

The algorithm proceeds in stages, each stage transforming the working subexpression forms to a collection of parent forms. The algorithm stops when the working database is empty. At a given stage, it iterates through the collection of all working forms. For each form in the working collection marked for study at
5 this stage, it considers the collection of all parent expressions of that expression. This is available because associated with a form is a list of pointers to its occurrences in the pool.

Among those parent method invocations, it identifies the unique forms, *i.e.* the
10 unique combinations of method name and method arguments which have the given subexpression as a first level subexpression. These unique invocation patterns are referred to as parent forms. If there are no parent forms, the subexpression is deleted from the working database. If there is exactly one parent form, the subexpression is replaced in the working database by its parent
15 form, the parent form being marked for processing only at the next stage, and the pointers to the occurrences of the parent form being properly calculated, using the previously available pointers to the children occurrences. If there is more than one parent form, then a new maximal form is recognized. It is assigned a maximal-form ID number, and a wrapper transformation is made on each
20 expression that references the form. That is, in all those expressions where the form occurs, a wrapper is inserted around the form according to the recipe:

(Maximal-Subexpression \$ID# (quote \$\$))

25 where ID# is replaced by the ID number of the identified maximal-form, \$\$ refers to the occurrence of the maximal-form itself, and the (quote) form is the means of preventing immediate evaluation, as described above.

The working forms database is then expanded to include each unique parent
30 form of the recognized maximal-form, with the newly added items marked for

evaluation at the following stage, and with a list of pointers to the occurrences of each parent form in the advice pool.

At the conclusion of this algorithm, there is a collection of transformed
5 expressions in which maximal common subexpressions have been identified and where only nonredundant evaluation is performed.

The reader may wish to verify that the algorithm produces exactly the desired result on the pool of five relevance clauses indicated earlier.

10

Alternates to Binary Relevance Determination

The invention contemplates a situation where messages arrive and computations are performed to evaluate certain assertions with the general goal of notifying the
15 user about certain associated messages, where the timing, format, and other attributes of the notification, including the decision to notify or not, are influenced by the results of the specified computations. The broader notion of influencing relevance appraisal may be implemented by a slight variation on the system described above.

20

The invention, in one embodiment, obtains appraisals of relevance according to non-binary criteria. A well formed phrase in the relevance language results in numerical values rather than Boolean values. Boolean True is viewed as equivalent to the numerical value 1.0, and the Boolean False as equivalent to the
25 numerical value 0.0. Suppose that certain clauses in a body of advice yield Boolean values, but other clauses yield numbers taking values between 0 and 1. A value between 0 and 1 is interpreted as indicating a degree of relevance that lies intermediate between certain relevance and certain irrelevance. In one embodiment, the user interface presents to the user advisories graded according
30 to degree of relevance, with those having value 1.0 at the top of the list and those

having value 0.0 at the bottom. This type of variation, extending Boolean to Real, is well known under the name fuzzy logic.

5 In a different embodiment, the outcome of relevance determination is a categorical label. In this embodiment, True and False are two labels, and the user interface is keyed to display messages labeled True. However, there are labels, such as Attractive Offer or Chronic Household Situation Needing Eventual Attention. Such labels result from evaluation of relevance clauses and, depending upon the user interface attached to the invention, such labels lead to
10 different methods of notification or different methods of presentation than other kinds of labels. The implementation of a centralized coordination authority such as advisories.com offers a mechanism for publication and coordination of such labels. The implementation of user side filtering allows the user to associate means of notification to various labels, which means include the possibility of no
15 notification.

In one embodiment of the invention, a layer of extra analysis is inserted between relevance appraisal and user interface. Thus, the result of relevance computation may be filtered based on user preferences and on observation of
20 the user. Thus, the relevance computation, rather than determining uniquely the notification status of messages, influences the notification process. For example, a user side filtering method (see above) whereby a user suppresses the display of certain messages which are nominally relevant may be implemented. In one embodiment, such censoring mechanisms are applied automatically. An advice
25 reader or other application contains a module to observe user behavior and make inferences about user preferences which can drive such censoring mechanisms. Similarly, in one embodiment, prioritization mechanisms are applied automatically. An advice reader or other application contains a module to observe user behavior and make inferences about user priorities, so that among

relevant messages those which are more likely to be of interest to the user are displayed earlier or more prominently.

Alternate Message Formats

5

Alternate to MIME Wrappers

The disclosed preferred embodiment uses MIME, a well known Internet standard, as a means of packaging advisories for transport across the Internet and other
10 digital transport media.

Another well known means for packaging textual information for remote interpretation is the XML language. This language also makes possible hierarchical messaging, and is able to accommodate message components of
15 the types enumerated above.

There are many implementations of the basic arrangement disclosed herein. Whether using well known protocols such as MIME and XML or proprietary protocols, they constitute implementations of the invention.

20

Substitutes for Three-Part Messaging

The invention is discussed in terms of a three-part message, containing humanly interpretable information, a relevance clause, and computer interpretable
25 information. These three logically connected components need not be packaged in the same physical message. There needs to be only an association between these parts. For example, the ASUP protocol sends abstracts containing only message identifiers and the relevance clause separately from the message body, consisting of explanatory content, software, and references. Under ASUP,
30 relevance evaluation drives a second reader-server interaction, where the

associated message body is obtained. In other implementations, an even looser association between relevance clause and content is maintained, where a relevant result initiates exploration of a whole sequence of messages.

5 Substitutes for Relevance Language

The relevance language is a convenient means of describing the state of a consumer computer and its environment. However, other languages can be modified into forms which enable computed-relevance messaging.

10

JAVA Model

The JAVA programming language is a well known and widely available tool for specifying computations.

15

In one embodiment of the invention, the role of the relevance language is played using software tools implemented in the JAVA programming language. Owing to the popularity of JAVA this might find wide acceptance among software developers and other computer professionals.

20

In the currently understood best method of developing this implementation, a special variant of JAVA, RELEVANCE-JAVA is developed, with its own specialized resources and evaluated by a specialized variant of the JAVA machine. The intent of this special version is to provide some of the same privacy and security characteristics as the relevance language described earlier. RELEVANCE-JAVA supplies three specific features which make it very useful:

25

- * Specialized inspector libraries. Special JAVA objects and classes developed to enable the determination of properties of the consumer computer. These inspect file system, system settings, and related properties of the computer

30

and its environment. This is effected by turning on certain features in the JAVA virtual machine which enable access of machine characteristics.

- 5 * Privacy Restrictions. While RELEVANCE-JAVA is able to learn a great deal about the user machine, it does not have the ability to transmit any gathered information back to the author. This is effected by limiting the installed objects and classes and turning off certain features in the JAVA virtual machine.
- 10 * Security Restrictions. While RELEVANCE-JAVA is able to learn a great deal about the user machine, it does not have the ability to modify the machine, *i.e.* to modify files and to affect the system settings.

20 The three part messaging model described above is conducted as follows: One part consists of humanly interpretable explanatory content; one part consists of RELEVANCE-JAVA code specifying conditions under which a message becomes relevant on certain consumer machines; and one part of computer interpretable code, perhaps in a different dialect of JAVA, able to cause effects on the consumer machine after consumer approval.

20

Visual Basic Model

25 The Visual Basic programming language is a well known and widely available tool for specifying computations.

25

In one embodiment of the invention, the role of the relevance language is played using software tools implemented in the Visual Basic programming language. Owing to the popularity of Visual Basic this finds wide acceptance among software developers and other computer professionals.

30

In the currently understood best method of developing this implementation, a special variant of Visual Basic, RELEVANT-BASIC is developed with its own specialized resources and evaluated by a specialized variant of the Basic interpreter. The intent of this special version is to provide some of the same
5 privacy and security characteristics as the relevance language described earlier. RELEVANT-BASIC supplies three specific features which make it very useful:

- 10 * Specialized inspector libraries. Special Visual Basic functions and data types are developed to enable the determination of properties of the consumer computer. These have the ability to inspect file system, system settings, and related properties of the computer and its environment.
- 15 * Privacy Restrictions. While RELEVANT-BASIC is able to learn a great deal about the user machine, it does not have the ability to transmit any gathered information back to the author. This is effected by limiting the installed objects and classes and turning off certain features in the BASIC interpreter.
- 20 * Security Restrictions. While RELEVANT-BASIC is able to learn a great deal about the user machine, it does not have the ability to modify the machine, *i.e.* to modify files and to affect the system settings.

The three part messaging model is conducted as follows: One part consists of humanly interpretable explanatory content; one part consists of RELEVANT-
25 BASIC code specifying conditions under which a message becomes relevant on certain consumer machines; and one part of computer interpretable code, perhaps in a different dialect of Visual Basic, able to cause effects on the consumer machine after consumer approval.

UNIX Model

The UNIX Shell, in its variant implementations, may be viewed as a scripting language, a well known and widely available tool for examining properties of a file system and specifying computations.

In one embodiment of the invention, the role assigned to the relevance language is instead played by software tools implemented in the UNIX shell and associated UNIX Tools. Owing to the popularity of UNIX in its variant forms, this might find wide acceptance among software developers and other computer professionals.

In the currently understood best method of developing this implementation, a special variant of the UNIX Shell, RELEVANT-Shell is developed with its own specialized resources and evaluated by a specialized variant of the Shell interpreter. The intent of this special version is to provide some of the same privacy and security characteristics as the relevance language described earlier. RELEVANT-Shell supplies three specific features which make it useful:

- Specialized inspector Applications. Special applications are developed to enable the determination of properties of the consumer computer. These have the ability to inspect file system, system settings, and related properties of the computer and its environment. These are known to RELEVANT-Shell.
- Privacy Restrictions. While RELEVANT-Shell is able to learn about the user machine, it does not have the ability to transmit any gathered information back to the author. This is effected by disabling access to certain communications and networking features in the shell interpreter.
- Security Restrictions. While the applications reachable through RELEVANT-Shell are able to learn about the user machine, they do not have the ability to

modify the machine, i.e. to modify files and to affect the system settings, except through standard mechanisms, such as creating temporary files in standard locations such as tmp and subject to resource metering.

- 5 The three part messaging model is conducted as follows: One part consists of humanly interpretable explanatory content; one part consists of RELEVANT-Shell code specifying conditions under which a message becomes relevant on certain consumer machines; and one part of computer-interpretable code, perhaps in a different dialect of Shell or other UNIX-interpretable code, able to cause effects
10 on the consumer machine after consumer approval.

Alternate State Description

- The possibility of alternate methods of describing the state of the consumer
15 computer is described above. It is possible to describe the state without using an overall relevance language if one has available a community of watchers, each with their own peculiar interfaces. The relevance language is then replaced by whatever means of expression by which the said application modules are invoked and controlled.

20

Relevance-Mediated Processes

- The description of the invention has taken the stance that the purpose of
25 relevance evaluation is to mediate the decision to notify a consumer about the existence of a message. To that end, the advice reader application functions as a messaging center, and advisories play a role analogous to messages in e-mail, USENET news, and other messaging modalities, in that they are read by the user as part of a user defined schedule. In this viewpoint, the user is a manager of his computer, his property, and his affiliations, and he reads advice which
30 helps him with his concerns in that managerial role.

However, there are other non-managerial settings in which relevance can drive the presentation of information to a consumer as an integral part of certain other processes in which the consumer is engaged.

5

- * Guidance. The consumer is the user of a computer applications program, and relevance based messaging provides guidance to the consumer at the moment before performing a certain action or at the moment after performing a certain action.

10

- * Composition. The consumer is reading a document using a display application on the computer, and relevance based content adaptation shapes the document so that the humanly interpretable message targets directly the characteristics of the reader.

15

In fact, all such applications are embodiments of the invention. Computed relevance messaging is of value much more broadly than in the managerial mode described above.

20 Relevance-Guided Computer Interaction

The following is an example showing how an advisory is used to guide a user in the operation of a piece of software.

- 25 Consider the following problem: A certain dangerous e-mail message has been obtaining wide distribution. When received by a user with the e-mail program Eudora 4.0, the user sees an innocent looking mail message including an attachment with an invitation to the user to open the attachment. The attachment is actually a maliciously prepared document which, if opened, can cause damage
- 30 to the user's computer.

The discussion below describes one implementation of relevance based messaging which helps to deal effectively with this situation. Under that implementation, an author writes an advisory which is evaluated for relevance
5 before a user of Eudora opens an attachment. The relevance clause inspects various attributes of the contemplated action and precisely targets an attempt to open an attachment with certain attributes. The advisory then returns text to the mail application which the mail application displays to the user.

10 In one embodiment, the desired effect may be produced using an inter-application communication framework as follows:

- The mail reader application has a special collection of relevance evaluation events, *i.e.* predefined events which are well known to authors of advisories.
15
- Whenever one of these events occurs, the mail reader notifies the advice reader of the event via a standard event notification protocol.
- The advice reader maintains event pools, *i.e.* advisories intended for
20 evaluation upon receiving notice of certain events.
- The advice reader evaluates the advisories in an event pool upon receiving notice of the corresponding event.
- 25 • The advice reader notifies the user of a relevant message by either:

Notifying the user of the application directly, employing standard user interface devices of the advice reader; or

Sending the relevant messages to the mail reader. The mail reader then displays those messages for the user, according to the user interface standards of that application.

- 5 The choice between these methods of notification is made under the control of user preferences, author preferences, or application defaults.

This event-driven framework is particularly powerful when:

- 10 * The application sending an event signal includes descriptive information about the event. In the mail reader context, the event Eudora About to Open Attachment is accompanied by information about the sender of the mail, information about the name of the attachment file, information about the sender of the mail, and information about attributes of the attachment file.

15

- * The advice reader contains an inspector library which refers to properties furnished by the application, e.g. mail sender and file name.

In this context, if someone wants to warn every user receiving mail from
20 king@athens.gr with an attachment named trojan.txt that he should not open the attachment, it is possible to author a relevance clause targeting the advisory to those people about to open such an attachment. The routing of advisories to advice event pools is handled through the header line mechanism of MIME and the message line variations discussed above. A simple header line of the form
25 advice-event-pool:, followed by the name of a predefined advice event, indicates the desired routing.

Relevance-Adapted Communication

The following is an example showing how relevance is used to customize the distribution of a body of information (see Fig. 19):

5

Consider the following problem: A certain publisher wants to create an electronic document whose content is tailored to the reader, for example because it consists of advertising which is more suitable for some readers than others, or because it consists of technical information which is more suitable for some readers than others. However, an ideal customization requires intimate knowledge of the configuration and details of the consumer's preferences, possessions, and affiliations, information which is not likely to be made available by consumers.

15 The discussion below describes an implementation of a system using the relevance evaluation components of invention. This implementation allows the publisher to create relevance adapted documents, allowing solution of the problem. The publication is distributed as a digital document containing embedded within it references to many possible variations in content. The selection among possible variants is driven by relevance clauses. The components of the document that actually appear on the users display are those which are selected based on intimate knowledge of the characteristics of the user.

25 The following is one implementation of such a system: A certain base document processing target format is chosen. Suppose for concreteness this is HTML. A special source format is then defined, consisting of documents. In the present context, this is referred to as PRE-HTML. This source format 194 offers the possibility of arranging many hierarchically nested fragments of modified HTML in a linear order. Each component of such an arrangement is protected by one

or more relevance clauses. The components of the source format differ from HTML in that they also offer embedded include expressions from the relevance language.

5 The advisory author writes the document with relevance clauses and inspector clauses 191. To create a custom document for a specific user, the source format document is transported to the user computer 192, and the document in source format is compiled into a custom target format document 195. The target format document is then processed by the intended target document processing system,
10 producing a display of a customized document 193.

The compilation step is the step where the customization occurs and bears closer examination. As the source document is processed, various components are encountered. Those which are protected by relevance clauses which
15 evaluate to False or at any rate not to True are discarded. They do not appear in the final target format file. Those which are protected by relevance clauses which evaluate to True are retained. They do appear in the final target format file. Each retained component is processed before placement in the target document file. If any include expressions are identified in the file, then those expressions
20 are evaluated, and the results are interpolated into the target document file.

This solves the problem of customized document preparation because the relevance language enables the provider to prepare documents which are customized as if the author had access to detailed intimate knowledge of
25 properties of the consumer's computer and environment, but it does so without the need for the consumer to reveal that intimate information to the provider.

This embodiment of the invention posits a provider with information which is presented to various consumers in precisely defined circumstances, and it uses
30 the relevance guarded messaging model described above. Here, the gatherer,

the watcher, and the notifier have different structure than they do in the invention as described above, but at an abstract level their functions are similar. For example, the tool which compiles a source format document into a target format document plays the role of both watcher and notifier in the five-part model
5 discussed above, while the target document processing system plays the role of user interface for the notifier. The role of gatherer is played by whatever system or systems bring the source format document into the consumer environment.

There are privacy considerations in this sort of customized documentation. The
10 use of HTML as a target language, for example, means that there is a possibility of leaks.

