z* document member of the same patent family</p>														
<p>Date of the actual completion of the international search 2 May 2001</p>		<p>Date of mailing of the international search report 12/06/2001</p>												
<p>Name and mailing address of the ISA European Patent Office, P.O. Box 5818 Patentlaan 2 NL - 2280 LV Rijswijk Tel: (+31-70) 340-2040, Tx: 01 651 epo nl, Fax: (+31-70) 340-3015</p>		<p>Authorized officer Glaser, N</p>												

Form PCT/ISA/210 (second sheet) (July 1992)

INTERNATIONAL SEARCH REPORT

 International Application No.
 PCT/US 00/34949

C. (Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category *	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	PATENT ABSTRACTS OF JAPAN vol. 016, no. 547 (P-1452), 17 November 1992 (1992-11-17) & JP 04 205368 A (SEGA ENTERP LTD), 27 July 1992 (1992-07-27) abstract	1, 19, 25, 37
P, X	US 6 088 731 A (HOLVATH ZOLTAN ET AL) 11 July 2000 (2000-07-11) column 4, line 49 -column 5, line 33 column 5, line 57 -column 7, line 57	1, 19, 22, 25, 37
P, X	EP 0 993 197 A (CSELT CENTRO STUDI LAB TELECOM) 12 April 2000 (2000-04-12) page 2, line 3 - line 40 page 3, line 55 -page 4, line 52	1-42
A	US 5 367 454 A (OMURA KENGO ET AL) 22 November 1994 (1994-11-22) abstract column 4, line 12 -column 5, line 10	1-42
A	WO 99 35600 A (SES CANADA RESEARCH INC) 15 July 1999 (1999-07-15) the whole document	1-42
A	FR 2 719 401 A (KHAN ALEXANDRE) 3 November 1995 (1995-11-03) page 6, line 1 - line 7	1-42

Form PCT/ISA/210 (continuation of second sheet) (July 1993)

page 2 of 2

 Unified Patents
 Exhibit 1002
 MSLV INV 003540
 Page 1764 of 2584

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No.
PCT/US 00/34949

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 9855903 A	10-12-1998	AU 7692898 A EP 1008019 A	21-12-1998 14-06-2000
WO 9605685 A	22-02-1996	CA 2197455 A	22-02-1996
US 5630017 A	13-05-1997	US 5689618 A US 5613056 A	18-11-1997 18-03-1997
JP 04205368 A	27-07-1992	NONE	
US 6088731 A	11-07-2000	AU 3862899 A EP 1073979 A WO 9956228 A	16-11-1999 07-02-2001 04-11-1999
EP 0993197 A	12-04-2000	JP 2000113216 A	21-04-2000
US 5367454 A	22-11-1994	JP 6012401 A	21-01-1994
WO 9935600 A	15-07-1999	AU 1956799 A EP 1046119 A	26-07-1999 25-10-2000
FR 2719401 A	03-11-1995	NONE	

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
20 September 2001 (20.09.2001)

PCT

(10) International Publication Number
WO 01/69436 A1

- (51) International Patent Classification: G06F 17/30
- (21) International Application Number: PCT/FI01/00274
- (22) International Filing Date: 19 March 2001 (19.03.2001)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
20000637 17 March 2000 (17.03.2000) FI
- (71) Applicant (for all designated States except US):
CODEONLINE OY (FI/FI); Ukonvaaja 2 A, FIN-02130
Espoo (FI).
- (72) Inventors; and
- (75) Inventors/Applicants (for US only): HÄMÄLÄI-
NEN, Matti (FI/FI); c/o Codeonline Oy, Ukonvaaja

2 A, FIN-02130 Espoo (FI). PRIHA, Ilkka (FI/FI);
c/o Codeonline Oy, Ukonvaaja 2 A, FIN-02130 Espoo
(FI). YLI-KREKOLA, Juho (FI/FI); c/o Codeonline
Oy, Ukonvaaja 2 A, FIN-02130 Espoo (FI). KUGGE,
Stephane (FR/FI); c/o Codeonline Oy, Ukonvaaja 2 A,
FIN-02130 Espoo (FI).

(74) Agent: BERGGREN OY AB; P.O. Box 16, FIN-00101
Helsinki (FI).

(81) Designated States (national): AE, AG, AL, AM, AT, AU,
AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU,
CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM,
HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK,
LB, LS, LT, LE, LV, MA, MD, MG, MK, MN, MW, MX,
MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SE, SL,
TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

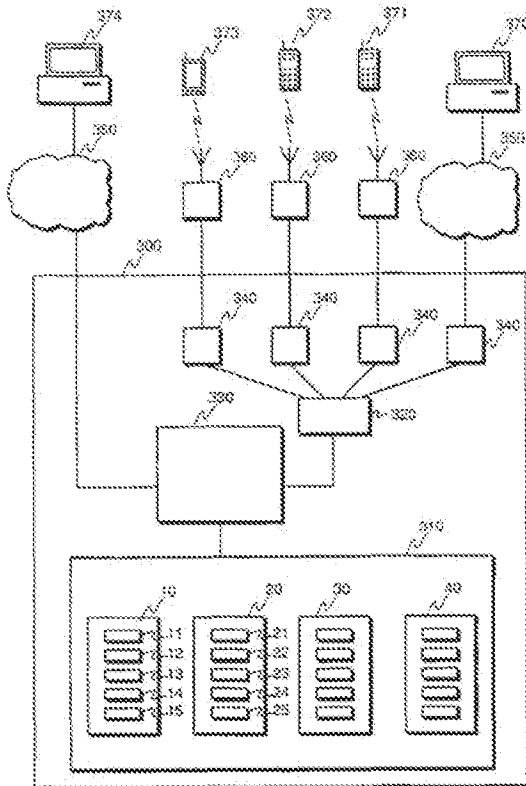
(84) Designated States (regional): ARIPO patent (GH, GM,
KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian
patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European

[Continued on next page]

(54) Title: A METHOD AND A SYSTEM FOR PROVIDING INTERACTIVE QUESTION-BASED APPLICATIONS



WO 01/69436 A1



(57) Abstract: The invention relates to methods and systems for providing interactive question-based applications over communication networks. According to the invention, information specifying questions and answers are stored in a first data unit, actions specifying how questions should be presented to users are stored in a second data unit, and rules specifying how presented material should be treated for display on different types of terminals are stored in a third data unit. This separation of content, functional logic, and presentation style simplifies the creation of new questionnaires and their delivery to different terminals over different communication networks.



patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BI, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

Published:

----- *with international search report*

A method and a system for providing interactive question-based applications

BACKGROUND OF THE INVENTION

5

1. Field of the Invention

The invention relates to methods, systems and computer program products for providing interactive question-based applications using data communication networks. Especially, the invention is related to such a method, system and computer product as specified in the preamble of the independent claims.

2. Description of Related Art

15 Different types of questionnaires, voting systems and interactive games, which can be used over data communication networks, are in use. The prior art solutions for creating questionnaires and other interactive question-based applications have, however, many drawbacks. In one typical approach, the questions and the logic how questions are presented to users, i.e. the behavior of a questionnaire, is hard coded
20 into the program performing the questionnaires, making the creation of new questionnaires difficult and time consuming. Some systems exist, in which changeable sets of questions can be presented through a fixed front-end, displaying the questions in a fixed, predefined format. In such systems the changing of questions is relatively straightforward, but changing of the behavior of the
25 questionnaires is burdensome. Typical questionnaire systems can only be used by one type of terminals, for which a particular questionnaire system is created.

It is possible to make, for example, a questionnaire form using HTML (HyperText Markup Language). Various logical elements are defined in HTML. A questionnaire
30 having certain information content and certain structure may be specified using HTML tags placed properly in a file specifying the information content. Typically such HTML questionnaires are static, so that a same questionnaire is presented to all users. Furthermore, there is need for a dedicated program in the server, which processes answers to a specific HTML questionnaire. If there is need to present the
35 same questionnaire to users of, for example, WAP (Wireless Application Protocol) enabled mobile devices, typically a second version of the questionnaire has to be provided using proper WML (Wireless Markup Language) tags.

Structured documents are known from creating and presenting information. A structured document is a document, where the information content of the document and the format, in which the document is presented to a user, are separated from the logical structure of the document. The use of structured documents has many advantages: By explicitly marking the logical structure (elements) of a document, we enable computer programs to identify and automatically process the elements, such as search for certain elements and convert to different presentation formats. Documents can be transformed, merged, and edited automatically. Also, it is easy to convert from one structured document to another.

In interactive applications, the use of a more advanced structured questionnaire form than a HTML/WML form may ease the problems related to presenting a static questionnaire to users and to specifying new content for an existing static questionnaire. It does not, however, resolve the problems relating to processing the answers of the questionnaires or, in more versatile interactive applications than static questionnaires, to controlling online behaviour of the application.

SUMMARY OF THE INVENTION

An object of the invention is to provide a flexible method, system and computer program product for providing interactive question-based applications, an example of such an interactive question-based application being displaying questionnaires, obtaining answers and giving feedback to users. Advantageously easy creation of new interactive question-based applications is enabled.

The objects are reached by separation of information content, application behavior, and presentation style from each other. This allows easy creation of questions, of behavior and feedback logic, and easy integration of different types of terminals to the questionnaire system.

A method according to the invention is characterized by that, which is specified in the characterizing part of the independent method claim. A system according to the invention is characterized by that, which is specified in the characterizing part of the independent system claim. A computer program product according to the invention is characterized by that, which is specified in the characterizing part of the independent claim directed to a computer program product. The dependent claims describe some advantageous embodiments of the invention.

According to the invention, information specifying questions and answers are stored in a first data unit, actions specifying how questions and other information should be presented to users are stored in a second data unit, and rules specifying how presented material should be treated for display on different types of terminals are stored in a third data units. These first, second and third data units are logically separate data units, which are typically implemented for example, as separate files or as separate entries in a database. Alternatively it is possible that a first, second and/or third data unit are implemented by using a single file or a single database entry, where all data units reside. Also in this second option, as the data units are logically separated from each other and they can be easily identified and, for example, modified independently of each other, they are separate data units.

The separation of content, functional logic of an interactive question-based application, and presentation style simplifies the creation of new interactive question-based applications, such as new questionnaires. The separation of content allows, for example, easy and straightforward modification of the content. This means that it is possible easily to provide localized content for interactive question-based applications, for example, by modifying the facts present in the information or by changing the language, in which the content is provided. The separation of presentation style enables straightforward presentation of an interactive question-based application using various end-user devices.

The separation of the functional logic of an interactive question-based application from the content and presentation, allows, for example, that certain information content may be used in various interactive question-based applications. One advantage of the separation is thus that the re-use of information content is enhanced: it is possible, for example, to use certain first data unit comprising certain information content in connection with various interactive question-based applications, i.e. with various second data units. A second advantage is that it is easy to modify a functional logic of a certain existing interactive question-based application to produce a new or modified interactive question-based application. As the content and presentation are separated from the functional logic, the risk of causing a problem to presenting an application or modifying inadvertently the content of an application is minimal.

In some advantageous embodiments of the invention, the functional logic of interactive question-based applications is described using certain basic actions. This allows the functionality of various applications to be described unambiguously

using the same basic actions. By choosing these basic actions properly, it is possible to provide basic actions, which allow the construction of various interactive question-based applications, such as questionnaires, various examinations, or question-and-answer games. The functional logic of an interactive question-based application relates, for example, to rules of a certain game.

Furthermore, to construct a new game or other interactive question-based application using the specified basic actions is more straightforward than, for example, programming a new game from scratch. It is possible that the actions relating to a certain interactive question-based application are online compiled to some generic actions carried out by a computer. Alternatively, it is possible that the description of an interactive application using certain basic actions is used as an unambiguous specification for programming an application, which behaves as specified by the description of the interactive application. Even in this case the use of unambiguous description of functional logic provides a major advantage over a non-formal description of functional logic: a non-formal description, for example a description in a natural language such as in English, typically leaves room for misunderstandings or may be otherwise ambiguous. Importantly, such a non-formal description cannot be used as a basis for automatic processing, such as transformation of the description to a functioning application. One of the most advantageous features of using an unambiguous description of functional logic is thus that this makes it possible for a machine to carry out operations that in unstructured case would require human processing.

A further advantage of the invention is that providing basic actions for constructing logic of interactive question-based applications also makes it possible to construct tools for creating and editing interactive question-based applications by end-users who need not know the details of the implementation languages.

BRIEF DESCRIPTION OF THE DRAWINGS

Various embodiments of the invention will be described in detail below, by way of example only, with reference to the accompanying drawings, where

Figure 1 illustrates schematically the separation of content and functional logic of an interactive question-based application in accordance with the invention,

Figure 2 illustrates, as an example, one method according to the invention,

Figure 3 illustrates, as an example, schematically a system for providing interactive question-based applications,

5

Figure 4 illustrates the role and relationship of various data units in accordance with the invention,

Figure 5 illustrates, as an example, an action data unit relating to a simple questionnaire,

10

Figure 6a illustrates, as an example, a first part of a content data unit relating to said simple questionnaire,

Figure 6b illustrates, as an example, a second part of the content data unit illustrated in Figure 6a,

15

Figure 7 illustrates, as an example, the actions, which a system in accordance with the invention carries out when it processes the action data unit illustrated in Figure 5 and the content data unit illustrated in Figures 6a and 6b, and

20

Figure 8 illustrates a system according to an advantageous embodiment of the invention.

25 DETAILED DESCRIPTION OF THE INVENTION

1. A first group of advantageous embodiments

According to a first aspect of the invention, a method for providing an interactive question-based application over a communications network is provided. The communication network may be, for example, a packet data network or a mobile network. The interactive question-based application may be, for example, a questionnaire, where questions are presented to a user and answers are retrieved. According to an advantageous embodiment of the invention,

30

- 35 - at least one question is defined in a data unit 10 of a first type,
- at least one action for processing questions in a data unit of the first type for producing presentable questions are defined in a data unit 20 of a second type, and

- rules for processing presentable questions for display on a specific type of terminal; are defined in a data unit of a third type.

Figure 1 illustrates schematically the separation of content and functional logic. In Figure 1 two data units 10a, 10b of the first type, that is data units comprising content relating to interactive question-based applications, are illustrated. These first data units 10a, 10b comprise content data elements 11-15. Typically a content data element comprises a data fragment defining at least a question to be presented to a user. A content data element may have a data fragment defining a category of the question and typically it has a data fragment defining an identifier for the content data element. Furthermore, a content data element may comprise a plurality of data fragments defining some answer alternatives to be presented to a user. A content data element may comprise a data fragment defining a correct answer. Additionally, a data element 11-15 may comprise data fragments defining scoring information and/or feedback information relating to, for example, each answer alternative, which is presented to a user, or a data fragment defining scoring information and/or feedback information relating to a correct answer. Typically a data element 11-15 specifies – either explicitly using certain specific data fragments or implicitly using only the data elements described above – a question type, examples of which are a multi-choice question, a single-choice question or question, to which a free-text answer is expected. Examples of these content data fragments are illustrated below in connection with a content data element example. Some content data elements may comprise only data fragments defining feedback. Such content data elements are often presented to a user at the end of a interactive question-based application.

Two data units 20a, 20b of the second type are also illustrated in Figure 1. These second data units 20a, 20b correspond to functional logic of different interactive question-based applications, for example, to two different games having different rules. The data units 20a, 20b comprise action data elements 21-25, which are typically processed in a sequential order. An action data element may define, for example, one or more of the following actions: selection of one content data element (typically a question or feedback) from a content data unit; selection of a plurality of content data elements (for example, selection of a certain number of questions from a certain category or categories) from a content data unit; presentation of selected content data element(s) to a user; retrieval of user input to a presented content data element or elements; rules for calculating of scores (scoring model); or calculation of scores. Examples of these action data fragments are illustrated below in connection with an action data element example.

The two circles in Figure 1 illustrate, as examples, two interactive question-based applications, which are provided using data units 10a, 10b, 20a and 20b. The content data elements defined in data unit 10a may be combined with functional logic defined in data unit 20a, resulting in a first interactive question-based application. In this first application, the questions may be, for example, presented to a user one by one and the difficulty level of the questions may be increasing. The difficulty level may be defined in content data elements using specific content data fragments. The same content data elements defined in data unit 10a may be combined with functional logic defined in data unit 20b to produce a second interactive question-based application. This easy use of content defined in data unit 10a is an example of re-use of content information, which is one of the advantages of the invention.

It is evident that content defined in data unit 10b may typically be combined similarly with functional logics defined in data elements 20a and 20b. The content defined in data unit 10a may be, for example, specific to certain first geographical region and the content defined in data element 10b may be specific to a second geographical region. This is an example of providing localized content in a straightforward and easy manner. Alternatively, the factual content in data units 10a and 10b may be the same, but the language, in which the questions, answer alternatives, correct answers and/or feedback is specified, may be different. This is an example of providing an interactive question-based application in various languages in an easy manner.

According to a further advantageous embodiment of the invention, said at least one question in data units of the first type is defined in a first computer language, said actions for processing are defined in data units of the second type in a second computer language, and said rules for processing a set of presentable questions are defined in data units of the third type in a third computer language.

According to a further advantageous embodiment of the invention, answers obtained as a result of displaying questions are stored in a fourth computer language.

According to a further advantageous embodiment of the invention, said first, second, third, and fourth computer languages are defined in the Extensible Markup Language (XML). XML is a meta language, using which it is possible to define application specific languages. XML is a well-defined standard language, and there exist XML parsers and XML formatters. Therefore it may be advantageous to use as:

XML-based language for defining especially functional logic of an application, as a specification written in XML-based language to be converted into generic actions to be carried out by computers. This means that it is possible to interpret a file containing XML-based language and modify the behaviour of the application accordingly even at execution time, i.e. when the application is executed. The data fragments discussed above are typically some basic structures defined in XML for use in an action data unit and/or in a content data unit. The data elements discussed above are typically constituted of a combination of data fragments and information specifying values for at least some of the basic structures defined using the data fragments.

There exist various standard languages, which are often called style languages, for defining rules for processing information to be presented for display. Therefore the rules for processing information to be presented, including presentable questions, may be defined in a data unit of a third type using a standard style language. An example of such standard style language is XSLT (XSL Transformations). XSLT is a language for transforming XML documents into other XML documents. XSLT is designed for use as part of XSL, which is a stylesheet language for XML. In addition to XSLT, XSL includes an XML vocabulary for specifying formatting. XSL specifies the styling of an XML document by using XSLT to describe how the document is transformed into another XML document that uses the formatting vocabulary. XSLT is designed for use as part of XSL, which is a stylesheet language for XML. Alternatively, it is possible to use presentation rules specifically designed for interactive question-based applications. Such presentation rules may be defined, for example, using a specific XML-based presentation language.

According to a further advantageous embodiment of the invention, the method comprises at least the steps of

- processing at least a part of a data unit of the first type according to at least a part of the definitions in a data unit of the second type for producing presentable questions,
- processing said presentable questions according to at least a part of the definitions in a data unit of the third type for display on a specific type of terminal,
- transmitting processed presentable questions to a terminal of said specific type.

Figure 2 illustrates an example of a method 200 according to the invention. In this method, a content data unit comprising a plurality of content data elements is defined in step 201. A question to be presented to a user is defined in at least one of

these content data elements. A plurality of content data units may be defined. An action data unit comprising a plurality of action data elements defining at least some actions for processing content data elements is defined in step 202. A plurality of action data units may be defined. A presentation data unit comprising rules for processing presentable information for display on a specific type of terminal is defined in step 203. At least some of the rules relate to processing presentable questions. Steps 201-203 are carried out at least before an interactive question-based application is provided to users. If new content is to be provided or existing content is to be modified, step 201 may be carried out when the interactive question-based application is already in use. Similarly, if there is need, for example, to modify functional logic of an application, it may be possible to carry out step 202 when the application is already in use. If the presentation rules need to be modified, step 203 may be carried out when an application is already in use. It typically depends on the implementation details, do these modifications affect an interactive question-based application immediately or, for example, only after the new data units are compiled. Advantageously changes made to a content or action data unit affect those interactive question-based application sessions, which are started after the modifications or creation of the content/action data unit, without any compilation or re-start of the actual application software in a system providing the applications to users.

Steps 204-205 relate to interacting with a user. An action data unit specifies the order in which the action data elements are processed, as the action data unit describes the functional behaviour of an interactive question-based application. The action data elements may be processed in a sequential order, for example in that order in which they are defined in an action data unit. Alternatively, the user input may affect, which action data elements and in which order the order are to be processed. Also in this case the action data unit describes, how the user input or feedback to user input affects the behaviour of an application; this is done with control structures defining conditional actions. In step 204 an action data element selected from the action data unit. Thereafter action specified in said selected action data element is carried out in step 205. Typically the content data unit and the rules for processing information are involved in step 205. Some examples of actions that may be carried out in step 205 are the following: selecting at least one content data element; processing selected content data element(s) using presentation rules; transmitting presentable content data element(s) to a terminal; or receiving an answer as a response to presentable content data element(s). Steps 204 and 205 are repeated until end of the interactive question-based application is encountered. A

user may terminate an application, or the application may be finished as specified in the action data unit.

- 5 According to a second aspect of the invention, a system 300 for providing interactive question-based applications, for example for presenting questions and retrieving answers, over a communications network is provided. Figure 3 illustrates an example of such a system. According to an advantageous embodiment of the invention, said system comprises at least means for storing
- 10 - a data unit 10 of a first type having the definition of at least one question,
 - a data unit 20 of a second type having the definition of at least one action for processing of at least one question in a first data unit and for producing at least one presentable question, and
 - a data unit 30 of a third type having the definition of at least one rule for processing of a presentable question for display on a specific type of terminal;
 - 15 and
 - computer code means 320 for processing at least a part of a data unit of the first type according to at least a part of the definitions in a data unit of the second type for producing presentable questions,
 - computer code means 340 for processing presentable questions according to at least a part of the definitions in a data unit of the third type for display on a specific type of terminal, and
 - 20 - computer code means for transmitting processed presentable questions to a terminal of said specific type.
- 25 A system according to the invention may implement any method according to the invention. It may implement, for example, any of those preferred embodiments described in the appended dependent method claims. In such system, the computer code means 320 and 340 are typically arranged to carry out the corresponding method steps. The data units 10 is often a data unit comprising a plurality of content data element, and the data unit 20 is often a data unit comprising a plurality of
- 30 action data elements.

Figure 3 further shows various terminals 370, 371, 372, 373, 374 of users. A user of a PDA (personal digital assistant) device 373 receives questions formatted for a PDA by computer code means 340 arranged to format material for display on a PDA. The user of a WAP (wireless application protocol) enabled mobile terminal 372 receives questions formatted for a WAP terminal. A SMS (short message service) enabled mobile phone 371 receives questions as short text messages (SMS

35

messages). Further, the system is able to present, for example, questionnaires to a user connecting to the system via the internet 350 using his computer 370.

5 According to a further advantageous embodiment of the invention, the system further comprises computer program code means 330 for receiving an answer to a question displayed to a user, and computer program code means for storing received answers in a memory means 310, 40. Preferably, the answers are stored in a predefined format for easy post-processing. Typically answers are stored in data units 40 of a fourth type.

10 According to a further advantageous embodiment of the invention, the system further comprises computer program code means for receiving information comprising definition of a question, and computer program code means for storing received information comprising definition of a question in a data unit of the first type. Such means can be used for creating new interactive question-based applications, for example questionnaires. Preferably, such means are connected to a data communications network 350 in order to allow remote users 374 to create new questions or other content and/or to modify content. Typically such computer code means for receiving information are called interfaces.

20 According to a further advantageous embodiment of the invention, the system further comprises computer program code means for receiving information specifying action definitions, and computer program code means for storing specified action definitions in a data unit of the second type. Such means can be used for creating new behaviors for new or already existing sets of questions. Preferably, such means are connected to a data communications network in order to allow remote users to create new behavior logics.

30 According to a third aspect of the invention, a computer program product for a system for providing interactive question-based applications, for example for presenting questions and retrieving answers, over a communications network is provided. According to an advantageous embodiment of the invention, the computer program product comprises at least computer program code means for storing

- a data unit of a first type having the definition of at least one question,
- 35 - a data unit of a second type having the definition of at least one action for processing of at least one question in a data unit of a first type and for producing at least one presentable question, and

- a data unit of a third type having the definition of at least one rule for processing of a presentable question for display on a specific type of terminal;
- and
- computer code means for processing at least a part of a data unit of the first type
- 5 according to at least a part of the definitions in a data unit of the second type for producing presentable questions,
- computer code means for processing presentable questions according to at least a part of the definitions in a data unit of the third type for display on a specific type of terminal, and
- 10 - computer code means for transmitting processed presentable questions to a terminal of said specific type.

Computer program product according to the invention may implement any method according to the invention. It may implement, for example, any of those preferred

15 embodiments described in the appended dependent method claims.

According to a further advantageous embodiment of the invention, the computer program product is stored on a computer readable medium. The computer readable medium can be for example a hard disk or other electromagnetic mass storage

20 medium, a CD-ROM or another optical mass storage medium, or for example a semiconductor memory element.

2. A second group of advantageous embodiments

2.1. Describing functional logic unambiguously

25

The functional logic of interactive question-based applications may be advantageously described using certain basic actions, which are chosen properly. This allows the functionality of various interactive applications, for example the

30 rules and behaviour of various games, to be described unambiguously using the same basic actions. Examples of game rules, which may be described, are gradually decreasing/increasing difficulty levels; selecting questions randomly or from certain specific topics; and various scoring models, even when the individual score values are part of the data unit of first type.

35

An example of set of basic actions in an interactive question-based applications is the following: presenting plurality of questions (one or more questions), receiving an answer to said question(s), evaluating the answer, giving feedback, selecting next

action typically based on the evaluation of the answer, and allowing actions to be done by users at each different phase of an application. (for example, answering a question, asking for help, throwing a dice, saving a game or accepting a new player to a multiplayer game). Using these basic actions it is possible to construct, for example, various game logics. These basic actions typically correspond to action data elements. An action data element typically comprises a plurality of action data fragments, and same action data fragments may be used in various action data elements.

10 In addition to data elements defining actions, an action data unit typically comprises control structures, which define conditional actions for branching and looping. Examples of such control structures are the following control structures, which are familiar from general programming languages: while-break -structure, if-then-else -structure and switch-case -structure.

15 To describe the functionality of an interactive question-based application using some basic actions or basic action data elements is one way of unambiguously describing an interactive question-based application. An unambiguous description of functional logic of an interactive question-based application enables, for example, that various application concept providers may describe their applications in a uniform way. More specifically, it enables construction of an editing tool for application concept providers. A method and system, where functional description of an interactive question-based application is described unambiguously, thus allows the creation of various interactive question-based applications by various concept providers or, in principle, by anybody capable of using the unambiguous description, either via an editing tool or otherwise.

20 One way to unambiguously describe, for example, the functionality an interactive question-based application is to describe it using a computer-processable specification language. A computer-processable specification language has an unambiguously defined syntax and semantics, therefore it may be automatically processed by a computer program, and it uses the concepts of the application domain, therefore it can be easily understood and often also produced directly by a human user. The syntax determines the valid structures and sentences of a language and semantics specifies their meaning or interpretation. XML, which is briefly discussed above, is one meta language using which it is possible to define computer-processable specification languages.

In some cases it is more advantageous to use an unambiguous description of a functional logic of an interactive question-based application only as an unambiguous specification and to implement a program corresponding to said specification in a general programming language. Some examples of suitable programming languages are Java, C++, and C . Preferably, the basic actions of functional logic of an interactive question-based application are provided as software components that can be combined to implement the specified application behaviour. The use of general programming languages typically provides more efficient use of computing resources than using interpreted computer-processable specification languages. In some cases, it is possible to use a general programming language as a computer-processable specification language for directly describing functional logic of an application. This is especially the case, when Java is used.

2.2. Examples of XML-based languages for defining data content and functional logic

The information content of the interactive question-based applications is advantageously provided in a certain format or, in other words, described in an unambiguous way. The basic features for information content may be, for example, elements specifying a question, an answer, and a feedback. There may be various types of answers: single-choice, multi-choice or free text answers. Similarly as for providing an editing tool for interactive question-based application logic, when an unambiguous way to define the logic is provided, it is possible to provide an editing tool for creating and modifying information content of interactive question-based applications.

An example of an XML-based computer-processable specification language for describing information content of an interactive question-based application is a Data Definition Language (DDL), which is specifically designed for this purpose. This subsection of the description describes DDL (Data Definition Language) syntax and structure. The terms and concepts referring to objects in a are introduced below.

In DDL, a data unit refers to all the content contained in a single DDL question set, including the header and the body. A data unit consists of one or more data elements. In DDL syntax, a data unit is everything enclosed in a ddl tag. At its fundamental level, a data element consists of pieces of information (category, data type, unique identification, correct answers and initial/default answers), and

presentation or logical content (multimedia, labels, images, constants, variables) for a specific answer represented by a primitive data type (integer, double...). In DDL, a data element specifies at least a question type. This data element specifying a question type can nest other data elements to form more advanced, user-defined question types and create more complex data type structures of answers. In DDL syntax the following are data elements: form, int, double, long, string, boolean, date, choice and block. A data fragment is the smallest piece of DDL information or content. In DDL syntax, any attributes, empty elements or text nodes are data fragments.

The ADL language (see more detailed description below), or an ADL action data unit, references DDL data fragments in two different ways: (1) Actions can directly reference a DDL data fragment by using its unique identification string if such a unique identification string exists. (2) It is also possible to represent the DDL data unit as an XML DOM tree, and access the data fragments by visiting the tree nodes sequentially or searching for a specific XML attribute or element.

The header of a DDL document (data unit) consists of information that describes the questionnaire document. This is often known as meta-information (information about the document). Below is shown a sample of a DDL header, it is actually a complete DDL document without a body. The header information is very legible, and therefore the elements and attributes of the DDL header not discussed in more detail here.

```

25 <?xml version="1.0" encoding="ISO-8859-1"?>
   <ddl xml:lang="en" cache="yes">
     <title>A sample DDL questionnaire</title>
     <author>
       <name>CODEOnline Oy</name>
30   <url>www.codeonline.com</url>
       <email>info@codeonline.com</email>
     </author>
     <timeinfo format="yyyy-MM-dd hh:mm:ss" zone="-2"/>
     <created time="2001-26-02 19:45:00"/>
35   <modified time="2001-26-02 19:45:00"/>
     <form/>
   </ddl>

```

The body of a DDL document consists of all the data elements (questions) making up a form or a questionnaire. The questions with their correct answers and feedback are specified in the body of a DDL document. A form does not need to have correct answers and feedback, the processor could just save the answers to a database as in a survey case. Below is shown a sample of a DDL form that consists of a single data element; the form contains a multiple-choice question type with a single answer.

```

5 <form id="-1" type="-1" cat="-1">
  <choice id="123" cat="geography" mandatory="true">
10  <init>0</init>
    <correct>1</correct>
    <label>What's the capital of <emph cat="bold">Finland</emph>?</label>
    <file mime="image/gif">finland.gif</file>
    <boolean>
15  <label>Paris</label>
    </boolean>
    <boolean>
    <label id="h">Helsinki</label>
    </boolean>
20  <feedback>
    <text>You answered <use this="answer"/>.<br/></text>
    <isCorrect>
    <text>You know European geography!</text>
    <score>10</score>
25  </isCorrect>
    <else>
    <text>Wrong ! the correct answer was <use id="h"/>.</text>
    </else>
    </feedback>
30  </choice>
  </form>

```

The following data fragments have the same meaning wherever in a DDL document they are used: **id** is optional and defines a unique identification value. **type** is optional and specifies a type: in form it is the type of document (exam, quiz, survey, etc), in block it is the user data type of a complex block (fill-in, true-false, graphical) and in data it is the data type (integer, long, double, etc). **cat** is optional

and defines a category, giving the purpose (business, political, game, community name, etc) of the entity it is in.

All the DDL data elements or inputs that receive a user answer are described next.
5 **form** defines the body of the document. It can be considered as a complex block. **int**, **double**, **long**, **string**, **boolean**, **date** and **choice** are inputs for primitive data types, which are also called primitive blocks. They all have an XML attribute **mandatory**, which tells if it is mandatory to provide an answer for this block. **Choice** is a special block that represents a list of choices from which a single
10 answer can be selected; the answer is the integer index of the selected choice. **block** is used to nest blocks in order to allow a user to create more complex question types.

The following data fragments define the displayable (shown on a client terminal)
15 and logical (processor executes operations on it: transformation, storage...) content of the form. XML element **use** maybe used almost anywhere in a DDL document; it is used for example in cases where there is need to insert values that are not known at request time. Those values can be client/terminal/platform-dependent and are resolved for instance after a database query or after giving a random value. With
20 **use**, in addition to **id**, two other data fragments are used: **i** is used in cases when **id** or **this** are referencing a multi-valued data fragment, and it gives the index of the value to consider; and **this** is used to reference a data fragment of the same parent data element by calling its reserved name(**correct**, **init**, **answer**, **file**, **text**, **data**). In general, **i** is used to enter several values in multi-valued entities.

25 **data** is a data value of any type, and it defines constants used by the document processor to evaluate, for instance, feedback. Legal XML attributes for **data** are **id**, **type** and **cat**. **data** can be multi-valued and should use XML element **i** to specify all the values. When XML element **data** has a single value it is defined as a child
30 text node. The **use** XML element is allowed here.

file specifies the path of a resource file. It is part of the displayable content for a data element. This data fragment can be multi-valued. XML element **use** is allowed for **file**. **file** takes an XML attribute **mime** in addition to **id** and **cat**. **mime**, in turn,
35 defines the MIME type of the content of the file.

text is a data value of type string, which is also part of the displayable content for a data element. This data fragment can be multi-valued. XML element **use** is allowed

here. XML element **emph** is used as a child of XML elements **text** and **label** (explained below). Its purpose is to embed style statements inside displayable sentences. **br** is an empty XML element, which defines a platform/terminal independent break/carriage return.

