



(19) **United States**

(12) **Patent Application Publication**
Majoor

(10) **Pub. No.: US 2002/0029154 A1**

(43) **Pub. Date: Mar. 7, 2002**

(54) **MECHANISM AND METHOD FOR
DYNAMIC QUESTION HANDLING
THROUGH AN ELECTRONIC INTERFACE**

Publication Classification

(51) **Int. Cl.⁷ G06F 17/60**
(52) **U.S. Cl. 705/1**

(75) Inventor: **Johannes W.F. Majoor**, San Francisco,
CA (US)

(57) **ABSTRACT**

Correspondence Address:
LYON & LYON LLP
633 WEST FIFTH STREET
SUITE 4700
LOS ANGELES, CA 90071 (US)

An improved mechanism for intelligently presenting questions through an electronic interface is disclosed. A rule server receives information identifying a set of answers from a user corresponding to a first set of questions presented to that user. In response, analysis is performed upon the set of received answers. A first determination is then made of whether any questions from the first set of questions need to be answered again based upon rules governing responsive answers. Next, a second determination is made of the composition of a second set of questions. The first determination and second determination may be performed at a rule server or an application tier. The second set of questions is then presented to a user through an electronic interface. Suggested answers to the second set of questions may also be presented to the user through an electronic interface contemporaneously with the second set of questions.

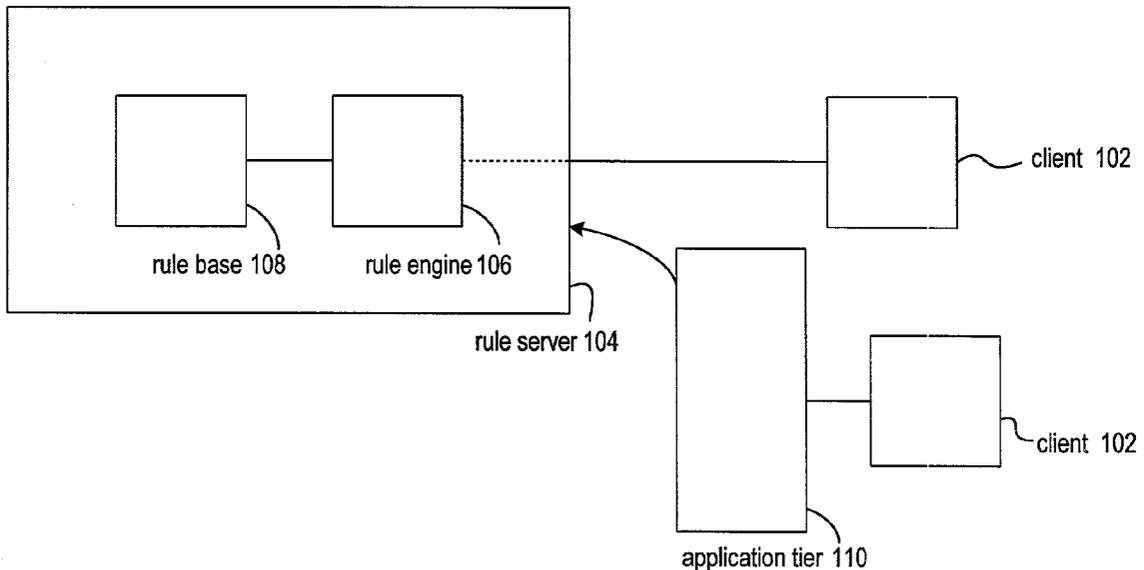
(73) Assignee: **HNC Software, Inc.**

(21) Appl. No.: **09/949,321**

(22) Filed: **Sep. 7, 2001**

Related U.S. Application Data

(63) Non-provisional of provisional application No. 60/231,161, filed on Sep. 7, 2000.