Other implementations of relevance driven document customization are possible. For example, one could develop a system in which the source document is not
15 compiled once and for all into a target document in a well known format but, rather the source document is structured for interactive interpretation. The following is an example: A source document consists of many pages of PRE-HTML. Embedded in the source document are conditional compilation blocks protected by relevance clauses, and include expression substitutions using the
20 relevance clauses, as described before. As the viewer goes through the document from page to page, each page is compiled from PRE-HTML to HTML and displayed as needed. Under this model, the user's path through the document is determined only at run time. For example, certain links in the document are relevance protected. The relevance expressions refer to attributes
25 of the environment that are changing as the reader progresses through the document, *i.e.* they are changing because the reader is progressing through the document. For example, a reader is prompted for information as part of his reading of the document and, as a result of the prompt, a site profile variable changes, causing pages visited later in the reading to change as a result.

30

Remote Access to Personal Information

The invention makes it possible for an advisory author to target situations based on an arbitrary combination of computationally verifiable conditions of the consumer computer and its environment. This environment may include data which may be of a personal nature. To the extent that certain kinds of personal data may be widely assumed to exist in a standard format on a substantial population of personal computers this creates the possibility of the invention being used to advise a substantial population of individuals on issues of a personal nature. Natural applications areas include:

- Personal Finance: If information about individual financial assets is assumed to exist on the consumer computer or in its environment in a standard format on a large collection of consumer computers, then advice authors can provide a large body of individuals timely and relevant advice about their bank account management or about their investment portfolio.

- Personal Health Issues: If information about individual medical records is assumed to exist on the consumer computer or in its environment in a standard format on a large collection of consumer computers, then advice authors can provide a large body of individuals timely and relevant advice about drug interactions, or about interactions between genetic or blood type information and drugs.

This creates an unprecedented opportunity, i.e. the ability to offer highly targeted advice without compromising individual privacy. Although the advice author is authoring detailed assertions about the finances or health of the consumer, and although it requires intimate knowledge of sensitive personal information to evaluate those assertions, the system itself is not revealing this information back

to the author. The consumer may, in some circumstances, choose to reveal such information after reading a relevant advisory.

Such applications are limited by the need for consumers to capture and maintain accurate data in a standard format about items which concern the consumers and which are accessible in a means well known to advice providers. It would be highly desirable to remove the data management and data input burden under this arrangement, so that consumers are not required to become data managers. In particular, it would be highly desirable for the professional organizations responsible for maintaining accurate data about their customers to be the locus of responsibility for data integrity. For example:

- Pharmacies maintain records about their customers.
- Doctors maintain records about their patients.
- Financial institutions maintain records about their clients.

These actors are paid, in part, for keeping accurate and timely records about their patients, customers, or clients.

It would be highly desirable for consumers to have access to some key information that is maintained for them by the professional organizations with which they are affiliated. For example:

- Instead of a consumer entering into his computer data about his drug prescriptions, it would be desirable for the needed data to be obtainable from the pharmacy automatically on demand by the consumer computer.

- Instead of a consumer entering into his computer data about his stock portfolio and manipulating it daily, it would be desirable for any needed data to be obtained from the financial institution automatically on demand by the consumer computer.

5

- Instead of a consumer entering into his computer data about his health records and manipulating the data as they change, it would be desirable for any needed data to be obtained from the medical institution automatically on demand by the consumer computer.

10

The following is a solution to this problem using the invention:

- A standard collection of remote medical records inspectors, remote financial records inspectors, and remote drug prescription inspectors is developed, and their syntax and use is published. These inspectors have both server side components and client side components, to be described later.
- Advice authors write advice concerning various issues associated with such personal information.
- Certain doctors, financial institutions, and pharmacies install server side components at computers in their offices. They advertise to the public the availability of remote information access.
- The consumer who is interested in benefiting from advice written using remote information access approaches the financial institution, doctor, or pharmacy and authorizes participation of his own information in the server software.

15

20

- The consumer subscribes to certain advice sites whose advice includes advice making use of the remote inspectors. The subscription is initialized appropriately so that the consumer computers advice reader make use of the information.

5

- Such advice is periodically evaluated according to the advice pool in which the advice is placed. Evaluation causes the consumer computer to establish connections to remote computers to obtain needed information. For example, the remote drug prescription inspector library on the consumer machine establishes a connection with the pharmacy information server and performs certain queries to check if the consumer has certain problematic prescription combinations.

10

The following is an example of an advisory that is written using this system:

- 15 Suppose that a certain pharmaceutical manufacturer provides an antidepressant drug to its patients, and that it is discovered that patients who also use a certain anti-inflammatory may experience difficulties. In practice, one prescription might be due to a psychiatrist and the other by an orthopedist who might not be aware of the patient's other medical prescriptions. The manufacturer authors an advisory referring to the dangerous combination as follows:

20

exists pharmacy prescription "Xanax" and exists pharmacy prescription "Buterin"

- 25 The manufacturer includes a description of the potentially dangerous combination for a message body. When the advice reader on the consumer computer encounters this relevance clause, it contacts the pharmacy server with queries for pharmacy prescription Xanax and pharmacy prescription Buterin. It determines the relevance of the advisory based on this. It notifies the consumer of the situation if it turns out to be relevant.

30

An important issue in determining the consumer acceptance of this system is the ability of the system to protect consumer privacy. To this end, the interaction between client and server is carefully protected:

- 5
- The connection between consumer client and pharmacy server is secured by standard cryptographic means (e.g. SSL protocol).
 - The identity of the client requesting the information is authenticated by the pharmacy server by standard cryptographic means.

10

By these devices, the pharmacy server avoids revealing information about a person except to the advice reader on that person's computer. The advice reader on that person's computer does not reveal information so received, at least under ordinary operations.

15

The following is a convenient interaction protocol for such remote inspectors. In this protocol, it is simple to make the client side software. The client transmits, over a secure link, ASCII strings describing the queries exactly as they are described in the surface language. In the above example, the client transmits
20 pharmacy prescription Xanax." The server parses this using a miniature version of the relevance clause parser evaluator. The server knows that this clause refers to the prescription records of Joseph A. Patient because of the initial authentication work and, using standard database inquiry methods, searches the pharmacy database for an entry indicating that Mr. Patient had a pharmacy
25 prescription to Xanax. The server then returns True or False as an ASCII string, and the client parses this string and returns the corresponding Boolean to the advice reader.

Bi-Directional Communications

An intent of the invention is to allow only one way communication, taking information from advice provider to advice consumer, but not allowing information to leak back from consumer to provider. The phrase one way membrane evokes this.

However, there are numerous situations where this model is restrictive. For example, in certain situations consumers are willing to cooperate with providers, particularly when they receive a benefit from cooperating. An example is when consumers want to get technical support to solve a specific problem which existing advisories do not address. For the sake of solving their problem, they are willing to disclose various pieces of information about their configuration to the solution provider. In other situations, advice consumers subscribing to a certain site are actually employees of the organization which operates the advice site, and so they are willing to share information with that particular advice provider.

Open Bi-Directional Communications

The phrase open bidirectional communications refers to a setting where the invention is run and the communications are typically one way, but occasionally there are processes which feed back information to the advice provider, and the process takes place in the clear with the consumer computer identity explicitly available to the provider.

Questionnaires

In one implementation (see Fig. 20), a particular document type is defined, referred to as a questionnaire 200, containing text together with comments,

together with distinguished Include-Expressions. Suppose, that Include-Expressions are delimited by double Dollar Signs as in \$\$\$. The Include-Expressions are written in the relevance language, and need not evaluate to True or False. For example, they are string- or integer- valued. Suppose also
5 that comments are preceded by %-signs.

An example questionnaire is:

% Data needed by ABC Corporation to
10 % Diagnose the XYZ Problem
inventory of User Computer Configuration:
Computer Manufacturer: \$\$ Manufacturer of Computer \$\$
Model: \$\$ Model of Computer \$\$
OSVersion: \$\$ version of Operating System \$\$
15 RAM: \$\$ System Ram \$\$
Disk: \$\$ size of boot volume \$\$

This questionnaire contains text, such as computer manufacturer, as well as Include-Expressions, such as manufacturer of computer. The intent of the
20 questionnaire is that information about the type of computer and about certain features be collected by the advice reader using its rich library of inspectors.

The following is an example showing how questionnaires are used: A questionnaire such as that above is authored by an advice provider 200 and is
25 inserted inside the solution component of an advisory as a MIME component with distinctive content-type 201. The consumer sees a relevant advisory 202, accompanied by humanly interpretable content. The humanly interpretable content says:

5 You have the XYZ situation. In order to help you,
we at ABC Corp. need some information about this
situation -- information about your system setting. This information can be
automatically
gathered for you if you'll push the button on the left below.
You'll be given a chance to review the information
and then to approve its transmission to ABC Corp.

10 Below the advisory are two buttons: one saying Gather information and the other
saying Review Request. The first button signifies approval to gather the
information; the second button signifies a request to view the source file of the
questionnaire and thereby learn more about the provider's request to gather
data.

15 If the user approves 203, the relevance clauses in the questionnaire are
evaluated 204, for example using various inspectors 205, 206, and the
corresponding results are included in the result where the relevance clauses had
been. In the case of the previous example, this process produces:

20 % Data needed by ABC Corporation to
% Diagnose the XYZ Problem
Inventory of User Computer Configuration:
Computer Manufacturer: Toshiba
25 Model: T1200
OSType: Windows 98
OSVersion: 1.0
RAM: 64M
Disk: 2G

30

The user may be shown the results of the include process and given a chance to inspect the results and to relay the results to the advice provider. In one implementation, the results are presented to the user as part of a mailer window, showing the intended recipient of this information 207, and with a button at the bottom marked Send It 208.

By this device, the relevance language simplifies communications between advice provider and advice consumer, allowing inspectors to gather information needed by the advice provider that is difficult for consumers to gather for themselves. The provider is helped because it quickly and accurately obtains information that may be essential in the technical support process, and the customer is helped because the process removes a burden which he would have had of finding the correct data and of reporting it accurately.

For this method to work it must have consumer acceptance. Consumers are sensitive to the possibility of questionnaire spoofing, where a questionnaire purports to gather information of one kind, *e.g.* CPU type, while actually gathering information about another kind, *e.g.* VISA card number or passwords.

One technique to further consumer acceptance is for a privacy ratings service at a central site to certify questionnaires as being in accord with privacy standards when they are appropriate implementations of the randomized response protocol. Under existing Web protocols (see Khare, Rohit (1997) Digital Signature Label Architecture, The World Wide Web Journal, Summer 1997, Vol. 2, Number 3, pp. 49-64, O'Reilly, Sebastopol, CA, <http://www.w3.org/DSIG>) there is a method for the establishment of ratings services which can reliably certify that certain messages have certain properties. The credibility of such assertions, *i.e.* that they are actually made by the service and not by an impostor, is based on deployment of standard authentication and encryption devices. Applying this technology, a privacy ratings service is established at a central site, *e.g.* Better

Advice Bureau.org, to certify that certain questionnaires gather information in a fashion generally accepted as appropriate for the advertised task, and the information is used by the solicitor in a manner to protect individual identity. Advice authors seeking certification of the privacy respecting character of their questionnaires submit those messages to the certification authority, which studies the messages and, at its option, agrees to certify some of those messages as privacy respecting. In one embodiment of the invention, the user interface of the advice reader or similar component is configured to permit questionnaires to be displayed to users only when they have been credibly certified by a trusted privacy ratings service.

Mandatory Feedback

In one embodiment of the invention (see Fig. 21), open two-way communication is possible for the purposes of maintaining a relationship with a certain trusted provider.

This assumes a consumer situation different from the usual invention setting. In this variant setting, certain kinds of advice providers enjoy a special status, for example as employers or contractors, which allows them certain coercive privileges not ordinarily enjoyed by advice providers in other settings. These overlord advice sites 210 publish advisories that are gathered by a reader 211, which then performs a relevance evaluation on the advisory 212. Relevant messages are displayed 213 to the user and the user may approve or deny such action 214 as recommended by the advisory. A feedback path 216 enables user actions to be reported 215 to the overlord advice site

In this embodiment, any of the following options may be exercised:

- Certain advice site subscriptions are mandatory;
- 5 • Certain advice cannot be deleted by the user, advice by certain providers is not subject to user scheduling, prioritization, or deprecation;
- Certain advice generates automatic feedback from the user to the provider, concerning some or all of:
 - 10 (a) The consumer computer's identity;
 - (b) The relevance status of a certain advisory on that computer; and
 - 15 (c) The fact that a user has/has not taken a certain recommended solution in a certain advisory.

The feedback is transmitted by e-mail or by other convenient electronic means.

- 20 In this setting, a manager of many computers can:
- (1) write advisories destined to many machines he is managing;
 - (2) expect that the machines all receive the advisory; and
 - 25 (3) expect to receive, in return, information about the relevance and/or solution status of the advice on all those machines.

This set of functions may be implemented by modifying the basic advice reader
30 architecture discussed above (see Fig. 22).

- Advice sites 220 may be given a special overlord status (as discussed above in connection with Fig. 21) by configuring the subscription manager of the advice reader to enable such special status.

5

- A new message line type, Mandated-Action, is instituted and is used by advice sites with overlord status to label a message component with a special keyword phrase as invoking a certain coercive privilege:

10 Not user deleteable labels a message as not deletable by the user through the advice reader user interface 221;

 On relevance 222, Evaluate questionnaire 223 and mail back 224 labels a message as requiring immediate notification 225 of the author via a feedback path 226 upon relevance, the notification involving first
15 processing of a questionnaire filling in the various include fields and second transmitting the information to the author;

 Mail back on user acceptance labels a message as requiring immediate
20 notification of the author upon user accepting a proposed action by selecting the action button of an associated advisory;

 Mail back on user refusal labels a message as requiring immediate
notification of the author upon user accepting a proposed action by
25 selecting the action button of an associated advisory. The advice reader is modified in the appropriate way to carry out the indicated function when a message with overlord status is received and processed.

Masked Bi-Directional Communications

It is possible to enable bidirectional communications while preserving some degree of privacy protection by masking the identity of the respondent.

5

Masking Via Anonymous Communications and Privacy Ratings

In one implementation (see Fig. 23), an advice provider 231 obtains detailed information from consumer computers while communicating with consumers anonymously, thus enabling consumers to protect their own privacy. This embodiment of the invention limits the scope of communications so that when messages return to the advice provider:

10

- Message headers contain no information uniquely identifying the respondent;
- Message bodies themselves contain no information uniquely identifying the respondent; and
- The process has these components:

15

20

An advice provider 231 authors a document such as a questionnaire as described above, for gathering information automatically or an HTML form for gathering information by consumer interview. The user's advice reader 232 gathers this information.

25

Upon determining relevance 233:

If the document is a questionnaire, the advice reader fills in the appropriate include fields.

30

If the document is an HTML form, the consumer fills in the appropriate survey questions.

5 The document is e-mailed to the provider via anonymous routing along feedback paths 235, 236 through a certain centralized site, e.g. the Better Advice Bureau, advisories.com, or another site 230 offering identity protection via anonymous remailer or functionally equivalent services.

10 The final stage of this process removes information about the identity of the consumer, by stripping such identity from the message headers. Consumers are expected to have confidence in the fundamental validity of this approach because they understand that the centralized site has an incentive to protect the integrity of the process.

15 The consumer himself is responsible for ensuring that the message body is free of identifying information. For example, if the consumer responds to an HTML form asking for his name and address, then he is not protecting his own identity. If the consumer forwards a questionnaire containing identifying information, such as IP address, then he is not protecting his own identity.

20 In one implementation, the consumer protects his privacy with the help of a privacy ratings service at a central site. Under existing internet protocols (see Khare, Rohit, *Digital Signature Label Architecture*, The World Wide Web Journal, Vol. 2, Number 3, pp. 49-64, O'Reilly (1997) <http://www.w3.org/DSIG>) there is a
25 method for the establishment of ratings services which reliably certifies that certain messages have certain properties. The credibility of such assertions, *i.e.* that they are actually made by the service and not by an impostor, is based on deployment of standard authentication and encryption devices. Applying this
30 technology, a privacy ratings service is established at a central site, e.g. Better Advice Bureau.org, to certify that certain questionnaires do not contain devices

soliciting sensitive information. Advice authors seeking certification of the privacy respecting character of their messages submit those messages to the certification authority which studies the messages and, at its option, agrees to certify some of those messages as privacy respecting. In one embodiment of the invention, the user interface of the advice reader or similar component is configured to permit questionnaires and forms to be displayed to users only when they are credibly certified by the privacy ratings service.

Masking Via Randomized Response

10

In one implementation, an advice provider obtains detailed information from consumer computers while enabling consumers to protect their own privacy. This embodiment of the invention limits the scope of communications so that when messages return to the advice provider:

15

- Message bodies themselves contain no information which can be reliably inferred to reflect the true state of the consumer's computer or environment.