5

score is a short name of a **data** value of type **double** and category "score". XML element **use** and XML attribute **id** are optional for **score**. **label** is a data value of type **string** and category "label". It is used to legend any input/block. XML elements **use**, **emph** and **br** as well as XML attribute **id** are legal and optional for **label**.

10

answer is a data value of same type as the parent block and of category "answer". It is used to redefine a user answer. XML attribute **id** is optional for **answer**. **init** is a data value of same type as the parent data element and of category "init". It is used to give an initial/default value to an input component. **correct** is a data value of same type as the parent data element and of category "correct". It is used to specify the correct answers for the parent data element. This data fragment can be multi-valued. XML element **use** and XML attribute **id** are legal and optional for **init** and **correct**.

15

20 The following elements and attributes define the feedback for any data element. XML element **feedback** starts the feedback section. It consists of content data fragments (**data**, **text**, **file**) systematically selected whatever the answer to an input, and a combination of the XML elements described below. XML element **isCorrect** wraps the content (**data**, **text**, **file**) that is selected, if the answer matches one of the values specified in the **correct** data fragment of the same parent data element. XML
25 element **case** is a more general form of **isCorrect**. It wraps the content (**data**, **text**, **file**) that is selected if the statement is true (can use ADL statements or script languages). XML element **else** is the same as XML element **case**; it is evaluated only if the previous **case** or **else** statement was false. When an **else** statement is true,
30 the processor jumps to the next **case** element if any.

An example of an XML-based computer-processable specification language for describing functional logic of an interactive question-based application is an Application Description Language (ADL), which is specifically designed for this
35 purpose. The basic data fragments and data elements in ADL are described below. ADL is designed to be used together with DDL (Data Definition Language) and with appropriate style language, such as XSLT.

The terms and concepts referring to ADL document objects below in this Section 2.1 are introduced here. In ADL case, a data unit represents all the content contained in a single ADL object. A data unit consists of one or more data elements. In XML based ADL syntax, a data unit is everything enclosed in **adl** tag denoting
5 the root element of the document. In ADL case, the data element can be either a variable or an action. In XML-based ADL syntax this means an element identified by a start tag and end tag. Data Fragment is the smallest piece of ADL. In XML-based ADL syntax this means attributes or primitive data elements.

10 ADL is a computer-processable specification language that is independent of the platform and implementation language. The actions relevant to question-based interactive applications comprise references to content data (DDL language element or fragment) and basic actions. Basic actions include control structures, logical operations, XML document management and evaluation and scheduling actions.
15 The basic set of actions can be extended as required by a particular application type.

Following are examples of the control structures that can be used to specify such actions as branching and skipping in the application logic, such as questionnaire processing: **If** structure, **Switch-Case-Else** structure and **While-Break** structure.

20 Following are examples of the logical operations that provide a means for logical comparisons: **Equal** checks whether all the 'indexed' parameters, e.g. children values, are equal and it is usually used inside **If** and **Break** structures. **And**, **Or** and **Not** are the basic logical operators. The variable values can be set and read using
25 **Set** and **Get** operations; **Set** sets the variable value in the parent action's environment.

Following are examples of the XML document management that can be used to manage any XML document, e.g. content data in DDL: **XPath** Searches or
30 addresses a part of an XML document and **Xslt** makes transformation to an XML document.

Following are examples of managing or referencing to content data (DDL). **Ddl.Select** selects content data elements to be presented from a content data unit.
35 **Ddl.QuestionAmount** returns the amount of data elements in a content data unit that can be presented as a question. **Ddl.Send** returns all selected content data elements so that style can be applied and the elements can be sent to the user.

Ddl.Feedback returns all content data elements that present feedback to certain (or latest) user answer.

- The following are examples of ADL evaluation, extension and scheduling. Execute
 5 executes actions returned by the children actions, and it is useful with e.g. Compile
 action. Compile compiles a string into ADL action; the string to be compiled is the
 concatenation of the results of all children actions. Install installs a new action to
 the language from a non-ADL code, e.g. from Java class. Parameter name is the
 name of the new action type, and parameter type tells the implementation language.
 10 Wait pauses the execution of ADL code.

ADL syntax can be expressed in XML even though other languages, such as Java,
 are also possible. The syntax that is chosen typically depends at least on the
 following points: A human that writes ADL may choose a syntax he is most familiar
 15 with. A computer tool (concept editing tool) that helps to generate the application
 logic writes syntax that is most convenient for the program to write and modify. The
 implementation of the feedback architecture (ADL processor, Game platform ...) may
 be such that certain syntax is recommended (e.g. from the server performance
 point of view).

20 The ADL language references DDL data fragments in two different ways. Actions
 can directly reference a DDL data fragment by using its unique identification string
 if it exists. It is also possible to represent the DDL data unit as an XML DOM tree,
 and access the data fragments by visiting the tree nodes sequentially or searching for
 25 a specific XML attribute or element.

A variable may be defined in ADL in the following way: `<var1 value="zip"/>` or
`<var1>zip</var1>`. An action may be defined in the following way:
`<action_name some_attr="" another_attr="">`
 30 `<any code>`
`</action_name>`

Reference to a DDL element may be made in the following way. A content data
 management action may reference a DDL data element directly by `<Ddl.Select`
`id="question1"/>`. An XML document management action may reference DDL from
 35 the DDL's XML DOM tree representation:

```
<XPath document ="myDdl.ddl">choice[@id="123"]</XPath>
```

While there may be different syntaxes used in particular implementation, these processing rules describe how the XML syntax of ADL code is typically executed. (1) The processing of XML code starts from the document element ('Adl'). (2) Each XML element is an action that can be processed. The XML element name defines the action to be used. The action defines what happens when it is processed. (3) The attributes and child nodes of the XML element define parameters to the action. The XML attributes are parameters with name (like method parameters in Java), the attribute values are set as local variables to the action before processing it, and these parameters can be in any order. The XML child nodes are 'indexed' or 'unnamed' parameters (like parameters for Lisp statements), these parameters may be referred to by the action with an index and there can be any amount of these parameters. (4) The action itself defines what is done with the parameters and what effect their values have. (5) Common way of processing an action is the following: First all child nodes are processed one by one; each child node has its own local processing environment. The return values of each child node may or may not be stored to be used later. Finally the action itself acts. The environment's variable values and child node return values are usually used here. (5) The user of the ADL processor may initialize some variable values before calling the processor. When the ADL processor has finished processing the return value of the document element or any values of the document element's environment may be used. (6) An action may pause the execution of ADL code. The user of the ADL processor may then decide whether or when to continue.

Variables are processed in ADL typically in the following way. Variable values may be set in attributes, in other words by initialising a local variable, or by giving the variable name as an element. Variables may contain any datatype. Each action has its own environment. The environments are linked so that the environments parent environment is the parent action's environment. If the value for a variable ??? is not found from the environment, then search goes automatically to the parent environment. When setting a value for a variable, the search also goes upwards until the variable is found.

DDL and ADL are described here as examples; the invention is not restricted to these specific computer-processable specification languages. Furthermore, it is evident to a person skilled in the art that XML, for example, may be used to define other computer-processable specification languages. The syntax presented above for DDL or ADL is also only as an example.

2.2. Role and relationship of the content, action and presentation data units

Figure 4 illustrates the role and relationship of the various data units according to the invention. Figure 4 illustrates a content data unit 10, which may be, for example, a DDL data unit; an action data unit 20, which may be, for example, an ADL data unit; and a presentation data unit 30, which may be, for example, an XSLT style sheet. In addition to these content, action and presentation data units it is often advantageous to use a further data unit, a configuration data unit 50. A configuration data unit typically corresponds to a specific interactive question-based application. The data elements in this configuration unit 50 may define, for example: languages, in which the data content is available; content data units, which may be used in connection with this interactive question-based application; action data unit relating to this application; terminals or end-user devices, for which there is support for this interactive application; presentation data units relating to said terminals; timing properties (when the interactive question-based application is available; when results/analysis is available); and/or answer storing methods (not stored, first trial is stored, last trial is stored, all trials are stored). Instead of having a separate configuration data unit, it is possible to include the configuration information to an action data unit.

A system selects an action data unit and a content data unit, as specified typically in a configuration data unit. These three data unit specify an interactive question-based application in accordance with the invention. Thereafter the system selects and processes content data element in the content data unit according to the actions defined in the action data unit. The resulting data blocks are converted to a suitable format using a presentation data unit. Typically there is at least one presentation data unit for each terminal type, which a system in accordance with the invention supports.

2.3. Simple questionnaire as an example

Figure 5 illustrates, as an example, an ADL data unit 20, describing a simple questionnaire. This ADL data unit 20, which is presented in Figure 5, has the following data elements defining actions: Action data element 21a defines that first two unasked questions of a content data unit are selected. Action data element 22a defines that the selected questions are presented to a user. Action data element 23 defines that the application waits for user input. Action data element 21b defines that all questions relating to category "personal" are selected from a content data

unit. Action data element 22b defines that feedback relating the questions, which were latest presented to the user (the two questions defined to be selected in action data element 22a), is given and current selected questions are presented to a user. Action data element 22c defines that feedback relating to the questions, which were latest presented to a user (personal questions), is given. The action data elements 22a, 22b and 22c illustrated in Figure 5 indicate that XSLT is used to process selected content data elements into presentable information. Action data element 24 defines that a test email is sent to a certain address. The actions defined by action data elements illustrated in Figure 5 are carried out in the sequential order, in which they occur in the action data unit. In the ADL there are control structures defining conditional actions for branching and looping.

Figures 6a and 6b illustrate, as an example, a DDL data unit 10, which is one possible content data unit relating to the ADL data unit presented in Figure 5. In Figure 6a, content data element 600 defines the beginning of a DDL form. Content data element 11 defines a question, whose answer is a free-text string, and whose category is "personal". The label tags are used to define the presented question: "What's your name:". Content data element 12 defines a single-choice question, where the answer options presented to a user are Male and Female. The corresponding actual answers, which the interactive application later may later use, are Mister and Miss. Figure 6b illustrates the rest of this DDL data unit. Content data element 13 defines question "How many sides can you find in a hexagon", whose answer is expected to have an integer value. The correct answer is defined using correct tags. The feedback for a correct answer is defined to be 10 points and for an incorrect answer 0 points. Content data element 14 defines question "What's the capital of Finland:", whose answer is a case-insensitive string. The feedback is similar as in content data element 13. Content data element 15 defines feedback for the whole form, and content data element 601 defines the end of the form.

Figure 7 illustrates the actions a system in accordance with the invention carries out, when it processes the action data unit 20 illustrated in Figure 5 and content data unit 10 illustrated in Figures 6a and 6b. In step 701 the system selects an XSLT file, which corresponds to the terminal to which the information is to be sent. When a session is started, the type of the terminal is known typically from the communication protocol (for example, HTTP, one of Short Message Service Center Access protocols, or X.25) using which a request to start a session is received. In step 702, the system selects, as action data element 21a specifies, two DDL data elements from a DDL data unit; said two DDL data elements corresponding to two

unasked questions. In step 703, the system applies (as specified in ADL data element 22a) the selected XSLT file to the selected DDL data elements 13 and 14, which are illustrated in Figure 6b. The result is presentable information suitable for display on the specific terminal, and in step 704 the presentable information is sent to the terminal, as specified in ADL data element 22a. In step 705 the system waits for an answer, as specified in the first ADL data element 23 illustrated in Figure 5. In step 706 the system selects, as specified in ADL data element 21b, DDL data elements corresponding to questions in category "personal". These DDL data elements are data elements 11 and 12 illustrated in Figure 6a. In step 707 the system collects answers to information sent to a terminal in step 704, and evaluates feedback relating to DDL data elements 13 and 14 (as specified in ADL data element 22b). In step 708 the system constructs a DDL block containing the questions and answers of DDL data elements 11 and 12 and feedback relating to DDL data elements 13 and 14. These actions are specified in ADL data element 22b. In step 709, the system applies the selected XSLT file to the DDL block and in step 710 it sends presentable information to the terminal. These actions are also specified in ADL data element 22b. In step 711, the system waits for an answer. In step 712 it collects the answers relating to DDL data elements 11 and 12, and evaluates (as specified in ADL data element 22c) feedback to the whole DDL form using the feedback to the DDL data elements 11 and 12 and scores, which are calculated using feedback to DDL data elements 13 and 14. In step 713 the system constructs a DDL block comprising the feedback to the DDL form, in step 714 the selected XSLT file is applied to the DDL block, and in step 715 the resulting presentable information is sent to the terminal. These actions are defined in ADL data element 22c. In step 716, the system sends email to a service administrator, as specified in ADL data element 24 in Figure 5.

Figure 8 illustrates schematically some services provided by a system 800 according to one advantageous embodiment of the invention. The system 800 comprises, as an example, databases 801a, 801b and 801c for storing content data units 10, action data units 20 and presentation data units 30. It is alternatively possible that these data units are stored in one database or that some of these data units are stored in other storage means (for example, as files on a disk) relating to the system 800. The system 800 typically comprises similar program code means 320, 340 as system 300 illustrated in Figure 3. The services illustrated in Figure 8 comprise the interactive question-based applications 830, communication services 840 for delivering the applications to users and production interface 820 for creating and/or modifying the content and/or actions data units using a content/action editing tool 810. The

production interface 820, which is typically implemented as program code means for receiving content/action data units, receives information delivered using certain protocols 821. Figure 8 presents HTTP as an example. The content/action editing tool 810 may be a batch authoring tool (e.g. Excel template), interactive standalone editor or a form available on the terminal for instant input. The content/action editing tool 510 may reside on a separate workstation, and created content/actions is typically transmitted over a data network via the production interface 810 to a database 801 or to other storage means.

10 If the functionality of interactive question-based applications is implemented using, for example ADL and specific interpreters, it is possible to store to a database also data units 20. The behaviour of the interactive question-based application can then be modified at execution time by modifying and interpreting the ADL action description. This also makes it possible to create and add new interactive applications to the system by users. The ADL description (file) can be edited either directly or by using an editing tool and taken into use while the application is in use. The changes affect only new session and existing ones continue to use the old one until they end. A production interface 820 may be used by various content providers and/or concept providers. Alternatively it is possible that separate interfaces and editing tools are provided for content and actions.

If the functionality of interactive question-based applications is implemented using some general programming language, such as Java, an ADL or other unambiguous description of an application is typically programmed to program code separately. In this case it typically is not possible to modify online the data elements 20, which are program code. We may use ADL as the interpreted computer-processable specification language that is used in the actual system, or only as a definition language for the implementation in another language, such as Java – or the computer-processable specification language can use the syntax of Java.

30 In addition to production interface 820, there may be an analysis service and interface 850. This interface allows, for example, a concept provider or customer to monitor the use of a certain interactive question-based application. If a communication protocol 851, which the interface 850 supports, is HTTP (as Figure 8 illustrates as an example), the analysis service may be accessed via a browser. Furthermore, an administration service and interface 860 may be provided. These services are used for adding new applications, for starting and stopping them and for managing the users of the applications. Similarly as the analysis services, the

administration services may be accessed using a browser. It is possible that the analysis and administration services and interfaces support also or alternatively other communication protocols than HTTP.

5 Figure 8 illustrates also various communication services 840, which a system 800 may support, for delivering the interactive question-based applications to a user. As examples, Figure 8 mentions SMS, HDML (Handheld Device Markup Language), WAP and Web. Various communication protocols 841 are presented, as examples, in Figure 8; 841a is Short Messages Service Center Access protocol, 841b is HTTP
10 and 841c is packet data network protocol X.25. A further example of a communication delivery service is interactive TV and related communication protocol.

The invention can be used in many different situations, where questions are asked
15 from users, their answers are recorded, and some kind of feedback is given back to the users. The invention can be used for example in gallups, personnel questionnaires, voting, and in many other situations. The inventive system provides an easy way of creating new questionnaires and defining a new logic and/or new content for new or already existing questionnaires.

20 The invention can be used to support instant creation and publishing of interactive question-based games and other interactive question-based applications, such as surveys. Furthermore, the invention can be used to provide the end-user (consumer) generated interactive question-based applications, where end users (for example, a
25 user group) can generate an application for their own use. It is possible to create games or other applications, where both game logic and content can be defined according to the need, on-the-fly. Examples of such applications comprise: event related games created based on the current situation (typical examples are sport: events and TV program related games and surveys) and creation of *ad-hoc* games by
30 end users to a peer group (typical examples include a community of users interested in a particular topic, such as golfers or football fans, or a group in social context, such as in a party or other gathering.

The elements of the system, which enable this instant creation of interactive
35 question-based applications, include the elements described earlier: editing tools for defining content and/or logic for an application; production interface for content and/or concept uploading; storage of the uploaded application; possibility to start sessions relating to the uploaded application without any management or

configuration procedure (supported especially by the online interpretation of computer-processable specification of the logic of an application); and communication services for presenting an application to end users via various communication networks to various terminals.

5

In view of the foregoing description it will be evident to a person skilled in the art that various modifications may be made within the scope of the invention. While some preferred embodiments of the invention have been described in detail, it should be apparent that many modifications and variations thereto are possible, all of which fall within the true spirit and scope of the invention.

10

Claims

1. Method (200) for providing interactive question-based applications over a communications network, characterized in that the method comprises the following steps:
- 5
- defining (201) at least one question in a data unit of a first type,
 - defining (202) in a data unit of a second type at least one action for processing questions in a data unit of the first type for producing presentable questions, and
 - defining (203) in a data unit of a third type rules for processing presentable
- 10 questions for display on a specific type of terminal.
2. A method according to claim 1, characterized in that
- the data unit of the first type comprises a plurality of first data elements defining data content for the interactive question-based application,
- 15
- the data unit of the second type comprises a plurality of second data elements defining actions, and
 - the data unit of the third type comprises rules for processing presentable information for display on a specific type of terminal.
- 20
3. A method according to claim 2, characterized in that the data unit of the second type further defines the order, in which the second data elements defining actions are to be processed.
4. A method according to claim 3, characterized in that the data unit of the
- 25 second type further comprises control structures defining conditional actions.
5. A method according to claim 2, characterized in that the method further comprises the step of:
- carrying out (205) actions defined in at least some second data element belonging
- 30 to said plurality of second data elements.
6. A method according to claim 5, characterized in that said step of carrying out actions comprises at least one of the following substeps:
- selecting at least one first data element in the data unit of the first type, resulting in
- 35 selected first data element(s),
- processing selected first data element(s) using rules comprised in the data unit of the third type, resulting in presentable first data element(s) for display on a terminal of a certain type,

- transmitting presentable first data element(s) to a terminal of said certain type, or
- receiving an answer as a response to presentable first data element(s).

5 7. A method according to claim 2, characterized in that at least one of the first data elements comprises at least one of the following data fragments:

- a data fragment defining a correct answer,
- a data fragment defining a plurality of answer alternatives,
- a data fragment defining feedback a correct answer, or
- a data fragment defining feedback to answer alternatives.

10

8. A method according to claim 2, characterized in that at least one of the second data elements defines at least one of the following actions:

- selection of one first data element from a data unit of the first type,
- selection of a plurality of first data elements from a data unit of the first type,
- 15 - processing of selected first data element(s) using a data unit of the third type,
- transmission of selected processed first data element(s) to a user,
- retrieval of user input to presented first data element(s), or
- calculation of scores according to defined rules.

20 9. A method according to claim 1, characterized in that in the method,

- said at least one question in data units of the first type is defined in a first computer-processable specification language,
- said actions for processing are defined in data units of the second type in a second computer-processable specification language, and
- 25 - said rules for processing a set of presentable questions are defined in data units of the third type in a third computer-processable specification language.

10. A method according to claim 9, characterized in that said first and second computer-processable specification languages are defined in the XML language.

30

11. A method according to claim 9, characterized in that said first computer-processable specification language is defined in the XML language and said second computer-processable specification language has the syntax of a general programming language.

35

12. A method according to claim 11, characterized in that said second computer-processable specification language uses the syntax of Java.

13. A method according to claim 9, **characterized** in that the method further comprises the step of:

- storing answers obtained as a result of displaying questions in a fourth computer-processable specification language.

5

14. A method according to claim 13, **characterized** in that said first, second, third, and fourth languages are defined in the XML language.

15. A method according to claim 1, **characterized** in that the method further comprises the steps of:

10

- processing at least a part of a data unit of the first type according to at least a part of the definitions in a data unit of the second type for producing presentable questions,

- processing said presentable questions according to at least a part of the definitions

15

in a data unit of the third type for display on a specific type of terminal, and

- transmitting processed presentable questions to a terminal of said specific type.

16. A method according to claim 1, **characterized** in that said method further comprises the step of:

20

- providing an editing tool for producing and/or modifying an interactive question-based application, said editing tool enabling the creation and/or modification of a data unit of the first type and/or a data unit of the second type.

17. A method according to claim 16, **characterized** in that said method further comprises the steps of:

25

- receiving from an editing tool a data unit of the first type and/or a data unit of the second type, and

- initiating a new interactive question-based application session after the receipt of data unit(s) from said editing tool, said session using the received data unit(s).

30

18. A method according to claim 1, **characterized** in that the terminal of said specific type is a mobile communication device.

19. A method according to claim 1, **characterized** in that the terminal of said specific type is a television set or a set-top-box of an interactive television with appropriate feedback device, such as remote controller.

35

20. A method according to claim 1, characterized in that the terminal of said specific type is a computer.
21. A method according to claim 1, characterized in that said method further
5 comprises the step of:
- defining an unambiguous description for functionality of said interactive question-based application,
and in that said at least one action is defined using said unambiguous description.
- 10 22. A method according to claim 21, characterized in that said unambiguous description is defined using an XML-based language.
23. System (300) for providing interactive question-based applications over a communications network, characterized in that said system comprises at least
15 means for storing
- a data unit (10) of a first type having the definition of at least one question,
- a data unit (20) of a second type having the definition of at least one action for processing of at least one question in a first data unit and for producing at least one presentable question, and
20 - a data unit (30) of a third type having the definition of at least one rule for processing of a presentable question for display on a specific type of terminal;
and
- computer code means (320) for processing at least a part of a data unit of the first type according to at least a part of the definitions in a data unit of the second type
25 for producing presentable questions,
- computer code means (340) for processing presentable questions according to at least a part of the definitions in a data unit of the third type for display on a specific type of terminal, and
- computer code means for transmitting processed presentable questions to a
30 terminal of said specific type.
24. A system according to claim 23, characterized in that the computer code means for processing at least a part of a data unit of the first type comprise an Extensible Markup Language parser and an interpreter for a Extensible-Markup-
35 Language-based language.

25. A system according to claim 23, characterized in that the computer code means for processing at least a part of a data unit of the first type comprise a program execution environment.
- 5 26. A system according to claim 25, characterized in that the program execution environment is Java Run-time Environment.
27. A system according to claim 23, characterized in that the system further comprises
- 10 - computer program code means for receiving an answer to a question displayed to a user, and
- computer program code means for storing received answers in a memory means.
28. A system according to claim 23, characterized in that the system further comprises
- 15 - computer program code means for receiving information comprising definition of a question, and
- computer program code means for storing received information comprising definition of a question in a data unit of the first type.
- 20 29. A system according to claim 23, characterized in that the system further comprises
- computer program code means for receiving information specifying action definitions, and
- 25 - computer program code means for storing specified action definitions in a data unit of the second type.
30. Computer program product for a system for providing interactive question-based applications over a communications network, characterized in that
- 30 the computer program product comprises at least
- computer program code means for storing
- a data unit of a first type having the definition of at least one question,
- a data unit of a second type having the definition of at least one action for processing of at least one question in a data unit of a first type and for producing at
- 35 least one presentable question, and
- a data unit of a third type having the definition of at least one rule for processing of a presentable question for display on a specific type of terminal;
- and

- computer code means for processing at least a part of a data unit of the first type according to at least a part of the definitions in a data unit of the second type for producing presentable questions,
 - computer code means for processing presentable questions according to at least a part of the definitions in a data unit of the third type for display on a specific type of terminal, and
 - computer code means for transmitting processed presentable questions to a terminal of said specific type.
- 10 31. Computer program product according to claim 30, characterized in that it is stored on a computer readable medium.