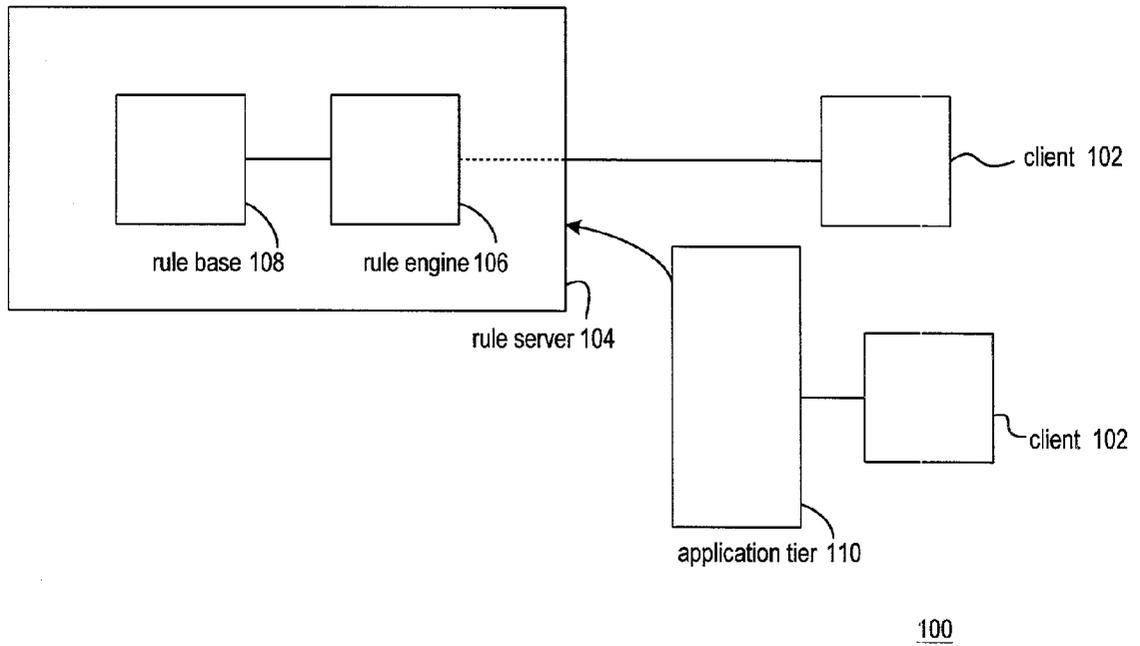


FIG. 1

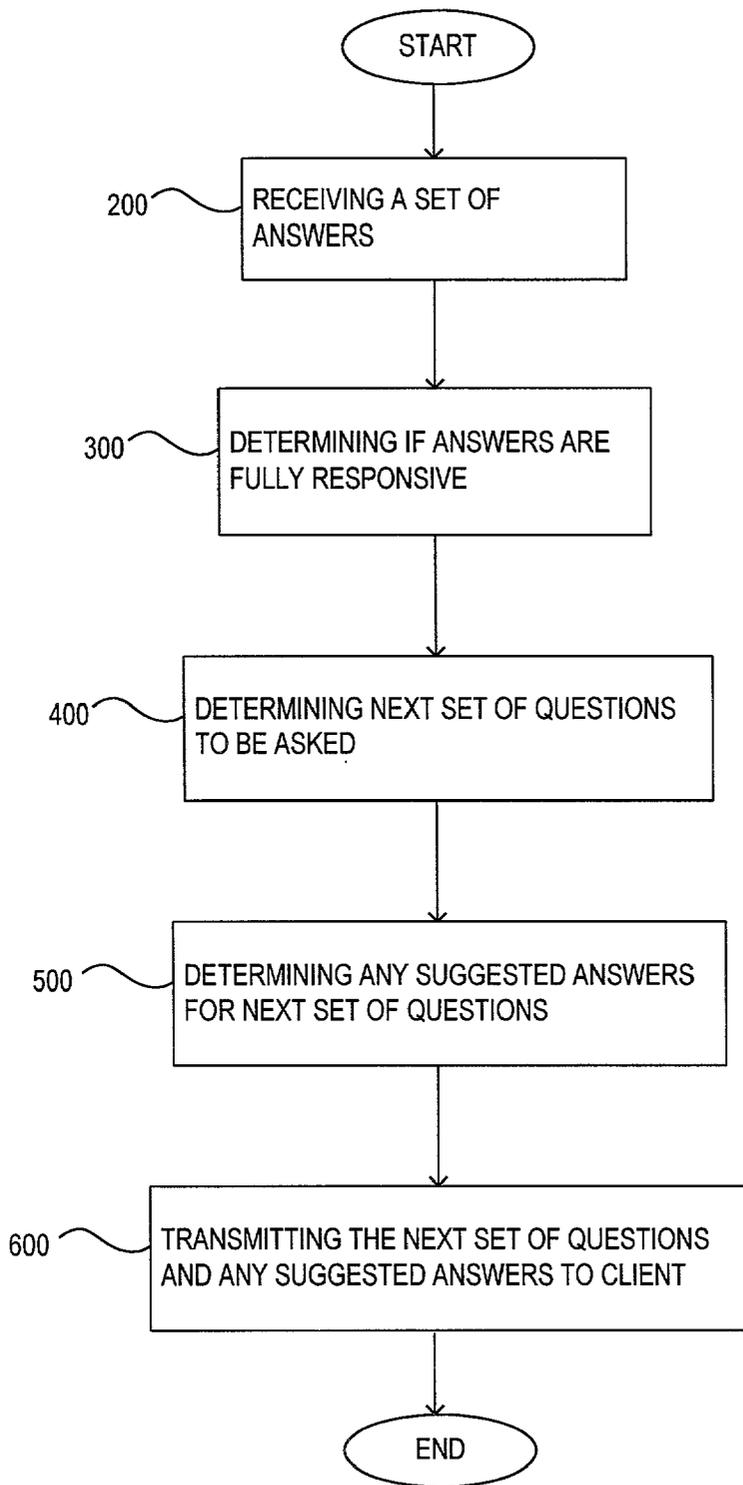


FIG. 2

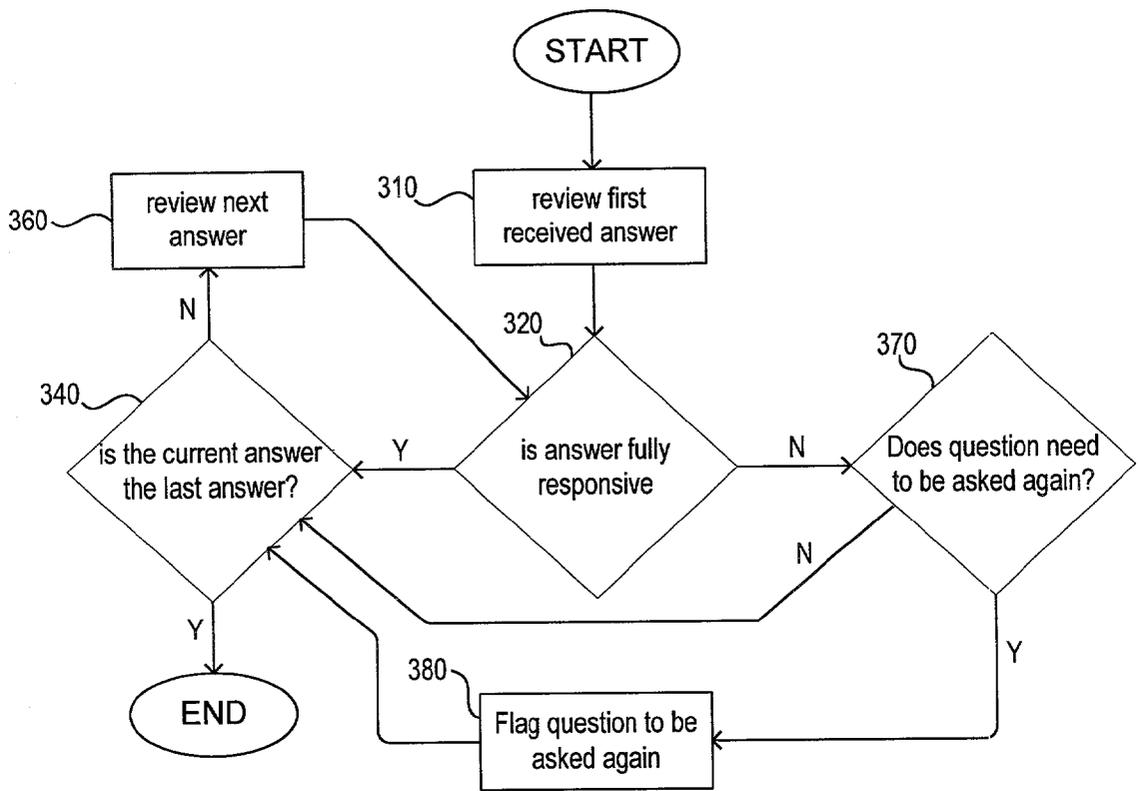


FIG. 3

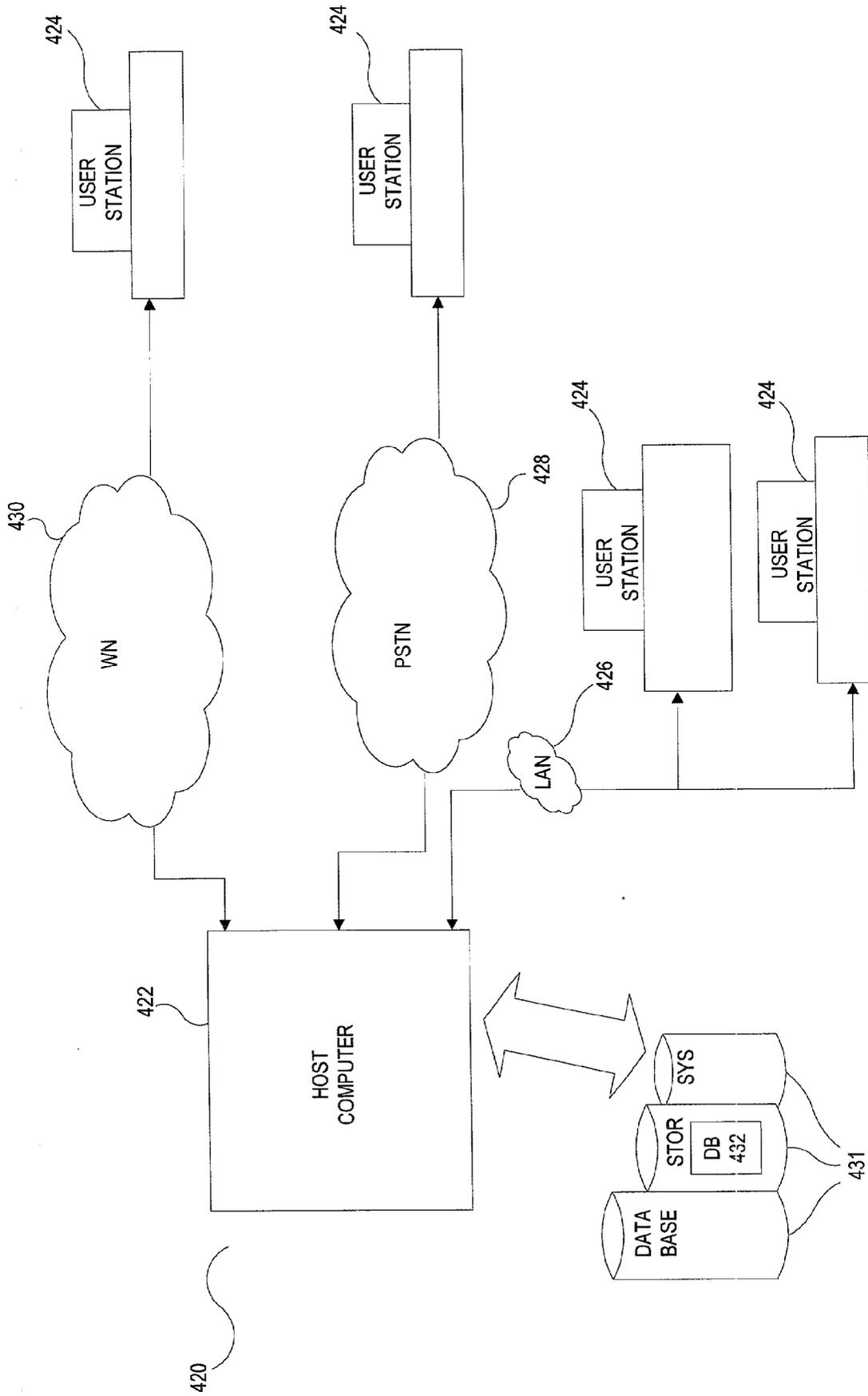


FIG. 4

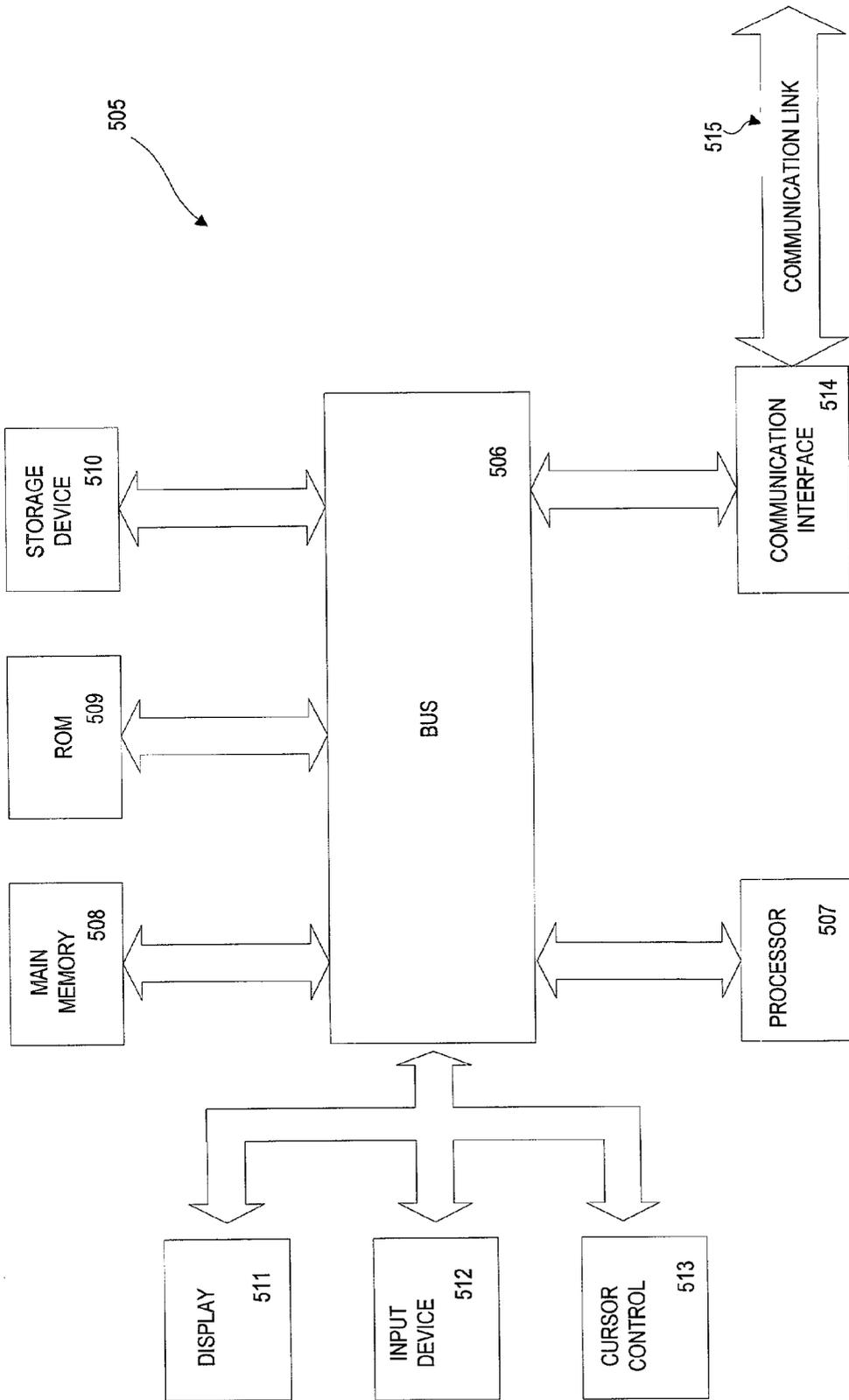


FIG. 5

MECHANISM AND METHOD FOR DYNAMIC QUESTION HANDLING THROUGH AN ELECTRONIC INTERFACE

RELATED APPLICATION DATA

[0001] This application claims priority to U.S. provisional application Serial No. 60/231,161 filed on Sep. 7, 2000, which is incorporated herein by reference in its entirety.

BACKGROUND AND SUMMARY

[0002] This invention relates generally to computer systems that utilize rule-based mechanisms to facilitate or implement an application or business process. A type of system that often uses a rules-based approach is an expert system or knowledge-based system, which attempts to mimic the thought process that an expert would utilize to address tasks in a particular field. Computer systems that employ rules-based mechanisms, e.g. expert systems, are employed in a wide variety of settings, such as computer systems that bank loan officer uses for guidance in approving and rejecting loan applications. Other examples of disciplines in which rules-based systems are used include automobile repair, medical diagnosis, oil exploration, financial planning, chemical analysis, surgery, weather prediction, computer repair, computer system configuration, operation of nuclear plants, interpreting government regulations, tax preparation, as well as many others.