In certain embodiments, the technique is supplemented by the use of centralized anonymous communications and centralized privacy certifications.

20

The process has these components:

- An advice provider authors a document similar to a questionnaire as described above, for gathering information automatically, however obeying additional constraints.

25

- The advice reader fills in the appropriate include fields, randomly changing the answers, and changing the correct answers to incorrect answers, depending on a random mechanism.

30

- The resulting document is returned to the author.

5 In one implementation, the process by which the information is returned is made anonymous. The document is addressed to a certain centralized site, e.g. the Better Advice Bureau, or advisories.com, or another site offering identity protection via anonymous remailer or functionally equivalent services. This final stage of this process removes information about the identity of the consumer by stripping such identity from the message headers.

10

The following discussion describes the concept of randomly changing the answers in more detail: Suppose that only questionnaires with Boolean values are allowed, although more general questionnaires are allowed with extra work. The relevance evaluation component of the advice reader evaluates the Boolean expressions indicated in the include fields. However, it does not always insert the result in the outgoing message. Refer to R as the value obtained by relevance evaluation. Instead of always substituting a representation of R in place of the include field, the advice reader conducts a two stage stochastic experiment. At the first stage, it obtains a random Boolean X from a random number generator, the random Boolean being equally likely to be True or False. The value of X is kept private, and drives a decision at the first stage. In this decision, if X is True, the decision is taken to insert a representation of R in the include field. If X is False, the decision is taken to obtain a second Boolean Y, again equiprobable, and to insert a representation of Y in the include field. As a result, in any specific message, it is impossible to say whether the answer obtained at the relevance evaluation stage (R) is True or False on the basis of that message alone because the reported value is equally likely to be R or Y, and the variable X driving the choice between R and Y is not divulged.

25
30 This provides a degree of privacy protection for the consumer.

At the same time, this randomized response communications protocol makes it possible for the questionnaire author to obtain information reliably about the population of users while not revealing information about specific users. If π denotes the fraction of users in the sample with a certain characteristic, and p denotes the fraction of True responses received, then:

$$E(p) = 1/4 + \pi/2$$

where $E(\cdot)$ denotes mathematical expectation.

From $p = E(p)$ (the law of large numbers), π can be estimated by:

$$\hat{\pi} = 2(p - 1/4).$$

15

For example, if 61% of the responses are True, one estimates that 72 % = 2(61 % - 25 %) of the sample has the given characteristic.

There are extensions of the method to non-Boolean variables and to multiple item responses.

For this method to work it must have consumer acceptance. One technique to further consumer acceptance is for a privacy ratings service at a central site to certify messages as being in accord with privacy standards when they are appropriate implementations of the randomized response protocol. Under existing internet protocols (see Khare, Rohit, *Digital Signature Label Architecture*, The World Wide Web Journal, Vol. 2, Number 3, pp. 49-64, O'reilly (1997) <http://www.w3.org/DSIG>) there is a method for the establishment of ratings services, which reliably certifies that certain messages have certain

properties. The credibility of such assertions, *i.e.* that they are actually made by the service and not by an impostor, is based on deployment of standard authentication and encryption devices. Applying this technology, a privacy ratings service is established at a central site, *e.g.* Better Advice Bureau.org, to
5 certify that certain questionnaires use randomized response techniques appropriately and protect individual identity. Advice authors seeking certification of the privacy respecting character of their messages submit those messages to the certification authority which studies the messages and, at its option, agrees to certify some of those messages as privacy respecting. In one embodiment of the
10 invention, the user interface of the advice reader or similar component is configured to permit questionnaires and forms to be displayed to users only when they have been credibly certified by the privacy ratings service.

Network Management

15 The following discussion describes two important variations of the basic invention which are useful in problems of network management, *i.e.* management of large networks of computational devices.

20 Mandatory Advice

In the basic description of the invention, it is assumed that advice is offered as a convenience to a human consumer who acts in a managerial role to read and act appropriately at his option (see Fig. 24).

25 There are settings where the basic communications model described earlier can be usefully modified so that there is no user review of certain advisories. As an example of one such setting, a network administrator 240 supervises a large network of communicating computational devices, each one in a potentially
30 different and dynamically changing state. The network administrator wants

certain devices to perform a certain operation, but does not know which devices those are.

5 In this setting, it is valuable to have an advice reader program 241 which obtains and reviews 242 advisories, but which automatically applies the indicated solution operator 244 when relevance 243 is determined. This enables the network administrator to write a general advisory targeting many machines but not knowing in advance which machines those turn out to be, and obtain the desired functionality on those machines. A solution or communications log 245
10 may optionally be mailed back to the network administrator via a feedback path 246.

Examples of scenarios where this functionality is useful include:

- 15 * Target all machines whose security settings do not match a certain administrator defined standard. Reimpose the required settings on all such machines.
 - * Target all machines with a copy of a certain file. On such machines, replace
20 the file with an updated version.
 - * Target all machines which have less than a certain amount of free space on local disk. On such machines, purge the tmp volume.
- 25 Other examples can be supplied, including examples outside the technical support application. For example, in a setting where office appliances are computational devices, network management involves tasks concerning the maintenance and monitoring of assets and their use.

In the currently understood best implementation of this variation, there are several changes to the invention:

- 5 * The advice reader is implemented as a faceless application with no user interface component.

- * The advice reader typically receives advisories by messaging mechanisms alternative to the usual subscription model, for example by e-mail or other diffusion mechanism.
- 10 * The message format omits the humanly interpretable content.

- * The message format includes a message component containing a software tool, such as a script or executable binary, or a reference to a software tool, such as a URL or a file system pathname, providing functionality to be
15 invoked automatically in case a certain condition becomes relevant.

Certain features may be included in this variant:

- 20 * Security Feature. The advice reader includes an authentication feature to verify the identity of the advice site attempting to exert coercive privilege.

- * Bi-directional Communication Feature. The advice reader includes the ability to communicate back to the advice Author when the advice Author requires
25 this, as indicated by a Mandated-Action: message line.

Master-Slave Configuration

In the description of the invention, it is assumed that advice is offered as a convenience to a human consumer, who acts in a managerial role to read and act appropriately at his option. In the description, it is assumed implicitly that the consumer is the manager of a personal computer and its environment.

There are settings where the basic communications model described earlier can be usefully modified to reflect the needs of managers of large collections of computational devices. As an example of one such setting (see Fig. 25), a network administrator 250 supervises a large network of communicating computational devices 251-253, each one in a potentially different and dynamically changing state. The network administrator wants to have an advice reader which functions as a master reader 254, in which each entry he sees in the master user interface summarizes the relevance status of advice on many machines 255, 256 simultaneously. This allows the manager to overview 257, 258 and to make decisions about accepting or rejecting advice on many machines at once.

In this setting, the network administrator's workstation is a master machine and the computational devices he manages are slave machines. It is very desirable to have a master advice reader program running on the master machine and which obtains advisories, and which then communicates with the slave machines, each one running a slave relevance evaluator and slave action implementer, and which summarizes the results of the interaction. These slave relevance evaluators accept messages from the master advice reader. The messages consist of wrapper information and individual relevance clauses. The slaves evaluate the relevance clauses in the environment defined by their machines and transmit the resulting values to the master. The master reader then studies the results so obtained and, according to a special master user interface, presents to

the network administrator a summary of master relevant messages. A message is deemed master relevant if the associated relevance clause is true on any slave machine. The network administrator studies the master relevant messages and may accept the proposed actions associated with some of them. When he does
5 so, the master reader communicates with the slave action evaluator on slave machines on which a relevant result is obtained, relaying the recommended action part of the advisory, and indicating that the action should be taken. Each slave action evaluator contacted in this way then applies the indicated solution within the environment provided by that machine.

10

In this setting, a network administrator subscribes to advice and plays the role of managing the advice process in place of all the users of the slave machines. If a piece of advice, when relevant under the ordinary invention, suggests to a user that certain software should be updated on that user's machine, then the same
15 advice is presented to the network administrator instead when some machine on the network should have an update, and it effectively proposes that the corresponding software on every such machine be updated.

In the currently understood best implementation of this variation, there are
20 several changes to the usual invention model:

- The slave relevance evaluator and slave action implementor are implemented as faceless applications with no user interface component.
- 25 • The slave relevance evaluator and slave action implementor typically receive advisories by messaging mechanisms alternative to the usual subscription model, for example by e-mail or other diffusion mechanism.
- The message format for communications between master reader and slave
30 relevance evaluator omit the humanly interpretable content.

- 5 * The message format for communications between master reader and slave action implementor include a message component containing a software tool, such as a script or executable binary, or a reference to a software tool, such as a URL or a file system pathname, providing functionality to be invoked automatically.

10 In addition, certain variations may be exercised as well. The slave advice evaluator and slave action implementor include cryptographic authentication features to verify the identity of the master attempting to exert coercive privilege.

Owing to the difference in outlook that a network administrator has, the Master user interface has features not ordinarily available in the invention. These include:

- 15 * Machine List Display. To display a list of all the machines on which a given advisory is relevant. To decorate this list by including other characteristics of the machines.
- 20 * Machine List Filtering. To apply selection mechanisms to the list of relevant machines, allowing to apply the recommended action only to a selected subgroup of machines within the relevant group. Particularly useful is the ability to intersect a list of machines with a predefined list, e.g. a list of machines in a certain operational division, a list of machines in a certain location, or a list of machines arising as relevant in some other advisory. It is also important to allow the list of machines to be expanded beyond the relevant machines, allowing both editing by hand or concatenation with some other list of machines, for example a predefined list, or a list of machines relevant for some other advisory.

30

The logical structure described is that of a single body of advisories evaluated for relevance in a collection of different contexts, where the results in all those different contexts are gathered together in one single master user interface. This logical structure makes sense in other settings. For example, in the example of
5 drug interactions discussed above, the pharmacist is an administrator, the body of advisories that he has received from pharmaceutical manufacturers are a body to be applied in many different contexts, and each of his customers database records provide a unique context for interpretation of the advisories. Here, the context is not of individual machines but individual records in a database. The
10 master user interface is the basis for another variation of the invention, *i.e.* operating with a specialized database inspector, the master advice reader obtains a list of all the patients for each advisory for whom a given advisory is relevant. The user interface displays only master-relevant information to the pharmacist, *i.e.* advisories relevant for some patient in the database. The
15 pharmacist then views the relevant advisories and inspects a list of associated patients.

Although the invention is described herein with reference to the preferred embodiment, one skilled in the art will readily appreciate that other applications
20 may be substituted for those set forth herein without departing from the spirit and scope of the present invention. Accordingly, the invention should only be limited by the Claims included below.

CLAIMS

1. A communications system, comprising:
an advice provider which broadcasts information over a communications
5 medium;
an advice consumer for gathering said broadcast information from said
communications medium; and
a reader associated with said advice consumer for determining relevance
of said broadcast information;
10 wherein said advice consumer is advised of said information only if said
information meets certain predetermined relevance criteria.
2. The system of Claim 1, wherein said reader further comprises:
means for providing relevant information to said advice consumer without
15 revealing any aspect of said advice consumer's identity to said advice provider.
3. The system of Claim 1, wherein relevance of said information to said
advice consumer is based upon any of the properties of an advice consumer's
computer, said computer's contents or state, or the properties of a local
20 environment associated with said computer.
4. The system of Claim 1, wherein said advice provider specifies an
audience for whom said information is potentially relevant by referring to
25 properties of an advice consumer which are used to determine the relevance of
said information to said advice consumer.
5. A communications method, comprising the steps of:
preparing a message at an advice provider location;

broadcasting said message to potential advice consumers anonymously using a network;

processing said message at an advice consumer location;

determining whether said message is relevant to said advice consumer;

5 wherein said message is broadcast to advice consumers to whom it is relevant without requiring that said advice consumers divulge their identity or attributes to said advice provider.

6. In a system including computational devices connected by a
10 communications network, a communications apparatus for linking an information provider to information consumer, comprising:

specific units of advice to be shared;

digital documents conveying said advice;

an advice provider for broadcasting said advice in the form of advisories;

15 an advice consumer for receiving said advisories;

wherein advisories are broadcast over said communications network from said advice provider to said advice consumer; and

a communications protocol for narrowly-focused targeting of said
advisories to said advice consumer by automatically matching advisories with an
20 advice consumer for whom said advisories are relevant.

7. The apparatus of Claim 6, further comprising:

an advice reader associated with an advice consumer computer for
performing relevance determination.

25

8. The apparatus of Claim 7, wherein relevance determination is
automatically performed based on a combination of conditions, including any of
hardware attributes, configuration attributes, database attributes, environmental
attributes, computed attributes, remote attributes, timeliness, personal attributes,
30 randomization, and advice attributes.

9. The apparatus of Claim 6, wherein said advisories comprise:
digital documents which contain an explanatory component describing in
5 terms said advice consumer can easily understand the reason that said advisory
is relevant and the purpose and effects of the action which is being
recommended to said advice consumer.
10. The apparatus of Claim 7, said advice reader comprising:
10 a gatherer for gathering advisories to which said advice consumer
subscribes;
a subscription manager for entering subscriptions to advisories based on
information in at least one advice consumer site definition file;
an unwrapper for parsing said advisories;
15 a module for determining the relevance of said advisories, said
determination being made either continuously, at scheduled intervals, or under
user manual control;
a user interface that receives relevant advisories; and
a display and management system that displays relevant advisories for
20 inspection by said advice consumer.
11. A communications apparatus, comprising:
an advisory comprising:
a relevance clause comprising an assertion about the state of an
25 advice consumer computer, its contents, or environment which can be
automatically evaluated by comparing said assertion with said advice
consumer computer's actual state;
a message associated with said relevance clause whose suitability
for the consumer is determined at least partially by the evaluation of said
30 relevance clause;

a gatherer for assuring that relevance clauses flow into said advice consumer computer from various locations;

5 a watcher for evaluating relevance clauses by comparing them with an actual state of an advice consumer environment, and by inspecting properties of said advice consumer computer and its environment and checking if these point towards or away from relevance; and

a notifier for displaying messages to an advice consumer under at least partial guidance of an evaluated relevance clause.

10 12. The apparatus of Claim 11, said advisory further comprising any of:

a wrapper for packaging information in said advisory for transport and subsequent decoding;

a from line fort identifying an advice author;

a subject line fort identifying the concern of said advisory;

15 a relevance clause for specifying conditions under which the said is relevant;

a message body for providing explanatory material explaining to said advice consumer what condition is relevant, why said advice consumer is concerned, and what action is recommended; and

20 an action button for providing said advice consumer with the ability to invoke an automatic execution of a recommended action.

25 13. A mechanism for encoding one or more digital documents conveying advice for transport across computer networks and other digital transport media, comprising:

a wrapper for packaging said one or more digital documents for transport and subsequent decoding;

a from line component for identifying an advice author;

30 a subject line component for summarily identifying said advice subject matter;

a relevance clause component for specifying conditions under which said advice is relevant;

5 a message body component for providing explanatory material explaining to a user what condition is relevant, why said user is concerned, and what action is recommended; and

an action button component for providing said user the ability to invoke an automatic execution of said recommended action;

10 wherein said one or more digital documents comprises broadcast messages that do not have an intended recipient or list of recipients.

14. The mechanism of Claim 13, said one or more digital documents further comprising any:

an expires-when clause for causing a document to expire if said clause evaluates to true;

15 an evaluate-when clause for causing a document to be evaluated for relevance if it evaluates to true;

a requires-inspector-library clause for giving the name of a specific inspector library and a URL where it can be found, wherein said specific inspector library must be installed for relevance to be evaluated correctly;

20 a refers-to clause for giving keyword labeling of systems referenced by a condition associated with a document; and

a solution-affects clause for giving keyword labeling of effects of a recommended response.

25 15. The mechanism of Claim 13, wherein said message body further comprises:

digital authentication features appended to said message to insure its integrity and authenticity.

16. The mechanism of Claim 13, wherein said one or more digital documents are packaged as text files which are a valid instance of a MIME file.

17. The mechanism of Claim 16, wherein said MIME file further comprises any of:

header lines for specifying that a message body may be preceded by an extensive message header consisting of a variety of header lines, where individual lines begin with a well known phrase and contain addressing, dating, and related commentary;

10 means for creating new message lines in messages, including any of means for embedding said new message lines in messages and means for registering said new lines with MIME authorities;

means for offering different versions of a same message, with a destination picking an appropriate display method; and

15 a digesting mechanism for packing several complete MIME messages into a single file for Internet transport.

18. The mechanism of claim 13, further comprising:

20 one or more ratings blocks for containing information which rates said one or more digital documents according to criteria including any of privacy, security, and usefulness.