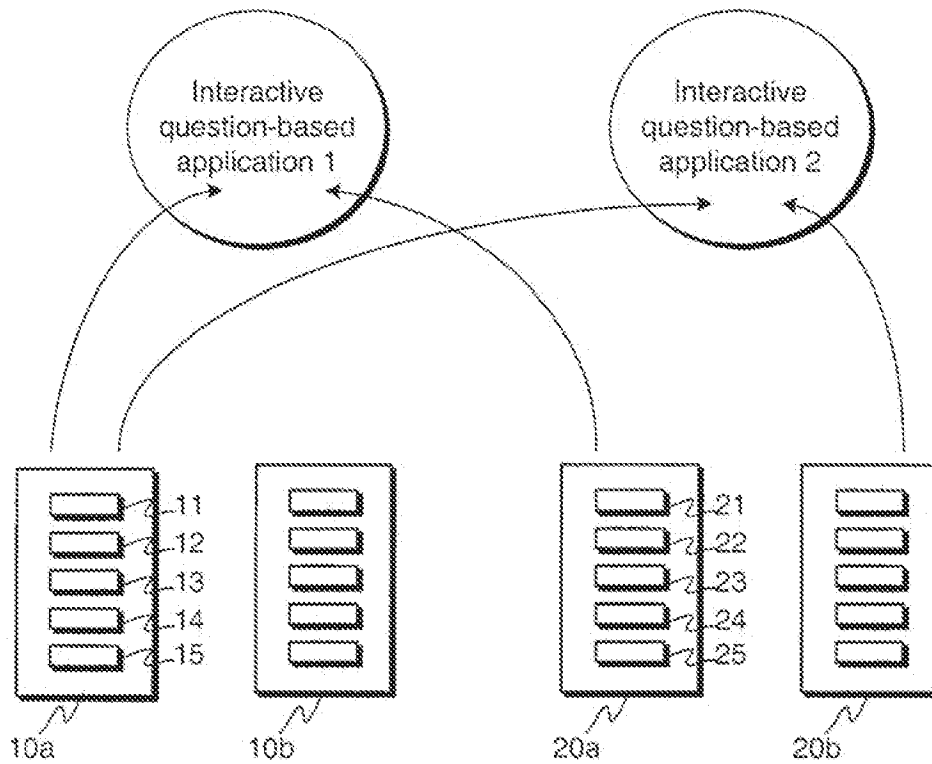


Fig. 1

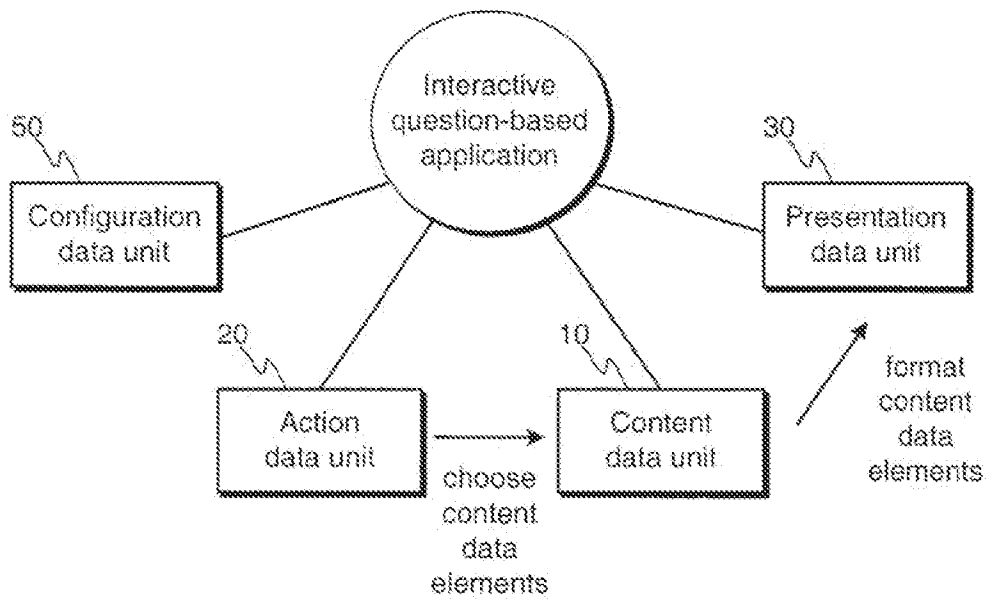


Fig. 4

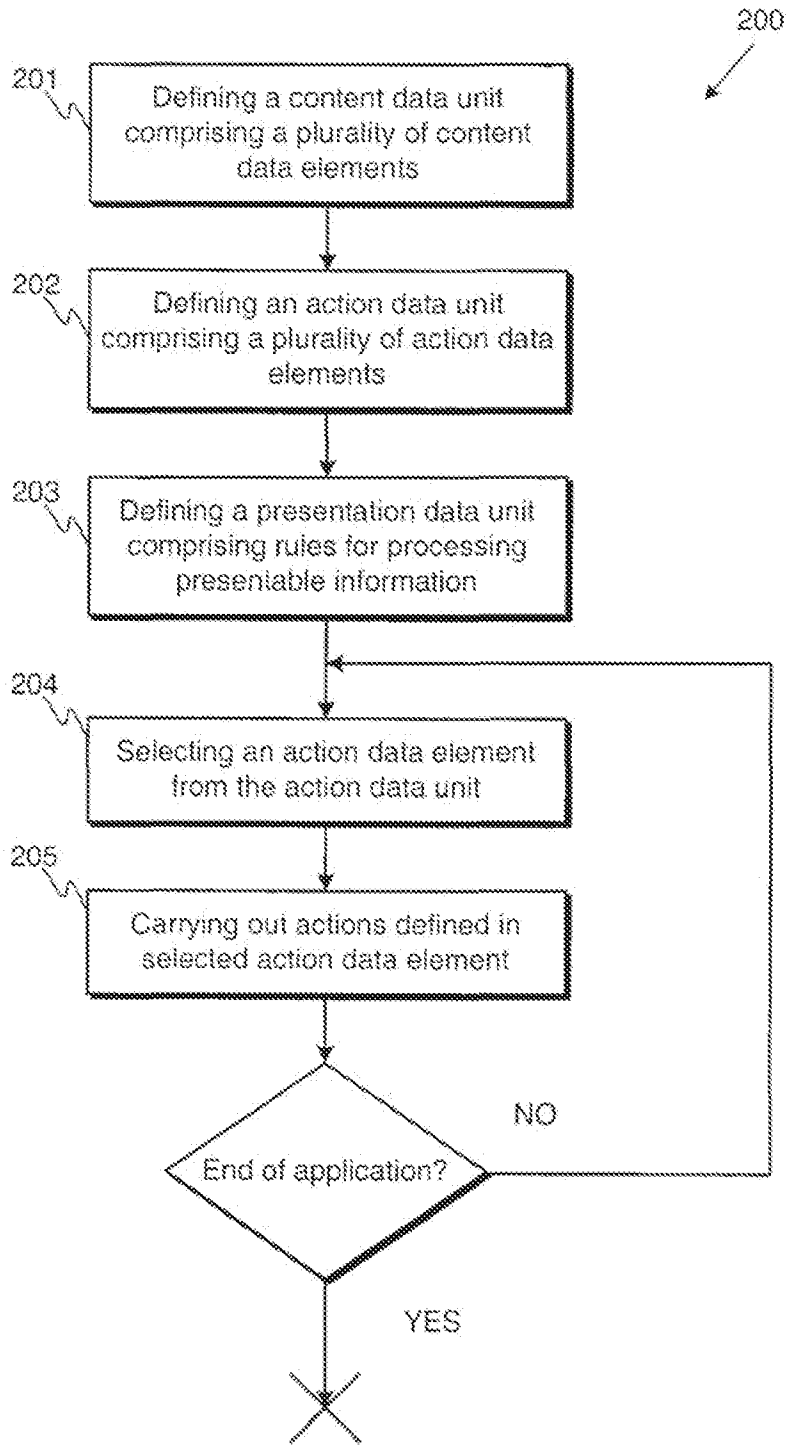


Fig. 2

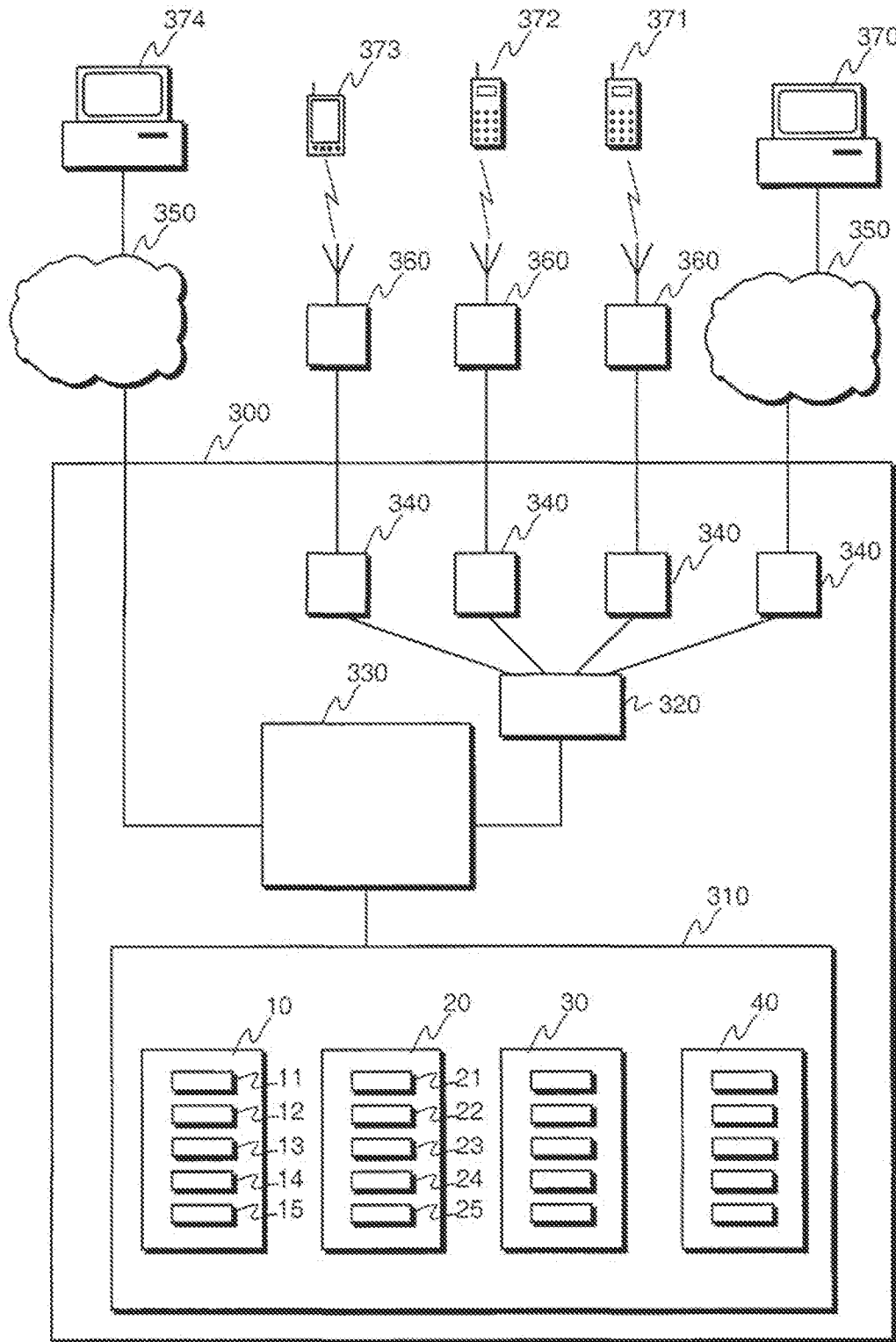


Fig. 3

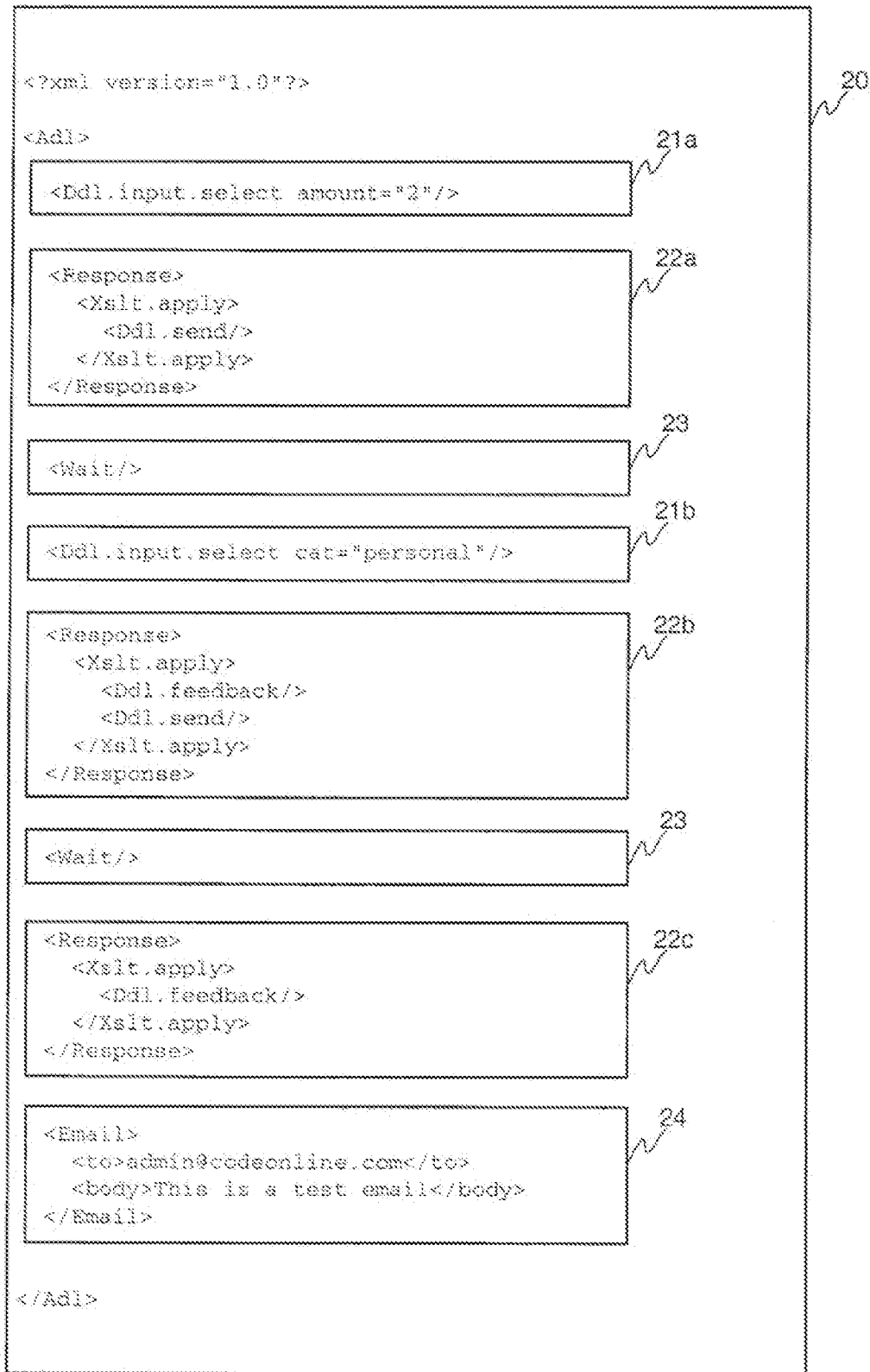


Fig. 5

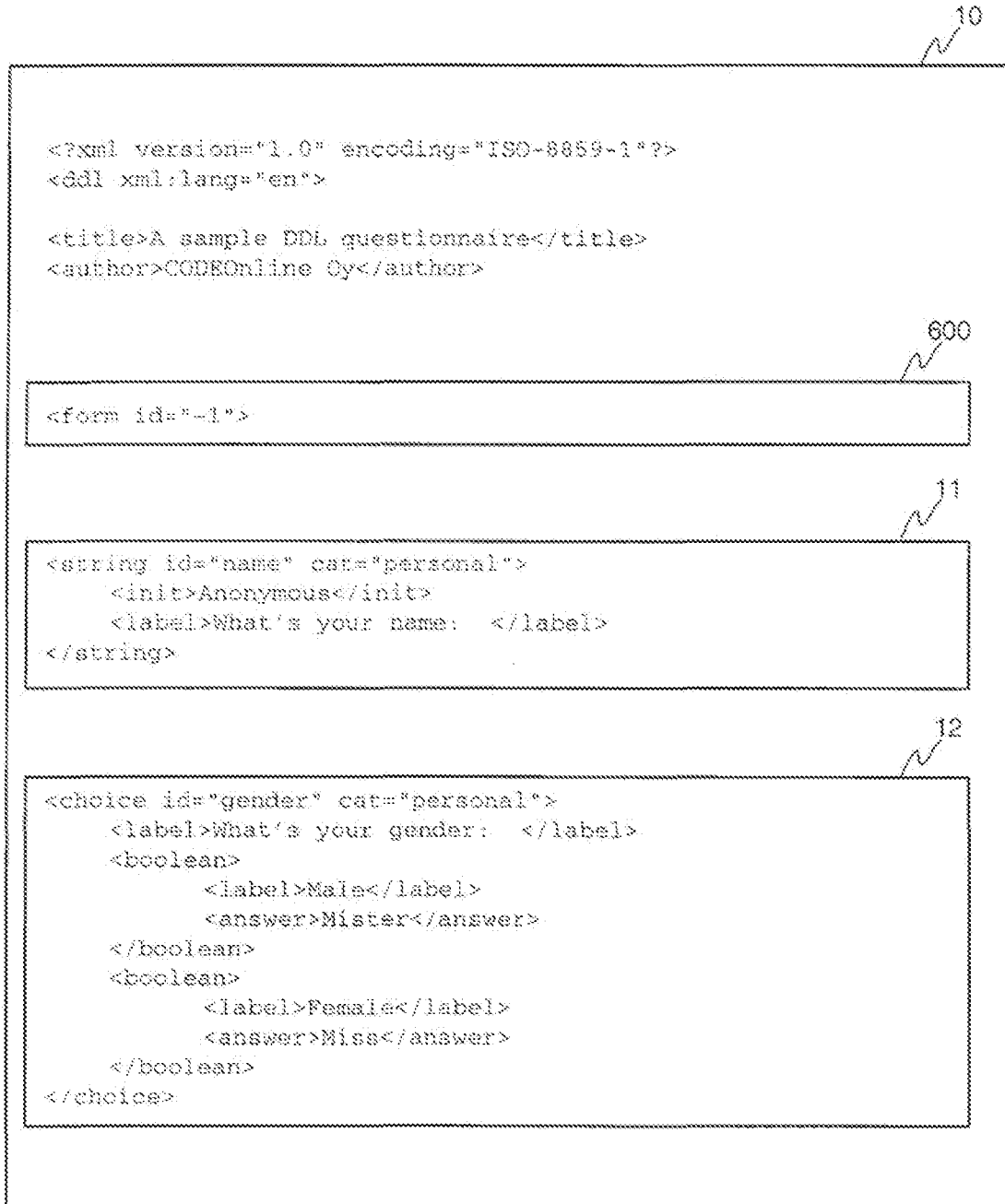


Fig. 6a

6/8

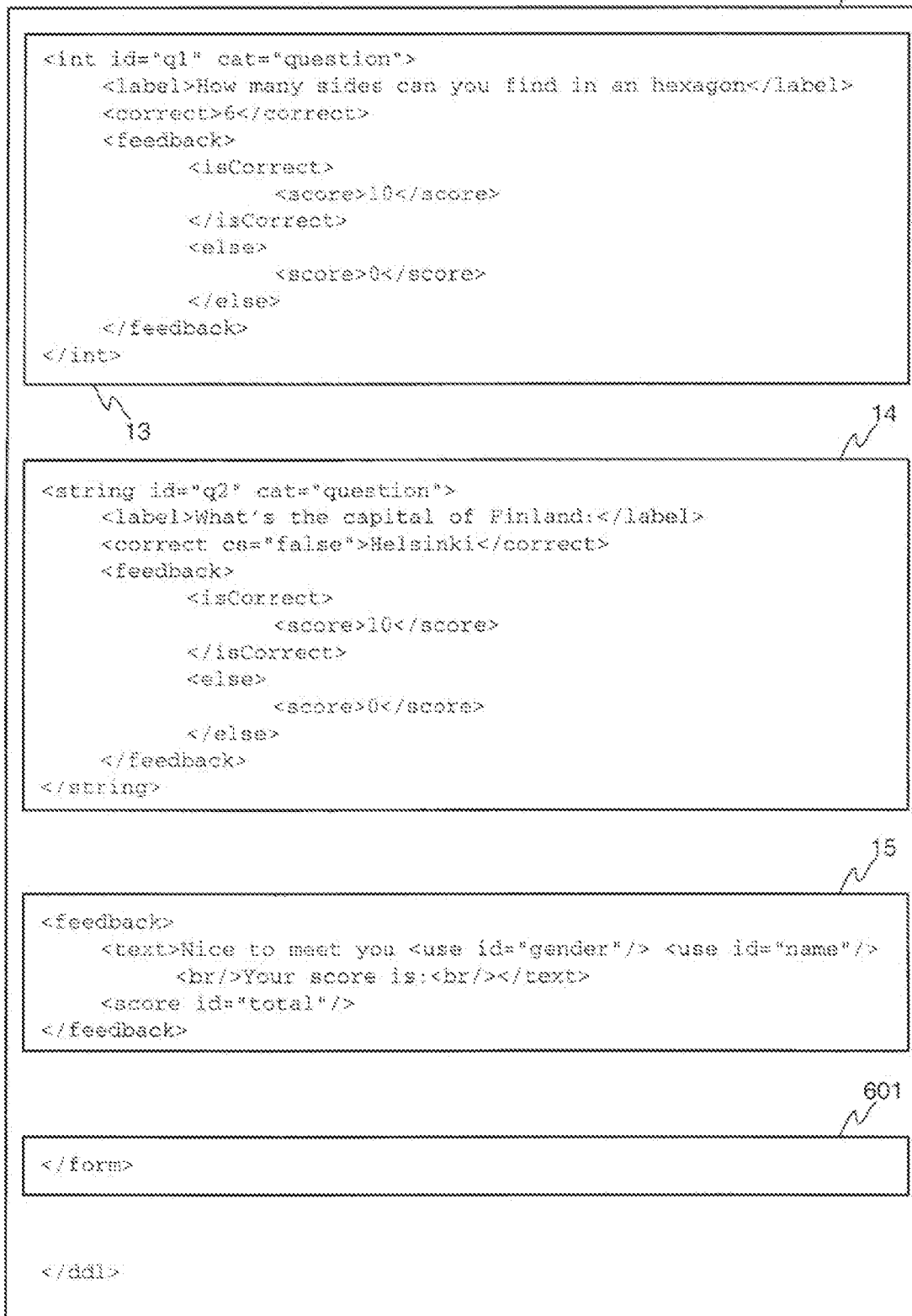


Fig. 6b

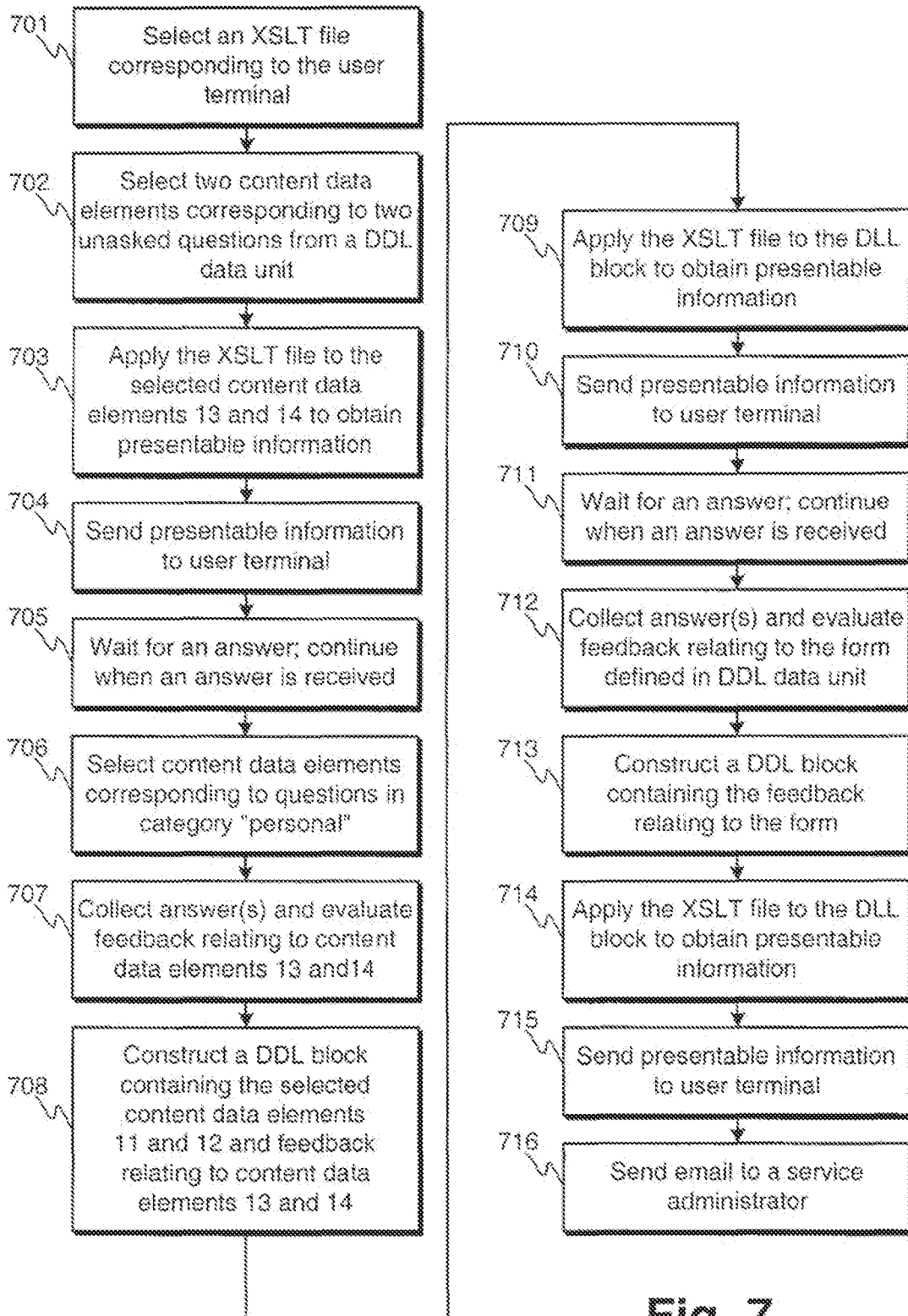


Fig. 7

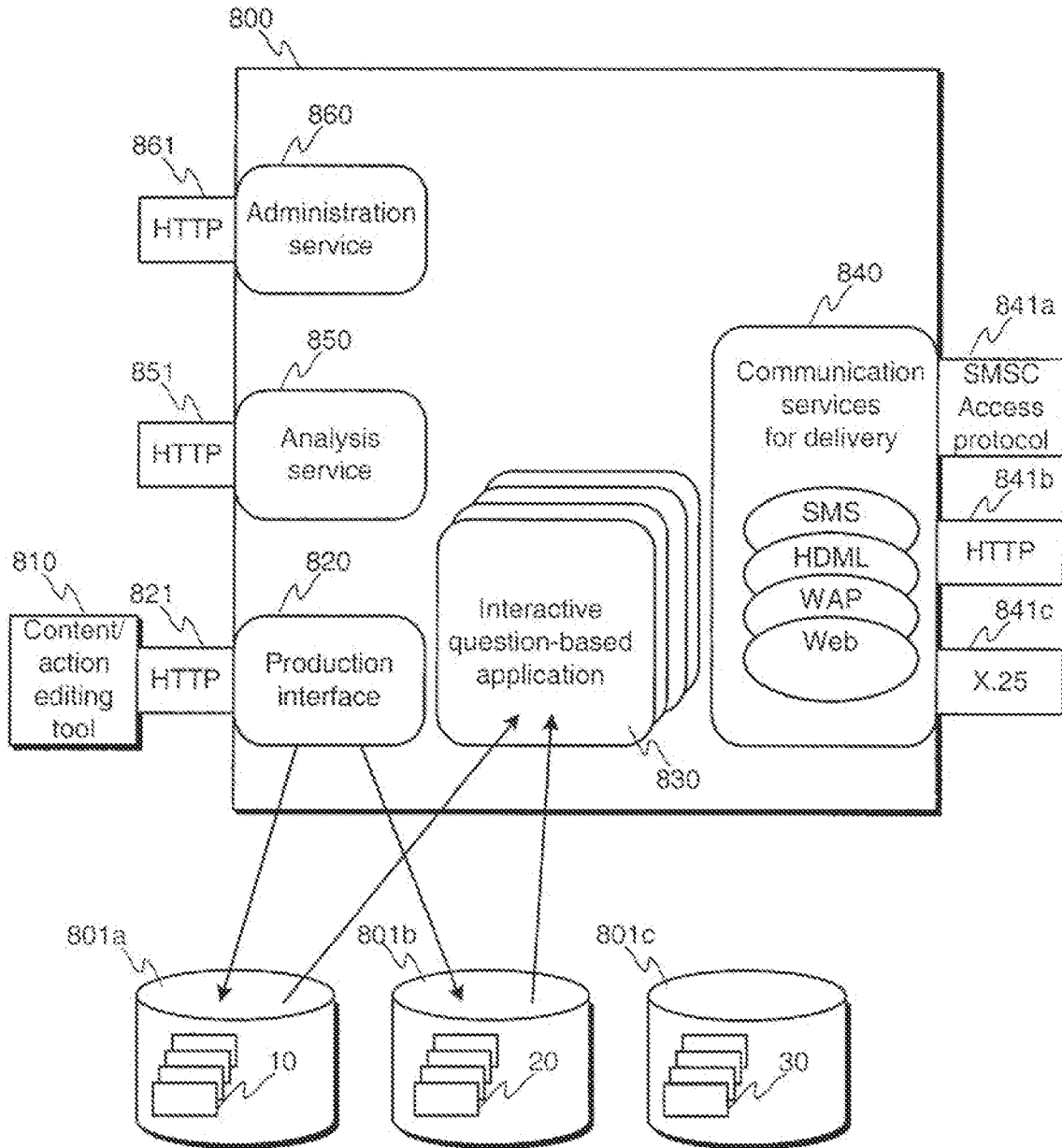


Fig. 8

INTERNATIONAL SEARCH REPORT

International application No.

PCT/FI 01/00274

A. CLASSIFICATION OF SUBJECT MATTER		
IPC7: G06F 17/30 According to International Patent Classification (IPC) or to both national classification and IPC		
B. FIELDS SEARCHED		
Minimum documentation searched (classification system followed by classification symbols)		
IPC7: G06F		
Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched		
SE,DK,FI,NO classes as above		
Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)		
EPO-INTERNAL, WPI-DATA, PAJ		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 6012098 A (ELIAS N. BAYEH ET AL), 4 January 2000 (04.01.00), column 1, line 7 - line 11; column 1, line 58 - line 61; column 3, line 30 - line 53, column 3, line 63 - column 4, line 47; column 8, line 13 - line 35; figure 4-5; claims 1-8; abstract	1-31
A	WO 0045304 A1 (OBJECT DESIGN, INC.), 3 August 2000 (03.08.00), page 2, line 29 - page 3, line 12; page 6, line 35 - page 7, line 23, claim 1, abstract	1-31
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> See patent family annex.		
* Special categories of cited documents: "A" document defining the general state of the art which is not considered to be of particular relevance "E" earlier application or patent but published on or after the international filing date "L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) "O" document referring to an oral disclosure, use, exhibition or other means "P" document published prior to the international filing date but later than the priority date claimed "T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention "X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone "Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art "&" document member of the same patent family		
Date of the actual completion of the international search		Date of mailing of the international search report
26 June 2001		28-06-2001
Name and mailing address of the ISA: Swedish Patent Office Box 5055, S-102 42 STOCKHOLM Facsimile No. +46 8 666 02 86		Authorized officer Pär Heimdahl/MN Telephone No. +46 8 782 25 00

Form PCT/ISA/210 (second sheet) (July 1998)

INTERNATIONAL SEARCH REPORT
Information on patent family members

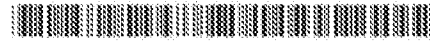
29/05/01

International application No.

PCT/FI 01/00274

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 6012098 A	04/01/00	NONE	
WO 0045304 A1	03/08/00	AU 2633800 A	18/08/00

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
4 October 2001 (04.10.2001)

PCT

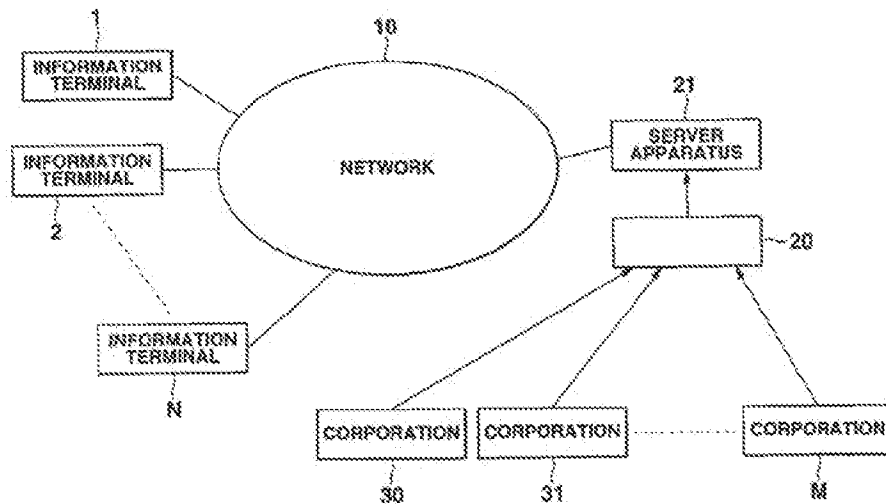
(10) International Publication Number
WO 01/73628 A2

- (51) International Patent Classification: G06F 17/60 [JP/JP]: 1297, Nakagomicho, Akishima-shi, Tokyo 196-0022 (JP); KATO, Yoshiyuki [JP/JP]: 2-32-10, Nangai, Higashiyamato-shi, Tokyo 207-0014 (JP).
- (21) International Application Number: PCT/JP01/02696
- (22) International Filing Date: 27 March 2001 (27.03.2001)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
2000-89042 28 March 2000 (28.03.2000) JP
2000-291877 26 September 2000 (26.09.2000) JP
- (71) Applicant (for all designated States except US): CASIO COMPUTER CO., LTD. [JP/JP]: 6-2, Hon-machi 1-chome, Shibuya-ku, Tokyo 151-8543 (JP).
- (72) Inventors; and
- (73) Inventors/Applicants (for US only): NAKAZAWA, Eiji
- (74) Agents: SUZUYE, Takehiko et al.; c/o SUZUYE & SUZUYE, 7-2, Kasumigasaki 3-chome, Chiyoda-ku, Tokyo 100-0013 (JP).
- (81) Designated States (national): AU, CA, CN, KR, ES.
- (84) Designated States (regional): European patent (DE, FR, GB).

Published:
— without international search report and to be republished upon receipt of that report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: SETTLEMENT SYSTEM AND SERVER APPARATUS



(57) Abstract: There is disclosed a settlement system in which predetermined image data is transmitted beforehand to a purchaser client (1, 2, ...N) as a substitute for a commodity ticket, a coupon ticket or cash, and the client transmits the image data to a virtual shop when purchasing a commodity in the virtual shop opened on Internet, and can settle commodity purchase on the Internet (10), so that the client can use the image data to buy the arbitrary commodity on the Internet. Moreover, a server apparatus (21) for transmitting the image data to the purchaser client is provided with a program for verifying and settling whether or not the same image data as the image data transmitted to the client has been received, so that the image data can be prevented from being mistaken or abused.

WO 01/73628 A2

When one participate in the virtual shop to purchase the commodity, a credit card is frequently utilized to pay for the commodity. However, in case of payment via the credit card, an identification number or another secret data has to be informed. A general user which purchases the commodity (hereinafter referred to as a client) is disadvantageously worried about safety of the secret data, which inhibits the virtual shop from being further developed.

Moreover, in many of general shops for actually selling the commodity, a predetermined sheet is stamped for every purchase of the commodity, and a discount ticket or coupon is distributed, so that a customer becomes a repeater to visit the shop again. However, it has been difficult to employ such system in the virtual shop on Internet, because an article corresponding to the stamp, the discount ticket, the coupon, or the like cannot be distributed.

On the other hand, for the application to a quiz, for example, the prize can be obtained for nothing, but the prize is sometimes not necessarily what one wants to have, and an unnecessary article is obtained in some cases. Therefore, if the won prize can be what one wants to have, popularity is further enhanced, and the network technique of Internet is further developed, but such system has not been realized.

Disclosure of Invention

The present invention has been developed to solve the aforementioned problem, and an object thereof is to provide a settlement system in which a service
5 corresponding to a stamp, a discount ticket, a coupon, or the like can be presented even on a network.

Moreover, an object of the present invention is to provide a server apparatus for managing the settlement system on network.

10 Furthermore, an object of the present invention is to provide a system in which a desired commodity can be obtained in various applications, a settlement system in which no uncertainty is given to a client in a virtual shop, or a server apparatus for managing the
15 system.

To achieve the aforementioned objects, according to the present invention, there is provided a settlement system comprising: transmission means for transmitting image data including a money amount to a
20 purchaser client via network; reception means for receiving the image data including the money amount from the purchaser client via the network; verification means for checking whether or not the image data including the money amount received from the purchaser
25 client via the reception means is the transmitted image data including the money amount; and settlement means for allowing the purchaser client to settle an account

in a range of the money amount via the image data including the money amount when the verification means verifies that the image data is the transmitted image data including the money amount.

5 That is, in the present invention, instead of sending an article to a client, image data including the money amount is transmitted to the client as money on network, and the client can advantageously use the image data to purchase an arbitrary commodity on the
10 network. Moreover, for a transmitter, it is checked whether or not the same image data as the transmitted image data including the money amount is received, and the account is settled, so that the image data can be prevented from being mistaken or abused.

15 Brief Description of Drawings

For a better understanding of the present invention, reference is made of a detailed description to be read in conjunction with the accompanying drawings, in which:

20 FIGS. 1 to 14 show a first embodiment of the present invention, and FIG. 1 is a system diagram showing a relation among an information terminal, network and server.

FIG. 2 is a block diagram showing a detailed
25 constitution of a server apparatus 21 of FIG. 1.

FIG. 3 is a diagram showing a detailed constitution of questionnaire data storage means 26 of

the server apparatus 21.

FIG. 4 is a flowchart showing a procedure and content of questionnaire advertisement according to the present invention.

5 FIG. 5 is a diagram showing a display example of questionnaire advertisement open to the public on a home page of a service company.

FIG. 6 is a diagram showing a display example of image data for use in the present system.

10 FIG. 7 is a diagram showing another display example of image data for use in the present system.

FIG. 8 is a diagram showing details of a winner DB27 of the server apparatus 21.

15 FIG. 9 is a diagram showing a display example of a virtual shop home page of the service company on Internet.

FIG. 10 is a diagram showing details of a commodity DB28 of the server apparatus 21.

20 FIG. 11 is a diagram showing a display example of a commodity order table of the home page of the service company on Internet.

FIG. 12 is a similar diagram showing the display example of the commodity order table.

25 FIG. 13 is a diagram showing a display example of electronic mail transmission data to the service company.

FIG. 14 is a flowchart of a processing performed

by control of a processing server 24.

FIGS. 15 to 29 show a second embodiment of the present invention, and FIG. 15 is a system diagram showing the relation among the information terminal,
5 network and server.

FIG. 16 is a block diagram showing a detailed constitution of a server apparatus 51 of FIG. 15.

FIG. 17 is a block diagram showing further detailed constitution of the server apparatus 51.

10 FIG. 18 is a diagram showing a detailed constitution of a member data memory 71 disposed in a data storage section 59 of the server apparatus 51.

FIG. 19 is a diagram showing a detailed constitution of an image data memory 80 disposed in the
15 data storage section 59 of the server apparatus 51.

FIG. 20 is a diagram showing a detailed constitution of an order reception data memory 91 disposed in the data storage section 59 of the server apparatus 51.

20 FIG. 21 is a diagram showing a detailed constitution of a member service image data memory 100 disposed in the data storage section 59 of the server apparatus 51.

FIG. 22 is a flowchart for setting and storing
25 image data displayed in the home page into the image data memory 80.

FIG. 23 is a diagram showing a display screen for

processing the flowchart of FIG. 22.

FIG. 24 is a diagram showing the image data set and stored by the processing of the flowchart of FIG. 22.

5 FIG. 25 is a similar diagram showing the image data set and stored by the processing of the flowchart of FIG. 22.

FIG. 26 is a similar diagram showing the image data set and stored by the processing of the flowchart
10 of FIG. 22.

FIG. 27 shows a display example of the virtual shop disposed in the home page.

FIG. 28 is a flowchart of a processing performed by control of the server apparatus 51.

15 FIG. 29 is a detailed flowchart of an order reception processing of FIG. 28.

Best Mode for Carrying Out of the Invention

Referring now to drawings, various preferred embodiments of the present invention will be described
20 in detail.

First Embodiment

An example of a first embodiment of the present invention will be described hereinafter with reference to the drawings.

25 In FIG. 1, personal computers 1, 2, ..., N as respective client information terminals in individual houses and offices are connected to a network 10 such

as Internet via a public telephone network (not shown) and the like by a contract made with a provider (not shown). The network 10 is connected to a computer including a server apparatus 21, described later, of a service company 20 which advertises, for example, for questionnaire, commodity present, quiz, monitor, and the like. In the server apparatus 21, a home page or the like is opened to invite public participation in questionnaire inquiry, commodity present, quiz, and monitor (hereinafter referred to as "client participation") via Internet. Moreover, the server apparatus can receive and store an electronic mail as an answer to each client advertisement.

Request content data from respective corporations 30, 31, ..., M which ask for the client participation, for example, in questionnaire inquiry are sent to the service company 20 via mail, facsimile, electronic mail, and the like. The service company 20 prepares and opens to the public a home page for collecting answers to the questionnaire inquiry based on these data. Additionally, a virtual electronic shopping mall called a virtual shop is also opened in the home page as described later.

FIG. 2 shows a detail of the server apparatus 21 of the service company. The server apparatus is provided with: a Web server 22, connected to the network 10 such as Internet, for transmitting and

receiving data with respect to the network 10 via the telephone network; an electronic mail server 23 for transmitting/receiving an electronic mail; and a managing/processing server 24 for managing data of a database server 25 described later and using the
5 respective data to perform various processings. These servers are connected to one another via a cable, or the like.

Additionally, excluding the database server 25, as
10 not shown, each of the servers is provided with a display apparatus such as a CRT or a large-sized LCD, input means such as a keyboard, mouse, and scanner, means for attaching an external memory, an internal memory, and CPU, and executes respective functions.