[0003] A typical rules-based approach employs one or more sets of rules ("rulesets") that define actions or results that occur upon specified input values or parameters. A rule server is a computing device that performs the functionality of applying a set of rules to values or parameters. Rule servers may solicit these values or parameters by posing questions, either to a human user or to another computer entity. An exemplary rules-based product embodying this approach is the Blaze Advisor™ product available from HNC Software, Inc. of San Diego, Calif.

[0004] While a rules-based approach is useful, there are drawbacks with respect to efficiency and ease of use. Computer systems that have implemented standardized forms, such as a loan application, often present questions without intelligence or forethought. In other words, new questions are presented that do not consider prior answers to past questions. Additionally, questions may be automatically presented in static question sets consisting of too few or too many questions. For example, related questions may be asked sequentially, one after another, on different screens, or an overwhelming number of unrelated questions may be presented on the same screen. As questions are presented without regard of their relevance to the user, many presented questions are inappropriate, which wastes time and computer resources. Moreover, if all the answers to a set of questions in a transaction are not fully responsive, the transaction may be aborted, which further wastes time and computer resources.

[0005] Therefore, it is desirable to implement an improved method and mechanism for question handling through an electronic interface. In aspects of the invention, questions are presented by a rule server based upon prior responses or knowledge of a user. In another aspect of the invention, the rule server dynamically determines the number and arrangement of questions presented. In a further aspect of the

invention, the rule server suggests possible answers to the posed questions. Embodiments of the invention enable a transaction to avoid aborting when a complete set of responsive information used in the processing of that transaction is not available. Note that the principles disclosed herein are equally applicable to perform question handling in other types of computer systems. Further details of aspects, objects, and advantages of the invention are described in the detailed description, drawings, and claims.

DETAILED DESCRIPTION OF THE DRAWINGS

[0006] The present invention is illustrated by way of example, and not by way of limitation, in the figures of the accompanying drawings and in which like reference numerals refer to similar elements and in which:

[0007] FIG. 1 is an illustrative example of an architecture that can employ one embodiment of the present invention;

[0008] FIG. 2 is a flow chart illustrating the high level operational overview of a rules-based system 100 according to an embodiment of the invention;

[0009] FIG. 3 is a flow chart illustrating the steps of determining if answers are fully responsive and valid according to an embodiment of the invention;

[0010] FIG. 4 is a diagram of a system architecture with which the present invention can be implemented; and

[0011] FIG. 5 is an additional diagram of a system architecture with which the present invention may be implemented.

DETAILED DESCRIPTION OF EMBODIMENT(S)

[0012] With reference to FIG. 1, there is shown an example of an architecture that can be employed for a rules-based system 100. In the system shown in FIG. 1, a client 102 of a computer-based application connects to a rule server 104 to perform a rules-driven operation. The client 102 may be either a local or a remote client that connects via a network connection. A particular human operator, or user, may use one or more client 102. A client 102 may be in data connection with a rule server 104 or an application tier 110. For example, client 102, is in data connection with rule server 104, and client 102₂ is in data connection with application tier 110. An application tier 110 is one or more software programs that are functionally coordinated with one another. For example, an application tier may comprise one or more CRM or ERP software systems. The rule server 104 comprises or connects to a rulebase 108 that contains one or more rulesets. A rule engine 106 is the mechanism that applies a selected ruleset to input received from client 102 to generate a set of results. A ruleset is a set of encoded rules that is applied by a rule engine 106 to input in generating a result set.

[0013] The rule server 104 includes processing entities to perform work on behalf of clients 102. Various types of processing entities are usable in the invention, such as processes, threads, or agents. Such processing entities are referred to herein as rule service agents. The rule server 104 can be configured to provide a pool of rule service agents to handle service requests in parallel from multiple clients 102. Multiple rule service agents are available to handle service

requests in parallel from multiple clients **102**. Even with only a single rule service agent available, concurrent client requests can be handled using a queuing arrangement for client requests. In an embodiment, a single dedicated rule service agent performs service requests on behalf of a single client. In an alternate embodiment, a single rule service agent can process service requests on behalf of multiple clients.

[**0014**] While only one client **102** is depicted in **FIG. 1** for clarity, it should be understood that multiple clients **102** may be connected to the rule server **604**. Similarly, the rule server **104** may be comprised of multiple rule engines **106** and multiple rule bases **108**, but for clarity, only one rule engine **106** and one rule base **108** is shown in **FIG. 1**.

[**0015**] The rule server **104** may be implemented using any well-known programming language. In one embodiment, the rule server **104** is implemented using an object-oriented programming language that can be compiled on any number of machine formats. Accordingly, the rule server **104** may be deployed in a variety of computing environments.

[**0016**] The high level operational overview of a rules-based system **100** according to an embodiment of the invention will now be presented with reference to **FIG. 2**, which is a flow chart illustrating the same. The rule server **104** receives a set of answers from a user responding to a set of asked questions **200**. After receiving a set of answers from a user, the rule server **104** determines if the received answers to the set of asked questions are fully responsive **300**. For example, a question might be incorrectly answered, partially answered, or not answered at all.

[**0017**] Subsequently, the rule server **104** determines the next set of questions to be asked to the user **400**. In other words, the next set of questions is dynamically generated based upon prior received answers. Next, the rule server **104** determines any suggested answers to the next set of questions to be asked **104**. Finally, rule server **104** transmits the next set of questions and any suggested answers to a client associated with the appropriate user **600**. Having thus described the operation of rule server **104** from a high level, each step illustrated in **FIG. 2** will now be discussed below in greater detail.