19. A method for encoding one or more digital documents conveying advice for transport across computer networks and other digital transport media, said process comprising the steps of:

25 packaging said one or more digital documents for transport and subsequent decoding;

identifying an advice author;

summarily identifying said advice subject matter;

30 specifying conditions under which said advice is relevant;

providing explanatory material explaining to a user what condition is relevant, why said user is concerned, and what action is recommended; and

providing said user the ability to invoke an automatic execution of said recommended action;

5 wherein said one or more digital documents comprises broadcast messages that do not have an intended recipient or list of recipients.

20. The method of Claim 19, said one or more digital documents further comprising any of:

10 an expires-when clause for causing a document to expire if said clause evaluates to true;

an evaluate-when clause for causing a document to be evaluated for relevance if it evaluates to true;

15 a requires-inspector-library clause for giving the name of a specific inspector library and a URL where it can be found, wherein said specific inspector library must be installed for relevance to be evaluated correctly;

a refers-to clause for giving keyword labeling of systems referenced by a condition associated with a document; and

20 a solution-affects clause for giving keyword labeling of effects of a recommended response.

21. The method of Claim 19, wherein said message body further comprises:

digital authentication features appended to said message to insure its integrity and authenticity.

25

22. The method of Claim 19, wherein said one or more digital documents are packaged as text files which are a valid instance of a MIME file.

23. The method of Claim 22, wherein said MIME file further comprises any of:

header lines for specifying that a message body may be preceded by an extensive message header consisting of a variety of header lines, where individual lines begin with a well known phrase and contain addressing, dating, and related commentary;

5 means for creating new message lines in messages, including any of means for embedding said new message lines in messages and means for registering said new lines with MIME authorities;

means for offering different versions of a same message, with a destination picking an appropriate display method; and

10 a digesting mechanism for packing several complete MIME messages into a single file for internet transport.

24. The method of claim 19, further comprising:

15 one or more ratings blocks for containing information which rates said one or more digital documents according to criteria including any of privacy, security, and usefulness.

25. A method for inspecting any of the properties of a computer, said computer's configuration, contents of said computer's storage devices, said 20 computer's peripherals, said computer's environment, or remote affiliated computers, comprising the steps of:

inspector dispatcher

providing at least one inspector which includes an inspector library and associated methods; and

25 evaluating subexpressions with said at least one inspector;

inspector dispatcher wherein said inspector performs any of mathematico-logical calculations, executes computational algorithms, returns the results of system calls, accesses the contents of storage devices, and queries devices or remote computers.

30

26. The method of Claim 25, further comprising the step of:
providing an inspector dispatcher associated with an advice client
computer for continually performing relevance determination;
wherein said relevance determination is driven by a database of relevance
5 clauses which can be continually evaluated; and
wherein said inspector library contains executable code which is invoked
by said inspector dispatcher as part of said relevance determination process.
27. The method of Claim 25, further comprising the steps of:
10 sending certain relevance clauses to a remote location;
evaluating said clauses; and
returning said clauses after a user is made aware of what is being
transferred; wherein properties of said remote location are learned.
- 15 28. The method of Claim 25, wherein relevance evaluation is driven in a
master-slave relationship by a master machine which tells a slave machine to
evaluate a relevance clause.
29. The method of Claim 26, wherein said method dispatch module performs
20 the steps of:
parsing a clause in a relevance language;
generating a list of method dispatches in response to said parsing step,
wherein specific methods are called in a specific order with specific argument
lists; and
25 systematically carrying out a sequence of method dispatches in an
appropriate order.
30. An inspector for inspecting any of the properties of a computer, said
computer's configuration, contents of said computer's storage devices, said

computer's peripherals, said computer's environment, or remote affiliated computers, said inspector comprising:

an inspector library containing executable code which is invoked as part of a continual relevance evaluation process; and

5 one or more inspector methods for performing any of mathematico-logical calculations, executing computational algorithms, returning the results of system calls, accessing the contents of storage devices, and querying devices or remote computers.

10 31. The apparatus of Claim 30, further comprising:

an inspector dispatcher associated with an advice client computer for continually performing relevance determination, wherein said relevance determination is driven by a database of relevance clauses which can be continually evaluated;

15 wherein said inspector library contains executable code which is invoked by said inspector dispatcher as part of said relevance determination process.

20 32. The apparatus of Claim 30, wherein certain relevance clauses are sent to a remote location, evaluated, and returned, after a user is made aware of what is being transferred, wherein properties of the remote location can be learned.

25 33. The apparatus of Claim 30, wherein relevance evaluation is driven in a master-slave relationship by a master machine which tells a slave machine to evaluate a relevance clause.

34. The inspector of Claim 30, said inspector library further comprising any of:
a declaration of a [Phrase] to be used in a relevance language;
an association of said [Phrase] to a specific method;
a declaration of a new data type to be used in an evaluation process;

a declaration of a calling prototype of said specific method, including a number and required data types of arguments to be supplied to said specific method;

a declaration of a result data type of said specific method;

5 an implementation of said specific method in executable form;

a declaration of special hooks associating code to be called on events, said events including any of inspector dispatcher initialization, inspector dispatcher termination, beginning of inspector dispatcher main evaluation loop, and ending of inspector dispatcher main evaluation loop;

10 a declaration of special hooks associated with creation and maintenance of special caches associated with said specific method; and

an implementation of special event methods and cache methods in executable form.

15 35. The inspector of Claim 30, further comprising:

a module for linking said inspector library into said inspector dispatcher with all declarations evaluated, resulting in changes to said inspector dispatcher's internal data structures, wherein new method invocations become available to said inspector dispatcher.

20

36. The inspector of Claim 35, further comprising:

a syntax table for providing said resulting changes to all allowed phrases and associated data types on which they operate; and

25 a dispatch table for systematically determining an associated executable method for given phrase and data types.

37. The inspector of Claim 30, wherein a plurality of inspector libraries are installed in an instance of said inspector dispatcher to define a set of recognized [Phrase]s in a relevance language, a set of allowable data types at evaluation
30 time, and a set of methods associated with those data types.

38. The inspector of Claim 30, wherein said inspector libraries are linked into said inspector dispatcher at the time said inspector dispatcher is initialized; and wherein declaration routines are invoked, new [Phrase]s are installed in a lexical analysis table of a relevance language, and said new [Phrase]s are associated to certain method invocations when said linking occurs.

39. The inspector of Claim 30, further comprising any of:
a base layer comprising a mechanism for elementary operations including any of arithmetic and logic, which are system-independent;
a system-specific layer associated with a specific operating system;
one or more vendor-specific layers for providing access to specific hardware devices and software products; and
additional layers as appropriate, based on other advice providers.

15

40. The inspector of Claim 30, wherein said inspector comprises any of:
a system specific inspector for accessing properties of an operating system and allowing advice to be written to verify the existence and configuration of attached devices and other subsystems;
a registry inspector for enabling a relevance language to refer to and evaluate properties of a registry database;
a preferences inspector for enabling a relevance language to refer to and evaluate properties of a preferences file of a specific application;
a database inspector for enabling a relevance language to access fields in a database; and
a user profile inspector for enabling a relevance language to refer to data stored in a user profile.

20
25

41. The inspector of Claim 30, further comprising:

a template file for describing a collection of variables to which an advice provider plans to refer in advisories;

wherein said template file is placed at an advice site and is automatically gathered at a synchronization time;

5 wherein said template file is used to drive an editing module on a client computer which presents a user with a list of template variable names and a list of their current values or blanks if they have not previously been defined, and

wherein said user can fill in said blank fields and edit other fields.

10 42. The inspector of Claim 30, wherein said inspector comprises:

a remote inspector for inspecting properties of other communicating devices;

wherein said remote inspector inspects any of:

remote physical measurements;

15 remote device queries;

remote computation;

remote database queries; and

remote relevance invocation.

20 43. The inspector of Claim 30, wherein said inspector comprises:

a program log inspector for enabling a relevance language to refer to data stored in a specific log file or files associated with any specific application, wherein said specific log file may comprise any of a web browser log, a telecommunications log, a fax log, or a clickstream log.

25

44. In a system including computational devices connected by a communications network, said system comprising a communications apparatus for linking an information provider to information consumer, said communications apparatus comprising specific units of advice to be shared, digital documents
30 conveying said advice, an advice provider for broadcasting said advice in the

form of advisories, an advice consumer for receiving said advisories, wherein advisories are broadcast over said communications network from said advice provider to said advice consumer, a communications protocol for narrowly-focused targeting of said advisories to said advice consumer by automatically

5 matching advisories with an advice consumer for whom said advisories are relevant, and an inspector dispatcher associated with an advice client computer for continually performing relevance determination, wherein said relevance determination is driven by a database of relevance clauses which can be continually evaluated, at least one inspector, comprising:

10 an inspector library and associated methods for evaluating subexpressions with said at least one inspector;

wherein said inspector library contains executable code which is invoked by said inspector dispatcher as part of said relevance determination process; and

wherein said inspector performs any of mathematico-logical calculations,

15 executes computational algorithms, returns the results of system calls, accesses the contents of storage devices, and queries devices or remote computers.