The database server 25 is provided with
15 questionnaire data storage means (hereinafter referred to as questionnaire DB) 26 for storing data such as questions and answers regarding questionnaire performed by the service company. The questionnaire DB 26 is
20 constituted, for example, of a large-capacity RAM, ROM, hard disk or optical magnetic disk as shown in FIG. 3.

In FIG. 3, in a questionnaire name area 26A, a type of questionnaire to perform, and questionnaire name data are stored. In a questionnaire period area
25 26B, questionnaire period data is stored. Moreover, in a request corporation name area 26C, a name of a corporation or a group which has requested the service

company for the questionnaire is stored.

In respective answer area rows 26D, 26E, 26F, ..., data of those who answer the questionnaire are stored (one row indicates the data of one person). As shown
5 without any reference numeral, areas for storing each name, address, age, and sex are disposed. Moreover, areas for storing respective answers to each question of the questionnaire (answers 1, 2, 3, ...) are disposed.

10 Furthermore, as described later, an article corresponding to a prize is sent to a winner among the questionnaire answering people by lot. In a last storage area 26G of each row, a won money amount is stored.

15 In an area 26H, total data of all answers to the questionnaire is stored. The managing/processing server 24 totals all answers of the answer areas 26D, 26E, 26F, ... and stores a result in the area. As shown in FIG. 3, the total number of applicants, the
20 number of applicants by sex, the number of applicants by age, and collected number of data of answer contents to the respective questions, and the like are stored.

As described above, the same storage areas as the storage areas 26A to 26H are disposed in the
25 questionnaire DB 26, and questionnaire data regarding a large number of questions are stored.

Additionally, in addition to the questionnaire

inquiry, to invite the client participation in applications for the commodity present, quiz, monitor, and the like, data of the participation content, and answer content are similarly stored.

5 Turning back to FIG. 2, the database server 25 is provided with winner data storage means (hereinafter referred to as the winner DB) 27 to store data such as prizes sent to winners selected by lot from those who answers the questionnaire. The database server is also
10 provided with: commodity data storage means (hereinafter referred to as commodity DB) for storing image data of a commodity photograph of the virtual shop, and name, price, and the like of the commodity; and mail data storage means (hereinafter referred to as
15 the mail DB) 29 for storing the data, and the like transmitted/received via the electronic mail.

Similarly as the questionnaire DB 26, each of the winner DB 27, commodity DB 28 and mail DB 29 is constituted of the large-capacity RAM, ROM, hard disk
20 or optical magnetic disk. Details of the winner DB 27 and commodity DB 28 will be described later.

FIG. 4 shows a procedure and content for inviting the participation in the questionnaire. This processing is performed in order of steps (1) to (15)
25 with elapse of time.

First, a corporation who wants to perform questionnaire inquiry informs the service company of

a questionnaire content and a commodity or a prize to be presented to a person who makes an application for the questionnaire step (1).

The service company receives such information
5 and places questionnaire advertisement in its own home page step ((2)), or opens a virtual shop (step(3)). Additionally, if the virtual shop is already open, the virtual shop does not have to be newly opened.

FIG. 5 shows a display example of questionnaire
10 opened to the public in the home page of the service company. Therefore, the client making application for the questionnaire displays this home page in the client's personal computer (step (4) browses questionnaire page of FIG. 4), inputs an answer to
15 the question, address, name, age, sex, and mail address in frames (step (5) prepares questionnaire answer), and clicks a transmit button, so that the input data is sent to the server apparatus 21 of the service company via the electronic mail, or the like by
20 the network (6).

Subsequently, answer data sent from the respective clients are successively stored/held in the mail DB 29 via the electronic mail server 23.

After the questionnaire period, the processing
25 server 24 collects the all answer data, and stores all answer data and totaled data in the questionnaire DB 26. Moreover, the service company chooses a winner

from the applicants by lot, announces the winner in the home page, and writes won money amount data in the money amount storage area 26G of the questionnaire DB 26 (7).

5 In an example of FIG. 3, Mr. B whose data is stored in the area 26E has won 10,000 yen, and Messrs. A and C has drawn blanks.

 Subsequently, the processing server 24 performs a processing of transmitting the image data (icon image data) including the won money amount to the winner via
10 the electronic mail (step (8)). FIGS. 6 and 7 show the image data (electronic money icon) including the money amount to be transmitted, FIG. 6 shows bit map data including characters indicating the money amount of
15 10,000 yen, and FIG. 7 shows the bit map data including characters indicating the money amount of 1,000 yen. These data are stored beforehand in a memory (not shown) of the processing server 24.

 The processing server 24 encrypts verification
20 code data, identification data, and the like comprising character data such as numerical values and characters, attaches these data to the image data, and sends the image data. Alternatively, the processing server 24 uses an electronic watermark technique to attach the
25 encrypted watermark data to the image data, and sends the image data.

 Additionally, the processing server 24 writes data

into the winner DB 27. FIG. 8 shows a detail of the winner DB 27. In FIG. 8, a questionnaire name area 27A is an area in which questionnaire name data is stored, and a questionnaire period area 27B is an area in which
5 questionnaires period data is stored. Moreover, a request corporation name area 27C is an area in which the name of a corporation or a group which has requested the service company for the questionnaire is stored.

10 In the next area 27D, a winner's name (name B), Mr. B's address, age, sex, electronic mail address, and other data are stored. Moreover, in a money amount area 27E, Mr. B's won money amount, for example, 10,000 yen is stored. Thus, it is stored that the
15 image including the money amount of 10,000 yen of FIG. 6 has been sent to Mr. B.

Furthermore, in an identification data storage area 27F, encrypted data attached to the image sent to Mr. B, or identification data attached with
20 an electronic watermark is stored. The identification data differs every time the data is issued from the processing servers 24, even if the data is the image data indicating the same money amount of 10,000 yen.

Therefore, in an area 27G, data of a date at which
25 the image data is sent to Mr. B is stored, and in an area 27H, date data of an effective period during which Mr. B can use the image data is stored. That is, Mr. B

can use the image data as described later to purchase the commodity at the virtual shop on the network, and the effective period is determined. Moreover, in a flag area F0, it is stored whether or not Mr. B has
5 purchased the commodity with the image data. When the area F0 indicates 0, he has not purchased the commodity yet. When he purchases the commodity, 1 is stored in the area F0.

Furthermore, the price of the commodity purchased
10 by Mr. B is not limited to 10,000 yen. For example, when he purchases the commodity of 7,500 yen, the image data for 2,500 yen is newly sent as a change. Mr. B can use the image data to purchase the commodity.

The presence of the change is stored in the flag
15 area F1. A date at which the commodity of 7,500 yen is purchased is stored in a purchase date storage area 27J, and 1 is stored in the flag area. In the next row, for 2,500 yen, the content of data similar to that described above is stored. Additionally, in FIG. 8, it
20 is stored that Mr. K has won 50,000 yen and that the image data has been sent to him.

When the image data is sent, each client who has answered the questionnaire can know that he has won money, and the won money amount (yen). Subsequently,
25 in order to purchase the commodity with the image data, the client opens the page of the virtual shop in the home page of the service company on Internet

(step (9)).

FIG. 9 shows the page of the virtual shop of the service company, and commodity data presented by each corporation is exhibited. That is, a commodity
5 photograph image (photograph 1, photograph 2 ...), and data of the photograph image, such as commodity name, model number, and price are displayed. The commodity DB 28 of FIG. 2 stores commodity data shown in FIG. 9, and is constituted as shown in FIG. 10. In FIG. 10,
10 the commodity photograph image (photograph 1, photograph 2 ...), and data such as commodity name, model number and price are stored. Moreover, a storage area of a name of a maker company having presented the commodity (28A), a storage area of the number of
15 commodities in stock (28B), and another storage area are disposed.

In FIG. 9, when the commodity to purchase is determined, and a cursor is aligned with a display button (not shown) with a mouse, and clicked, input
20 display is changed to an order table shown in FIG. 11. Here, the present date, and the name, model number, price, and the like of the commodity to purchase are set (selection or key input), and the electronic money icon is clicked as a payment method, the display
25 changes to that shown in FIG. 12. Subsequently, an electronic mail address "ankeito@xyz.co.jp" of the service company is clicked in accordance with

description, and electronic mail transmission data to the service company is displayed as shown in FIG. 13.

In FIG. 13, data inputted in FIG. 11 is copied as it is, total amount is also displayed, and a frame for
5 inputting a transmission client address, name, and mail address is further displayed. Subsequently, the client's address, name, and mail address are inputted into the frame, and the electronic money icon already sent as the won money amount is attached (step (10)).

10 Thereafter, when the transmit button is clicked, display content of FIG. 13 and the electronic money icon are transmitted to the service company via the electronic mail (step (11)).

It is verified in the service company whether or
15 not the electronic money icon sent via the electronic mail is authentic. This processing is performed as shown in a flowchart of FIG. 14 under the control of the processing server 24.

In FIG. 14, first in step S1, it is judged whether
20 or not the same data as those of the address, name, and electronic mail address of the electronic mail sent from the client exist in the winner DB 27. Subsequently, when the agreeing data exist, that is, when answer to step S2 is yes, the processing advances
25 to step S3. It is then detected whether or not the flag area F0 indicates 0. That is, in the money amount data sent to the client and stored in the money amount

area 27E, it is detected whether or not the flag area
F0 indicates 0, that is, whether or not there is an
electronic money icon unused by the client in the
virtual shop. If yes (Y), the processing advances to
5 step S4.

In the step S4, it is judged whether or not the
sent total amount of the commodity purchase is equal to
or less than the money amount of the unused money
amount data. If yes (Y), the processing advances to
10 step S5.

In the step S5, it is judged whether or not the
identification data attached to the sent electronic
money icon agrees with the identification data stored
in the identification data storage area 27F. When the
15 data agree with each other, it is judged that the
electronic money icon sent to the winner is returned as
it is and that the electronic money icon is authentic.
Then, the processing advances to the following step S6.

In the step S6, a difference between the money
20 amount data stored in the money amount area 27E and the
sent total amount of the commodity purchase, that is,
a change is calculated. If the difference is equal
to or more than 100 yen in the following step S7,
a carry-forward processing is performed in the
25 following step S8. In the carry-forward processing of
the step S8, the image data including the money amount
corresponding to the amount of the change (electronic

money icon) is transmitted to the client, and it is registered in the winner DB 27 that the new electronic money icon has been sent.

After the processing ends, or when the amount is equal to or less than 100 yen in step S7, the processing advances to step S9 to prepare a list of purchaser name, address and commodity.

Additionally, if the answer to any one of the steps S2 to S5 is no (N), a mail indicating a reason for the result is transmitted to the client (steps S10 to S13).

As described above, when the order for commodity purchase based on the authentic electronic money icon is received and the list of the name, address and ordered commodity is prepared, the list is sent to the corporation together with the total result of the questionnaire (steps (13) and (14)). Subsequently, the corporation sends the ordered commodity to each client based on the list (step (15)). This means completion of settlement of the commodity purchase by the purchaser using the electronic money icon.

Additionally, in the aforementioned embodiment, the winner is extracted from those who have made application for the questionnaire to answer the questions, so that the winner can buy the commodity with the electronic money icon. This can similarly be performed for the free present, the quiz, or

the applicant for the commodity monitor. Moreover,
the money icon data can be stored in recording mediums
such as a floppy disk, and a semiconductor (IC) memory,
and mailed to the client, or handed over to the client
5 at a window.

Further, the present invention can be applied not
only as a reward or a prize of the questionnaire, quiz,
or the like but also as a so-called prepaid system in
which the electronic money icon for purchasing the
10 commodity from the virtual shop on network such as
Internet is purchased beforehand by cash and the
commodity is purchased from the virtual shop with the
purchased electronic money icon. In this case, the
recording medium in which the electronic money icon is
15 stored may be sold beforehand.

Moreover, the virtual shop is opened on the home
page of the service company, but may be opened in the
home page of each corporation so that the electronic
money icon can be used to purchase the commodity.

20 Second Embodiment

A second embodiment of the present invention will
be described hereinafter with reference to the
drawings. Additionally, the same constituting portions
as those of the first embodiment are denoted with the
25 same reference numerals and detailed description
thereof is omitted.

In the second embodiment, a company or a maker

sells the commodity in the virtual shop (virtual mall) on Internet, and the client makes an order for the commodity so that an actual commodity is sent to the client. A network for this system is shown in FIG. 15.

5 In FIG. 15, a pizza shop 50 actually sells pizza, and also includes a server apparatus 51 connected to the network 10 such as Internet. Pizza orders from the personal computers 1, 2, ..., N of the clients who have already made contracts as members are received via the
10 network 10, and the ordered pizzas are cooked and sent to clients' homes. In this case, payment for the ordered pizza is settled by a pre-contracted financial institution 52, and the payment can partially or wholly be settled with service image data described later.

15 FIG. 16 shows a detail of the server apparatus 51 of the pizza shop 50. The server apparatus is provided with: a Web server 53, connected to the network 10 such as a public telephone network and Internet, for transmitting/receiving data with respect to the network
20 10 via the telephone network; an electronic mail server 54 for transmitting/receiving the electronic mail; and a managing/processing server 55 for managing data of a database 56 described later and using the respective data to perform various processings. These servers are
25 connected to one another via the cable, or the like.

FIG. 17 is a detailed circuit block diagram, for example, of the Web server 53 among the respective

servers 53 to 55. Additionally, the electronic mail server 54 and managing/processing server 55, and the Web server 22, electronic mail server 23 and managing/processing server 24 shown in FIG. 2 have constitutions similar to the constitution shown in FIG. 17, except for various programs and processing functions controlled by the programs. Therefore, detailed description of the constitution thereof is omitted.

10 The Web server 53 is provided with a controller (CPU) 61, ROM 62, data storage unit 63, driver 65 for driving a storage medium 64 to read/write a stored content, display 66, input unit 67, input/output unit 68 for transmitting/receiving data with respect to external apparatuses, and communication controller 69, and these respective units are connected to one another via a bus 70. Moreover, the bus 70 is also connected to the database 56 of FIG. 16.

20 The CPU 61 reads the system program and various control programs stored in the ROM 62, develops the programs in a work area (not shown) disposed in the storage unit 63 to execute the processing, and drives/controls the respective units. Moreover, the CPU 61 reads and executes the corresponding application from application programs stored in the ROM 62 or the storage medium 64 in response to an input signal transmitted via the input unit 67 and various accesses

transmitted via the communication controller 69, and executes various processings such as data updating described later.

Furthermore, the CPU 61 outputs an input content
5 from the input unit 67, various data stored in the storage unit 63 or the database 56, data stored in the storage medium 64, or content transmitted/received via the communication controller 69 to the display 66
such as CRT or liquid crystal display, and displays
10 the content or the data in a display screen of the display 66.

The ROM 62 stores various system programs and controls programs executed by the CPU 61, data regarding these programs, and the like.

15 The storage unit 63 forms a storage area for storing the data processed by the program executed by the CPU 61, application program and data read from the storage medium 64, and the like, and a work area for temporarily storing the processing data during
20 processing of various programs.

The driver 65 drives the storage medium 64 with the programs and data pre-stored therein to write and read these programs and data. The storage medium 64 is
25 constituted of a magnetic or optical recording medium, or a semiconductor memory for storing the system program, various application programs for the system, database update processing program, and data processed

by the respective processing programs. For example,
a floppy disk, CD-ROM, MD, IC card memory, or the like
may be used. This storage medium 64 can be disposed on
the driver 65 in a fixed or detachable/attachable
5 manner.

The input unit 67 is provided with various input
devices such as a keyboard including character keys,
numeric keys and various function keys, and pointing
devices such as a mouse and a tablet, to generate an
10 operation signal in accordance with an input operation
in these input devices and outputs the signal to the
CPU 61.

The display 66 is provided with the display screen
and includes a CRT, a liquid crystal display, or the
15 like as described above, and displays various images on
the display screen based on display information
inputted under the control of the CPU 61.

The input/output unit 68 inputs data from the
external apparatus, such as digital image data from an
20 electronic still camera and image scanner, and document
data from various information apparatus terminals, via
a cable or a wiring (not shown) and stores the data in
the storage unit 63, or outputs the data from the
storage unit 63.

25 The communication controller 69 is constituted of
a modem, a terminal adapter (TA), and the like, and
connected to the Internet network via communication

circuits such as the public telephone network and ISDN network. The communication controller 69 controls the network to perform communication.

Turning back to FIG. 16, the Web server 53 is
5 provided with the program for opening the home page of the pizza shop 50 on the network 10, and this home page constitutes a virtual shop for selling pizza (accepting orders) on the network. Moreover, the database 56 is provided with a data storage unit 57 for storing
10 display data to be displayed in the home page. Additionally, each of the data storage unit 57 and respective data storage units 58 to 60 described later is constituted, for example, of the large-capacity RAM, ROM, hard disk or optical magnetic disk.

15 The data storage unit 58 of the database 56 stores mail data transmitted/received via the electronic mail server 54. The data storage unit 59 is provided with respective data storage units (memories) for storing member data, received order data, and the like. The
20 respective servers 53 to 55 control reading/writing of the data. This structure and function will be described later in more detail. The data storage unit 60 stores other data.

FIGS. 18 to 21 show details of respective memories
25 of the data storage unit 59. These memories include a member data memory 71 (FIG. 18) for storing various data of pre-registered members, an image data memory 80

(FIG. 19) for pre-storing a large number of pieces of commodity (pizza) image data among the display data to be displayed in the home page as described above, a received order data memory 90 (FIG. 20) for storing content data when an order for the commodity is received via the network, and a member service image data memory 100 (FIG. 21) for storing that a discount service image has been supplied to a member having purchased the commodity.

10 In the member data memory 71 of FIG. 18, each row is a storage area for one member. Each row is constituted of: a member number storage area 72 for storing a member number which differs with each member; a member name storage area 73 for storing a member name; an address storage area 74; a phone number storage area 75; a mail address storage area 76 for storing an electronic mail address of the member; a financial institution storage area 77 for storing a name of a pre-contracted financial institution which settles payment of the ordered pizza; and a settlement data storage area 78 for storing settlement data by the financial institution, such as a credit card verification code number and identification number for payment from a bank account via the credit card.

25 In the image data memory 80 of FIG. 19, a large number of pieces of image data displayed in the virtual shop, that is, commodity (pizza) image data and

associated data of the display data displayed in the home page by the server apparatus 51 of the pizza shop 50. The memory is constituted of a large number of areas 81A, 81B, 81C ... for storing respective commodity image data (image data obtained by photographing the commodity with a digital still camera, for example) or arbitrarily prepared icon image data, and each row of the image content data storage area forms a storage area for one piece of image data. The image content data storage area includes: an image number storage area 82 for storing an image number which is set to differ with each image data; an address storage area 83 for storing storage address data A001, A002, A003, ... of storage areas 81A, 81B, 81C, ... in which the respective image data are stored; an identification data storage area 84 for identifying the respective image data as described later in detail; and a content data storage area 85 for storing content data of each commodity image data. Moreover, some of the respective stored image data are used in services such as commodity discount. Therefore, there are provided an object commodity data storage area 86 for storing data of the commodity to be discounted, a preparation date data storage area 87 indicating a date on which an image is prepared, and an effective period storage area 88 indicating an effective period of the service.

FIG. 20 shows a detailed constitution of

the received order data memory 90 for storing the data when the order for the commodity is received from the member via Internet 10. The memory includes: a date storage area 91 for storing data of the date on which
5 the order is received; a member number storage area 92 for storing a member number of the member having ordered the commodity; an ordered commodity storage area 93 for storing an image number of the ordered commodity as ordered commodity data; a number storage
10 area 94 and amount storage area 95 for storing the number of ordered commodities and a total amount (unit price \times number), respectively; and a payment method storage area 96 for storing data of a payment method for settling payment of the amount stored in the amount
15 storage area 95.

Moreover, when the commodity is purchased, a service image such as a discount for each commodity is transmitted to the member having purchased the commodity. Therefore, there is also provided a service
20 image storage area 97 for storing an image number data of the service image to be transmitted.

FIG. 21 shows details of the member service image data memory 100 for managing and storing the transmitted service image data for each member. A storage area only for one member is shown in FIG. 21.
25 In the member service image data memory 100, a member number storage area 101, member name storage area 102,

member address storage area 103 and phone number
storage area 104 are disposed for one member.
Moreover, data storage areas regarding a plurality of
service images transmitted to the member are also
5 disposed for the member. For each image data, the area
includes an image number storage area 105 of the
transmitted image data, an identification data storage
area 106 for storing image identification data, a
transmission date storage area 107 for storing a date
10 on which the image is transmitted, and a use date
storage area 108 for storing a date on which the image
data for receiving the transmitted service is used to
receive the service.

Although FIG. 21 shows the area only for one
15 member, the member service image data memory 100
includes the aforementioned storage areas for all the
members to which the service images have been
transmitted.

An operation in the second embodiment constituted
20 as described above will next be described.

First, member contracts for accepting pizza orders
from the personal computers 1, 2, ... N via the network
10 are made beforehand between the clients and the
pizza shop 50. Personal information for specifying the
member, settlement data such as financial institution
25 data for settlement and verification code or
identification number data for use in the settlement,

and the like are stored beforehand in the member database memory 71 of FIG. 18.

FIG. 22 shows a flowchart of an image setting/storage processing in which display commodity image data for use in display of the home page (virtual electronic shop) transmitted/received via the Web server 53 are set/stored beforehand in the image database memory 80. For example, a program for executing this processing is stored beforehand in the ROM 62 (FIG. 17) of the managing/processing server 55 or the Web server 53, and this processing flow is executed.

Therefore, when an image setting start button (not shown) of the input unit 67 of FIG. 17 is operated, the display 66 forms an image setting screen as shown in FIG. 23. That is, as described later, displayed are a display area 109A of an image read via the input/output unit 68, a display area 109B of image number data inputted via the input unit 67, an identification data display area 109C, a content data display area 109D, an object commodity display area 109E, and an effective period display area 109F.

In this state, first, in the image setting storage processing flow of FIG. 22, in step S20 an image reading processing is executed. In this processing, for example, the image data of the commodity photographed with the electronic still camera (not

shown) is read via the input/output unit 68 of FIG. 17, stored in the work area of the storage unit 63, and further displayed in the image display area 109A of the display 66. FIG. 23 shows that an image of a
5 one-eighth piece of the entire pizza is photographed with the camera, read, and displayed. In the next step S21, an image number is inputted. For the image number input, the number different from the number of the image number data already stored in the image number
10 storage area 82 of the image database 80 is inputted via the input unit 67, and this inputted data is displayed in the image number data display area 109B.

In the next step S22, identification data is inputted. By this identification data, one image is
15 distinguished from the other image. For example, upper four digits indicate character/numeric value data (ID21, ID22, ID23, ID51 ...) which differ with each read image, and specify a difference of the read image. Additionally, lower four digits are all inputted as
20 0000 for a reason described later in detail. The inputted identification data is displayed in the display area 109C, and the flow advances to step S23. In the step S23, the content data, object commodity data, and effective period data are inputted, and the
25 inputted data are displayed in the display areas 109D, 109E and 109F.

When the data input ends in this manner, the data

is OK, and a setting button (not shown) of the input unit 67 is operated (step S24), the flow advances to step S25. In the step S25, the respective inputted data are stored in areas of the image data memory 80 in which data are not stored yet. That is, the read image data is stored, for example, in the image data storage area 81B (it is assumed that no image data has been stored here). Moreover, the inputted image number "0002", identification data "ID220000", content data "10% OFF", object commodity "pizza", and effective period "2002.1.30" are stored in a second row as shown in FIG. 19 (it is also assumed that no image data has been stored here). Furthermore, preparation date "2000.5.21 (May 21, 2000)" is stored in response to a present date signal from a clock circuit (not shown).

In the next step S26, memory address data "A002" of the storage area 81B of the extracted image data is stored in the address storage area 83 (second row), and this image setting storage flow ends.

The image data for use in the home page is successively stored in the image data memory 80 in this manner. For example, an image P0 taken into the storage area 81A is image data of one entire pizza shown in FIG. 24. The content data of this image data is stored in an image number "0001" row (first row), image data P1 of the one-eighth piece of the entire pizza described above and shown in FIG. 25 is stored in

the storage area 81B (content data is stored in a second row), and image data P2 of a one-fourth piece of the entire pizza shown in FIG. 26 is stored in the storage area 81C (content data is stored in a third row). Moreover, the processing of reading the image data in the step S20 may be other than reading the image data via the input/output unit 68. For example, the image pre-stored in the storage unit 63, or a bit map image prepared by oneself may be read. For example, in the storage area 81D, instead of the image data read via the input/output unit 68, an icon image P3 indicating the money amount as shown in FIG. 6 is transferred from another memory and stored (content data is in a fourth row).

The image data P0 of the whole pizza of FIG. 24 is displayed (exhibited) as an exhibition image of the commodity for sale in the virtual shop, but the image data P1 of the one-eighth piece of pizza, image data P2 of the one-fourth piece of pizza and icon image P3 including the money amount is displayed as a service image which can be used as a discount ticket for discounting a charge, exchange ticket to be exchanged for the commodity, coupon ticket, point ticket, or another service ticket. Moreover, since 10% discount only for the next purchase of pizza is possible with the image P1 of the one-eighth piece of pizza, "10% OFF" is set/stored in the content data storage area 85.

Moreover, since a discount object is only the next
pizza purchase, "pizza" is set/stored in the object
commodity storage area 86. Moreover, the image P2 of
the one-fourth piece of pizza is the same as the image
5 of the one-eighth piece except a discount ratio.

On the other hand, with the icon image P3 of the
storage area 81D, 100-yen discount is possible with all
commodities other than the pizza (it is assumed that
the commodities other than the pizza are also
10 exhibited, waiting for orders in the virtual shop), and
"all commodities" is therefore set as the object
commodity.

FIG. 27 shows an example of a displayed pizza
image which is prepared using the image data stored in
15 the image data memory 80, and displayed as the
commodity for sale in the virtual shop of the pizza
shop 50 on Internet. In FIG. 27, the image P0 of the
whole pizza shown in FIG. 24 and stored in the storage
area 81A of the image data memory 80 is displayed in
20 the display area 110 to introduce the commodity. In
the display area 111, the image P1 shown in FIG. 25 and
stored in the storage area 81B and characters
indicating the content of the image are similarly
displayed. Moreover, in the display area 112, the
25 image P2 shown in FIG. 26 and stored in the storage
area 81C and the characters indicating the content of
the image are similarly used to prepare and display

the discount image. Therefore, the client contracted
as the member browses the home page of the virtual shop
in his own information terminal 1, 2, ... or N, and
operates a click button 113 with "Click here" displayed
5 thereon as shown in FIG. 27 (the mouse is used to align
the cursor with the button and click the button).
Thus, the name of the commodity to purchase and
personal data of the member himself are transmitted in
the similar manner as shown in FIG. 13. In FIG. 13,
10 there is no member number input area. But in this
case, an input area of the member number or the
identification number indicating that the client is the
member may be disposed in addition to those shown in
FIG. 13, and such data is transmitted. Furthermore,
15 when a service image is already obtained, the image is
attached to the data and the data is transmitted.

FIG. 28 shows a flowchart of an order reception
and response processing in which the client transmits
the order for the pizza to the virtual shop as
20 described above, and the server apparatus 51 receives
the order and transmits a response to the client.
In the processing, the Web server 53 informs the
managing/processing server 55 of reception of the
order, and the managing/processing server 55 processes
25 the received order according to program software stored
in the ROM 62. Alternatively, the program software for
executing the flow of processing may be stored in

the ROM 62 of the Web server 53, and the Web server 53 executes the processing.

It is judged in step S30 whether or not received data is pizza order data. With the order data, it is
5 judged in the next step S31 whether or not the order is transmitted from the already registered member, by comparing the transmitted member data with the data stored in the member data memory 61. When the order is transmitted from the member, the flow advances to an
10 order reception processing of step S33. When the data is not transmitted from the member, or when it is judged in the step S30 that the order data is not normal, the flow advances to step S32, in which the client having transmitted the data is notified of an
15 error.

Detailed steps of the order reception processing of the step S33 are shown in FIG. 29. First in step S40, the respective transmitted data are stored in the received order data memory 90. That is, received order
20 date, client member number data, ordered commodity data, quantity, amount, and the like are stored in the corresponding storage areas of the received order data memory 90. Moreover, for the payment method, at this point of time, a method predetermined by the contract is stored. Furthermore, it is judged whether or not
25 there is service image data to be distributed for the ordered commodity (this is judged because the storage

unit stores beforehand that the service image of 10% OFF is given to the order for two 3,000-yen pizzas as shown in FIG. 27). When there is the service image data to be distributed, the image number of the service image data is stored in the service image storage area 97. For example, when the client orders two tomato pizzas shown in FIG. 27, the client is given one service image P1 of 10% discount. Therefore, when the order for two tomato pizzas shown in FIG. 27 is transmitted from the client, it is judged that the service is applied, and the image number data 0002 of the service image P1 of 10% discount is automatically stored in the service image storage area 97.

It is then judged in step S41 whether or not the service image data is attached to the received data. When the data is attached, the identification data of the image data is stored in the payment method storage area 91 of the received order data memory 90, and the flow advances to the next step S42. That is, as described later, the identification data is attached to the image to be transmitted as the service image, and the image is transmitted. Therefore, if the received service image indicates the transmitted service image data itself, the identification data must exist in the image data. Therefore, the identification data is extracted from the image data, and the identification data is also stored in the payment method storage

area 91.

It is then judged in the next step S42 whether or not the stored identification data is authentic, that is, whether or not the data is the service image data transmitted to the client having transmitted the present order during the previous order. That is, the member service image data memory 100 stores the image number of the previously transmitted image data, and the identification data attached to the image number data by the electronic watermark technique for each client as the member. It is therefore judged whether the presently received and stored identification data agrees with the identification data of the image data stored in the member service image data memory 100 and previously transmitted to the member.

When the identification data agree with each other, it is verified that the service image data presently transmitted from the client is the service image data transmitted to the same client during the previous order. If the data do not agree with each other, the flow advances to an error processing of the step S43. If it is verified that the data is authentic, however, the flow advances to step S44. In this step S44, on the assumption that the image data is normal, and used in payment or settlement for the pizza order, date data of that day is written in the use date storage area 108 corresponding to the storage

area of the member service image data memory 100 in which the same identification data is stored, and the flow advances to the next step S45.

For the step S45, when the flow advances to the
5 step S45 from the step S44, it is judged whether or not a balance payment amount excluding a service amount indicated by the service image can be drawn from the contracted financial institution. When the flow advances to the step S45 from the step S41, the total
10 amount of the ordered commodity has been drawn from the financial institution. That is, it is judged whether the total payment is OK with the service image or by the settlement of the financial institution. If the payment is OK, the flow advances to the step S46.
15 If the payment is not OK, the flow advances to the error processing of the step S43. In the error processing of the step S43, the reason for the error is notified to the client when the service image transmitted in the step S42 is judged not to be normal,
20 or when the total payment is judged not to be OK in the step S45.

After judging that there is no problem in payment for the ordered pizza in the step S45, it is judged in
25 step S46 whether or not there is service image data to be transmitted to the client having transmitted the order. This is judged by judging whether there is image number data in the service image storage area 97

of the received order data memory 90. That is, in the step S40 of storing the received data, when the order for the pizza is transmitted together with the service image, the image number of the service image is stored in the service image storage area 97. Therefore, it is judged here whether or not the image number data is stored.

Subsequently, when it is judged that there is the service image to be transmitted, the flow advances to step S47. In the step S47, first the image number stored in the service image storage area 97 is stored in the image number storage area 105 of the member having transmitted the order in the member service image data memory 100. For example, in FIG. 20, for an order for commodity "0001" by a client with member number "12...01", the service image P1 with image number "0002" is transmitted. In this case, first the image number "0002" is stored in the image number storage area 105 of the member having transmitted the order (member number "12...01") in the member service image data memory 100.

Subsequently, the lower four digits of the identification number data of the service image P1 to be transmitted are changed, and the data is stored in the identification data storage area 106. That is, identification data "ID220000" is applied to the image number "0002" as stored in the identification data

storage area 84 of the image data memory 80 of FIG. 19. Moreover, the data is distinguished from the other image data by the upper four digits "ID22" of the data.

Therefore, to store the identification data of the image number "0002" in the identification data storage area 106 of the member service image data memory 100, the upper four digits "ID22" are used as they are. Even with the same image number "0002", for the lower four digits, the same number is avoided, and the digits are changed during numbering for each transmission to the client. That is, when the image number "0002" is transmitted as the service image data for a first order, the identification data "ID220001" is transmitted. When the service image data is transmitted for the next order, the lower four digits are successively changed, and the identification data "ID220002" is transmitted. In this case, even with the image data of the same image number "0002", 9999 service images can be transmitted by changing the identification data. After the identification data is stored in the identification data storage area 106 in this manner, the date on which the data is transmitted is stored in the sending date storage area 107, and the order reception processing thereby ends.

After the order reception processing of FIG. 29 ends, the flow advances to an image data transmission processing of step S34 of FIG. 28. In the image data

transmission processing, the service image is actually transmitted to the client having transmitted the order. The image data P1 with the image number stored in the image number storage area 105 of the member service
5 image data memory 100, for example, with the image number "0002" as described above is read from the image data storage area 81B of the image data memory 80. The identification data stored in the identification data storage area 106 in accordance with the image number,
10 for example, the identification data "ID220087" as described above is attached to the read image data, for example, in a watermark state using the electronic watermark technique. Alternatively, to further enhance security, the identification data "ID220087" is
15 subjected to a special encryption processing, encrypted and attached to the image data in the electronic watermark state. The image data is then transmitted to the client having transmitted the order (the client with the member number 12...01 in the above example)
20 via electronic mail.

In the next step S35, a commodity sending processing for cooking and sending the ordered commodity, that is, the tomato pizza shown in FIG. 27 is performed, and the order reception and response
25 processing thereby ends.