[**0018**] The process of receiving a set of answers from a user responding to a set of asked questions **200** may be performed by a variety of mechanisms. In one embodiment, the rule server **104** employs defined application programming interface, or API, which defines procedures which facilitate input to and output from the rule server **104**. As the invention may be implemented using a variety of clients, application tiers **110**, and rule servers **104**, it may be desirable to employ an API posing questions or receiving responses because the API can be modified to accommodate a variety of implementations. For example, it may be desirable to employ an API that accommodates integration with a legacy system.

[**0019**] According to embodiments of the invention, the functional steps illustrated in **FIG. 2** may be performed within a single transaction. Rules defined in rule base **108** contain procedures used to continue processing a transaction when information needed to complete the execution of the transaction is not available. For example, if a transaction requires a complete set of address information, and trans-

action only has access to an incomplete set of address information, then rules defined in the rule base **108** are used to continue processing the transaction such that it is not aborted. According to embodiments of the invention, the transaction may be processed with an incomplete set of information, the required information may be obtained from storage, or a question may be posed to the user to obtain the needed information.

[**0020**] According to embodiments of the invention, the interaction between clients **102** and the rule server **104** is optimized to enhance performance. As a transaction may be processed with an incomplete set of information, a client **102** may answer a set of questions in full, partially, or not at all, and the rule server **104** may process the transaction without interruption. By avoiding a separate interaction between the rule server **104** and each client **102** for each question, network traffic and the risk of complications within the system **100** is reduced.

[**0021**] The process of determining if the received answers to the set of asked questions are fully responsive **300** will now be explained with reference to **FIG. 3**, which is a flow chart illustrating the same. The rule server **104** reviews the first received answer in the set of received answers **310**. The rule server **104** then determines if the answer is fully responsive **320**. For example, the answer may be answered incorrectly, partially answered, or not answered at all. Rules defined in the rule server **104** determine what constitutes a responsive answer. These rules may be arbitrarily complex, and may consider any number of answers. For example, an answer providing a residential zip code in Iowa might be identified by the rule server **104** as unresponsive if it is known the residence is located in Montana. In another example, an answer providing a residential zip code with only four digits might be identified by the rule server **104** as unresponsive, as zip codes have five digits. Accordingly, the rules check the syntax and content of the answer to ensure both are appropriate for an answer.

[**0022**] If the rule server **104** determines the answer is fully responsive and valid then, then the rule server **104** determines if the present answer is the last answer to be reviewed **340**. If the current answer is not the last answer to be reviewed, then the rule server **104** reviews the next answer in the set of received answers for review **360**. After the next answer is reviewed, the rule server **104** repeats step **320** by determining if the newly current answer is fully responsive.

[**0023**] If any answers are determined to be not fully responsive in step **320**, then the rule server **104** determines if the corresponding question which prompted the answer needs, to be answered again **370**. As the question was not answered correctly, the rule server **104** defaults to posing the question again; however, asking a question again may not always be necessary or desirable, as when questions are optional. For all optional questions that are not responded with a fully responsive answers, rules in the rule server **104** determine if it is necessary to ask those questions again. If a question needs to be re-asked, the rule server **104** flags the question to be re-asked so that the questions may be identified later **380**, and execution of the rule server **104** proceeds to step **340**. If a question does not need to be re-asked, the question is not flagged, and the execution of the rule server **104** proceeds to step **340**. After all the answers have

been reviewed, the process of determining if the received answers to the set of asked questions are fully responsive and valid concludes **350**.

[**0024**] After determining in the received answered are responsive and valid, the next set of questions to be asked by the rule server **104** are determined **400**. Initially, questions that were flagged in step **380** are identified for inclusion into the next set of questions to be asked. New questions to be asked are dynamically determined based upon a master list of questions and analysis performed on the received answers. In other words, the next set of questions from the master list presented to the user will change based upon how the first set of questions was answered. Rules defined in the rule server **104** are consulted to determine the composition of the next set of questions.

[**0025**] For example, if a person was filling out an electronic form and was asked if they were under 18 years of age, and answered "No", thereafter questions pertinent only to a person under 18 years of age would not be presented to that person. Thus, if all the possible questions that could be asked of a user were arranged in a tree structure, then branches corresponding to questions that are indicated by the user to be inapplicable are not traversed. In other example, if a user's answers identify him or her as belonging to a certain profile or demographic, then only questions appropriate for that profile or demographic may be presented to the user. Accordingly, only questions with relevance to the user based upon prior answers are presented to the user.

[**0026**] In embodiments of the invention, the number and arrangement of the questions in the set of questions presented to the user are dynamic. For example, a set of 1000 possible questions that can be asked of a person may be divided into numerous smaller groups to be presented to users. One may be presented four screens with 250 questions each, 10 screens with 100 questions, and so on. In contrast to a static arrangement, in which the number of questions per screens are fixed or in which questions are mapped to a set of one or more screens, embodiments of the invention provide for dynamically determining the number and arrangement of questions per screen presented to the user. Thus, each user will be presented with different questions in different arrangements based upon different responses to questions.

[**0027**] The process of determining any suggested answers the next set of questions to be asked **500** will now be described. In embodiments of the invention, the rule server **104** suggests possible answers to the user in subsequent questions. In other words, questions are presented to the user with possible responses based upon how a user has previously answered. For example, if a person had previously indicated that her or she is a resident of San Francisco, and later was asked to name the airport from which they fly most frequently, then the San Francisco International Airport may be presented to the user as a default answer, given the close proximity of the airport to the user's residence. The rules employed by the rule server **104** in generating suggested answers may be of any complexity, and may consider any number of prior answers, external variables, external data sources.