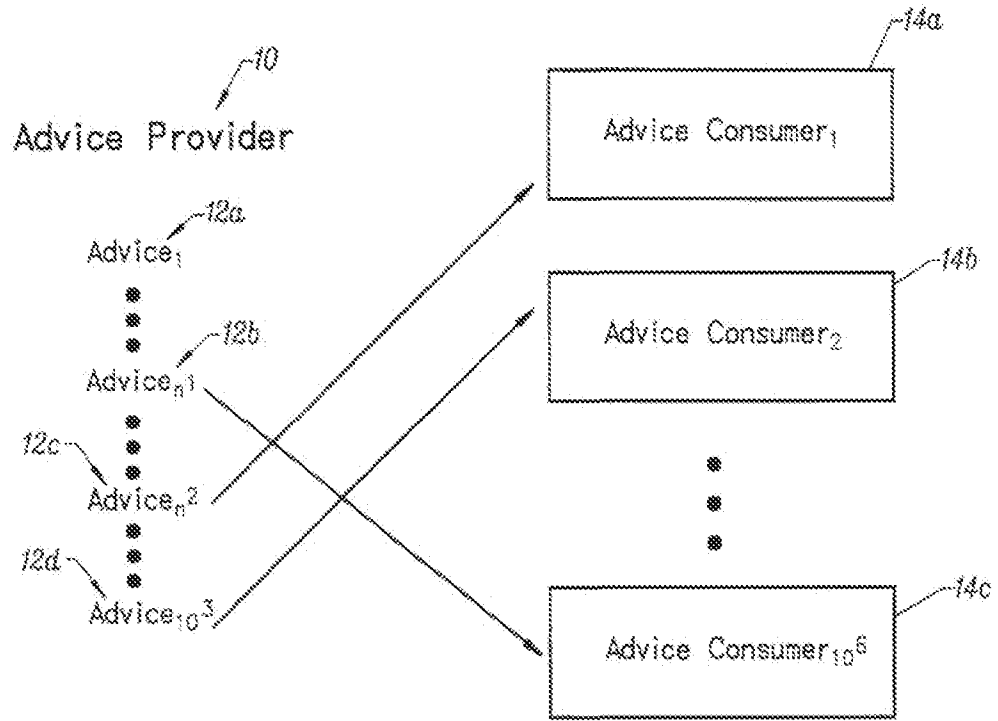


FIG. 1

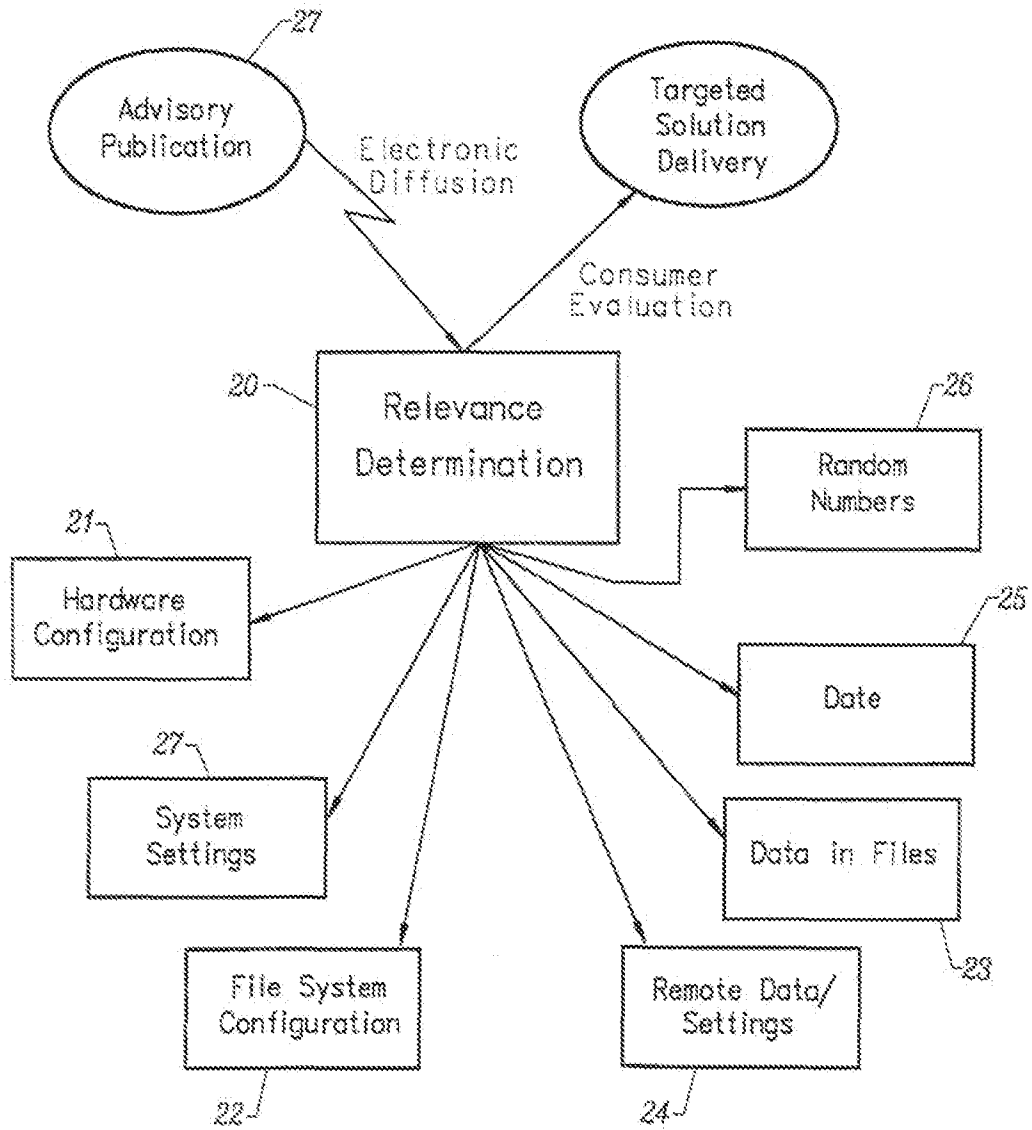


FIG. 2

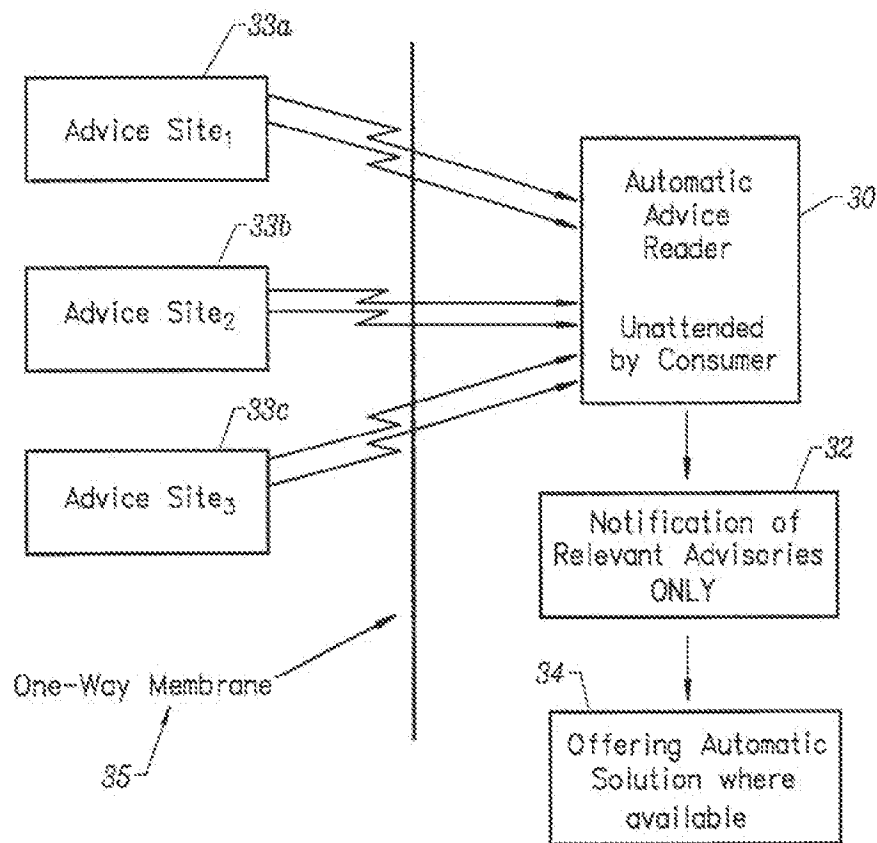


FIG. 3

4/25

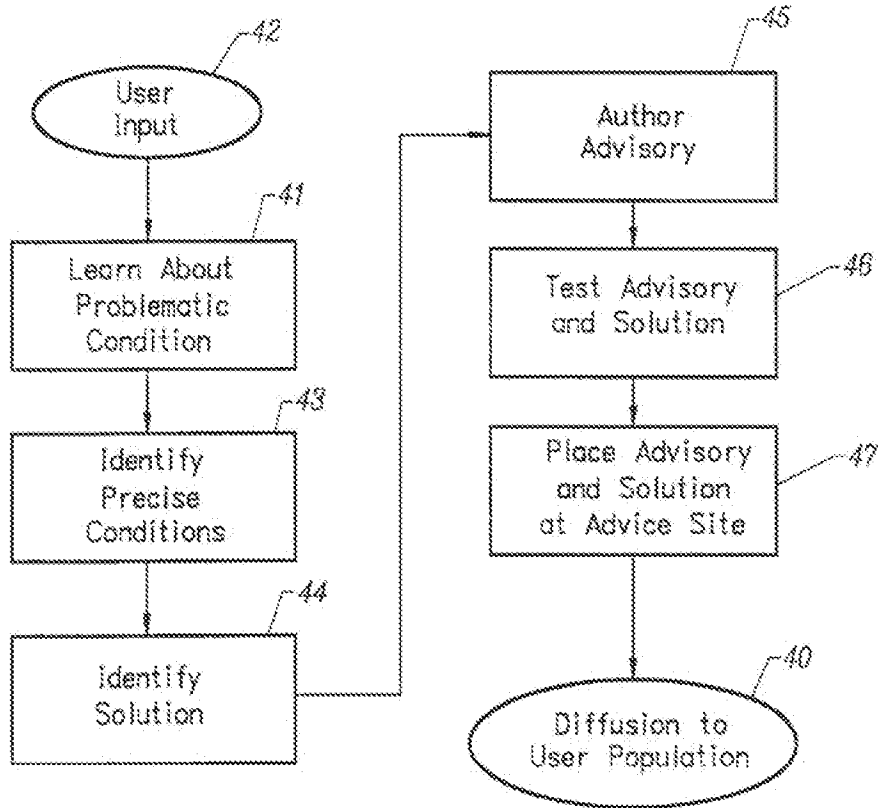


FIG. 4

5/25

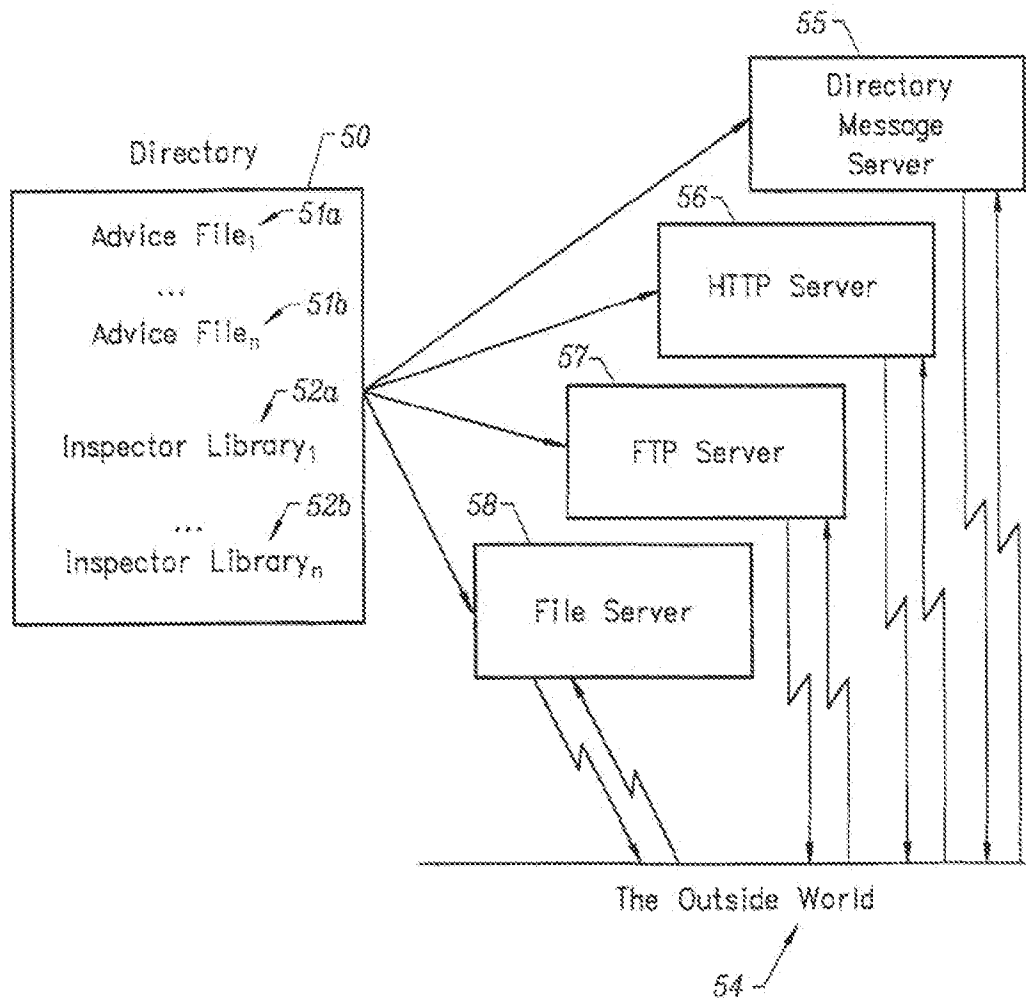


FIG. 5

6/25

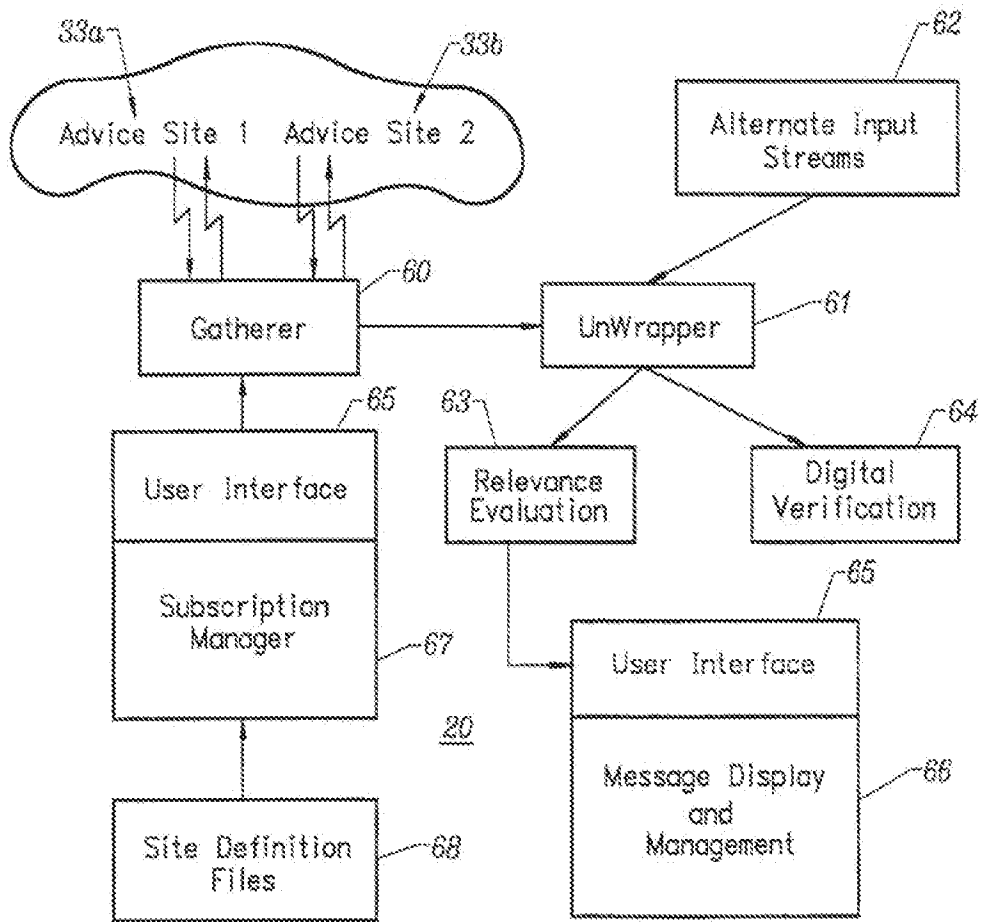


FIG. 6

7/25

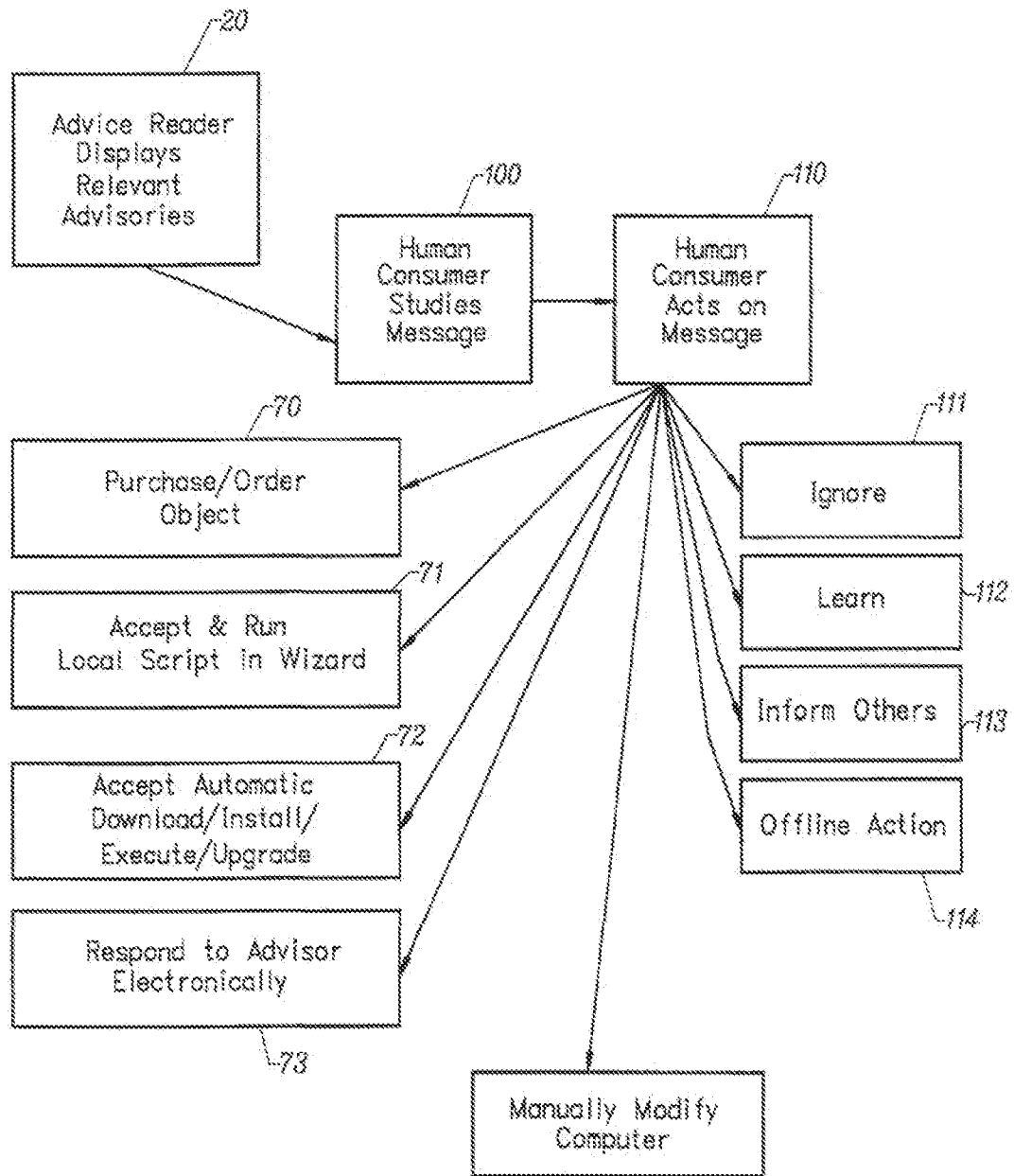


FIG. 7

8/25

DATE: SAT MAR 21 1998 17:54:06 - 0800
FROM: JOE ADVISOR <ja@advisories.com>
MIME-Version 1.0
ORGANIZATION: Universe Communications, Inc.

SUBJECT: You need to upgrade Advice Reader

RELEVANT-WHEN: version of file "Advice.exe"
< 5.0

CONTENT-TYPE: text/html; charset us=ascii

<HTML><BODY>
Click here to <A HREF = "http://advisories.com/d"
download the latest version of Advice Reader.
</BODY></HTML>

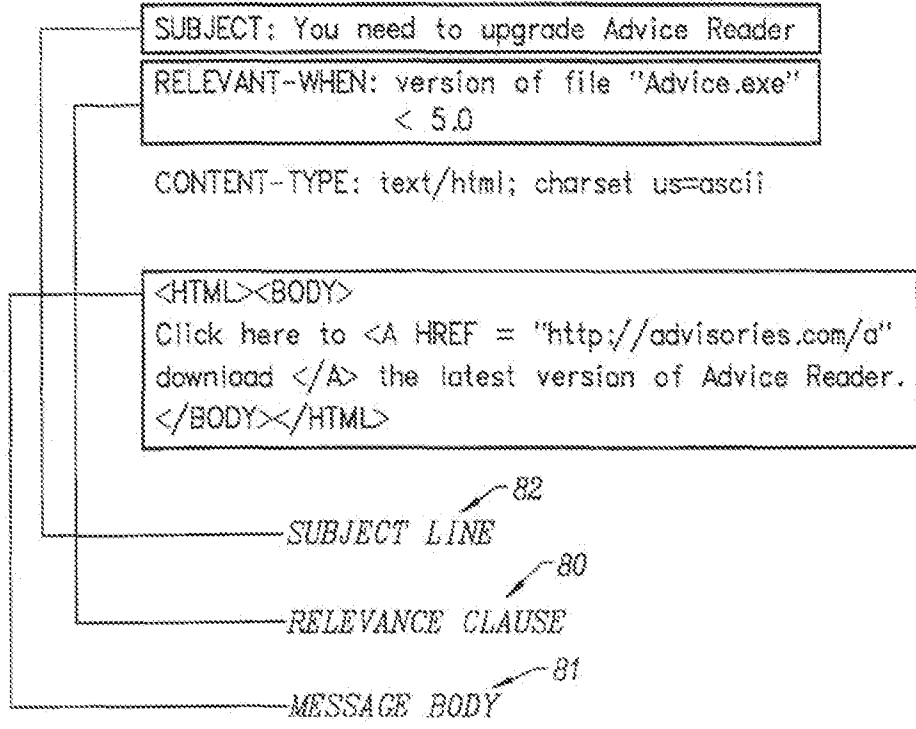


FIG. 8

9/25

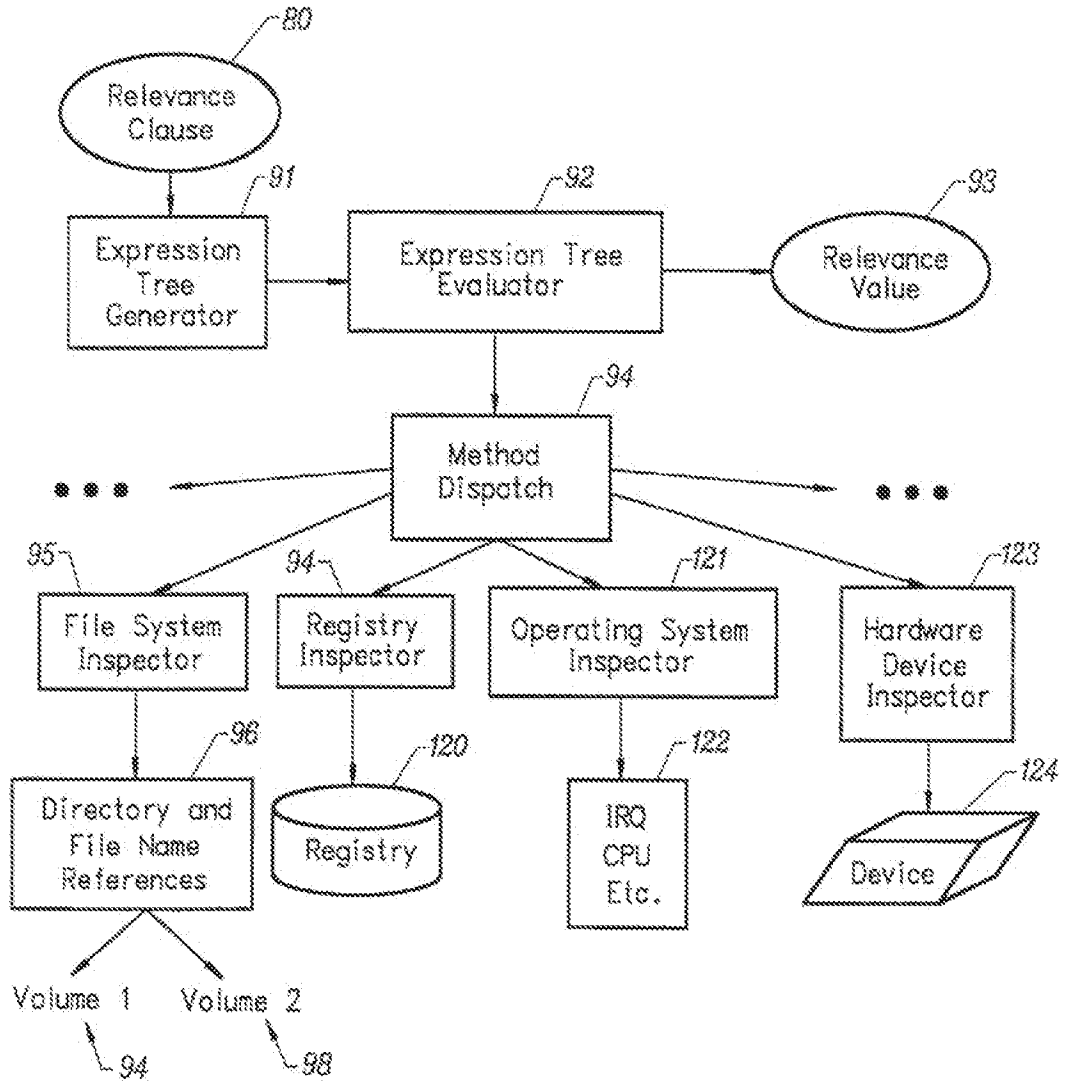


FIG. 9

10/25

Expression:

(not exists File 'System' of System Folder) or
 (not exists File 'MacTCP DNR' of System Folder)

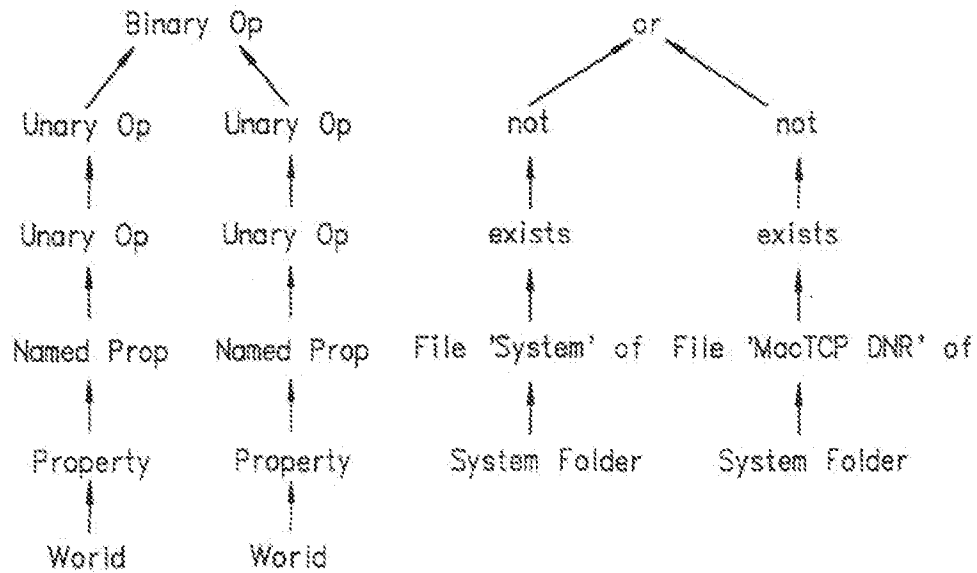


FIG. 10

11/25

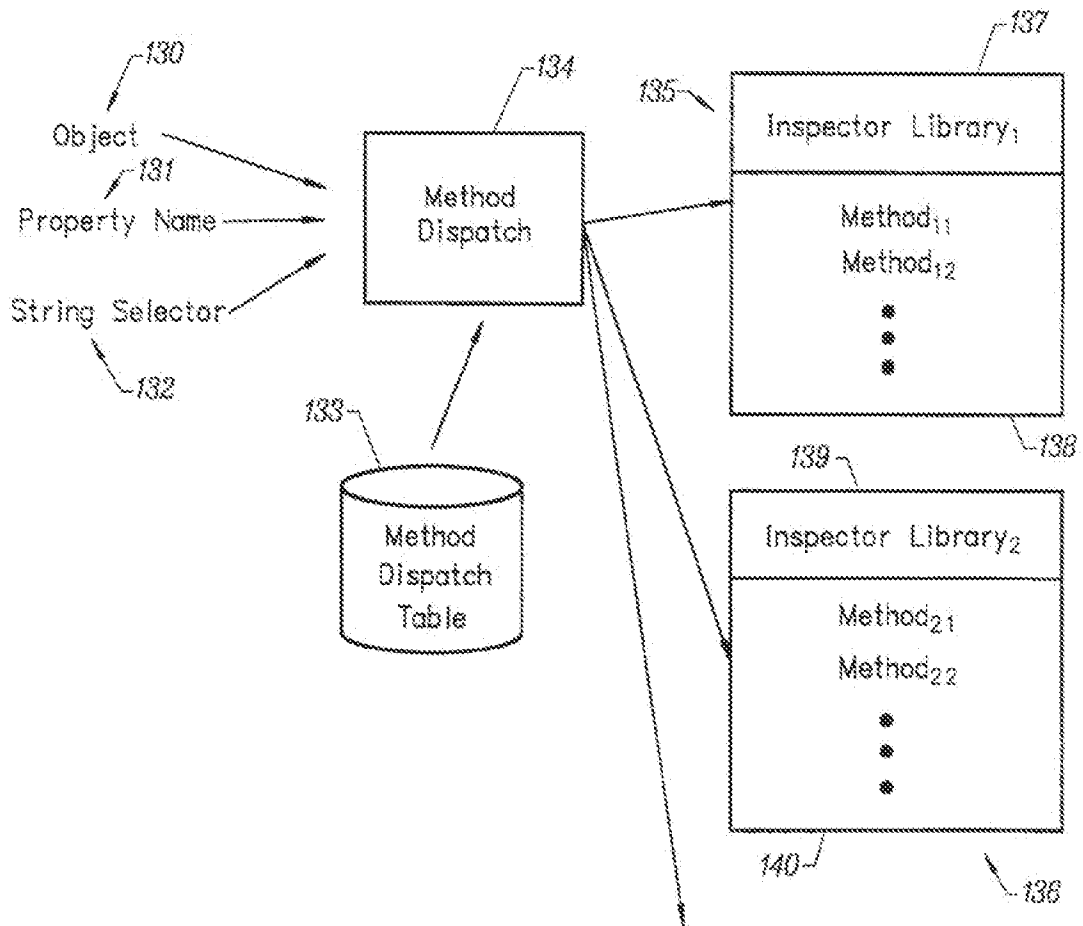


FIG. 11

12/25

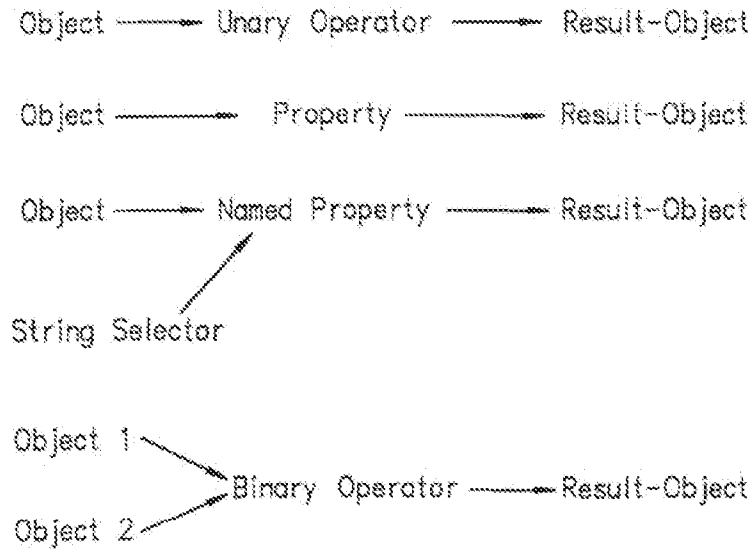


FIG. 12

13/25

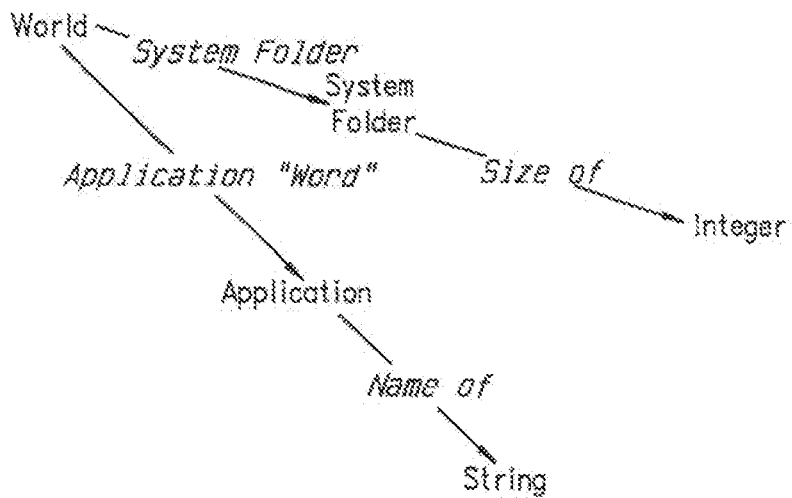
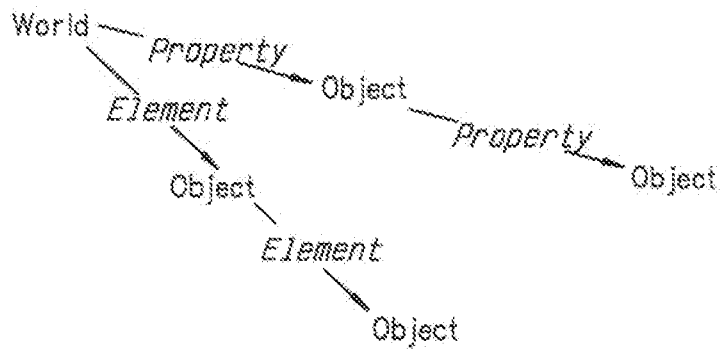