As described above, in the second embodiment, the image data displayed on the network can be treated as

the service image which can be used as the discount
ticket for discounting a selling price, exchange ticket
for use to be exchanged for the commodity, coupon
ticket used as one of a plurality of coupon tickets to
5 be collected for obtaining a special privilege, point
ticket or another service ticket.

Additionally, similarly as the first embodiment,
the service image of the second embodiment can be
utilized as a prize for the winner of the quiz or the
10 like, a free present, or a reward for the applicant for
the commodity monitor. Moreover, the service image
data can be stored in recording mediums such as a
floppy disk, and a semiconductor (IC) memory, and
mailed to the client, or handed over to the client at a
15 service window or the like.

The aforementioned first and second embodiments of
the present invention can variously be applied or
modified. For example, the virtual shop is opened in
the home page of the sales company. However, when a
20 specific corporation (such as a service company)
collects commodities of many corporations and opens the
virtual shop, the present invention can be applied
here.

Moreover, instead of the personal computers 1,
25 2, ..., N, the client information terminal may be, for
example, a portable handy phone or a portable personal
digital assistance (PDA).

As described above, in the present invention, the image data is transmitted to the purchaser client beforehand. When the purchaser client transmits the image data including the money amount, it can be
5 verified whether or not the image data is normal or authentic. Therefore, the purchaser client can use the image data to perform settlement for commodity purchase on the network, and the purchaser client can use the image data to buy the arbitrary commodity on the
10 network. Alternatively, on a selling side which has transmitted the image data to the purchaser client, it is verified whether or not the same data as the transmitted image data has been received before the settlement, and the image data can be prevented from
15 being mistaken or abused.

C L A I M S

1. A settlement system comprising:

transmission means for transmitting image data including a money amount to a purchaser client via a network;

reception means for receiving the image data including the money amount from the purchaser client via the network;

verification means for checking whether or not the image data including the money amount received from said purchaser client via the reception means is that transmitted image data including the money amount; and

settlement means for allowing said purchaser client to settle an account in a range of said money amount via the image data including the money amount when the verification means verifies that the image data is that transmitted image data including the money amount.

2. The settlement system according to claim 1, wherein said transmission means transmits said image data including the money amount to a winner client among a large number of clients having applied for participation in a questionnaire or a quiz or as a commodity monitor via said network.

3. The settlement system according to claim 1, wherein said transmission means attaches predetermined encryption data to said image data and transmits the

image data, and said verification means verifies said image data including the money amount by judging whether or not said predetermined encryption data is attached to the image data including the money amount
5 received from said purchaser client via said reception means.

4. The settlement system according to claim 3, wherein the encryption data is attached by electronic watermark means.

10 5. The settlement system according to claim 1, wherein said transmission means and said reception means comprise electronic mail means.

6. A settlement system in which a virtual shop is opened on a network and a purchaser client purchases a commodity and settles a purchase in said virtual shop,
15 said settlement system comprising:

image data issuance means for issuing image data including a money amount to the purchaser client;

reception means for receiving the image data including the money amount from the purchaser client
20 via the network;

verification means for checking whether or not the image data including the money amount received from said purchaser client via the reception means is that
25 issued image data including the money amount; and

settlement means for allowing said purchaser client to settle an account in a range of said money

amount via the image data including the money amount when the verification means verifies that the image data is that issued image data including the money amount.

5 7. The settlement system according to claim 6, wherein said image data issuance means stores said image data including the money amount in a storage medium and issues the image data.

10 8. The settlement system according to claim 6, wherein said issuance means attaches predetermined encryption data to said image data and issues the image data, and said verification means verifies said image data including the money amount by judging whether or not said predetermined encryption data is attached to
15 the image data including the money amount received from said purchaser client via said reception means.

 9. The settlement system according to claim 8, wherein the encryption data is attached by electronic watermark means.

20 10. The settlement system according to claim 6, wherein said issuance means and said reception means comprise electronic mail means.

 11. A settlement system comprising:
 image data issuance means for issuing image data
25 indicating a money amount and including encryption data as a winner prize or a reward prize to a client having made an application via a network;

reception means for receiving the image data indicating the money amount from the client via the network;

5 verification means for checking whether or not the image data indicating the money amount received from said client via the reception means includes said encryption data; and

10 settlement means for settling an account in a range of said money amount of said image data when the verification means verifies that said encryption data is included.

12. The settlement system according to claim 11, wherein said issuance means attaches predetermined encryption data to said image data and issues the image data, and said verification means verifies said image data including the money amount by judging whether or not said predetermined encryption data is attached to the image data including the money amount received from said purchaser client via said reception means.

20 13. The settlement system according to claim 12, wherein the encryption data is attached by electronic watermark means.

14. The settlement system according to claim 11, wherein said issuance means and said reception means 25 comprise electronic mail means.

15. A server apparatus comprising:

first image data storage means for storing image

data including a money amount to be transmitted to a purchaser client via transmission means on a network;

second image data storage means for storing image data including a money amount received from the purchaser client via the network;

verification means for checking whether or not the image data including the money amount stored in the second image data storage means is the image data stored in said first image data storage means and transmitted to said purchaser client; and

settlement means for allowing said purchaser client to settle an account in a range of said money amount by the image data including the money amount when the verification means verifies that the image data is that transmitted image data including the money amount.

16. A settlement system comprising:

transmission means for transmitting an image data to a purchaser client via a network;

reception means for receiving the image data from the purchaser client via the network; and

verification means for checking whether or not said image data received from said purchaser client via the reception means is the image data transmitted via said transmission means,

wherein a predetermined service is presented to said purchaser client when said verification means

verifies that the image data is that transmitted image data.

17. The settlement system according to claim 16, wherein said transmission means transmits said image data to the purchaser client having purchased a commodity in a virtual shop disposed on said network.

18. The settlement system according to claim 16, wherein said transmission means attaches predetermined encryption data to said image data and transmits the image data, and said verification means verifies the image data by judging whether or not said predetermined encryption data is attached to the image data including a money amount received from said purchaser client via said reception means.

19. The settlement system according to claim 18, wherein the encryption data is attached by electronic watermark means.

20. The settlement system according to claim 16, wherein said transmission means and said reception means comprise electronic mail means.

21. A settlement system in which a virtual shop is opened on a network and a purchaser client purchases a commodity in said virtual shop and settles the purchase, said settlement system comprising:

25 exhibition means for exhibiting an image data clearly indicating that a predetermined service can be received in said virtual shop;

image data issuance means for issuing said image data to said purchaser client when the purchaser client purchases the commodity in said virtual shop;

reception means for receiving the image data from
5 the purchaser client via the network; and

verification means for checking whether or not the image data received from said purchaser client via the reception means is that issued image data,

wherein said purchaser client can receive said
10 predetermined service when said verification means verifies that the image data is that issued image data.

22. A server apparatus comprising:

image data storage means for storing image data to
be transmitted to a purchaser client via transmission
15 means on a network;

reception means for receiving image data from said purchaser client via the network;

verification means for checking whether or not the image data received via the reception means is the
20 image data stored in said image data storage means and transmitted to said purchaser client; and

settlement means for presenting a predetermined service to said purchaser client during settlement when the verification means verifies that the image data is
25 that transmitted image data.

23. The server apparatus according to claim 22, wherein said transmission means transmits said image

data to the purchaser client having purchased a commodity in a virtual shop disposed on said network.

24. The server apparatus according to claim 22, wherein said transmission means attaches predetermined encryption data to said image data and transmits the image data, and said verification means verifies the image data by judging whether or not said predetermined encryption data is attached to the image data including a money amount received from said purchaser client via said reception means.

25. The server apparatus according to claim 24, wherein the encryption data is attached by electronic watermark means.

26. The server apparatus according to claim 22, wherein said transmission means and said reception means comprise electronic mail means.

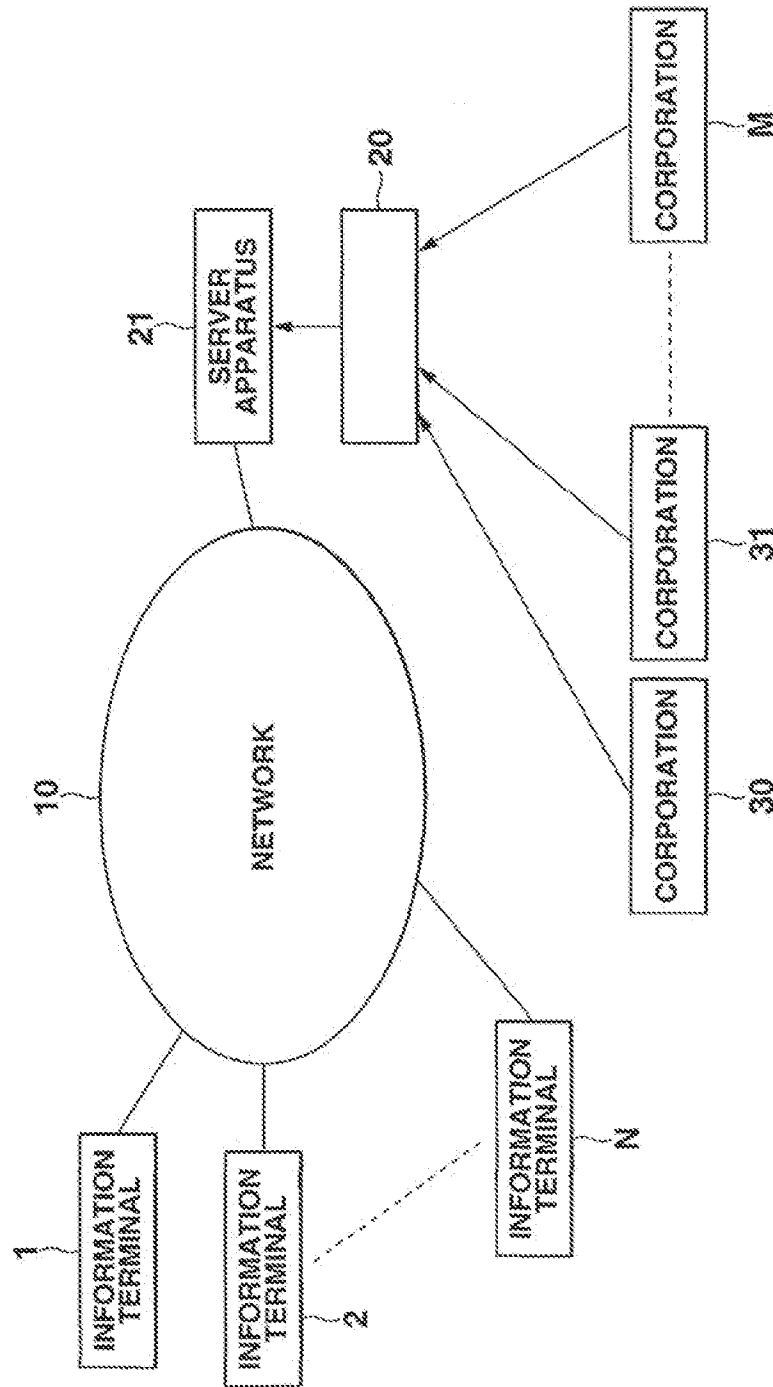


FIG.1

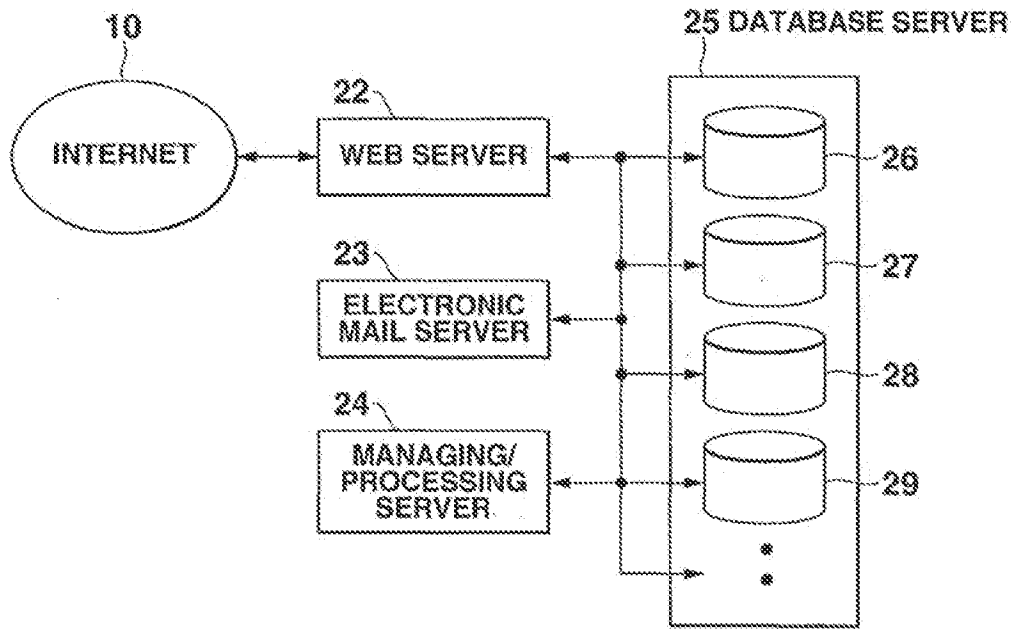


FIG.2

4/26

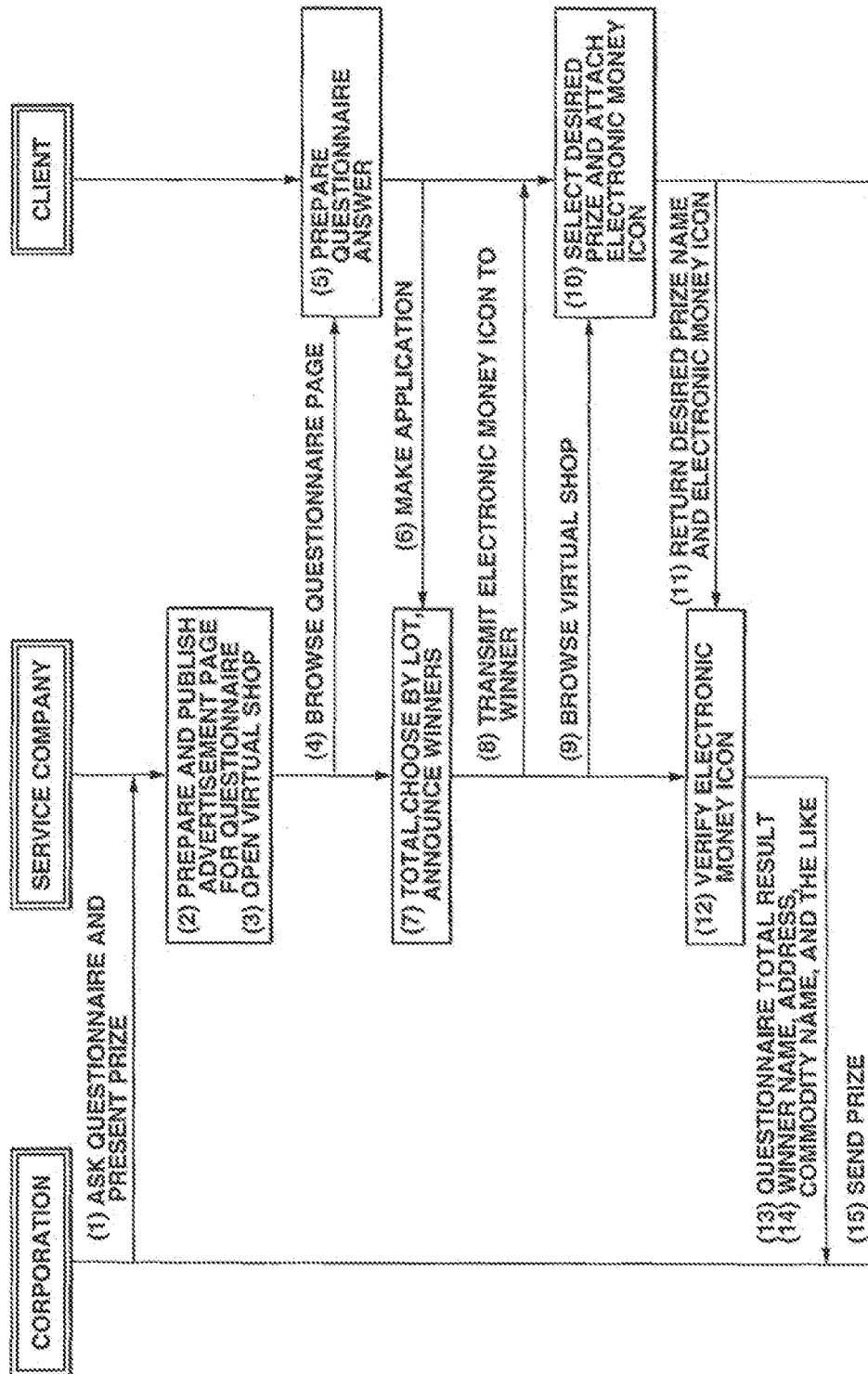


FIG.4

5/26

PLEASE ANSWER THE FOLLOWING QUESTIONNAIRE.
AN ELECTRONIC COMMODITY TICKET FOR 10,000 YEN IS GIVEN TO
50 WINNERS BY LOT. THE COMMODITY TICKET CAN BE USED ONLY
IN A VIRTUAL SHOPPING MALL OF THIS HOME PAGE.
ADDITIONALLY, INSTEAD OF ANNOUNCING THE WINNER, THE
ELECTRONIC COMMODITY TICKET IS SENT TO THE WINNER VIA
ELECTRONIC MAIL.

QUESTION 1) WHAT IS YOUR FAVORITE ... ?
IT IS

QUESTION 2) WHAT IS YOUR ... ?
IT IS

QUESTION 3) ...
:
:
:

ADDRESS

NAME AGE

MAIL ADDRESS : SEX

AFTER ANSWERING ALL THE QUESTIONS,
PLEASE CLICK A TRANSMIT BUTTON.

FIG.5

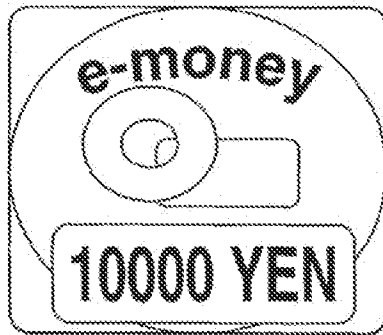


FIG. 6

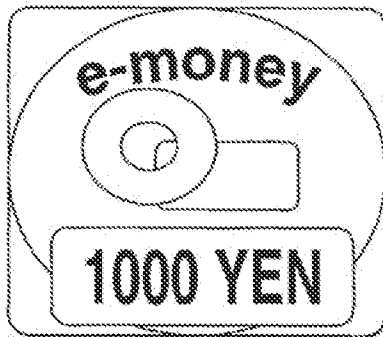


FIG. 7

7/26

USER'S FAVORITE	27F		27G	27B	27A	F0	27C	AC INDUSTRY Co., Ltd.
NAME B) ADDRESS	AGE	MAN	kuro@...	JANUARY 1 TILL MARCH 31, 2000	27D	F1	27E	
10000 YEN	ID10202485	2000.4.15	2000.10.15	1	2000.05.30	27J		1
2500 YEN	ID25860121	2000.05.31	2000.11.30	0	0			0
NAME K) ADDRESS	AGE	MAN	koba@...					
50000 YEN	ID52350202	2000.4.15	2000.10.15	0	0			0

FIG.8

8/26

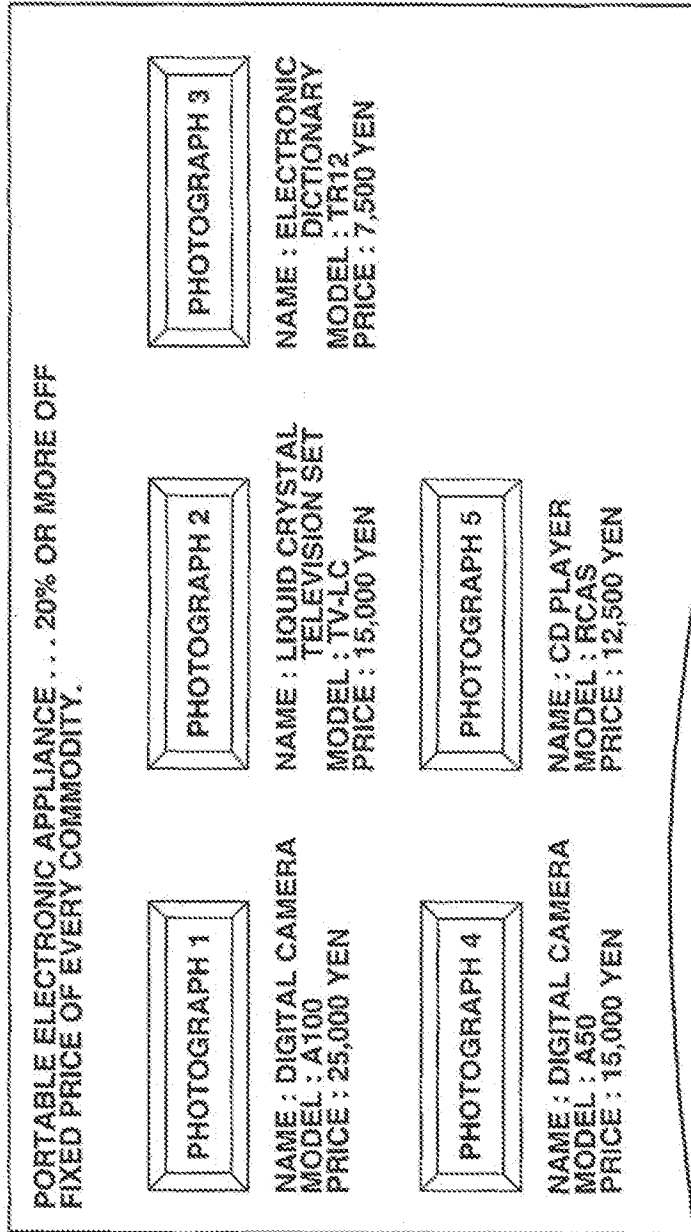


FIG.9

9/26

PHOTOGRAPH IMAGE 1	DIGITAL CAMERA		PHOTOGRAPH IMAGE 2	LIQUID CRYSTAL TELEVISION SET		
	A100	25,000 YEN		TV-LC	15,000 YEN	
PHOTOGRAPH IMAGE 3	25	S CAMERA Co., Ltd.	.	10	AA DENKI Co.	
	ELECTRONIC DICTIONARY			.		
	TR12	7,500 YEN		.		
	85	R INDUSTRY Co., Ltd.		.		
.	
.	
.	

28A

28B

FIG.10

10/26

ORDER TABLE . . . YOU CAN ORDER AS MANY AS FIVE COMMODITIES.

	DATE	NAME	MODEL No.	PRICE
1				
2				
3				
4				
5				

PAYMENT METHOD (PLEASE CLICK APPROPRIATE PLACE)

- REGISTERED MAIL
- PAYMENT TO BANK
- CREDIT CARD
- ELECTRONIC MONEY ICON
- .
- .

FIG.11

11/26

PAYMENT METHOD BY ELECTRONIC MONEY ICON

- YOU CAN USE AN ELECTRONIC MONEY ICON ONLY ONCE. SECOND AND SUBSEQUENT TRANSMISSIONS BECOME INVALID.
- IF THE OTHER PERSON USES THE ICON WITHOUT PERMISSION, ONLY FIRST USE IS EFFECTIVE. PLEASE HOLD THE ICON CAREFULLY TO AVOID MISAPPROPRIATION.
- THE ELECTRONIC MONEY ICON HAS ITS USE PERIOD. AFTER THE USE PERIOD, THE ICON BECOMES INVALID. PAY ATTENTION TO THE USE PERIOD.
- YOU CANNOT PURCHASE THE COMMODITY WITH A PRICE HIGHER THAN A MONEY AMOUNT INDICATED BY THE ELECTRONIC MONEY ICON.
- WHEN THERE IS A DIFFERENCE OF 100 YEN OR MORE BETWEEN THE MONEY AMOUNT OF YOUR SENT ELECTRONIC MONEY ICON AND THE PRICE OF YOUR PURCHASED COMMODITY, THE ELECTRONIC MONEY ICON INDICATING THE DIFFERENCE OF THE MONEY AMOUNT WILL BE SENT.

PROCEDURE

- (1) PLEASE CLICK THE FOLLOWING ELECTRONIC MAIL ADDRESS.
- (2) YOUR ORDERED CONTENT AND TOTAL MONEY AMOUNT WILL BE DISPLAYED.
- (3) PLEASE INPUT NECESSARY ITEMS.
- (4) PLEASE ATTACH AN ELECTRONIC MAIL ICON FOR USE.
- (5) PLEASE CLICK THE TRANSMIT BUTTON.

ADDRESS ankelito@xyz.co.jp

FIG.12

12/26

ADDRESS :	ankeito@xyz.co.jp		
CASE NAME :	COMMODITY PURCHASE		
DATE	COMMODITY	MODEL No.	PRICE
MARCH 15	ELECTRONIC DICTIONARY	TR12	7,500 YEN
TOTAL AMOUNT			7,500 YEN
YOUR ADDRESS	<input type="text"/>		
YOUR NAME	<input type="text"/>		
YOUR MAIL ADDRESS	<input type="text"/>		
PLEASE ATTACH THE ELECTRONIC MAIL ICON TO THE FOLLOWING AND CLICK THE TRANSMIT BUTTON.			
			<input type="button" value="TRANSMIT"/>

FIG.13

13/26

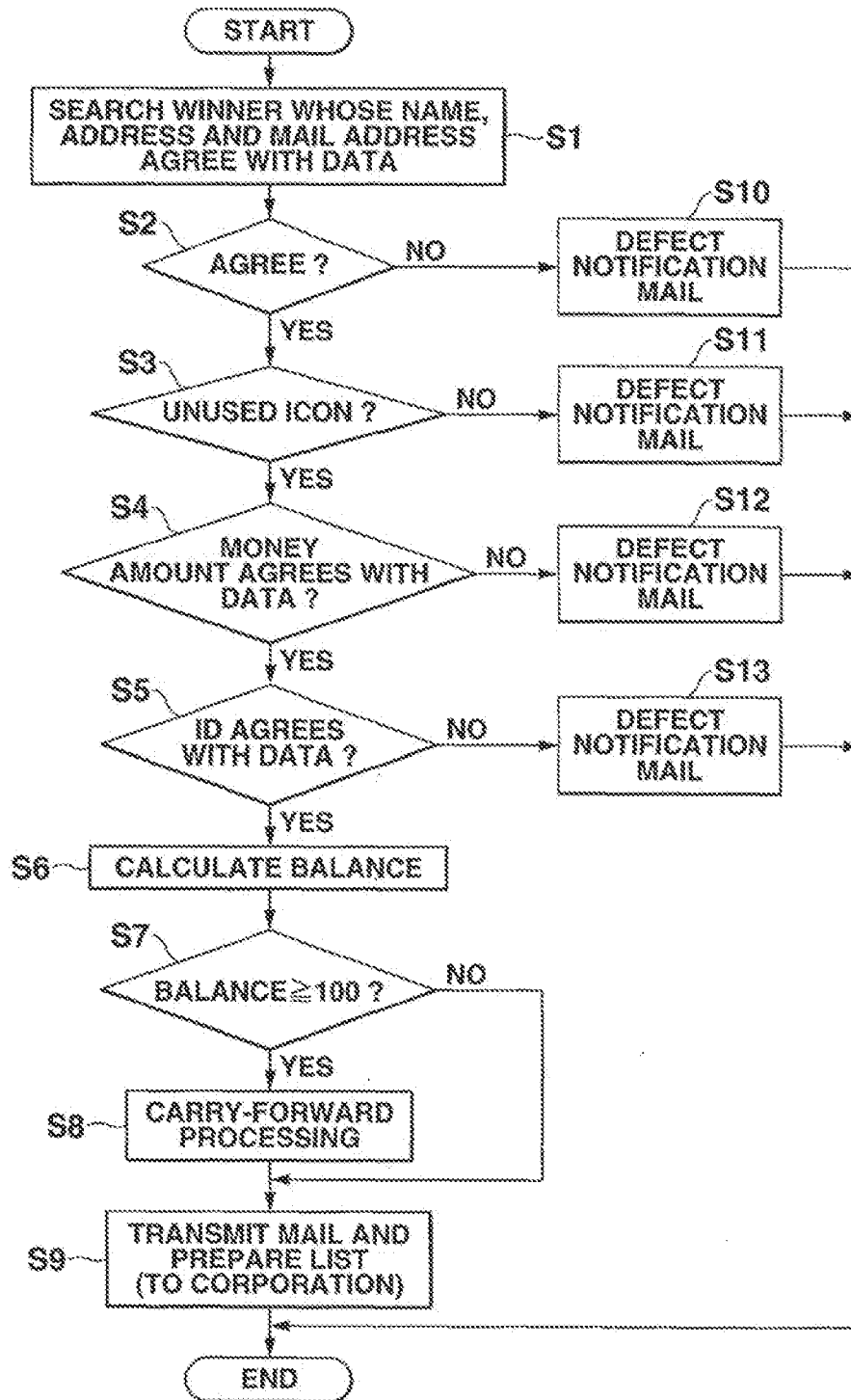


FIG.14

14/26

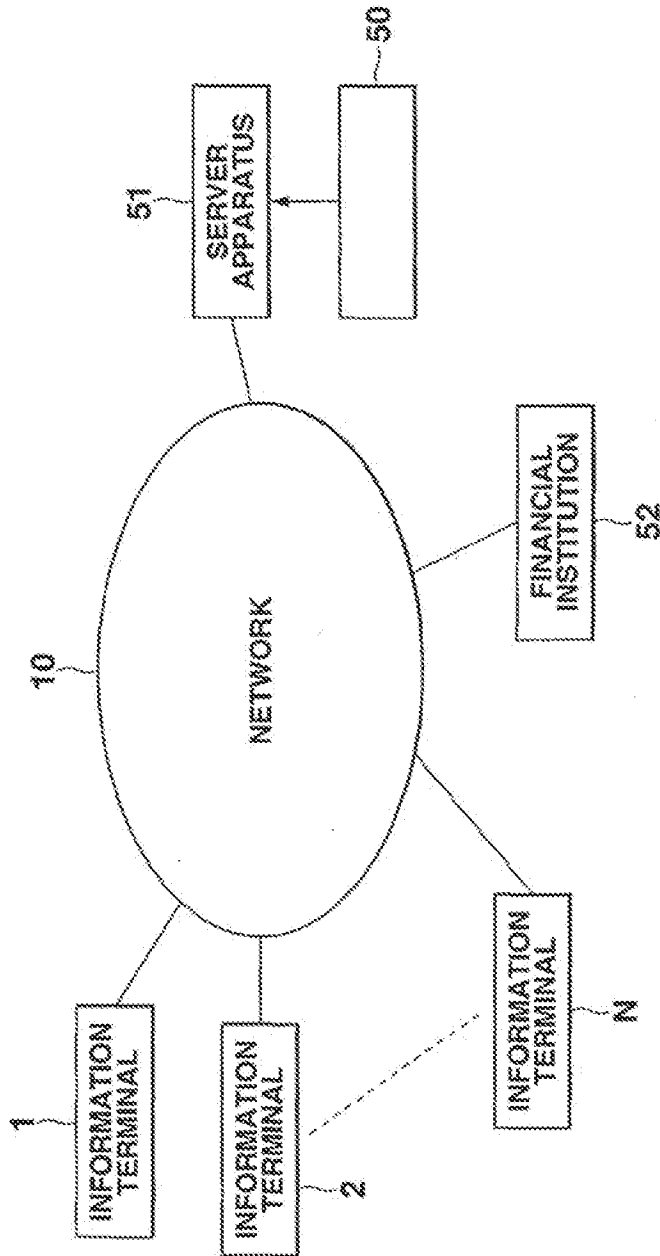


FIG.15

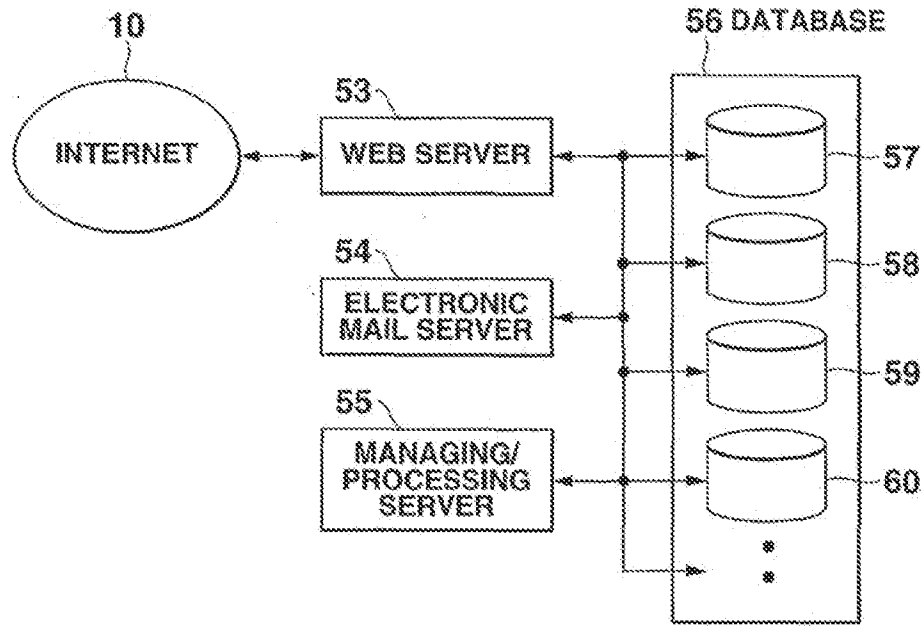


FIG.16

16/26

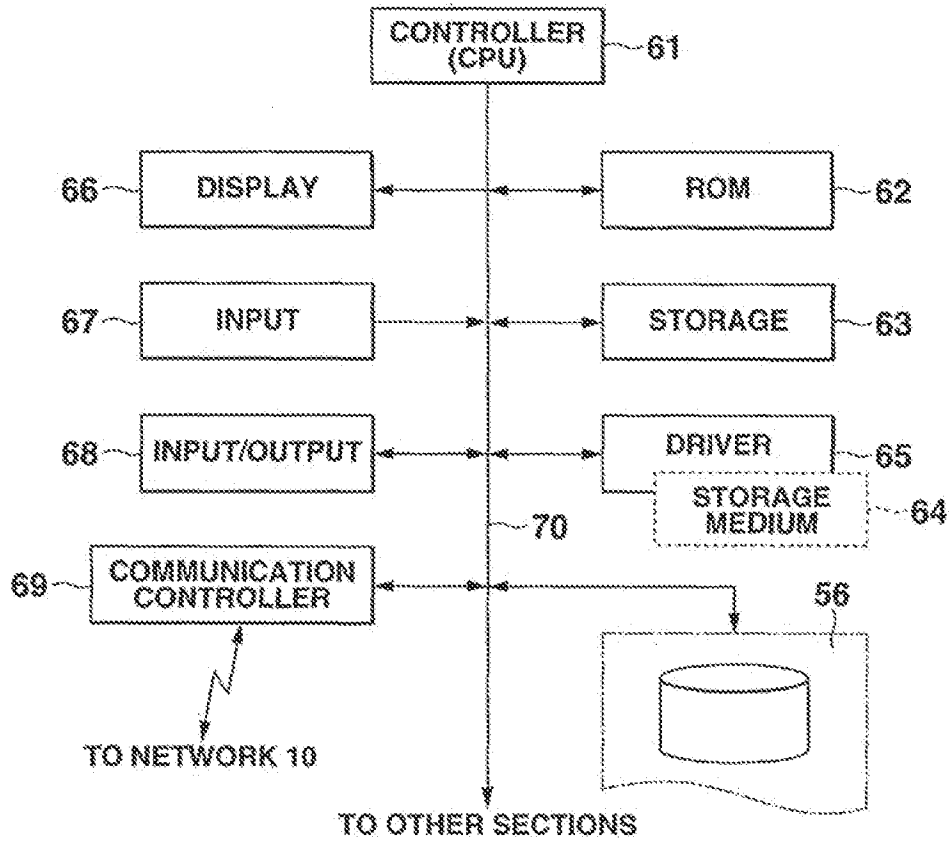


FIG.17

19/26

90 RECEIVED ORDER DATA MEMORY

DATE	MEMBER No.	ORDERED COMMODITY	NUMBER	AMOUNT	PAYMENT METHOD	SERVICE IMAGE
2000.8.20	12...01	0001	2	6000 YEN	CREDIT CARD	0002
2000.8.21	12...02	0008	1	2000 YEN	ID340123, CREDIT CARD	-