[**0028**] In embodiments of the invention, the rule server **104** may consult a plurality of external data sources in determining the suggested answers in step **500**. According to

embodiments of the invention, the rule server **104** may consult data resident upon client **102** in generating suggested answers. In other embodiments, rule server **104** may consult data resident in rule server **104** in generating suggested answers. In still other embodiments of the information, external data sources may be consulted, such as, for example, an external database in data connection with the rule server **104**.

[**0029**] Subsequently, the rule server **104** transmits the next set of questions to be asked and any suggested corresponding answers to a client **600** associated with the appropriate user. This information can be sent to the appropriate user using a variety of well-known mechanisms. In embodiments of the invention, the information in step **600** is transmitted to the appropriate user using the API of step **200**. The next set of questions to be asked and any suggested corresponding answers may be transmitted to a client directly, or through the application tier **110**. The next set of questions to be asked may be visually presented through an electronic interface, such as, for example, a web page or other such graphical user interface.

[**0030**] For ease of explanation, embodiments of the invention have been discussed in relation to a rule server **104**. However, functions of the rule server **104** may be performed by other processing entities in other embodiments of the invention. For example, functionality performed by the rule server **104** may be performed by the rule engine **106** in embodiment, of the invention. Accordingly, it is contemplated that various processing entities, such as the rule server **104**, and rule engine **106**, among others, may perform the steps discussed herein to practice the invention.

SYSTEM ARCHITECTURE OVERVIEW

[**0031**] Referring to **FIG. 4**, in an embodiment, a rule server **104** and a client **102** may be implemented in a computer system **420**, wherein a rule server **104** is implemented in a host computer **422** and a client is implemented in an individual user station **424**. A computer system **420** includes a host computer **422** connected to a plurality of individual user stations **424**. In an embodiment, the user stations **424** each comprise suitable data terminals, for example, but not limited to, e.g., personal computers, portable laptop computers, or personal data assistants ("PDAs"), which can store and independently run one or more applications, i.e., programs. For purposes of illustration, some of the user stations **424** are connected to the host computer **422** via a local area network ("LAN") **426**. Other user stations **424** are remotely connected to the host computer **422** via a public telephone switched network ("PSTN") **428** and/or a wireless network **430**.

[**0032**] In an embodiment, the host computer **422** operates in conjunction with a data storage system **431**, wherein the data storage system **431** contains a database **432** that is readily accessible by the host computer **422**.

[**0033**] In alternative embodiments, the database **432** may be resident on the host computer, stored, e.g., in the host computer's ROM, PROM, EPROM, or any other memory chip, and/or its hard disk. In yet alternative embodiments, the database **432** may be read by the host computer **422** from one or more floppy disks, flexible disks, magnetic tapes, any other magnetic medium, CD-ROMs, any other optical

medium, punchcards, papertape, or any other physical medium with patterns of holes, or any other medium from which a computer can read.

[0034] In an alternative embodiment, the host computer 422 can access two or more databases 432, stored in a variety of mediums, as previously discussed.

[0035] Referring to FIG. 5, in an embodiment, each user station 424 and the host computer 422, each referred to generally as a processing unit, embodies a general architecture 505. A processing unit includes a bus 506 or other communication mechanism for communicating instructions, messages and data, collectively, information, and one or more processors 507 coupled with the bus 506 for processing information. A processing unit also includes a main memory 508, such as a random access memory (RAM) or other dynamic storage device, coupled to the bus 506 for storing dynamic data and instructions to be executed by the processor(s) 507. The main memory 508 also may be used for storing temporary data, i.e., variables, or other intermediate information during execution of instructions by the processor(s) 507.

[0036] A processing unit may further include a read only memory (ROM) 509 or other static storage device coupled to the bus 506 for storing static data and instructions for the processor(s) 507. A storage device 510, such as a magnetic disk or optical disk, may also be provided and coupled to the bus 506 for storing data and instructions for the processor(s) 507.

[0037] A processing unit may be coupled via the bus 506 to a display device 511, such as, but not limited to, a cathode ray tube (CRT), for displaying information to a user. Sin input device 512, including alphanumeric and other keys, is coupled to the bus 506 for communicating information and command selections to the processor(s) 507. Another type of user input device may include a cursor control 513, such as, but not limited to, a mouse, a trackball, a fingerpad, or cursor direction keys, for communicating direction information and command selections to the processor(s) 507 and for controlling cursor movement on the display 511.

[0038] According to one embodiment of the invention, the individual processing units perform specific operations by their respective processor(s) 507 executing one or more sequences of one or more instructions contained in the main memory 508. Such instructions may be read into the main memory 508 from another computer-usable medium, such as the ROM 509 or the storage device 510. Execution of the sequences of instructions contained in the main memory 508 causes the processor(s) 507 to perform the processes described herein. In alternative embodiments, hard-wired circuitry may be used in place of or in combination with software instructions to implement the invention. Thus, embodiments of the invention are not limited to any specific combination of hardware circuitry and/or software.

[0039] The term "computer-usable medium," as used herein, refers to any medium that provides information or is usable by the processor(s) 507. Such a medium may take many forms, including, but not limited to, non-volatile, volatile and transmission media. Non-volatile media, i.e., media that can retain information in the absence of power, includes the ROM 509. Volatile media, i.e., media that can not retain information in the absence of power, includes the main memory 508. Transmission media includes coaxial cables, copper wire and fiber optics, including the wires that comprise the bus 506. Transmission media can also take the

form of carrier waves; i.e., electromagnetic waves that can be modulated, as in frequency, amplitude or phase, to transmit information signals. Additionally, transmission media can take the form of acoustic or light waves, such as those generated during radio wave and infrared data communications.