FIG. 13

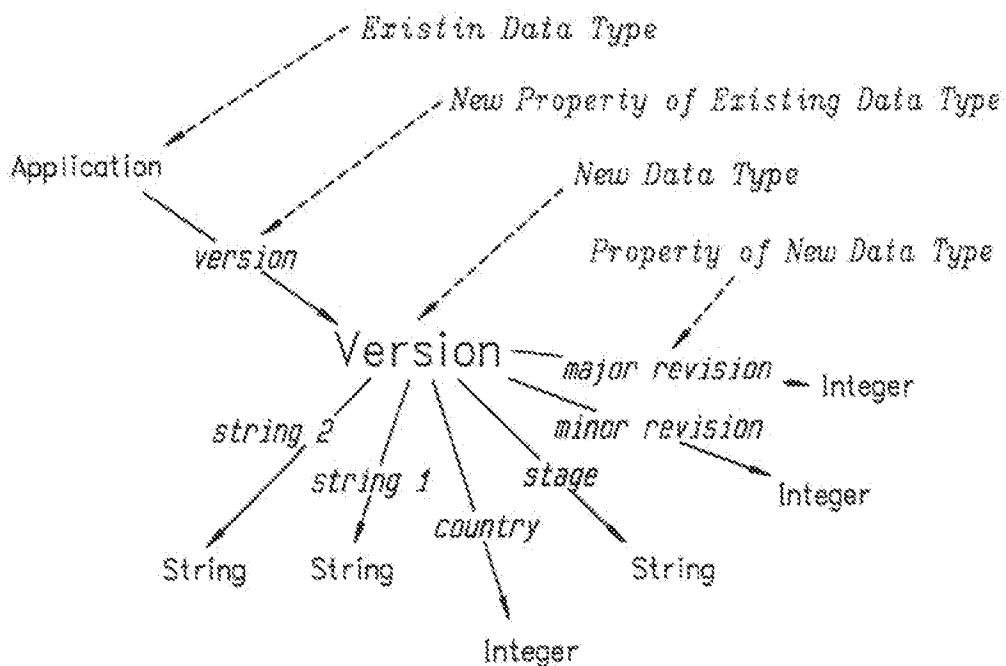


FIG. 14

15/25

Inspector Library

Macintosh "Version" Inspector

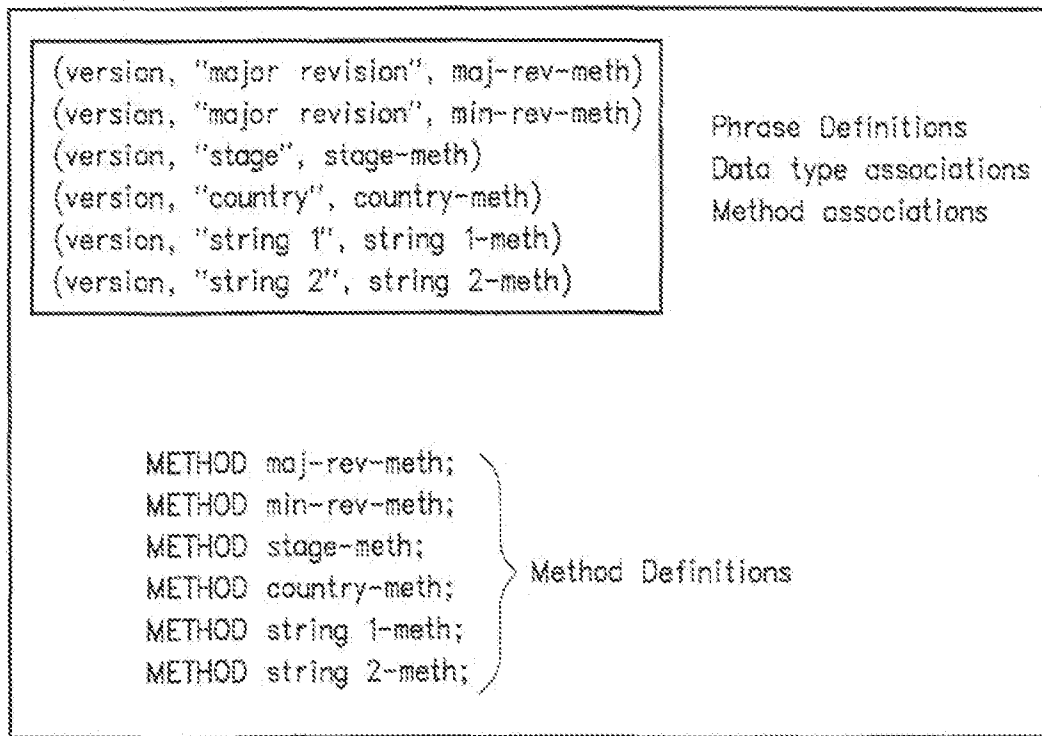


FIG. 15

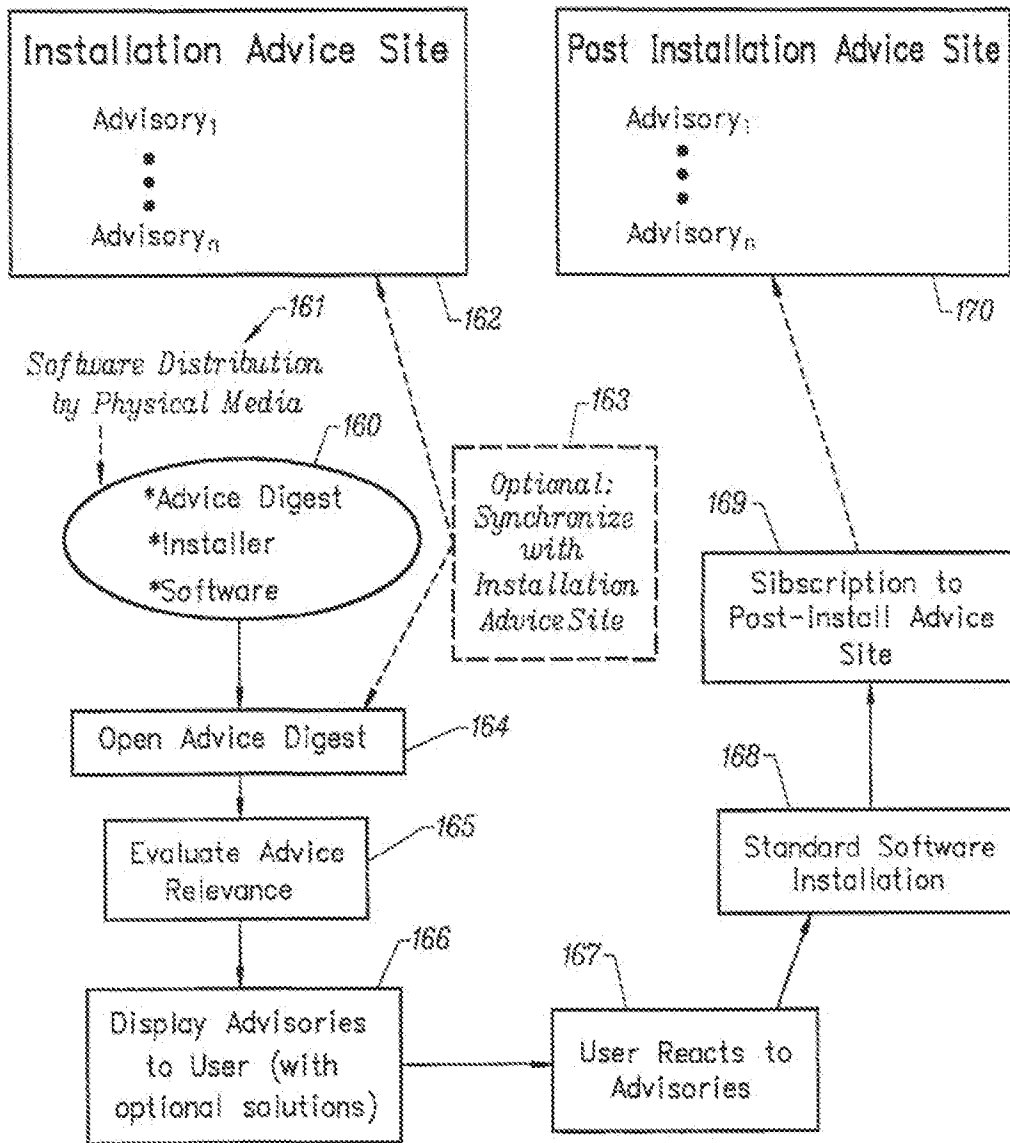


FIG. 16

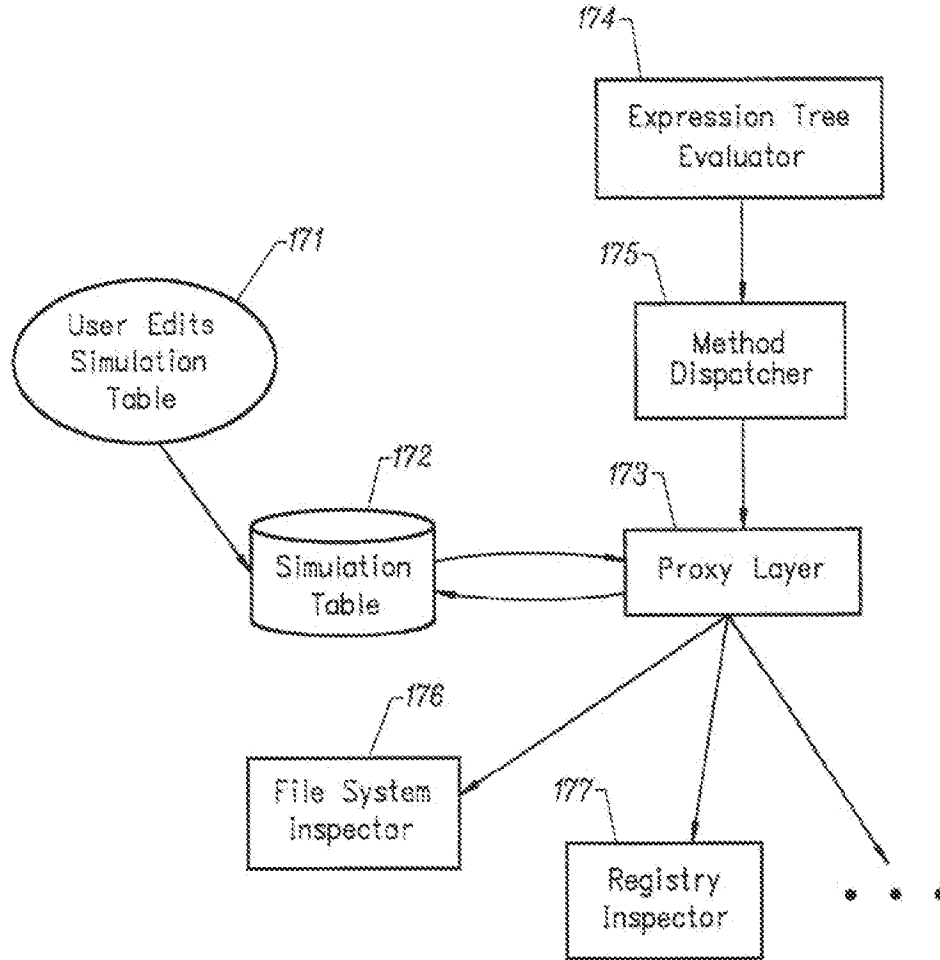


FIG. 17

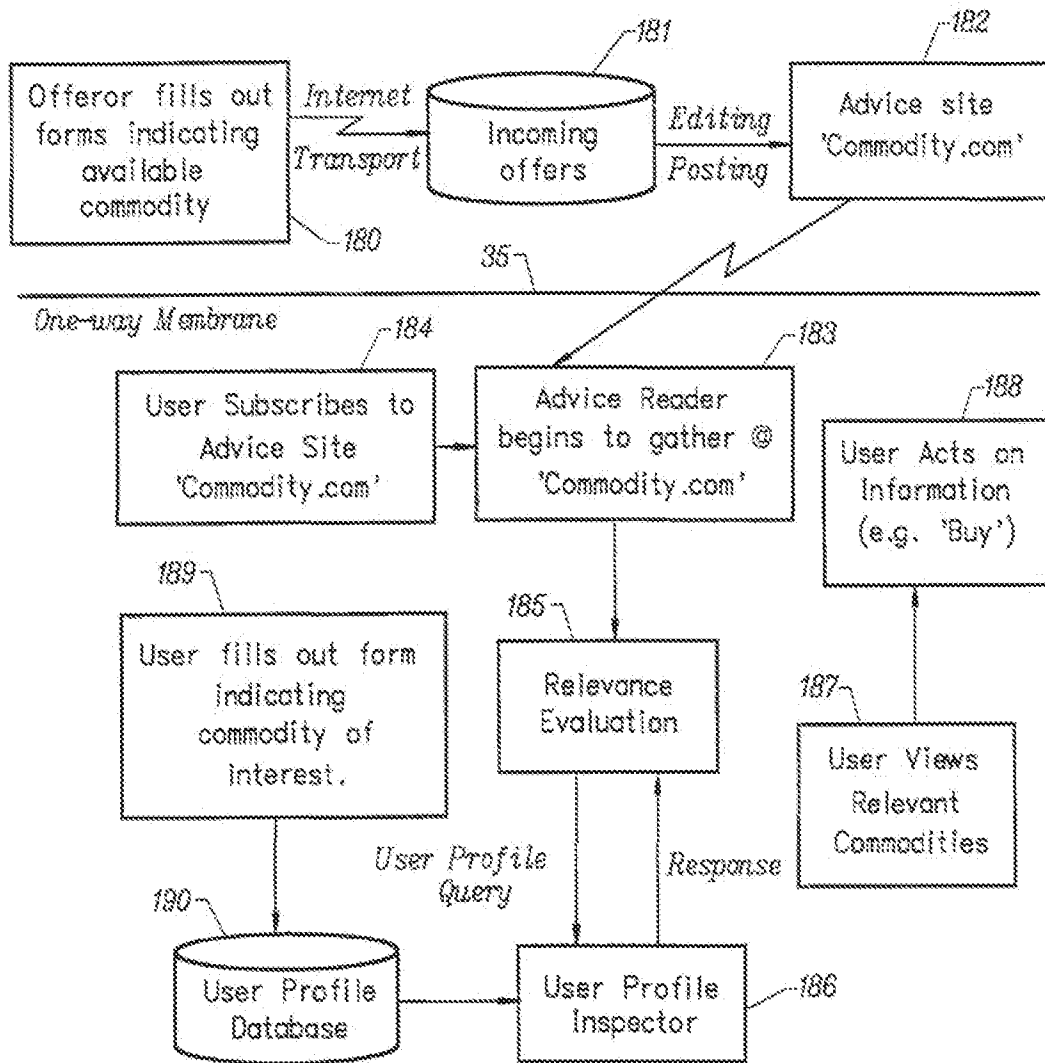


FIG. 18

19/25

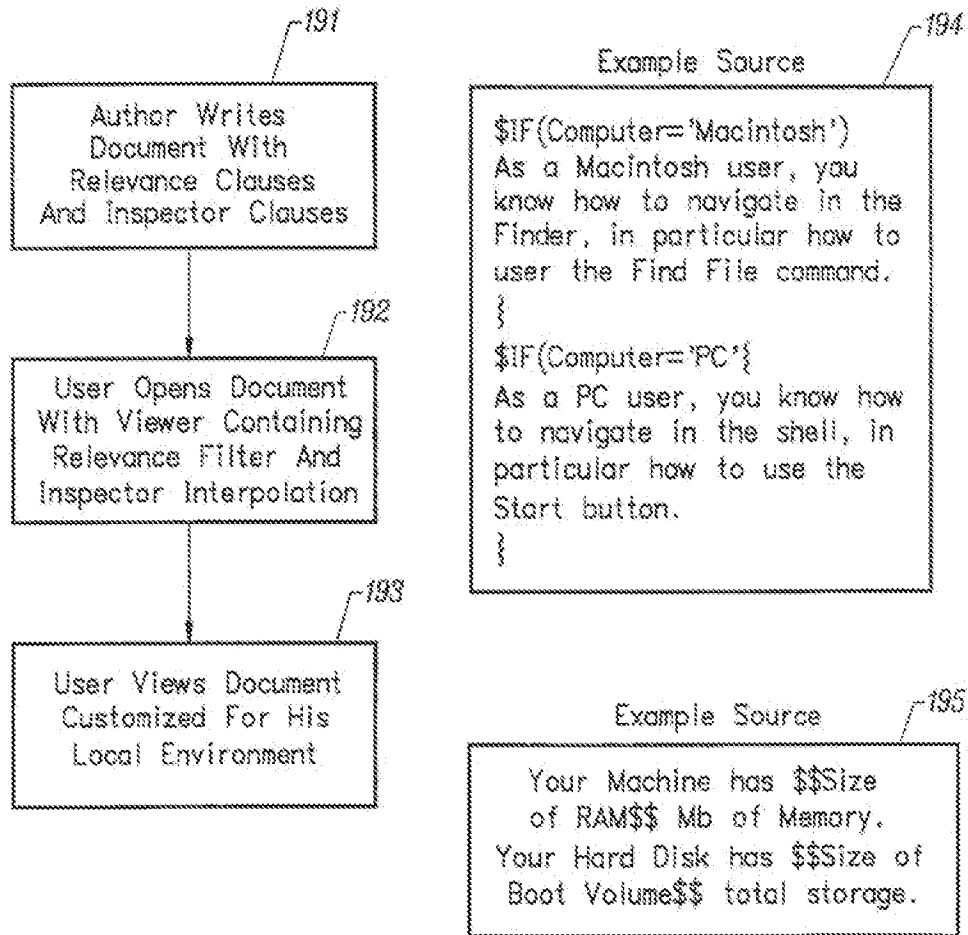


FIG. 19

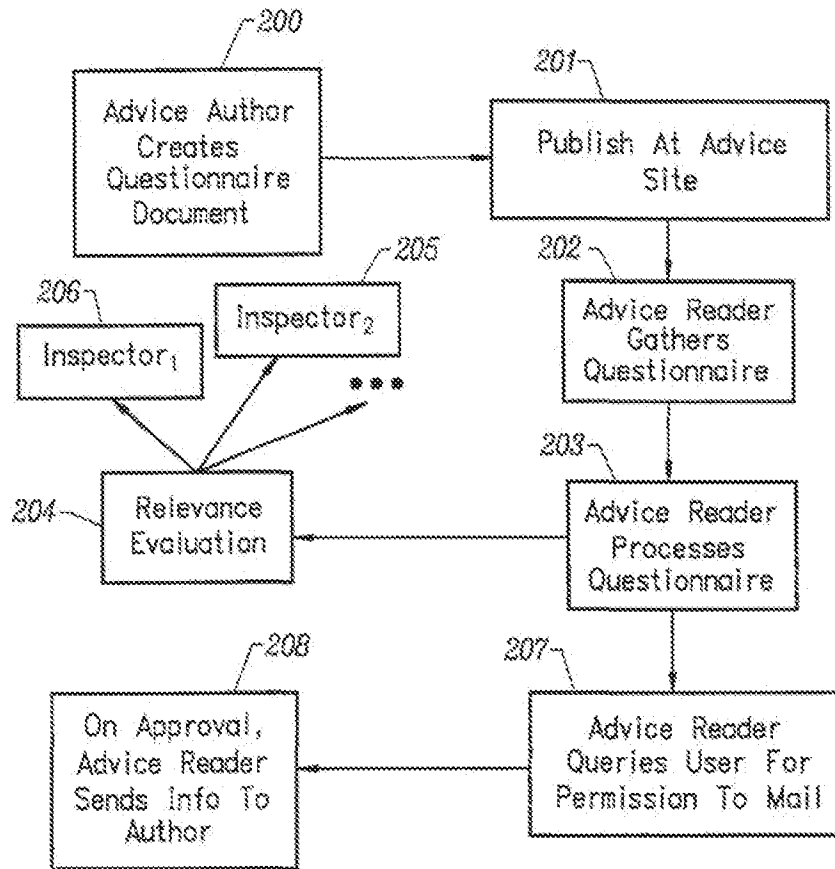


FIG. 20

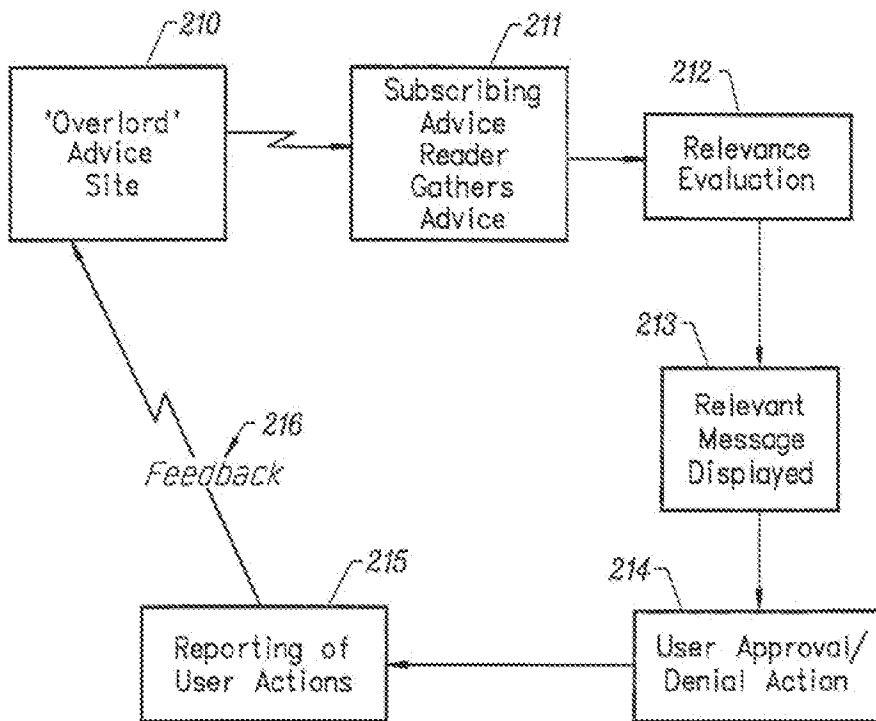


FIG. 21

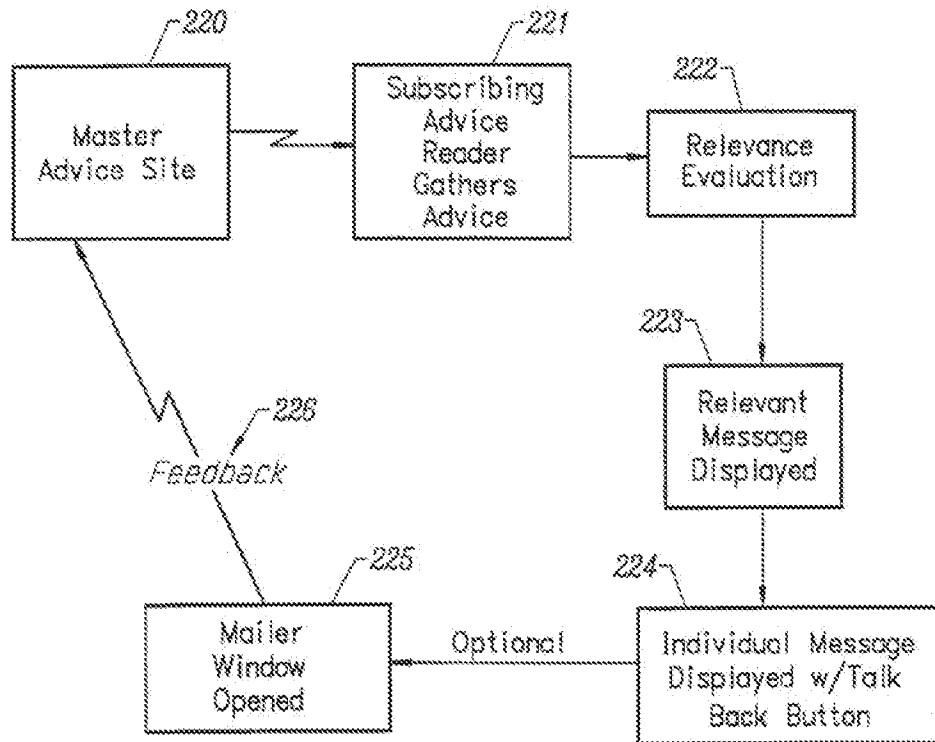


FIG. 22

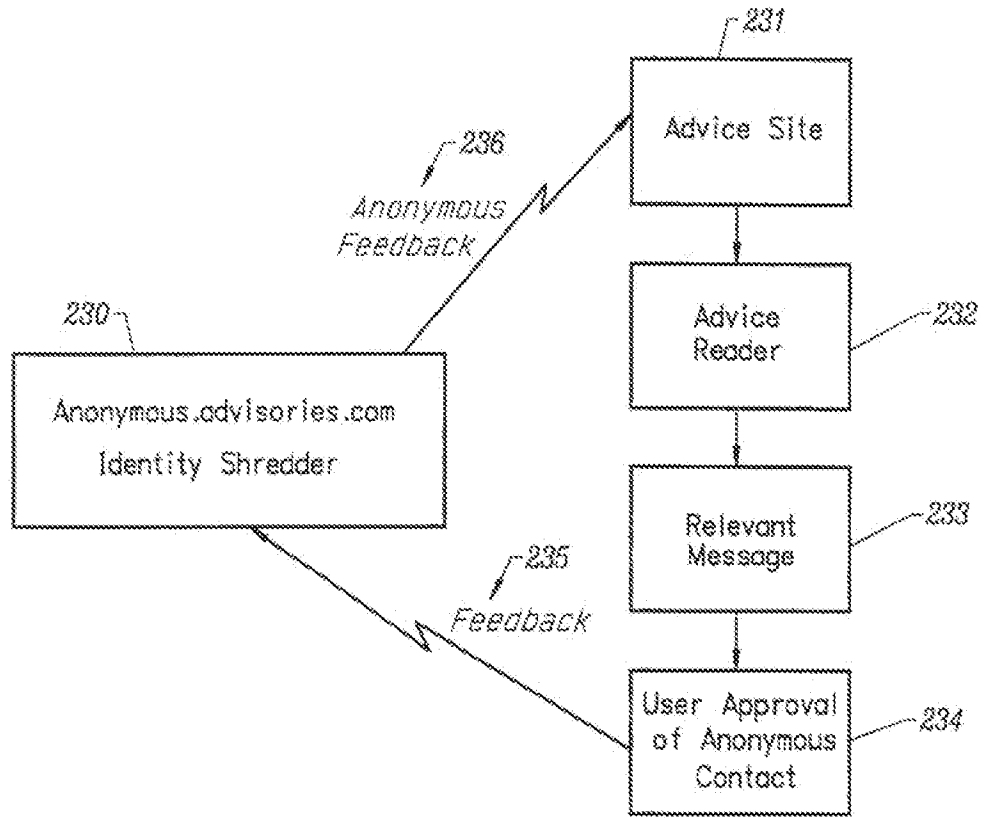


FIG. 23

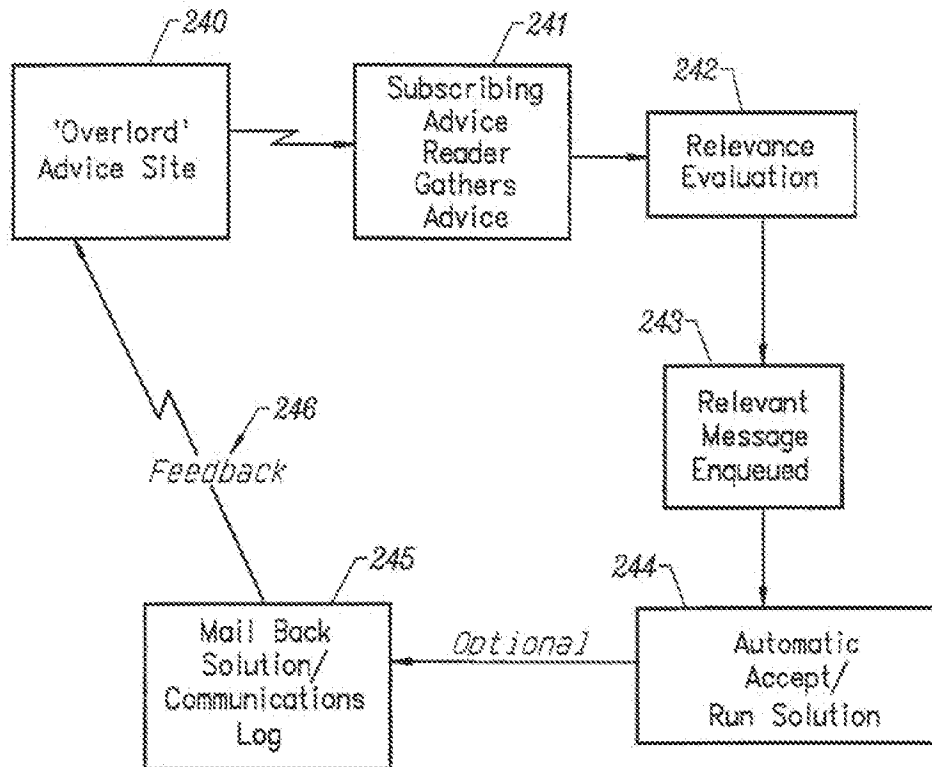


FIG. 24

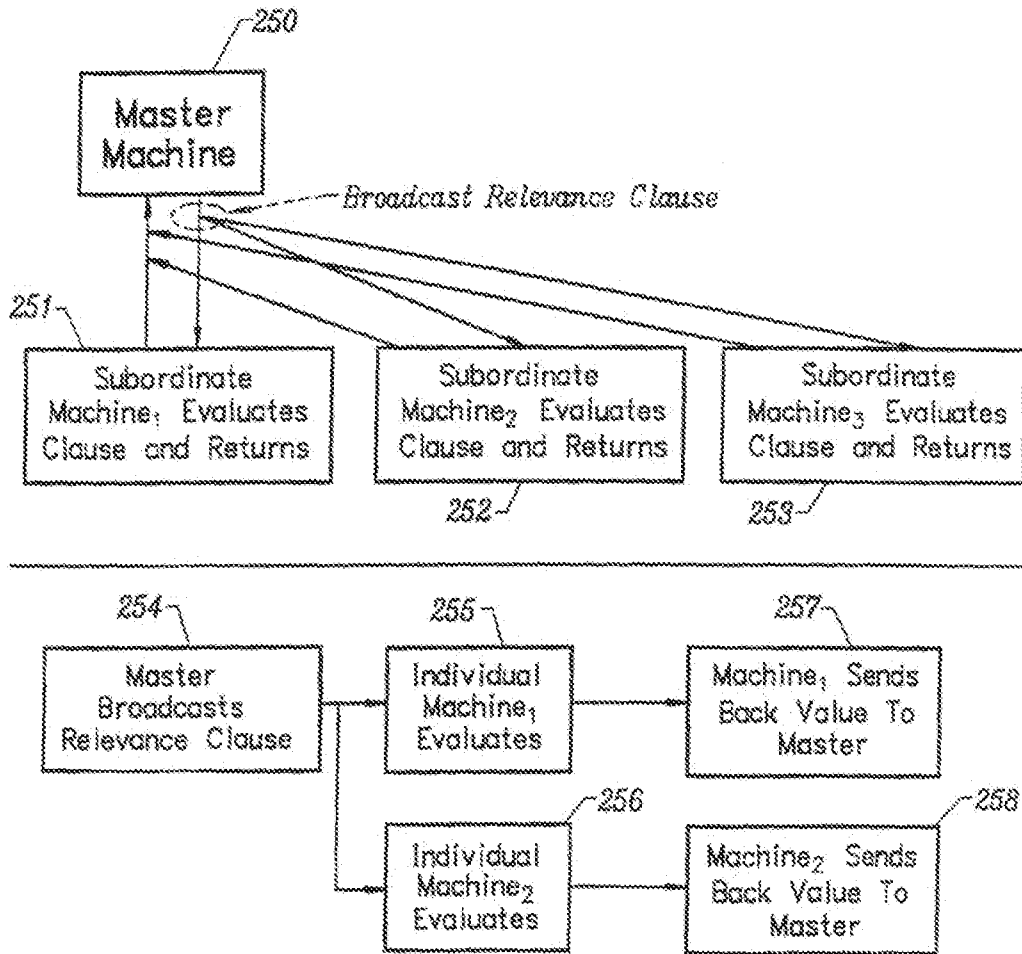


FIG. 25

INTERNATIONAL SEARCH REPORT

International Application No.
PCT/US 99/19751

A. CLASSIFICATION OF SUBJECT MATTER IPC 7 G06F17/60 H04L12/58				
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) IPC 7 G06F H04L				
Documentation searched other than minimum documentation in the extent that such documents are included in the fields searched				
Electronic data base consulted during the international search (name of data base and, where practical, search terms used)				
C. DOCUMENTS CONSIDERED TO BE RELEVANT				
Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim no.		
X	US 5 649 186 A (FERGUSON GREGORY J) 15 July 1997 (1997-07-15) column 2, line 51 -column 4, line 5 column 5, line 61 -column 7, line 6	1, 3, 4, 6-10, 13-24		
X Y A	WO 97 10558 A (BELL COMMUNICATIONS RES) 20 March 1997 (1997-03-20) page 3, line 20 -page 4, line 12 page 10, line 5 -page 14, line 20 --- -/-	1, 6-10 44 2-4		
<input checked="" type="checkbox"/> Further documents are listed in the continuation of box C.				
<input checked="" type="checkbox"/> Patent family members are listed in annex.				
* Special categories of cited documents:				
<table style="width: 100%; border: none;"> <tr> <td style="width: 50%; border: none; vertical-align: top;"> *A* document defining the general state of the art which is not considered to be of particular relevance *E* earlier document but published on or after the international filing date *L* document which may throw doubts on priority claims or which is cited to establish the publication date of another citation or other special reason (as specified) *Q* 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 </td> <td style="width: 50%; border: none; vertical-align: top;"> *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. *Z* document member of the same patent family </td> </tr> </table>			*A* document defining the general state of the art which is not considered to be of particular relevance *E* earlier document but published on or after the international filing date *L* document which may throw doubts on priority claims or which is cited to establish the publication date of another citation or other special reason (as specified) *Q* 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. *Z* document member of the same patent family
A document defining the general state of the art which is not considered to be of particular relevance *E* earlier document but published on or after the international filing date *L* document which may throw doubts on priority claims or which is cited to establish the publication date of another citation or other special reason (as specified) *Q* 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. *Z* document member of the same patent family			