FIG.20

100 MEMBER SERVICE IMAGE DATA MEMORY

101	102	103	104
MEMBER No.	NAME	ADDRESS	PHONE No.
12 . . . 01	NAME A CITY, TOKYO	042 . . .
IMAGE No.	IDENTIFICATION DATA	SENDED DATE	USED DATE
0002	ID220087	2000.8.20	-
0005	ID310156	2000.8.21	-

FIG.21

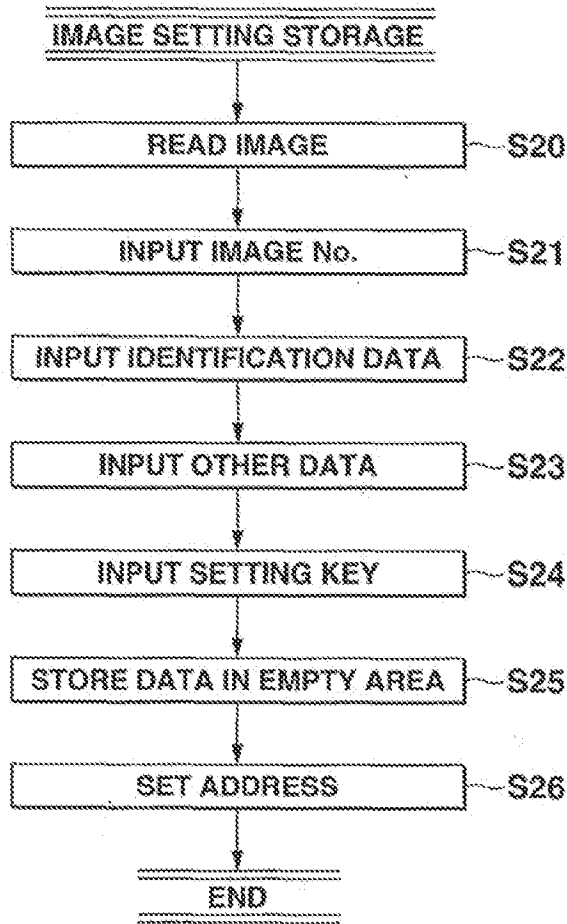


FIG.22

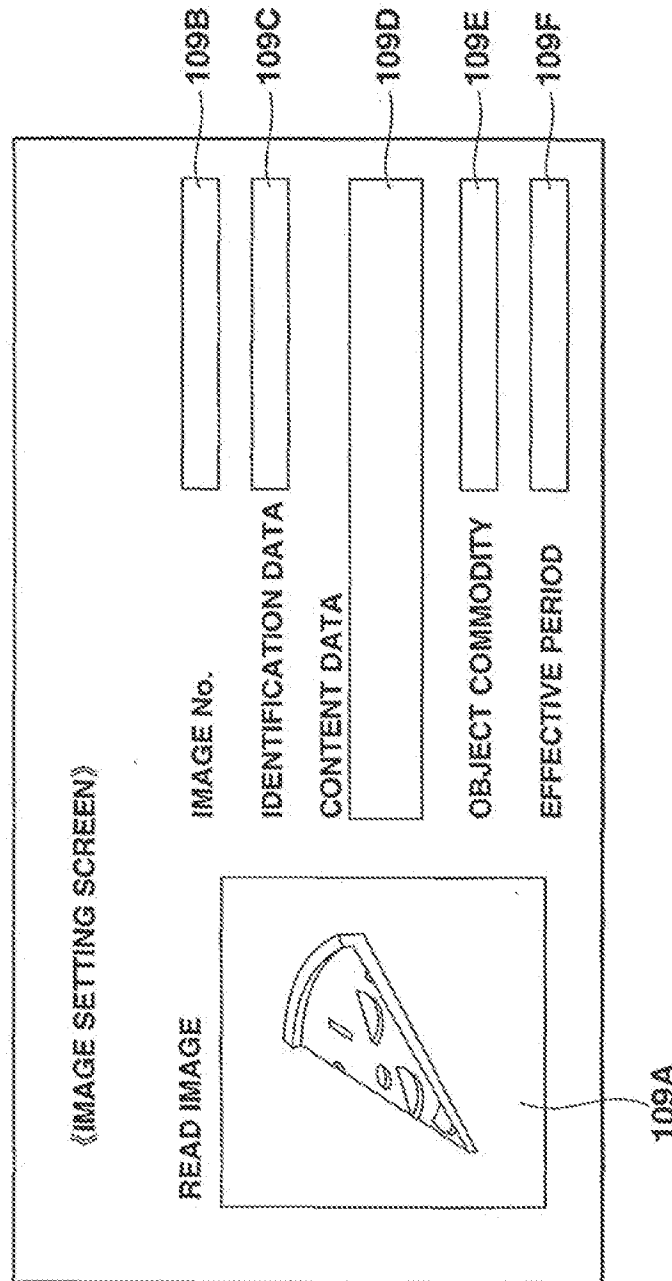


FIG.23

23/26

P0

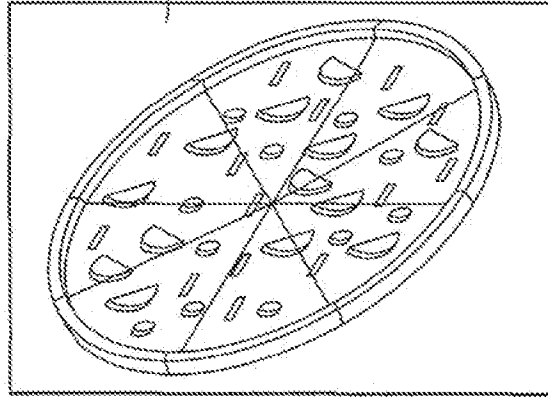


FIG. 24

P1

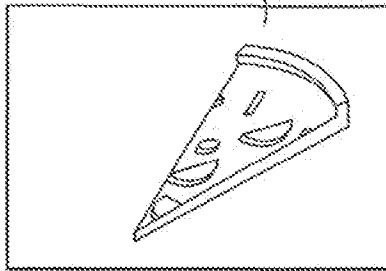


FIG. 25

P2

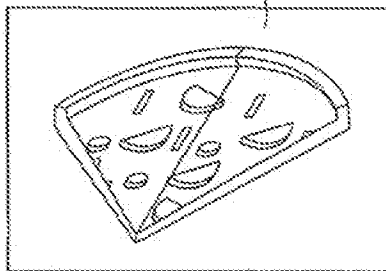


FIG. 26

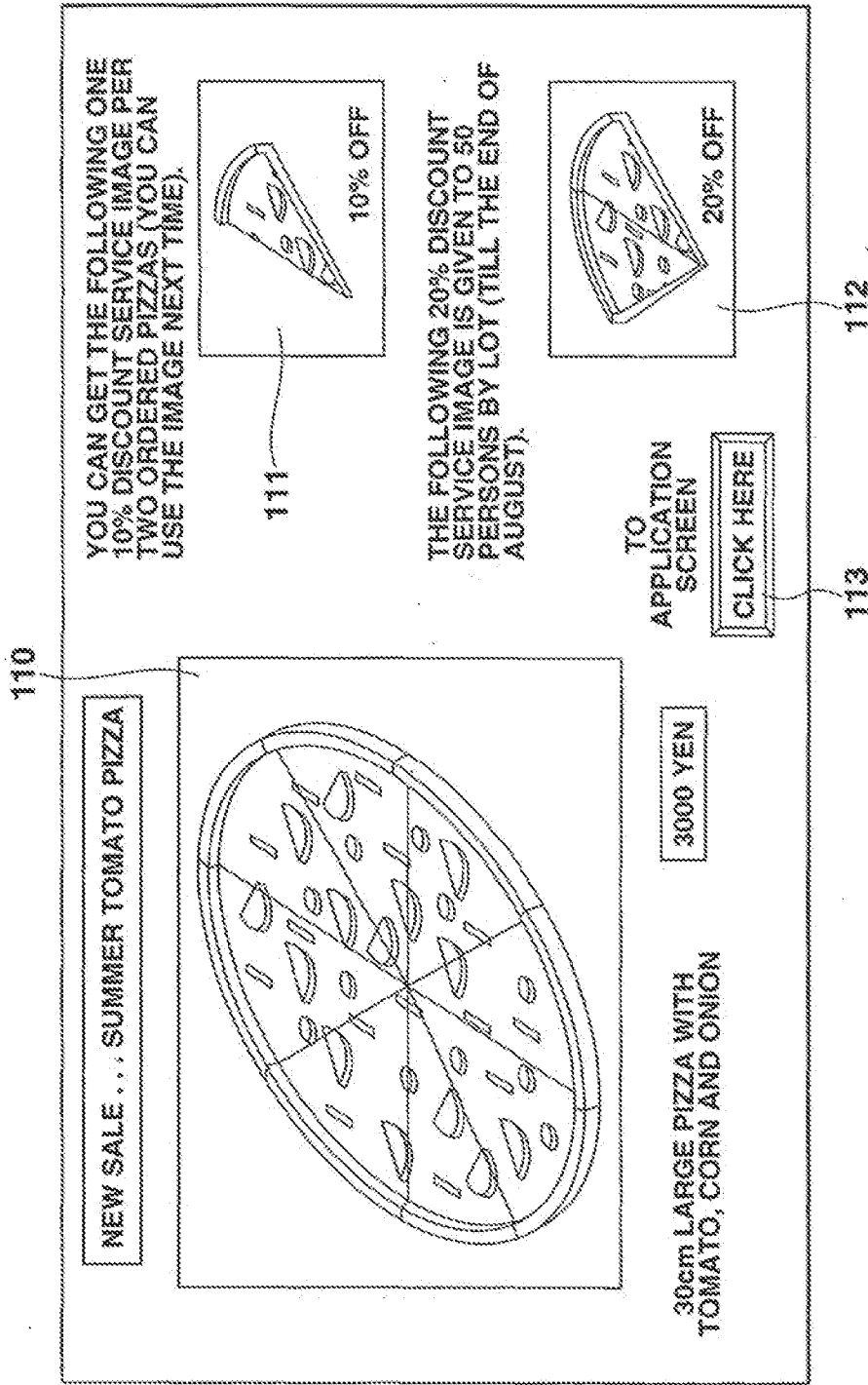


FIG.27

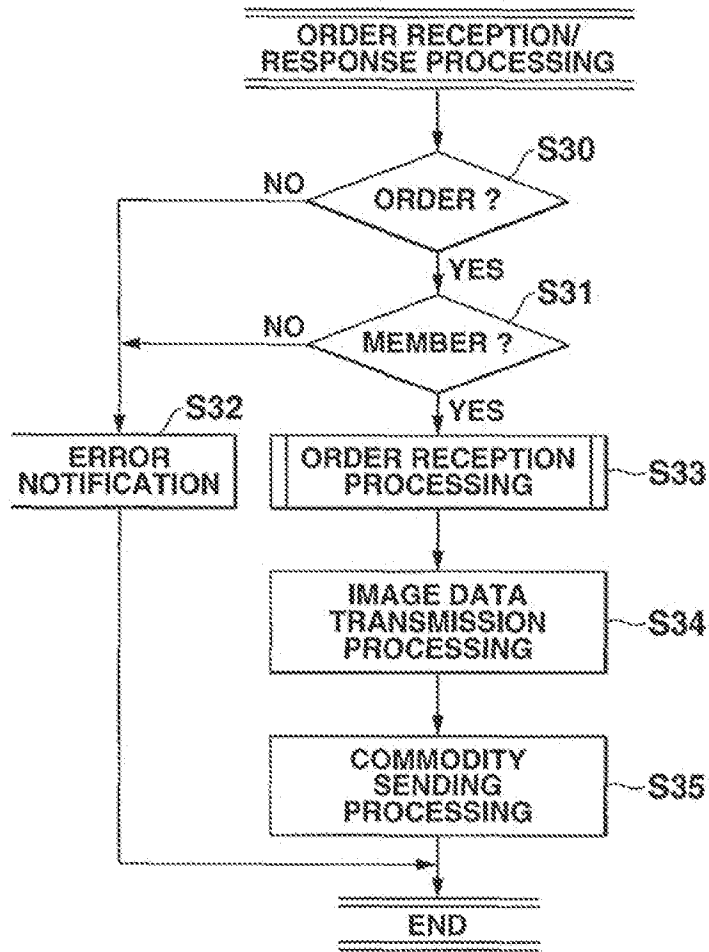


FIG.28

26/26

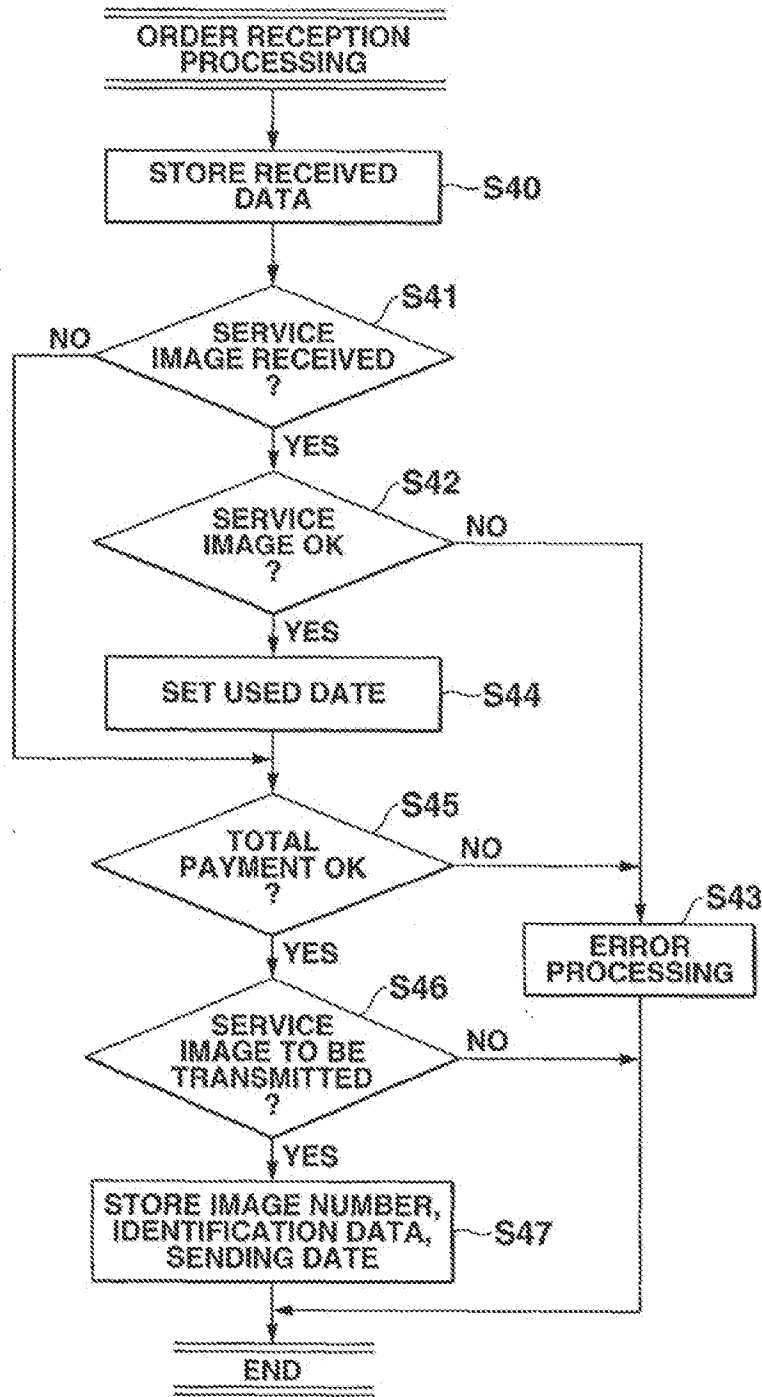


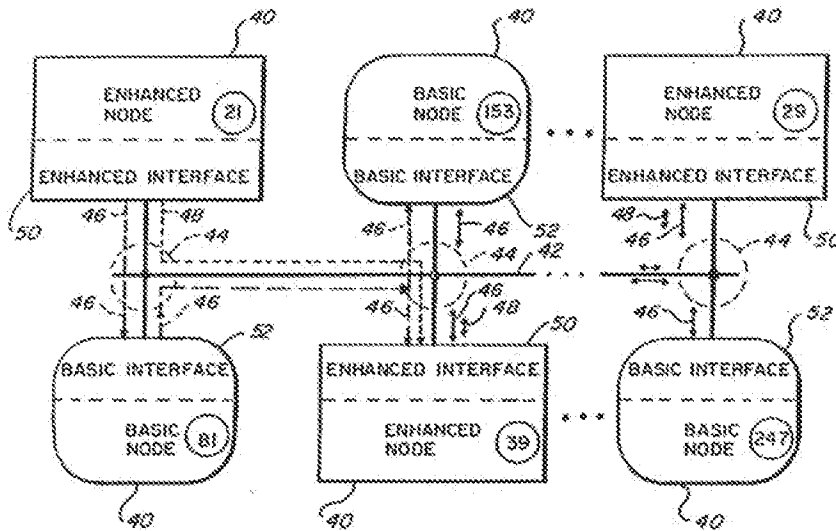
FIG.29



INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

<p>(51) International Patent Classification ⁵ : H04J 3/00</p>	<p>AI</p>	<p>(11) International Publication Number: WO 90/07830 (43) International Publication Date: 12 July 1990 (12.07.90)</p>
<p>(21) International Application Number: PCT/US89/05781 (22) International Filing Date: 15 December 1989 (15.12.89) (38) Priority data: 292,467 30 December 1988 (30.12.88) US (71) Applicant: DATAPOINT CORPORATION [US/US]; 9725 Datapoint Drive, San Antonio, TX 78284 (US). (72) Inventor: FISCHER, Michael, A. ; 2910 Hunter's Horn, San Antonio, TX 78284 (US). (74) Agents: CRANDELL, Ralph, F. et al; Holland & Hart, 555 17th Street, Suite 2900, P.O. Box 8749, Denver, CO 80201 (US).</p>		<p>(81) Designated States: AT (European patent), AU, BE (European patent), BR, CH (European patent), DE (European patent), DK, ES (European patent), FI, FR (European patent), GB (European patent), IT (European patent), JP, KR, LU (European patent), NL (European patent), NO, SE (European patent). Published <i>With international search report.</i> <i>With amended claims.</i></p>

(54) Title: DYNAMIC NETWORK RECONFIGURATION TECHNIQUE FOR DIRECTED-TOKEN EXPANDED-ADDRESS LAN



(57) Abstract

Additional nodes (21, 29, 39) beyond those permitted by the size of the address field of a standard network operating protocol, may be added to a directed token LAN, and the added nodes may be automatically and dynamically configured or reconfigured into a token passing loop. The added nodes (21, 29, 39), referred to as enhanced nodes, interoperably combine a standard reconfiguration sequence of a standard network operational protocol with an enhanced reconfiguration sequence of an enhanced protocol in order to send tokens to establish the next active nodes of the network and thereby establish the token passing loop through all of the standard (81, 153, 247) and enhanced nodes.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify States party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AT	Austria	ES	Spain	MG	Madagascar
AU	Australia	FI	Finland	ML	Mali
BB	Barbados	FR	France	MR	Mauritania
BE	Belgium	GA	Gabon	MW	Malawi
BF	Burkina Faso	GB	United Kingdom	NL	Netherlands
BG	Bulgaria	HU	Hungary	NO	Norway
BJ	Benin	IT	Italy	RO	Romania
BR	Brazil	JP	Japan	SD	Sudan
CA	Canada	KP	Democratic People's Republic of Korea	SE	Sweden
CF	Central African Republic	KR	Republic of Korea	SN	Senegal
CG	Congo	LI	Liechtenstein	SU	Soviet Union
CH	Switzerland	LK	Sri Lanka	TD	Chad
CM	Cameroon	LU	Luxembourg	TG	Togo
DE	Germany, Federal Republic of	LV	Latvia	US	United States of America
DK	Denmark	MC	Monaco		

DYNAMIC NETWORK RECONFIGURATION TECHNIQUE
FOR DIRECTED-TOKEN EXPANDED-ADDRESS LAN

Cross Reference to Related Applications

This is a continuation-in-part of U.S. Patent
5 Applications for LAN WITH INTEROPERATIVE MULTIPLE
OPERATIONAL CAPABILITIES, Serial No. 270,641, and
LAN WITH DYNAMICALLY SELECTABLE MULTIPLE OPERATIONAL
CAPABILITIES, Serial No. 270,804, both filed
November 14, 1988, and assigned to the assignee
10 hereof. The disclosures of these two applications
are incorporated herein by this reference.

This application also relates to the inventions
disclosed in U.S. Patent Applications for MULTIBIT
AMPLITUDE AND PHASE MODULATION TRANSCEIVER FOR LAN,
15 Serial No. 270,739, filed November 14, 1988, and
UNEVEN TOKEN DISTRIBUTION TECHNIQUE FOR TOKEN BASED
LOCAL AREA NETWORK, Serial No. 154,970, filed
February 11, 1988. The disclosures of both of these
applications are also incorporated herein by this
20 reference.

Background of the Invention

The present invention relates to a dynamically
self-configuring, directed token passing local area
network. A local area network is referred to herein
25 as "LAN" or "network". More particularly, the
present invention pertains to improvements in such
LANs which increase the logical limit of the number
of nodes which can be operative on the network, and
which allow all of the nodes, including additional
30 nodes, to be established in a token passing loop, on
a self-configuring and dynamic basis.

In general, a LAN is a communication medium and
means by which a relatively large number of
relatively closely physically located processor-
35 based communications devices can send and receive
message communications. The medium usually includes

electrical or optical cables, but may also include radio or open-air optical links over which signals are conducted. Each communication device and its interconnection to the medium is referred to as a "node". Messages are communicated by transmitting patterns of signal elements, known as frames, between nodes. Each node of the LAN has its own unique address or identification (ID). Typically, a frame includes address information of the transmitting or source node, known as the source ID or SID, and address information of the destination node to which the frame is addressed, known as the destination ID or DID. The transmitted frames are disregarded by all nodes except the destination node, but each node decodes the DID of the frame to the extent necessary to determine whether the frame is addressed to it.

Access to the LAN medium is controlled to assure that only one message is properly communicated at a time, thereby preventing two or more nodes from simultaneously transmitting frames and interfering with one another and with the proper operation of the LAN. One widely used technique of medium access control involves a "token". A token is a frame which is addressed to a specific destination node and which, upon receipt at the destination node, allows that node to have the exclusive right to initiate a message communication over the medium. The operational protocol of the LAN permits only the node in receipt of the token to initiate the message communication. Since the LAN typically uses only one token, there is no interference resulting from the simultaneous initiation of multiple message communications because only the node in receipt of the token can initiate message communications.

The operational protocol of the LAN causes the token to be passed among all of the active nodes in a regular, complete and even rotational pattern or sequence, called a "token loop" or a "token passing loop". The token loop allows each active node to have equal access to the medium for initiating message communications. Each active node receives the token in its turn in the token loop, regardless of whether or not that node has a message communication to initiate. If the node has a message communication to initiate, it does so after receipt of the token, and then passes the token to the next node in the loop after communication of the message has been completed. If the node does not have a message communication to initiate, the token is immediately passed to the next node in the token loop. Another type of LAN access control is contention arbitration or Carrier-Sense Multiple Access (predominantly CSMA/CD). With contention arbitration only those nodes which have messages to communicate vie for access to the LAN medium. When a node with a message to communicate detects that no other message communication is in progress over the medium, it initiates the message communication. So long as no other node also initiates a message simultaneously, the node which is first in time receives exclusive access to the LAN medium for the duration of its message. If two or more nodes initiate message communications simultaneously, a collision of the transmitted signals occurs and prevents the colliding messages from being correctly delivered. Collisions necessitate subsequent re-transmissions of the collided message communications. Thus, in contention arbitrated LANs, each node must contend with all other nodes for access to the LAN medium. Contention-type LANs

are not pertinent to the present invention, because contention-type LANs do not utilize tokens and the present invention pertains to token based LANs.

LANs can further be distinguished by the signal
5 connectivity or signal transmission patterns. Bus-type LANs connect all nodes to a single common logical or electrical point, although the single common point may be physically dispersed over a large number of locations. Bus-type LANs send
10 signals directly from the source node to the destination node, through the common connection point. The transmitted signals from any node are received about simultaneously by all the other nodes and are disregarded by all of the nodes except the
15 destination node. Since all signals are transmitted and received through a common connection point, the communication path between all the nodes of a bus-type LAN is bi-directional, usually half-duplex.

Ring-type LANs have all of their nodes
20 physically connected in a serial loop or ring. Each node always transmits to its neighboring node in the ring and receives from its other neighboring node in the ring. All the nodes in the ring serially pass the uni-directional transmissions from one to
25 another in this ring. The signals are sent through every intervening connected node in the ring between the source node and the destination node. All of the nodes operate in synchronism, with each node receiving and re-transmitting the signal. The
30 address information contained within the transmission causes only the destination node to utilize the information, while the other nodes simply pass the message on to their neighboring nodes in the ring. Thus, all transmissions in a
35 ring-type LAN circulate in an unalterable, predefined, unidirectional signal path through all

of the nodes around the ring. When tokens are employed in a ring-type LAN, the tokens are also passed in a logical sequence which is the same as the unalterable ring-like physical path.

5 The concept of "directed-token" applies to networks where the recipient of a token pass is determined by a destination address or DID in the token frame. A directed token approach is primarily used on token bus networks, where there is a logical
10 token passing loop operating on a physical bus. This technique can also be used on other broadcast-media networks such as radio-frequency or free-air optical links. In contrast, token passing on token-based ring networks is not of the "directed-token"
15 variety, because the token is passed by the physical transmission of the token frame from one node to the next adjacent node in the ring. The sequence of token passing on a ring is the physical connection order of nodes in the ring. As a result, token
20 frames on token-based rings generally do not include a destination address. Bus-structured networks and broadcast-media networks do not have a physical means of designating the destination of a token pass and therefore incorporate a destination address in
25 each token. In general, the improvements of the present invention are not readily applicable to token-based ring LANs.

 In some situations it becomes desirable to connect a large number of nodes to the LAN. The
30 maximum number of nodes on a LAN is due to either physical or logical restrictions. The physical restrictions pertain to electrical and/or cabling limits, as required by or for signal propagation and decay. The logical restrictions relate to how many
35 nodes may be addressed in the node addressing scheme of the LAN operational protocol. As a physical

restriction, a LAN bus which is wired as a single multi-drop cable is limited in the number of drops which may be installed on a cable segment, but a LAN bus which is wired as an unrooted tree has little or no practical limit on the physical number of nodes that may be attached, other than as is limited by the maximum end-to-end signal propagation delays which can be tolerated by the network operating protocol. A ring network has a physical limit on the number of allowed nodes based upon the maximum amount of synchronization fluctuation or clock "jitter" that is acceptable for the data recovery circuits of the node receivers. A small amount of time-base clock jitter accumulates at each node of the ring, which causes a practical limit on the number of nodes on the ring.

As a logical restriction, IEEE 802.3-type bus networks (e.g., contention arbitrated) are not practically limited in the logical number of nodes, since their address fields are 48 bits long, permitting roughly 144 trillion distinct node addresses. On the other hand, most non-IEEE 802 bus networks use address fields in the frames that are only 8 bits long. An 8-bit length of the address field limits the number of nodes and addresses to no more than 255, since each node requires its own separate address. Increasing the size of the address field to accommodate more than 255 nodes might be possible, but the network operating protocol will usually not accommodate expanded address fields.

Increasing the size of a directed token passing network by adding more nodes and expanding the address field without replacing the pre-existing nodes is not a complete solution to expanding the size of the network, because the network operating

protocol does not allow the additional nodes at the expanded addresses to be dynamically configured or reconfigured in the token passing loop.

Reconfiguration, as explained in greater detail below, is that process whereby all the active nodes are automatically and dynamically established in the token passing loop. The reconfiguration sequence under the standard or basic network operating protocol is effective only as to the nodes in the basic range of standard addresses and not to additional nodes in an extended or expanded range of addresses. All of the active nodes must be included in the token loop for the network to function properly. Replacing numerous pre-existing nodes in order to use expanded address space is economically unattractive.

Accommodating the basic network operating protocol for reconfiguration while still incorporating in the reconfiguration sequence those additional nodes at an expanded range of addresses beyond the original limit established by the basic address field, is believed to have been a significant impediment to expanding the size of a directed token LAN, prior to the present invention.

Brief Summary of the Invention

The present invention allows the number of nodes on a directed-token LAN to be increased beyond the limit of the standard or basic address field while permitting the standard or basic network operating protocol to be retained and used during a reconfiguration sequence of all of the nodes, including the additional nodes, in a token loop. A new reconfiguration sequence and operational protocol is incorporated in the additional or enhanced nodes. The new or enhanced protocol provides additional node addressing capability for

the enhanced nodes, and interoperatively interacts with the basic protocol to allow the token loop to be extended from the nodes in the basic range of addresses into the enhanced nodes in an enhanced or expanded range of addresses.

In accordance with its major significant aspects, the directed token LAN of the present invention comprises a plurality of at least three nodes and a network reconfiguration means associated with all of the nodes. Each of the nodes has its own unique address or ID. At least one of the nodes is a basic node having an ID within the basic range of permissible IDs. The basic range is typically established by the size of address field of frames transmitted in accordance with the basic network operating protocol. At least one of the nodes is an enhanced node and is capable of operating in accordance with the basic protocol and in accordance with an enhanced network operating protocol. The enhanced protocol provides one or more enhanced operational capabilities such as an increased data transfer rate or an improved message communication protocol. Each of the enhanced nodes has an ID within either the basic range or within an enhanced range. The enhanced range extends beyond the basic range. At least one of the enhanced nodes has an ID in the extended range. The network reconfiguration means executes a basic reconfiguration sequence and an enhanced reconfiguration sequence. Both reconfiguration sequences pass tokens to locate the next active node of the LAN to establish all of the active nodes in a token loop. Once the token loop is established, the token circulates directly to all of the active nodes in the loop so all of the active nodes, in their turn, can initiate message

communications in accordance with either the basic or enhanced protocols.

Execution of the basic reconfiguration sequence of the basic protocol establishes a basic segment of the token loop through the basic nodes in the basic range of IDs. The enhanced reconfiguration sequence begins executing upon the first enhanced node receiving the token. The first enhanced node to receive the token may be an enhanced node with an ID in the basic range, or when there are no enhanced nodes in the basic range, the first enhanced node in the extended range to experience a reconfiguration precedence timeout. Execution of the enhanced reconfiguration sequence of the enhanced protocol establishes the remaining enhanced segment of the token loop through the enhanced nodes and links the enhanced segment with the basic segment to complete the token loop.