[0040] Common forms of computer-usable media include, for example: a floppy disk, flexible disk, hard disk, magnetic tape, any other magnetic medium, CD-ROM, any other optical medium, punchcards, papertape, any other physical medium with patterns of holes, RAM, ROM, PROM (i.e., programmable read only memory), EPROM (i.e., erasable programmable read only memory), including FLASH-EPROM, any other memory chip or cartridge, carrier waves, or any other medium from which a processor 507 can retrieve information.

[0041] Various forms of computer-usable media may be involved in providing one or more sequences of one or more instructions to the processor(s) 507 for execution. For example, the instructions may initially be provided on a magnetic disk of a remote computer (not shown). The remote computer may load the instructions into its dynamic memory and then transmit them over a telephone line, using a modem. A modem local to the processing unit may receive the instructions on a telephone line and use an infrared transmitter to convert the instruction signals transmitted over the telephone line to corresponding infrared signals. An infrared detector (not shown) coupled to the bus 506 may receive the infrared signals and place the instructions therein on the bus 506. The bus 506 may carry the instructions to the main memory 503, from which the processor(s) 507 thereafter retrieves and executes the instructions. The instructions received by the main memory 508 may optionally be stored on the storage device 510, either before or after their execution by the processor(s) 507.

[0042] Each processing unit may also include a communication interface 514 coupled to the bus 506. The communication interface 514 provides two-way communication between the respective user stations 424 and the host computer 422. The communication interface 514 of a respective processing unit transmits and receives electrical, electromagnetic or optical signals that include data streams representing various types of information, including instructions, messages and data.

[0043] A communication link 515 links a respective user station 424 and a host computer 422. The communication link 515 may be a LAN 426, in which case the communication interface 414 may be a LAN card. Alternatively, the communication link 515 may be a PSTN 428, in which case the communication interface 514 may be an integrated services digital network (ISDN) card or a modem. Also, as a further alternative, the communication link 515 may be a wireless network 430.

[0044] A processing unit may transmit and receive messages, data, and instructions, including program, i.e., application, code, through its respective communication link 515 and communication interface 514. Received program code may be executed by the, respective processor(s) 507 as it is received, and/or stored in the storage device 510, or other associated non-volatile media, for later execution. In this manner, a processing unit may receive messages, data and/or program code in the form of a carrier wave.

[0045] In the foregoing specification, the invention has been described with reference to specific embodiments

thereof. It will, however, be evident that various modifications and changes may be made thereto without departing from the broader spirit and scope of the invention. For example, the reader is to understand that the specific ordering and combination of process actions shown in the process flow diagrams described herein is merely illustrative, and the invention can be performed using different or additional process actions, or a different combination or ordering of process actions. The specification and drawings are, accordingly, to be regarded in an illustrative rather than restrictive sense.

What is claimed is:

1. A method for presenting questions, comprising the steps of:

receiving a set of answers from a user corresponding to a first set of questions;

analyzing said set of answers using a rule server;

making a first determination of whether any questions from the first set of questions need to be answered again;

making a second determination of the composition of a second set of questions; and

presenting to said user said second set of questions through an electronic interface.

2. The method of claim 1, wherein said second set of questions includes any questions identified in said first determination.

3. The method of claim 1, wherein said first determination and said second determination are performed at an application tier.

4. The method of claim 1, wherein said first determination and said second determination are performed at said rule server.

5. The method of claim 1, wherein said second set of questions is dynamically generated.

6. The method of claim 1, wherein said second determination includes determining the number and arrangement of said second set of questions.

7. The method of claim 1, further comprising the step of:

making a third determination of a composition of a set of one or more suggested answers to one or more questions in said second set of questions.

8. The method of claim 7, further comprising the step of:

presenting to said user said set of one or more suggested answers through said electronic interface.

9. The method of claim 6, wherein said number and said arrangement of said second set of questions is different for different compositions of said set of answers.

10. A computer-readable medium carrying instructions for sharing information across a plurality of services provided over a wireless connection to a device, wherein said instructions include instructions for performing the steps of:

receiving a set of answers from a user corresponding to a first set of questions;

analyzing said set of answers using a rule server;

making a first determination of whether any questions from the first set of questions need to be answered again;

making a second determination of the composition of a second set of questions; and

presenting to said user said second set of questions through an electronic interface.

11. The computer-readable medium of claim 10, wherein said second set of questions includes any questions identified in said first determination.

12. The computer-readable medium of claim 10, wherein said first determination and said second determination is performed at an application tier.

13. The computer-readable medium of claim 10, wherein said first determination and said second determination is performed at said rule server.

14. The computer-readable medium of claim 10, wherein said second set of questions is dynamically generated.

15. The computer-readable medium of claim 10, wherein said second determination includes determining the number and arrangement of said second set of questions.

16. The computer-readable medium of claim 10, further comprising the step of:

making a third determination of the composition of a set of one or more suggested answers to one or more of said second set of questions.

17. The method of computer-readable medium of claim 16, further comprising the step of:

presenting to said user said set of one or more suggested answers through said electronic interface.

18. The method of claim 15, wherein said number and said arrangement of said second set of questions is different for different compositions of said set of answers.

* * * * *