Both reconfiguration sequences preferably involve transmitting tokens to an ID which is incremented relative to the ID of the transmitting node, and recording the ID of the next active node which accepts the token at the incremented ID, as determined by sensing network activity created by the next active node after it has accepted the token. Determining the next active node in this manner by executing the reconfiguration sequence completes the token loop, with the next active node ID being the destination ID of the token when it is passed in the token loop.

A reconfiguration precedence timeout procedure preferably initiates network or system reconfiguration. Preferably the nodes with IDs in the basic range initiate the reconfiguration sequences. Should no enhanced node be located in the basic range, an enhanced node in the enhanced

range will time out and initiate the reconfiguration sequence, even though some or all of the token loop may have been established through the basic nodes in the basic range. When no enhanced nodes are located
5 in the basic range, one of the enhanced nodes must respond to a token addressed to a predetermined ID in the basic range in order to complete the token loop, even though this same enhanced node will respond to message communications addressed to its
10 own ID in the extended range. Preferably, the predetermined address in the basic range to which the enhanced node responds for token passes, is the address normally reserved for broadcast message communications to the basic nodes.

15 Preferably, different types of tokens are transmitted during execution of the reconfiguration sequences. A basic token is transmitted by the basic nodes. Enhanced nodes recognize the basic token because the enhanced nodes operate in
20 accordance with the basic protocol. Enhanced nodes transmit extended tokens and enhanced tokens. Enhanced tokens are valid only in the enhanced protocol in accordance with which only the enhanced nodes also operate. Extended tokens appear as valid
25 communications in both the basic and enhanced protocols. Extended tokens are employed to link the completed enhanced segment of the token loop with the basic segment of the token loop and to communicate additional information which is
30 transparent to the basic nodes but which is recognized by other enhanced nodes.

The scope of the present invention is more precisely defined by the appended claims. A more complete understanding of the invention and its
35 embodiments can be gained from the following

detailed description and from the accompanying drawings.

Brief Description of the Drawings

5 Fig. 1 is an illustration of a bus-type LAN in which the present invention is incorporated, having a network medium connecting multiple nodes, including basic nodes and enhanced nodes.

10 Fig. 2 is a block-diagram illustration of a portion of Fig. 1, showing the basic and enhanced nodes in greater detail.

Fig. 3 is a generalized block diagram of an interface of a basic or enhanced node of the LAN shown in Fig. 2.

15 Fig. 4 is a block diagram of an enhanced interface of an enhanced node of the LAN shown in Fig. 2, which is capable of operating at a plurality of different rates and in accordance with a plurality of different protocols when communicating with other nodes of the LAN shown in Fig. 1.

20 Fig. 5 is a generalized illustration showing the fields of a token which is passed between all of the nodes of the LAN shown in Figs. 1 and 2.

25 Fig. 6 is a generalized bit pattern illustration of a basic frame known as a token (ITT), which is communicated between each interface of each basic node of the LAN shown in Figs. 1 and 2, represented in a basic symbol and basic interval format.

30 Fig. 7 is a bit pattern and field illustration of the token frame shown in Fig. 6, represented in an enhanced signal and enhanced interval format as decoded by the enhanced interfaces of each enhanced node of the network shown in Figs. 1 and 2.

35 Fig. 8 is a bit pattern and field illustration of a token frame shown in Fig. 6, known as an extended token (XITT), represented in enhanced

signal and enhanced interval format which is communicated between the enhanced interfaces of enhanced nodes and to the basic interfaces of the basic nodes of the network shown in Figs. 1 and 2, and which appears to the interfaces of the basic nodes as standard network signaling.

Fig. 9 is a bit pattern and field illustration of a frame shown in Fig. 6, known as a speed or enhanced token (XTOK), which is communicated between the enhanced interfaces of the enhanced nodes of the LAN shown in Figs. 1 and 2, and which is not decoded by, but which appears to, the interfaces of the basic nodes as standard network signaling.

Fig. 10 is an illustration of one example of a prior art token passing loop established by a basic reconfiguration sequence among only basic nodes of a network.

Fig. 11 is a flow chart illustrating the basic reconfiguration sequence of each interface of each basic node of the LAN shown in Figs. 1 and 2, to establish a token passing loop among only basic nodes as illustrated in Fig. 10.

Fig. 12 is an illustration of one example of a token passing loop established by the network reconfiguration means according to the present invention among the basic and enhanced nodes of the LAN shown in Figs. 1 and 2, where both the basic and enhanced nodes have addresses within the basic range of permissible addresses. Fig. 12 illustrates the situation where a basic node times out to initiate the reconfiguration. Fig. 12 also illustrates the types of tokens illustrated in Figs. 7 to 9 which are passed to establish the loop, and which are passed in the normal course of network operation after the token loop is established.

Fig. 13 is an illustration of another example of a token passing loop established by the network reconfiguration means according to the present invention among the basic and enhanced nodes of the LAN shown in Figs. 1 and 2, where the addresses of some of the enhanced nodes and all of the basic nodes are within the basic range of permissible addresses and the remaining enhanced nodes have addresses in an enhanced range of permissible addresses beyond the basic range of addresses. Fig. 13 illustrates the situation where a basic node times out to initiate the reconfiguration. Fig. 13 further illustrates the types of tokens shown in Figs. 7 to 9 which are passed to establish the loop, and which are passed in the normal course of network operation after the loop is established.

Fig. 14 is an illustration of another example of a token passing loop established by the network reconfiguration means according to the present invention among the basic and enhanced nodes of the LAN shown in Figs. 1 and 2, where the addresses of some of the enhanced nodes and all of the basic nodes are within the basic range of permissible addresses and the remaining enhanced nodes have addresses in an enhanced range of permissible addresses beyond the basic range of addresses. Fig. 14 illustrates the situation where an enhanced node times with an address in the basic range out to initiate the reconfiguration sequence. Fig. 14 further illustrates the types of tokens shown in Figs. 7 to 9 which are passed to establish the loop, and which are passed in the normal course of network operation after the loop is established.

Fig. 15 is an illustration of another example of a token passing loop established by the network reconfiguration means according to the present

invention among the basic and enhanced nodes of the LAN shown in Figs. 1 and 2, where only basic nodes have addresses within the basic range of permissible addresses and all of the enhanced nodes have addresses in the enhanced range of permissible addresses. Fig. 15 illustrates time-outs which occur first at one basic node and later at one enhanced node during which and by which the token passing loop is established during reconfiguration. Fig. 15 also illustrates the types of tokens shown in Figs. 7 to 9 which are passed to establish the loop, and which are passed in the normal course of network operation once the loop is established.

Fig. 16 is an illustration of one example of a token passing loop established by the dynamic reconfiguration sequence according to the present invention among only enhanced nodes of a LAN where all of the enhanced nodes have addresses in the enhanced range of permissible addresses. Fig. 16 also illustrates the types of tokens shown in Figs. 8 and 9 which are passed to establish the loop, and which are passed in the normal course of network operation once the loop is established.

Figs. 17A, 17B and 17C are complementary illustrations which form a flow chart illustrating the reconfiguration sequence performed by the enhanced interfaces of the enhanced nodes of the LAN shown in Figs. 1 and 2.

Detailed Description

The present invention applies to a local area network (LAN or "network") such as that shown in Figs. 1 and 2. The LAN comprises a plurality of nodes 40 which are all commonly interconnected to a communication medium 42. The communication medium 42 includes means by which signals are transmitted between the nodes 40. The communication medium may

take the form of a plurality of interconnected signal communication links, such as coaxial cables, twisted cable pairs, optical links, radio links, or combinations of these and others.

5 The LAN illustrated in Figs. 1 and 2 is a bus-type LAN, meaning that all of the nodes 40 are connected to a single logical point (the medium 42) and logically in parallel with one another. An essential characteristic of a bus-type LAN is that
10 each transmission by any node is directly received by all of the other nodes. Typically, the nodes are connected through connecting point devices known as hubs 44. A hub 44 is a means by which a plurality of signal communication links can be connected
15 together, thus connecting all the communication links to a single common logical point, the medium 42. Hubs facilitate cable management, signal amplification and/or fault isolation. Hubs neither interpret nor modify LAN communications. Each node
20 of a bus-type LAN may directly address and communicate with other nodes through the single logical point.

Each node 40 of the LAN has its own unique network address, known as an identification (ID).
25 This address or ID is assigned to the node at the time the node is physically connected to the LAN medium 42. The numbers enclosed within the circles and the boxes in the nodes 40 shown in Figs. 1 and 2 are representative examples of network addresses.

30 The nodes 40 communicate with each other by transmitting "frames" of data. Each frame typically include the address of the node to which the frame is destined, referred to as a DID. The node which initiates the communication, hereinafter referred to
35 as a "source" node, includes the DID of the node to which the transmission is destined as a part of the

frame transmitted over the medium 42. The node to which the communication is destined is referred to as a "destination" node. Since all of the other nodes on the LAN also receive the signals transmitted by the source node, the DID is utilized by each node on the network to recognize and accept only those transmissions addressed to it, while discarding or not recognizing the other transmissions not addressed to it. In addition, since some communications over the network involve multiple transmissions of signals between the source and destination nodes, the source node also frequently includes its own address (SID) in transmissions so the destination node can utilize that address when replying. Broadcasts, which are received by all nodes, and multicasts which are received by predefined groups of nodes, are also made possible by this addressing technique.

One well known, token-passing, bus-type LAN which has a basic address field limitation of eight bits is that manufactured and sold by the assignee hereof under its United States registered trademark ARCNET. An extensive amount of information has been published on the ARCNET LAN, both by the assignee of the present invention and by others. Components to implement the ARCNET LAN are commercially available from sources including the assignee and others. One source of information concerning the ARCNET LAN is the ARCNET Designer's Handbook published by Datapoint Corporation, San Antonio, Texas, copyright 1983. The basic aspects of the preferred embodiment described herein are those of the ARCNET LAN, although the present invention has applicability to directed-token LANs other than the ARCNET LAN.

Two different types of nodes 40 are present on the LAN in accordance with the present invention.

As is shown in Figs. 1 and 2, basic nodes at IDs 81, 153 and 247 and enhanced nodes at IDs 21, 29 and 39, are both connected in the LAN. The basic nodes have only a single basic or common operational capability, and therefore always operate in accordance with this common operational capability. The enhanced nodes have multiple different operational capabilities. One of the multiple operational capabilities available from each enhanced node is the common operational capability also present in each basic node. Thus, both the enhanced and the basic nodes have one common operational capability which may be used for communication.

Operational capability as used herein may refer to a variety of substantially different operational functionalities. Examples of multiple different operational capabilities contemplated by the present invention are different message communication protocols and different data transfer rates between the nodes. As an example, but not to be used to construe the extent of the difference in operational capabilities, the common data transfer rate may be at 2.5 million bits per second, while the enhanced data rate transfer capability may be 20 million bits per second.

In accordance with the present invention, the enhanced nodes of the LAN are capable of dynamically selecting among themselves which of the operational capabilities to employ in communicating with another enhanced node and with a basic node. The common operational capabilities of each basic node remain unaffected by the presence of the enhanced nodes, thus preserving the normal operational capability of the LAN and avoiding the necessity to replace the whole LAN to obtain enhanced communication

capabilities between a limited number of high performance nodes, i.e. the enhanced nodes. More details regarding the enhanced nodes and their enhanced capabilities are described in the two
5 previously mentioned applications LAN WITH DYNAMICALLY SELECTABLE MULTIPLE OPERATIONAL CAPABILITIES and LAN WITH INTEROPERATIVE MULTIPLE OPERATIONAL CAPABILITIES.

The common and enhanced operational
10 capabilities of the basic and enhanced nodes of the LAN are illustrated by the dash lines 46 and 48 shown in Fig. 2. The longer dash lines 46 represent the common operational capability. The enhanced node ID 21 may communicate at the common operational
15 capability with the basic node ID 81. Similarly, the basic node ID 81 can communicate only at the common operational capability with another basic node ID 153 and with an enhanced node ID 39. The shorter dash lines 48 illustrate an enhanced
20 operational capability which is used only by the enhanced nodes. The enhanced node ID 21 may communicate only with the other enhanced nodes, e.g. ID 39, at the enhanced capability. As is represented by the longer dash lines 46 and the
25 shorter dash lines 48, the enhanced nodes can communicate at both operational capabilities, while the basic nodes can communicate only with the common operational capability. Of course, the network medium 42 carries the signals representative of
30 either type of operational capability with equal facility.

Each node includes an interface 50 or 52 by which signals are applied to and received from the medium 42, as is shown in Fig. 2. Enhanced nodes
35 include enhanced interfaces 50, while the basic nodes include basic interfaces 52. Each node,

whether enhanced or basic, also includes a host computer or processor (not shown) which performs various data processing functions, or a controller performing various data transfer functions. For example, a node may include a personal computer, work station, a network server computer, or network connected I/O device, interface, sensor or actuator, or the like, which transmits and receives data over the medium 42.

10 The function of the interfaces 50 and 52 is to send the data over the medium, to receive the data from the medium, to receive the data to be sent on the medium from the host processor, and apply the data received from the medium to the host processor, so that the host processor can function in an efficient and reliable manner. Because each node includes an interface 50 or 52, the functionality of the interfaces is distributed throughout all of the nodes of the LAN.

20 The basic components of a typical interface 50 or 52 are illustrated in Fig. 3. A transceiver 54 applies the electrical, optical or other physical signals to the medium 42 and receives the signals from the medium 42. A physical level protocol interface 56 receives electrical signals from the transceiver 54 and applies electrical signals to the transceiver 54. The signals transmitted on the medium 42 are in serial form. One of the functions of the physical level protocol interface 56 is to convert the serial data stream into a parallel data stream for use by the other elements of the node, and to convert the parallel data stream from the other elements of the node into a serial data stream. The term "physical level" used in reference to the interface 56 is the well known physical layer in the seven layer reference model for network

communications. The physical level or layer is responsible for interfacing with the medium 42, detecting and generating signals on the medium, and converting and processing the signals received from the medium. In very general terms, the physical layer concerns the general encoding of network data into waveforms which will travel on the medium, and decoding those waveforms when received. The physical level protocol interface 56 and the transceiver 54 achieve these functions.

Each interface 50 or 52 also includes a link level protocol engine 58. "Link level" again refers to the standard seven layer reference model for networks, and generally relates to sending and receiving frames of data over the medium 42 and controlling access to the medium 42. Frames of data, as will be discussed below, relate to groupings of various physical level signals in such a way to achieve the desired network functionality. For example, all the functions involved in sending and receiving frames, including inserting starting delimiters, ending delimiters, and stripping these off once the data is received, are link level functions. Other link level functions are access control to the medium and the handling of affirmative and negative acknowledgements. The link level protocol engine 58 controls and executes the reconfiguration sequence functionality to which this invention relates. The higher levels of communication in the seven layer model are generally handled by the host or I/O processor of the node. Even though it is preferred to implement the functionality of the interfaces in a distributed manner in each node, some of this functionality, for example media access control, can be implemented on a centralized basis, as is known.

More reliable network interfaces generally provide a separate link level protocol engine 58, generally implemented as a micro sequencer operating from firmware. However, many of the link level functions could also be achieved by the host processor. Generally speaking, the advantages of providing a separate link level protocol engine 58 are that its functionality is generally independent from the host processor, and therefore it offers more reliability and interoperability in LAN functionality. The functionality of the link level protocol engine 58 in the common capability of operation is identical in all nodes, and its functionality is isolated and secure against possible malfunctions in the host software or hardware. A second reason for providing a separate link level protocol engine 58 is that the time dependent aspects of the operation of the host processor are isolated from the time dependent aspects of data communication over the LAN. Use of the separate link level protocol engine 58 avoids sporadic timing problems between the host processor and the signals on the LAN. Lastly, the use of the separate link level protocol engine 58 allows some of the functionality from the host processor to be off-loaded, thereby increasing the productivity of the host processor.

Basic interfaces 52 for the ARCNET LAN are commercially available from a wide variety of sources, and are known as resource interface modules (RIMs). The link level protocol engine 58 which is used on the ARCNET LAN is commercially available as an integrated circuit designated COM90C26 from Standard Microsystems Corp. and 90C26 from NCR Corporation. The physical level interface 56 used on the ARCNET LAN is commercially available as an

integrated circuit designated COM90C32 from Standard Microsystems Corporation and 90C32 from NCR Corporation.

5 An example of an enhanced interface 50 is shown in Fig. 4. The common and enhanced operational capabilities selected for illustration of the present invention are different data communication rates. Accordingly, the enhanced interface 50 includes a common rate transmitter 60 and an
10 enhanced rate transmitter 62, each of which operates at a different data rate. The transmitters are commonly connected to the network medium 42. A common rate receiver 64 and an enhanced rate receiver 66 are also included in the interface 50.
15 These receivers commonly receive signals from the network medium 42. The common rate transmitter 60, and the common rate receiver 64 operate at the common data rate. Therefore, the enhanced interface 50 will always be able to transmit and receive at
20 the common data rate. The enhanced transmitter 62 and the enhanced receiver 66 have a substantially different data rate communication capability, compared to the common rate. The data rate capabilities of the enhanced transmitter 62 and the
25 enhanced receiver 66 are preferably the same in all of the enhanced nodes.

The various transmitters and receivers illustrated in the enhanced node 50 may actually be separate items, as indicated in Fig. 4, or may be
30 integrated into a single device 68. One example of a transceiver capable of communicating at common and enhanced data rates is disclosed in the application for MULTIBIT AMPLITUDE AND PHASE MODULATION TRANSCEIVER FOR LAN, referenced above.

35 Each enhanced node 50 also includes a network protocol controller 70. In a typical case, the

network protocol controller 70 will control all of the physical and link level protocol functionality, leaving the network, transport and other higher levels of network functionality to the host processor of the node. The network protocol controller 70 is the preferred means for achieving the reconfiguration sequence functionality described herein.

The protocol controller 70 controls a transmitter selector 72 which in turn supplies a control signal 69 to the transmitters 60 and 62 to activate the selected transmitter. Data from the host computer is converted by the protocol controller 70 into the appropriate frame format for both the common and enhanced operational capabilities. The protocol controller 70 assures that all transmissions are in accordance with the established protocols for the selected operational capability. Media access is controlled by the protocol controller 70.

The enhanced interface 50 also includes a receiver selector-discriminator 74. Signals from the network medium 42 are preferably applied as control signals at 76 to the selector-discriminator 74. In the majority of cases, the signals supplied over the medium 42 will unambiguously identify the rate and protocol at which those signals are transmitted. The physical characteristics or signal elements of the signals may distinguish the data rates from one another. The signals form a control signal at 76 which allows the selector-discriminator 74 to select one of the data paths 78 and 80 from the receivers 64 and 66, respectively, which will be coupled through the selector-discriminator 74 over the data path 82 to the network protocol controller 70. Of course, the network protocol controller 70

removes the various link and physical level control information from the signals in the data path 82, and supplies those remaining signals to the host processor of the node. The selector-discriminator 5 74 discriminates among the various data rates present on the medium 42, and selects the appropriate receiver which supplies the data to the network protocol controller 74.

A capability table 84 is also connected to the 10 protocol controller 70, in the enhanced interface 50. The capability table is a random access memory (RAM) in which information is recorded regarding the capabilities and status of other nodes on the network. This information is made available to the 15 protocol controller 70 for use in selecting the transmitters and receivers for communication. The capability information is recorded in the table 84 in association with the ID of each node.

In addition to controlling the selection of the 20 transmitter, the data in the capability table 84 may also be used by the protocol controller 70 to supply a control signal at 86 to the receiver selector-discriminator 74 for selecting the appropriate one of the receivers to receive transmissions. The 25 control signal at 86 is used by the selector-discriminator 74 when the characteristics of the raw signals applied at 76 are insufficient to discriminate between multiple different data rates on the medium 42. Under such circumstances, the 30 protocol controller 70 would obtain information, either from the host processor, the capability table 84 or from other sources, that transmissions from a particular other node would be arriving at a particular rate. Under those circumstances, the 35 signal at 86 would select the appropriate receiver

to apply output signals over the data path 82 to the protocol controller 70.

5 The capability table 84 may not be needed in practicing this invention, as is described more completely in the applications LAN WITH DYNAMICALLY SELECTABLE MULTIPLE OPERATIONAL CAPABILITIES and LAN WITH INTEROPERATIVE MULTIPLE OPERATIONAL CAPABILITIES.

10 Based on the information recorded in the capability table 84 or otherwise obtained, the source node will generally transmit at the highest data rate that the destination node is capable of receiving during normal communications. However, it should also be recognized that a destination node
15 and source node could dynamically negotiate or establish a data rate on a communication by communication basis which is less than their maximum capabilities if such circumstances are appropriate. Examples of such circumstances might be where
20 optical or radio communication links are included in the medium 42 and atmospheric or other environmental influences have degraded the integrity of the communication link to a point where the high data rate is more likely to result in an unacceptable
25 amount of transmission errors.

The network protocol controller 70 is preferably implemented by a micro-sequencer operating from firmware. Alternatively, the majority of the functions of the network protocol
30 controller 70 could also be implemented on the software of the host computer of the node, but for the reasons previously mentioned, including reliability, compatibility and economical implementation, a separate network protocol
35 controller 70 is preferred. Details of the reconfiguration sequence functionality preferably

provided by means of the network protocol controller 70 to dynamically and automatically establish a directed token passing loop among basic and enhanced nodes, including enhanced nodes at addresses greater than those permitted in the basic range of permissable addresses, is set forth below.

In token passing LANs, the functionality for passing the token is typically distributed to each of the node interfaces 50 and 52 (Fig. 1). A token is a predetermined frame which, when received by the interface of a particular node identified by its network address, enables that particular node to initiate a message communication. Each interface includes its own components for identifying the network address of the next active node of the network and for passing the token to the next active node. Centralized token passing control techniques for LANs are also known. In centralized token passing control, a centralized node or processor controls the passage of the token from one active node to the other. The present invention has applicability to both distributed and centralized control arrangements for directed token passing in a LAN.

In addition to its own ID, the address of the next active node (NID) in the rotational sequence of token passing is established and maintained in each node interface 50 and 52. An active node is one which is currently able to participate in network communication but which may or may not have messages to communicate. Inactive nodes, that is those nodes which are not functioning at that time and are therefore not able to participate in network communication, are eliminated from the token passing loop. Only the active nodes participate in token passing.

Upon receipt of the token, the node initiates a message if it has a message to communicate. At the conclusion of the message, or if no message is to be communicated, the token is passed to the next active node in the rotational sequence. In this manner, the token is passed from active node to active node in an even rotational sequence or token passing loop.

The even rotational sequence of basic token passing is typically from a basic node of a lesser network ID to the next active basic node at the next higher network ID in the basic range of permissible addresses. A basic token passing loop is illustrated in Fig. 10. However, when a token loop is established through both the basic and enhanced nodes in accordance with the present invention, the token loop includes a basic segment through the basic nodes and an enhanced segment through the active enhanced nodes in the enhanced range of addresses. Once the token commences circulating through the active enhanced nodes in the enhanced range of addresses, the token passes from the active enhanced node of the lowest ID through all of the active enhanced nodes in an increasing ID sequence, and from the active enhanced node at the highest enhanced range ID back to the next active node in the basic range of permissible addresses. When the token reaches the active basic node of the highest ID, the token is passed to the active node of the lowest ID to commence the next token loop. Token passing loops involving both basic and enhanced nodes are shown in Figs. 12-15, and a token passing loop involving only enhanced nodes is shown in Fig. 16.

Each node stores the address (NID) of the next active node in the token loop. In this manner, each

active node knows the next active node in the loop to which the token is to be addressed when the token is passed. To establish the NID of the next active node for token passing purposes in all of the active nodes, network or system reconfiguration occurs. The present invention relates to automatically establishing or reconfiguring a token loop to include basic and enhanced nodes in the basic range of permissible addresses and enhanced nodes in an enhanced range of permissible addresses beyond the basic range of addresses. In the ARCNET LAN, the basic range of addresses is from 0 (indicating a broadcast, except in one situation described below) to 255. According to the present invention, the enhanced range is from 256 up to the maximum point which may be almost unlimited but which is selected for illustration of this invention to be 2047.

Network reconfiguration starts upon power on of the network, whenever a new node becomes active on the network, or whenever any node has not received the token in a predetermined period of time. At the beginning of the basic network reconfiguration sequence, the interface initializes its NID to its own ID. A reconfiguration precedence timeout procedure at each interface is used to select the basic node with the highest assigned ID, and that node commences sending a token. The first token is sent to an ID which is equal to its own assigned ID. Sending the first token to the node's own ID is convenient for implementation at the link level in the network protocol controller, but does not result in passing the token. The first functional token passing attempt, however, is to the NID which is the assigned ID of the node plus one. After sending that token, the interface waits for activity on the medium. Such activity occurs only where another

node has received the token and is sending a message or passing the token itself. If no activity is sensed within a predetermined time, the interface increments its NID and repeats the process. The process continues until the next active node is addressed and that node responds to the token by creating network activity. At that point, the interface which sent the token successfully to the next active node senses the activity and establishes the correct NID for that next active node in the token loop. The next active node repeats the procedure until it too has established its NID and successfully passed the token. Incrementing the NID in the basic range of addresses is performed modulo the size of the LAN address field (256 in the case of conventional ARCNET), to produce a wraparound from the highest ID in the basic range to a zero ID in the basic range ((255 + 1) modulo 256, is 0). All of the interfaces of all of the basic nodes of the LAN function in a similar manner. A somewhat different situation occurs in enhanced nodes, as is described below, but incrementing the NID of the enhanced nodes is still employed until the NID of each enhanced node is determined, which establishes the complete token passing loop through all of the active nodes.

Network or system reconfiguration can occur at any time to allow new active nodes to enter the token loop. When an interface is first powered on or when it has not received a token for a predetermined time period (as established by a lost token timer), it sends a reconfiguration burst. A reconfiguration burst is a unique signal pattern which is longer than any type of frame so it interferes with any communication which is attempted

or in progress. This interference prevents passing the token, thereby forcing system reconfiguration.

Another type of reconfiguration which can occur allows previously active but now inactive nodes to drop out of the token loop. This reconfiguration does not involve network-wide reconfiguration. When an active node becomes inactive and drops out of the rotational sequence, the attempted token pass to the newly inactive node will result in no interface receiving the token. The node which unsuccessfully attempted to pass the token to the previously active but now inactive node will sense a lack of activity that would have otherwise resulted if the node had accepted the token. After a predetermined time period which is shorter than the lost token time period before system reconfiguration occurs, the node which unsuccessfully attempted to pass the token will commence the reconfiguration sequence of incrementing its NID and sending tokens until a token is successfully passed, as in a system reconfiguration previously explained. However, once the token is successfully passed the token loop is re-established because all of the other nodes in the loop remain active and retain the NIDs they previously established during system reconfiguration. Thus, when a previously active but now inactive node drops out of the loop, only the preceding active node in the token loop will establish a new NID, thereby saving some of the time required to create a network-wide reconfiguration.

Communication between nodes occurs by sending and receiving frames. Since each frame is in reality a serial stream of signals on the LAN medium, each frame is separated by an interframe gap (IFG) of silence or absence of signals on the medium. The duration of the interframe gap is

usually established based on fundamental physics and relates to the propagation delays created in part by the physical size of the network. The purpose of the interframe gap is to allow the medium to quiesce after signals have been applied to it, and to allow the transceiver circuitry to be made ready for the next frame. For most LANs, the interframe gap is at least equal to the physical settling time of the medium.

5
10
15
20
25
30
35

A token frame 100 is illustrated in Fig. 5. The token is broken into a number of fields: a starting delimiter (SD) 102, a type field 104 (which for a token is designated EOT for "end of transmission"), two identical repeated fields 106 which each contain the destination address of the token (which is the NID of the next active node in the token loop), and an ending delimiter (ED) 108. The SD 92 is typically a fixed pattern of signals used to indicate that the frame is beginning and to provide the necessary synchronization or calibration information for the receiver at the destination node. The ED 108 is a pattern of signals (silence in the case of ARCNET) which is fixed in length and in content and serves to mark the end of the frame 90. The SD and ED are physical level protocol elements and are described in greater detail below. The NID of the token is repeated to prevent transmission errors from causing the wrong node to receive the token. If the two NID fields do not match the token is ignored. A field for the ID of the source node (SID) may or may not be present in a token depending both on the type of network and the type of token. For example, in most token based networks including the ARCNET LAN, an SID is not present in the token frame.

In order to successfully interoperatively incorporate two or more different operational protocols in a single LAN, the different communication protocols cannot interact with one another so as to cause errors or to generate spurious, invalid data, but yet the two protocols must create an appearance of valid activity in at least one (the common) and possibly both protocols. The protocols must not violate or conflict with one another. The protocols must be clearly separable, so the enhanced protocol and the enhanced aspects of an interface function reliably at the enhanced capability, and the basic or common protocol and basic aspects of the interface function reliably at the basic capability.

To successfully adapt a second different communication protocol and capability to a basic LAN as described herein, considerations must be given to compatibility and interoperability of the physical signals applied to the medium, the transparency of certain signals to the basic interfaces, the frames transmitted in each protocol, the basic and enhanced protocols, and other factors. These compatibility considerations are described in greater detail in the applications LAN WITH DYNAMICALLY SELECTABLE MULTIPLE OPERATIONAL CAPABILITIES, LAN WITH INTEROPERATIVE MULTIPLE OPERATIONAL CAPABILITIES, and MULTIBIT AMPLITUDE AND PHASE MODULATION TRANSCEIVER FOR LAN.

The standard electrical signal applied by the basic interfaces to the LAN medium to signify a one bit is a basic dipulse, in conventional ARCNET. The dipulse is applied for a time period or interval which in a standard ARCNET LAN is 200 nsec, followed by an equal length period or interval of 200 nsec of silence or absence of a physical signal following

each dipulse. To transmit a one bit digital indication in the basic network protocol, the dipulse is applied during an interval and is followed by the silence interval. To transmit a zero bit digital indication, two periods of silence exist sequentially. Thus in the basic and common protocol, the dipulse or a silence signal is applied in one interval followed sequentially by another interval of silence or absence of a signal. This pattern of interval pairs in which the second interval is always silence and the first interval is always a basic dipulse or silence is the basic or common signaling technique.

The basic token 100 is illustrated in Fig. 6 in terms of the actual coding for the fields 102, 104 and 106. Each basic dipulse is represented by a "1" signifying a single bit, followed by an interval of silence in which no marking is illustrated. A 0 symbol also represents a signal indicating the absence of a dipulse, indicative of a 0 bit. Each 0 bit is also followed by an interval of silence which is also not marked.

In order to communicate in a second enhanced protocol and to achieve an enhanced data rate capability, the second interval in each pair of intervals is utilized to present signals. Further, the signals applied during both signal intervals are amplitude and phase modulated to thereby achieve even greater data transfer capabilities, as is discussed in the applications LAN WITH INTEROPERATIVE MULTIPLE OPERATIONAL CAPABILITIES and MULTIBIT AMPLITUDE AND PHASE MODULATION TRANSCEIVER FOR LAN.

The token frames in the basic protocol and the enhanced protocol are shown in Figs. 6 to 9. Only the token frames are shown because the other frame

types of the basic protocol and the enhanced protocol are not affected by, and are not directly used in, the present invention. Three types of token frames are utilized in the present invention:
5 an ITT shown in Figs. 6 and 7; an XITT shown in Fig. 8; and a XTOK shown in Fig. 9.

The ITT (Invitation To Transmit) standard or basic token frame 100 shown in Figs. 6 and 7 is passed by basic nodes. The specific coding for the
10 ITT is illustrated. Fig. 6 illustrates the manner in which a basic interface interprets the ITT, where the second silent interval of each pair of intervals of the ITT is not recognized by the basic interface. Fig. 7 illustrates the manner in which each enhanced
15 interface interprets the ITT, where the second silent interval of each pair of intervals of the ITT is interpreted as a zero signal element. The interpretations by both the basic and enhanced nodes are without conflict.

The XITT (eXtended Invitation To Transmit) extended token frame 110 shown in Fig. 8 is passed between enhanced nodes and is interpretable by both basic nodes and enhanced nodes. When received by a basic node, an XITT appears indistinguishable from
25 an ITT, as illustrated in Fig. 7, because the signals in the second interval of each pair of intervals are not interpreted by the basic node. In an XITT, the fourth interval of the starting delimiter is a 1 rather than a 0, permitting the
30 XITT to be distinguished by enhanced nodes from the ITT (Fig. 7). However, additional information (called the "flag" and represented by "f"s) is encoded in the SD of the XITT in the 6th, 8th, 10th and 12th intervals. This additional information
35 does not conflict with interpretation of an XITT by standard nodes, but is decoded by enhanced nodes.

The last four bit-pairs ("1f") in the SD each include a bit, the "f", by which information on the capabilities of the enhanced node transmitting the XITT frame is communicated. These "f" bits are used
5 to encode information relating to the transmission speed and capabilities of the enhanced node transmitting the XITT, as well as transmitter and receiver status of that node. This coding is further explained in the co-pending applications for
10 LAN WITH DYNAMICALLY SELECTABLE MULTIPLE OPERATIONAL CAPABILITIES and LAN WITH INTEROPERATIVE MULTIPLE OPERATIONAL CAPABILITIES.

The XTOK (eXtended TOKen) enhanced or speed token frame 112 is shown in Fig. 9. Enhanced nodes
15 pass the XTOK to other enhanced nodes. The XTOK cannot be interpreted by basic nodes, but it does appear as network activity to basic nodes. The DID fields contain the address of the node to which the XTOK is directed. Each DID field contains 12 bits
20 rather than the 8 bits contained in the DID fields of the ITT and XITT frames. The XTOK provides the means by which enhanced nodes pass the token and, subsequently, communicate data packet frames, to enhanced nodes within the enhanced range of
25 addresses which are greater in number than the standard range of addresses (0-255). By using 11 bits of the DID address field in the XTOK 112, up to 2047 nodes may be individually defined and addressed.

30 Fig. 10 illustrates a token passing loop for nodes having addresses within the basic range of permissible addresses. If some of the nodes are enhanced nodes, one enhanced node may send an XTOK to a next subsequent enhanced node, but unless two
35 enhanced nodes are in sequential order, the token passing occurs by the basic nodes sending ITTs and

the enhanced nodes sending XITTs, as represented by the longer dashed lines 46.

Fig. 11 shows the flow of a basic reconfiguration sequence implemented as a part of the basic network operating protocol by the link level protocol engine 58 (Fig. 3) of each basic node. The information contained in the first two applications mentioned above is slightly different than that shown in Fig. 11, because aspects of the receive and transmit sequence which are not related to the reconfiguration sequence of this invention have been eliminated from Fig. 11. A convention used below is that the steps or functionality in each flow chart is represented by reference numerals and each reference numeral is enclosed within parenthesis in the following text.

The reconfiguration sequence is entered from power-on reset (120) or timeout (122) of a lost-token timer (which is generally called the RECON timer in ARCNET terminology). When either of these events (120 or 122) occur, a reconfiguration burst is sent (124) to halt other activity on the network and thereby force reconfiguration of the token loop.

Each interface then proceeds to read (126) its own assigned network address (ID) and set (128) its next ID (NID) register equal to its own assigned ID. The timeout-based selection of the node which will begin by each node initializing (130) a timer (sometimes referred to as the "reconfiguration precedence timer") to a value of $(255-ID) \times 146\text{usec}$. The node which will time out first is the active node with the highest assigned ID in the basic range of addresses. During the timeout period established (130), each node monitors (132), the network for activity and, if activity is detected, terminates (136) its reconfiguration processing and goes to its

normal receive sequence to decode the detected activity. The normal receive sequence is described more fully in the applications for LAN WITH DYNAMICALLY SELECTABLE MULTIPLE OPERATIONAL CAPABILITIES and LAN WITH INTEROPERATIVE MULTIPLE OPERATIONAL CAPABILITIES.

5 If the timer expires (134) before any network activity is detected, no other node has started. Therefore, the node which experienced the timeout has the highest ID of any active basic node and must commence the reconfiguration sequence. This node proceeds to send (138) an ITT frame (attempted token pass) to the current value in its NID register, then wait (140) up to 78.4 microseconds for subsequent activity (which would be a token pass by a node receiving this token). The time period of 78.4 microseconds is specific to the ARCNET LAN, but in more general applicability is a time period which is greater than that time period during which the destination node would normally commence transmissions and is less than the time period established (122) by the lost token timer. If such activity is detected, the node enters (136) the normal receive sequence to monitor the activity. A discussion of the normal receive sequence is not included here, but is included in the first two mentioned applications.

20 If no such activity is detected, the basic node increments (142) the NID by 1, modulo 256, and returns to send (138) an ITT to this incremented NID value. This looping (138, 140, 142) will continue until the next active NID value is found, as evidenced by the node detecting (140) activity when the next active node in the token loop starts transmitting the token.

The transmit sequence ends (144) by entering this loop (138, 140, 142) as the normal means of passing the token. In the usual case, a token will be passed to the NID, after which activity, either
5 in the form of a token pass or the transmission of some other frame will be observed within the 78.4 microsecond period. In the case where the NID node has become inactive since the last time it received the token, the final loop (138, 140, 142) of the
10 reconfiguration sequence is executed until the next active node is addressed and the new NID of this node is established. This describes the partial reconfiguration mentioned above.

The present invention provides a
15 reconfiguration means to establish a token loop to include basic and enhanced nodes in the basic range of permissible addresses, and to allow additional enhanced nodes at an enhanced range of addresses
20 beyond the basic range of addresses to be included on the network, and to automatically and dynamically include the additional enhanced nodes at addresses in the enhanced range in a token loop which encompasses basic and enhanced nodes. The token
25 loop is established by executing a reconfiguration sequence of a reconfiguration means which and interoperatively combines with the basic reconfiguration sequence available from the basic
operating protocol of the basic nodes. The nature of the token loops established and the tokens passed
30 during this reconfiguration sequence are illustrated in Figs. 12 to 16. The flow of the reconfiguration sequence of the present invention is illustrated in Figs. 17A, 17B and 17C.

In Figs. 12 to 16, circles are used to show
35 basic nodes and rectangles are used to show enhanced nodes. The numbers within these rectangles or

circles are the addresses of the nodes. The arrows between the nodes, or pointing to null, are token passes or unsuccessful token pass attempts, respectively, during the reconfiguration sequence.

5 Plain arrows represent ITTs, arrows with cross-hatching represent XITTs, and double lined arrows represent XTOKs.

Fig. 12 shows a case where all nodes, both basic and enhanced, have addresses or IDs within the basic range of permissible addresses, i.e. less than 256. A 7-node network is illustrated with 5 basic nodes and 2 enhanced nodes. The reconfiguration precedence timeout occurs at basic node 245 and node 245 proceeds to send ITTs and get no response from nodes at IDs 245 to 255, and then, following the modulo-256 incrementation of NID, sends ITTs to and gets no response from the nodes at IDs 0 through 6. Thereafter the token is successfully passed to node 7. Node 7 commences sending ITTs to addresses 7 through 34 with no response, before successfully passing the token to node 35.

10
15
20

Node 35, being the first enhanced node to receive the token, will not have detected the transmission of any enhanced protocol token frames (XITTs or XTOKs) since the beginning of the network reconfiguration sequence, because only ITTs have been sent so far during this reconfiguration sequence. Node 35 will then set its NID to 256, the first address in the enhanced range, and commence sending XTOKs to IDs 256 through 2047. XTOKs will be sent because no basic nodes may have addresses greater than 255. Because no enhanced nodes in the enhanced range of addresses are present on the network there will be no response to these XTOKs.

25
30
35

Node 35 will then commence sending XITTs to the nodes at IDs 35 through 65 before successfully

passing the token to node 66. The process continues through nodes 66, 142, 174, and 199 before the token loop is finally completed with an ITT to 245. The enhanced node 174 does not send any XTOKs because it has detected the previous enhanced activity of the XTOKs sent by node 35. Once the token loop is established, the token is passed only to nodes 7, 35, 66, 142, 174, 199 and 245. No attempts to pass the token to other nodes occur until the next reconfiguration sequence.

Fig. 13 shows the situation where both basic and enhanced nodes have IDs below 256, and enhanced nodes have IDs at 256 and above. In the Fig. 13 case the reconfiguration precedence timeout occurs at the basic node at ID 245. The token loop initialization begins with the node at ID 245. Tokens (ITTs) are sent until the nodes at ID 7 and at ID 35, in that order, have been established as a part of the token loop. When the first enhanced node at ID 35, receives the token, it commences sending XTOKs. XTOKs are sent unsuccessfully to IDs 256 through 295, followed by a successful pass of the token to the enhanced node at ID 296, thereby establishing a direct token pass from a node below ID 256 to an enhanced node above ID 256. Node 296, of course, has recognized previous enhanced activity, since it has detected the previous XTOKs sent by node 35. Node 296 commences sending XTOKs from IDs 296 to 312 before passing the token successfully to node 313. Node 313 finds node 442, and 442 finds node 507 in a similar manner. Node 507, in this example, unsuccessfully sends XTOKs to IDs 508 through 2047; at which point, upon detection that the NID has reached its maximum value (2047), node 507 restores its NID to the basic ID of the previous basic node (PID) which previously

successfully passed the token plus 1, i.e. PID+1
(which is ID 36 in this example). Node 507
commences sending XITTs starting at ID 36. All the
enhanced nodes on the network monitor the PID value
and have their PID registers set to the ID of the
last basic node which successfully passed the token.
The PID register of each enhanced node is not
updated when token passes to addresses greater than
256 are detected.

The XITT transmissions to IDs 36 through 65
will, in this example, receive no response. When
the XITT is successfully sent to node 66, the token
returns to the basic range of permissible addresses.
The further initialization of the token loop to
nodes 142, 174, 199, 245 will take place in the
basic manner, with the token loop being complete
when the token returns to node 245. Again, node 174
does not attempt to send XTOKs, because it has
previously detected enhanced activity. Until the
next reconfiguration, the token is passed directly
to the nodes 7, 35, 296, 313, 442, 507, 66, 142,
174, 199 and 245, in the token loop thus
established.

Fig. 14 shows another case where both basic and
enhanced nodes have IDs below 256 and other enhanced
nodes have IDs at ID 256 and above. Fig. 14 is
similar to Fig. 13 except that the reconfiguration
precedence timeout occurs at enhanced node 251
rather than at a basic node. When an enhanced node
with an ID less than or equal to 255 times out
before any other node with an address in the basic
range, the enhanced node immediately sets its NID
equal to 256 and commences sending XTOKs to
addresses in the enhanced range of permissible
addresses. In this case, enhanced node 251 sends
XTOKs to IDs 256 through 295 before a successful

token pass occurs to node 296. Reconfiguration from node 296 to node 313 to node 442 to node 507 is equivalent to that in Fig. 13. Node 507 sends XTOKs to IDs 507 through 2047 and then resumes token passing with XITTs starting at PID+1. At this point PID is still 0 because it is initialized to 0 at the beginning of the reconfiguration sequence. Since no token passes below ID 256 have occurred up to this point, no updates of PID have occurred, so PID remains at 0. As a result, the sending of XITTs begins at basic ID 1 and proceeds to ID 6 with no response before an XITT to node 7 bridges the token loop back to the basic range of addresses. Node 7 will then send ITTs to find node 35. Node 35 will find node 66, etc., around through nodes 245 to 251. Enhanced nodes 35 and 174 have both detected previous enhanced activity and, therefore, restrict their reconfiguration activities to sending XITTs to the basic range of addresses starting at their own ID. Once the token returns to node 251, subsequent token loops will involve passing the token only to nodes 296, 313, 442, 507, 7, 35, 66, 142, 174, 199, 245 and 251, until the next network reconfiguration.

Fig. 15 represents the case of what is termed a composite token loop. In this case, there are only basic nodes at addresses in the basic range below ID 256, and there are only enhanced nodes in the enhanced range at ID 256 and above. As will be seen two reconfiguration precedence timeouts will occur before the token passing loop is completely established. The first node to time out will be the highest addressed node in the basic range of addresses, in this example basic node 245. Node 245 sends ITTs to IDs 245 through 255 and 0 through 6 with no response before establishing its NID at node 7. This is illustrated by a dashed line between

nodes 245 and 7 because of the fact that this link in the token loop is only present until the second reconfiguration precedence timeout occurs, as discussed below. Nodes 7, 66, 142, and 199 in this example then proceed to establish a conventional ITT token loop of the type previously described only through the basic nodes.

The nodes above ID 256 will not detect any enhanced activity because there are no enhanced nodes at IDs below 256, so eventually the reconfiguration precedence timer of the lowest-addressed of the active enhanced nodes (in this example node 296) will time out. Node 296 sets a flag in a decode zero (DZ) register to indicate that this node is to respond to a token addressed to ID 0, checks a reconfiguration burst (RB) register to determine that a reconfiguration burst needs to be sent, and in response to these conditions sends a reconfiguration burst to destroy the token that is circulating only to the nodes in the basic range of addresses. Node 296 then commences sending XTOKs with its own ID, polling IDs 296 to 312 before a successful XTOK transmission to node 313 is achieved. Node 313 establishes node 442 in the token loop, and in turn node 442 establishes node 507 in the loop. Node 507 will detect no responses in sending XTOKs to IDs up to 2047, and then commences XITT polling at PID+1. PID+1 will be equal to 1 because PID was set to 0 when the node 296 sent the reconfiguration burst and destroyed token. Accordingly, XITT polling will begin at ID 1 and increment up to ID 6 before node 7 successfully responds to an XITT from node 507. The token loop will then be re-initialized between nodes 7, 66, 142, 199, and 245 in the standard manner. Node 245 will proceed to send ITTs to IDs 245 through 255

before the modulo-256 incrementation of NID results in an ITT to ID 0. The ITT to node 0 will be decoded by node 296 because node 296, which incurred the second reconfiguration precedence timeout, will have its decode zero (DZ) register set to recognize a response to a token addressed to ID 0. This token pass completes the token loop with the ID 0 pseudo-address, thereby completing closure of the composite token loop. The pseudo-address 0 is only detected by the lowest addressed active node in the enhanced range (node 296 in this example) and only for ITTs and XITTs.

A composite token loop is necessary because with only basic nodes below ID 256, no enhanced node is available in the basic range to send an XTOK directly to an enhanced node in the enhanced range of addresses. ID 0 is preferred for the pseudo-address because the use of any other ID value would use an address otherwise assignable to an actual node. Address 0 is acceptable for use as a token destination in the composite token loop, because no specific node may actually exist at that address, and because under the network operating protocol the use of address 0 for broadcasts applies only to the transmission of data packet frames and not to token frames. Accordingly the other nodes, except that enhanced node with its DZ register set, will not recognize a token pass to ID 0. Once the initialization of the token loop is completed when the token returns to node 296 after the timeout by node 296, subsequent token loops involve only the nine active nodes 296, 313, 442, 507, 7, 66, 142, 199 and 245, until the next network reconfiguration occurs.

Fig. 16 shows the case where only enhanced nodes exist on the network and all of them have

addresses in the enhanced range. In this case, there is only one reconfiguration precedence timeout and that is by the lowest assigned enhanced node ID, in this case node 296. Node 296 sends XTOKs upward
5 from its ID, and ultimately the token passing loop to nodes 313, 442, and 507 in the previously described manner. After node 507 accepts the token, it then sends XTOKs with no response up to ID 2047 and then commences sending an XITT to PID+1, which
10 will be 1. Node 507 will continue to send unsuccessful XITTs to all IDs 1 through 255. The modulo-256 incrementation of NID will then result in an XITT to 0. Because node 296 incurred the original timeout, its DZ register will be set to
15 respond to tokens addressed to ID 0, and node 296 will decode the node 0 pseudo-address, completing a token loop from 507 to 296 through pseudo-address 0. Once the initialization of the token loop is completed when the token returns to node 296, the
20 token loop involves only the four active nodes 296, 313, 442 and 507, until the next network reconfiguration occurs.

An optimization to the Fig. 16 situation can be made, when deemed to be a sufficiently common case, to allow enhanced nodes to resume XTOK polling at ID
25 256 to complete the token loop totally with XTOKs, without consuming the added time during network reconfiguration to send the null XITTs to IDs 1 through 255 and without consuming the added time
30 during network operation to send the XITT to 0, either of which will take longer than sending an XTOK directly from 507 to 296. The optimizing approach requires each enhanced node to determine that a XTOK has been unsuccessfully passed to ID
35 2047 when the RB register indicates that no reconfiguration burst has been sent. Under these

conditions XTOK polling will resume at ID 256. This optimization approach adds complexity to the enhanced reconfiguration sequence shown in Figs. 17A, 17B and 17C, described below. Ordinarily, networks having both enhanced and basic nodes below ID 256 will be far more common than networks with only nodes above ID 256, so the added complexity of the reconfiguration sequence will usually be unjustified.

Under very heavily loaded operating conditions, a basically configured LAN will pass the token with sufficient speed that the token will return to each node well before the lost token timer times out. The margin of extra time is sufficient so that the added enhanced nodes can also participate in network activities when the token loop is extended to include the added enhanced nodes. Under a worst case analysis when a maximum number of basic and enhanced nodes are active and each is communicating the maximum length messages, the lost token time out period of the nodes may need to be extended. In reality the worst case situation would virtually never occur, since a typical heavily loaded situation occurs when less than about 5% of the nodes are communicating messages. In any event, if the worst case situation did occur, a network reconfiguration would simply occur, and the probabilities are that, after reconfiguration, the worst case situation would have changed due to the dynamically changing use characteristics of the network. Another approach, rather than lengthening the lost token time out period, is to break one larger network into two or more smaller segments, since a network which incurs that much loading is generally an inefficient network.

The functionality of the enhanced reconfiguration sequence of the present invention is preferably achieved by means of the network protocol controller 70 (Fig. 4) in each enhanced node. Since the basic reconfiguration sequence of a basic node has been previously described in conjunction with Fig. 11, only the enhanced reconfiguration sequence associated with each enhanced node will be described below. The enhanced reconfiguration is interoperative with the basic reconfiguration sequence, so the protocol implemented by the link level protocol engine 58 (Fig. 3) of each basic node need not be modified in order to simultaneously accommodate the enhanced nodes in the enhanced range of addresses on a LAN.

The enhanced reconfiguration sequence begins, as is shown in Fig. 17A, in a similar manner to the normal reconfiguration sequence, with power-on reset (150) or the lost-token timer timeout (152) causing the transmission (154) of a RECON burst to terminate activity on the network and force a complete reconfiguration of the token loop. Each enhanced node then reads (156) its own ID, which in this case can be any value from 1 to 2047. The enhanced node then sets (158) several internal state variables to zero in similarly designated registers including: PID, the address of the previous basic node to hold the token; KA, a flag indicating whether or not any enhanced activity has been detected during this reconfiguration sequence, used to determine whether other enhanced nodes have received the token; and DZ, a flag which indicates whether this node is to decode token passes to network address 0, used to complete the token loop between basic nodes and enhanced nodes, as discussed above in conjunction with Figs. 15 and 16. These initial state variables

of zero indicate: that there is no valid basic node at which the token was last held (FID=0); that there has been no enhanced activity detected thus far during this reconfiguration sequence (XA=0); and
5 that token passes to address 0 should not be decoded (DZ=0). The enhanced node then initializes its NID register to its own ID value. The capability table is then initialized (160) such that all IDs less than or equal to 255 are marked as being basic nodes
10 and all IDs greater than 255 are marked as being enhanced nodes. The initial default conditions are thus established, which are assumed later in the sequence.

Process flow then bifurcates (162) based on
15 whether this node has an assigned ID greater than 255 or less than or equal to 255. In the case where the ID is less than or equal to 255, meaning that the enhanced node is present in the basic range, the enhanced node must participate in a conventional
20 token loop initialization, as well as providing the gateway to the enhanced range of addresses for other enhanced nodes if no other enhanced node has done so previously.

When the ID of the enhanced node is less than
25 or equal to 255, the reconfiguration precedence timeout period (164) is the same ($146\text{usec} \times (255 - \text{ID})$) as for basic nodes. If network activity is detected (166) before the timeout expires (168), the receive sequence shown in Fig. 17C is entered. If
30 this timeout expires (168) before any network activity is seen, the node timing out is the highest addressed active node below 256. This node, therefore, will start to establish the token loop.

If there are any enhanced nodes below ID 256,
35 it is advantageous to have one of them pass the token to the lowest ID active enhanced node above ID

255 to both minimize the reconfiguration time and maximize the efficiency of the resulting token loop. This is accomplished in the preferred embodiment by having the first enhanced node to receive the token during the reconfiguration sequence commence passing tokens to the enhanced range of addresses, starting at ID 256. If the node which times out is an enhanced node, which is the example contemplated by Fig. 14, the token loop initialization begins by an attempted token pass to ID 256. So, when the first node in the basic range to timeout is an enhanced node, it sets (170) its NID to 256, sets (170) the XA flag to 1 to handle the case that this is the only enhanced node on the network, and enters the polling sequence, shown in Fig. 17B.

The polling sequence commences, as is shown in Fig. 17B, with looking up (172) the capabilities of the node at the next ID (NID) in the capability table. If the NID is that of an enhanced node, an XTOK is sent (174) to the NID. Of course, for addresses above 256, the capabilities are, by default, enhanced, as was established during initialization (160, Fig. 17A) of the capability table. The enhanced node sends (174) an XTOK to the NID, listens (176) for activity on the network medium within 78.4 microseconds, and if activity is detected, proceeds to the receive sequence shown in Fig. 17C to process the activity. If no activity is detected (176) within 78.4 microseconds, the NID is tested (178) to determine if it has reached 2047. If the NID has not reached 2047, which in the preferred embodiment is the highest allowable ID in the enhanced range of addresses, the NID is incremented (180) and the polling sequence is repeated.

If the NID has reached 2047, a test is then made (182) to determine how to resume polling for addresses below 256. This resumption is based on the assigned ID of this particular node. If the node has an ID less than or equal to 255, as is the case in Fig. 14, normal polling for basic addresses is entered by setting (184) NID equal to ID and commencing with the normal XITT polling (187). On the other hand, if the ID is greater than 255 (182), then the NID is set equal to PID plus 1 (186), and normal XITT polling commences (187) with NID equal to PID+1. If there has been no activity by basic nodes, PID will be equal to 0, so the token polling of the basic range of addresses will begin at ID 1. Otherwise polling of basic addresses will commence at the last polled basic address plus 1.

Token polling to basic nodes by an enhanced node is by sending XITTs (187). XITTs are sent after determining (172) that the NID is that of a basic node (or a node of unknown type) from the capability table. After sending (187) the XITT, a determination is made (188) whether activity is present on the network medium within 78.4usec. If so, the receive sequence shown in Fig. 17C is entered. If not, the NID value is incremented (190) by 1, modulo 256, and the process flow (172, 187, 188) is repeated until the next active basic node is found. Whenever activity is detected (176, 188, Fig. 17B) after a token pass or while waiting for activity in the idle loop (192, Fig. 17C) during normal operation (described more completely in the application for LAN WITH INTEROPERATIVE MULTIPLE OPERATIONAL CAPABILITIES) the receive sequence shown in Fig. 17C is entered.

The first determination made (194) in the receive sequence (Fig. 17C) is whether the received

frame is a basic frame or an enhanced frame. This determination is made by checking (194) the third symbol of the frame. If this symbol is equal to a non-standard inverse-phase dipulse or enhanced signal element as described above and more fully in the applications for LAN WITH INTEROPERATIVE MULTIPLE OPERATIONAL CAPABILITIES and MULTIBIT AMPLITUDE AND PHASE MODULATION TRANSCEIVER FOR LAN, an enhanced frame is indicated. The XA flag is set (196) equal to 1 to indicate that enhanced activity has been detected on the network since network reconfiguration began. Setting the XA flag equal to 1 will prevent the enhanced node where XA has been set to 1 from attempting to initiate token polling of addresses above 255. Next, the header of the enhanced frame is received and decoded (198). If the header of the enhanced frame does not indicate an XTOK, the frame is processed (200) in a manner appropriate for that frame-type in the enhanced protocol. Frames which are not tokens are not part of the reconfiguration sequence of this invention, but are discussed more fully in the first two previously mentioned applications. If the frame is an XTOK, the DID of the XTOK is saved (202), the XTOK flag field is decoded (204), the rate and status capabilities of the enhanced node sending the XTOK are recorded (206) in the capability table based on the value of FID, which identifies the node to last receive the token, and then it is determined (208) whether the saved DID is greater than 255. If not, PID is set (210) equal to the saved DID; if so, PID is not updated, and a determination is made (212) whether the XTOK is addressed to this node. If not, the sequence enters (214) the idle loop, ending the reconfiguration process, if any, for this node.

If the XTOK is addressed to this node (212), the XA flag is tested (216). If the XA flag is equal to 0, meaning that the token has been received by this node before any enhanced activity was detected or generated on the network, the node sets (170, Fig. 17A) NID equal to 256, sets (170) XA equal to 1 to prevent an infinite loop in the case that this is the only enhanced node on the network, and enters the XTOK polling sequence (Fig. 17B) for the enhanced range of addresses. If XA is equal to 1, then the sequence for receipt of the token for conventional network activity is entered. The sequence involves restarting (218) the lost-token timer, determining (220) if there is a pending transmission, and, if so, entering the transmit sequence (222), which involves sending the packet with any necessary inquiries and acknowledgements. After executing the transmit sequence (222), of after detecting (220) that there is no pending transmission, the token is passed (Fig. 17B) based on the currently established NID value.

If the third symbol test (194) during the receive sequence (Fig. 17C) determines that a basic frame is received (based on there being a normal-phase dipulse in the third symbol as described more fully in LAN WITH INTEROPERATIVE MULTIPLE OPERATIONAL CAPABILITIES and MULTIBIT AMPLITUDE AND PHASE MODULATION TRANSCEIVER FOR LAN), the basic header is received and decoded (224). One aspect of this decoding is that an ITT or XITT will be accepted not only when addressed to the assigned ID of this node, but also when addressed to ID 0, in the case where the DZ flag is equal to 1. If the incoming frame is neither an ITT or a XITT, then it is processed (226) in the manner for a basic type of frame. If the frame is either an ITT or a XITT, the

DID is saved (228 or 230, respectively). In the case of an XITT the flags are decoded (232) and recorded (206) in the capability table at the PID. The saved DID is tested as being greater than 255 and, if not, PID is set (210) equal to the saved DID. On the path of a received ITT, the test (208) for the ID is not needed, because the ID cannot be greater than 255 in an ITT. In all cases, it is determined (212) if the token is addressed to this node. Of course, if the DZ flag is equal to 1 then tokens addressed to ID 0 are also treated as being addressed to this node. If the token is addressed to a different node, an exit is made to the idle loop (214). If the token is addressed (212) to this node, XA is tested (216) and processing proceeds as previously discussed.

At the bifurcation (162) following the initialization sequence (156, 158, 160), shown in Fig. 17A, the process flow for IDs greater than 255 begins. The RB flag is set (234) to 0. The RB flag is used in determining whether a supplemental reconfiguration burst is needed. The reconfiguration precedence timer is initialized (236), but, in this case, to a different value than for a basic node. The enhanced reconfiguration precedence timeout is set to $146\text{usec} \times \text{ID}$ rather than to $146\text{usec} \times (255 - \text{ID})$ for a basic node. This means that the timeouts for all enhanced nodes in the enhanced range of addresses are longer than any basic address timeouts, since the longest basic address timeout is $146\text{us} \times 254$ and the shortest enhanced address timeout is $146\text{us} \times 256$. Consequently the first enhanced node in the enhanced address range to time out will be the active node with the lowest assigned address greater than 255. For nodes in the basic address range the first

timeout occurs at the active node with the highest assigned address less than or equal to 255.

5 Once the reconfiguration precedence timer has been started (236), two separate tests occur during an activity monitoring loop. If enhanced activity is detected (238), as determined by the third symbol of the frame, the reconfiguration timeout sequence is ended, because activity from an earlier timeout of an enhanced node has been detected. In this case
10 entry is made to the receive sequence (Fig. 17C) immediately after the third symbol test (194) for processing of an enhanced frame. If basic activity is detected (240, Fig. 17A), the RB flag is set (242) to 1, but the reconfiguration precedence
15 timeout sequence is not terminated. There is no need to decode this detected basic activity during network reconfiguration, since basic activity cannot pass the token to a node with an address greater than 255. The RB flag will only be equal to 1 at
20 the time when the lowest-addressed active enhanced node above 255 times out, in the case where all active nodes below 256 are basic nodes, and, therefore, no enhanced node is available in the basic range of addresses to perform XTOK polling of
25 IDs 256 and above.

 If the reconfiguration precedence timer (236) times out (244) for a node above 255, this indicates that either no activity has been detected (238, 240) during the reconfiguration precedence timeout
30 interval (236), or that only standard (basic) activity has been detected. In either of these cases, it will be necessary to create the composite token loop -- where the last token pass connects back to enhanced nodes above ID 255 by a token pass
35 to ID 0. To enable this last token pass in a composite token loop to ID 0, the DZ flag is set

(246) to 1 to enable the enhanced node timing out to decode ITTs and XITTs addressed to ID 0. The RB flag is next tested (248). The RB flag will be 0 based on initialization and entry to this branch.

5 If only basic activity and no enhanced activity has been detected (240, 238 respectively), RB will be set to 1. If RB is 1, a supplemental reconfiguration burst is sent (250) by this node to destroy any partial token loop which may exist at

10 any nodes with IDs below 256. Otherwise, no reconfiguration burst is sent by this node. In either case this node then sets (252) NID=ID and enters the polling sequence (Fig. 17B) which, because the ID is greater than 255, will begin with

15 the enhanced path by sending (174) XTOKs to successively higher addresses in the enhanced range.

When all of the active nodes below 256 are basic nodes, there is no enhanced node to initiate polling at ID 256 and above. The token in the token

20 loop that has been established in the basic range of node addresses must be destroyed and, a complete token loop which includes the enhanced nodes in the enhanced address range must be generated, beginning with the lowest active enhanced address node and

25 then proceeding up through successive enhanced address IDs up to 2047 and then back around through addresses below 256.

It is possible, although not likely, that the second reconfiguration precedence timeout in a

30 composite token loop case analogous to Fig. 15 could occur in certain cases before the basic token loop segment through the basic range is fully established, based on the operational sequence shown in Figs. 17A, 17B and 17C. In such a worst case

35 situation of basic nodes below ID 256, the specified second reconfiguration precedence timeout period

will expire first. No operational problem results because the consequences of this worst case will still result in proper network operation. Any token loop segment for nodes below ID 256, whether
5 partially or fully initialized, will be destroyed by the supplemental reconfiguration burst in the process of establishing the composite token loop.

A worst case situation occurs when the only active nodes below 256 are at very low addresses and
10 the first node to timeout is a basic node. For example, consider only two nodes below ID 256 at IDs 1 and 2, and enhanced nodes at IDs greater than 255, including one at ID 256. The node at ID 1 is an enhanced node, and the node at ID 2 is a basic node.
15 The timeout at ID 2 will be 146×253 microseconds, which is only 3×146 microseconds before the timeout of the lowest possible node above ID 255 -- that of the node at ID 256. However, the token polling of IDs starting at ID 2 and going up through ID 255
20 will take nearly 30 milliseconds, which is much longer than the period before node 256 will time out because it sees only basic activity. Node 256 will generate the second reconfiguration burst and take over token loop initialization. Node 256, or some
25 similarly low-addressed enhanced node, takes over token loop initialization by polling upward with XTOKs, thereby initializing a token loop through all enhanced nodes above 256. Since no enhanced activity was detected below ID 256, which would have
30 averted the second timeout, the highest addressed active node in the enhanced address range commences XITF polling at ID 1, which will, ultimately, close the token loop through the active nodes in the basic address range. The enhanced activity in
35 establishing the token loop above ID 255 will prevent XTOK polling by any enhanced nodes

encountered in the basic address range and the token loop will finally be closed by the highest addressed node in the basic address range sending an ITT to ID 0 to create a composite token loop.

5 The network will initialize to a valid token loop even though there are cases where, had the initialization below 256 been allowed to complete, a bifurcation at an enhanced node would have prevented a need for the second reconfiguration burst. The process flow as shown in Figs. 17A, 17B and 17C is an effective compromise between the possibility of taking even longer to initialize and the occasional spurious or unnecessary worst case creation of a composite token loop.

10 By making the timeout on the branch (162, Fig. 17A) for IDs greater than 255, equal to ID X 146us plus a constant (approximately 61 milliseconds for ARCNET to completely initialize in the basic address range), all possible worst case token loop initialization situations below ID 256 would take place, before any timeouts above ID 256 take place while still providing for the orderly establishment of a token loop for the enhanced nodes at IDs above 256. However, the constant (e.g. 61ms) bias would significantly increase the time to reconfigure the network with either a composite token loop or with only nodes at IDs greater than 256 active on the network. Adding the constant bias would result in no substantial perceived benefit, since there is no functional disadvantage or operational requirement that requires the branch (162) for IDs greater than 255 to prevent the second timeout (236, 244) from ever preceding a timeout (164, 168) on the branch for IDs less than 256. Because of the use of XTOKs in the polling of addresses 256 to 2047, and the reduced turnaround time which is possible with the

enhanced signaling technique, described more completely in LAN WITH DYNAMICALLY SELECTABLE MULTIPLE OPERATIONAL CAPABILITIES and LAN WITH INTEROPERATIVE MULTIPLE OPERATIONAL CAPABILITIES, the full polling of the enhanced address range takes only about 20-1/2 milliseconds and overhead of about 37% is added to the time it takes to reconfigure in the non-composite token loop situation. Full polling of the enhanced address range is considerably shorter than a bias of 61 milliseconds. Of course, in the worst case composite token loop situation that can validly occur, as much as 300 milliseconds could be lost in timeouts. Therefore, a total reconfiguration time could get as high as about 380 milliseconds, which is considerable longer than the current ARCNET worst case of 61 milliseconds, but, since reconfigurations only occur when initializing or re-initializing the network, this is not an operational restriction that is likely to be noticeable to users.

The presently preferred embodiment of the present invention and its improvements have been described with a degree of particularity. It should be understood, however, that previous description has been made by way of preferred example and that the scope of the present invention is defined by the following claims.

WHAT IS CLAIMED:

1. A directed token local area network (LAN), comprising:

5 a plurality of at least three nodes,
each node having a unique address or ID, at least
one of the nodes being a basic node, all basic nodes
having IDs which fall within a basic range of
permissible IDs, at least one of the nodes being an
enhanced node, each enhanced node having an ID which
10 falls within an enhanced range of permissible IDs or
the basic range, all of the IDs in the enhanced
range being greater than those permitted in the
basic range, at least one of the enhanced nodes
having an ID in the enhanced range; and

15 reconfiguration means associated with all
of the nodes and operative for passing tokens to
locate the next active node and automatically
establish all of the active nodes in a token loop
according to the IDs of the nodes to enable all of
20 the active nodes to receive tokens during normal LAN
operation, said reconfiguration means executing a
standard reconfiguration sequence for locating and
establishing the active nodes in the basic range of
IDs in a basic segment of the token loop, said
25 reconfiguration means also executing an enhanced
reconfiguration sequence for locating and
establishing the active nodes in the enhanced range
of IDs in the other enhanced segment of the token
loop, and the execution of the enhanced
30 reconfiguration sequence commences with the first
enhanced node to receive the token in conjunction
with execution of the basic reconfiguration
sequence.

2. A LAN as defined in claim 1 wherein:
the execution of the enhanced