Date of the actual completion of the international search		Date of mailing of the international search report		
19 January 2000		03/02/2000		
Name and mailing address of the ISA European Patent Office, P. B. 5818 Patentlaan 2, NL - 3200 HV Rijswijk Tel. (+31-70) 340-2040, Tx. 31 651 600 01, Fax: (+31-70) 340-3018		Authorized officer Bowler, A		

INTERNATIONAL SEARCH REPORT

International Application No
PCT/US 99/19751

C (Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	PALME J ET AL: "Issues when designing filters in messaging systems" COMPUTER COMMUNICATIONS,NL,ELSEVIER SCIENCE PUBLISHERS BV, AMSTERDAM, vol. 19, no. 2, 11 January 1996 (1996-01-11), page 98-101 XP004032392 ISSN: 0140-3664 page 100, column 1-2	44
A	-----	30-43
X	US 5 491 791 A (KISTENMACHER JOHN C ET AL) 13 February 1996 (1996-02-13) column 2, line 47 -column 3, line 42	25,30
X	GABBER E ET AL: "How to make personalized Web browsing simple, secure and anonymous" FINANCIAL CRYPTOGRAPHY FIRST INTERNATIONAL CONFERENCE, FC '97. PROCEEDINGS, FINANCIAL CRYPTOGRAPHY FIRST INTERNATIONAL CONFERENCE, FC'97. PROCEEDINGS, ANGUILLA, 24-26 FEB. 1997, pages 17-31, XP002059819 1997, Berlin, Germany, Springer-Verlag, Germany ISBN: 3-540-63594-7 page 17-31	1,2,5
A	PATENT ABSTRACTS OF JAPAN vol. 018, no. 593 (E-1629), 11 November 1994 (1994-11-11) & JP 06 224938 A (FUJITSU LTD), 12 August 1994 (1994-08-12) abstract	1,2,5
P,X	US 5 892 909 A (GRASSO CHARLES A ET AL) 6 April 1999 (1999-04-06) column 4, line 28-63 column 6, line 56 -column 18, line 22	1,3,4, 6-44
	----- DATABASE INTERNET 'Online! PC World Communications Inc. http://www.tipworld.com , 1999 XP002128141	
	----- DATABASE INTERNET 'Online! New Technology Holdings LLC. http://select.infobeat.com , 1999 XP002128142	

INTERNATIONAL SEARCH REPORT

Information on patent family members

International Application No.

PCT/US 99/19751

Patent document cited in search report		Publication date	Patent family member(s)	Publication date
US 5649186	A	15-07-1997	NONE	
WO 9710558	A	20-03-1997	NONE	
US 5491791	A	13-02-1996	NONE	
JP 06224938	A	12-08-1994	NONE	
US 5892909	A	06-04-1999	NONE	

Form PCT/ISA/210 (patent family annex) (July 1992)

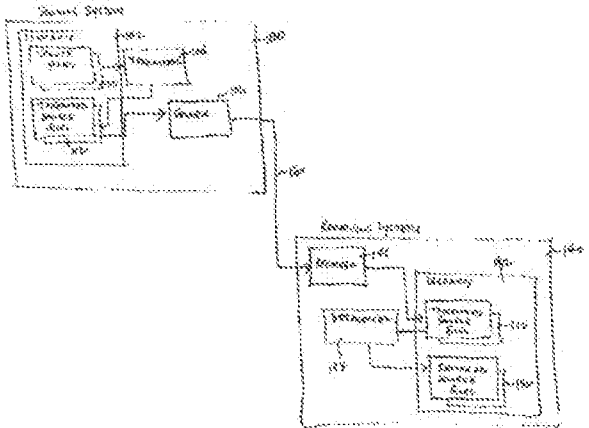


INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(51) International Patent Classification ⁶ : G06F 15/16, 7/00, 13/38		A1	(11) International Publication Number: WO 00/23905
			(43) International Publication Date: 27 April 2000 (27.04.00)
(21) International Application Number: PCT/US99/24919		(83) Designated States: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, 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, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YD, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, 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), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).	
(22) International Filing Date: 21 October 1999 (21.10.99)			
(30) Priority Data: 09/177,444 21 October 1998 (21.10.98) US			
(71)(72) Applicant and Inventor: PORTER, Swain, W. [US/US]; 12511 89th Court, N.E., Kirkland, WA 98034 (US).			
(74) Agents: AUYEUNG, Aloysius, T., C. et al.; Blakely, Sokoloff, Taylor & Zafman, 7th floor, 12400 Wilshire Boulevard, Los Angeles, CA 90025-1026 (US).			

Published
*With international search report.
Before the expiration of the time limit for amending the claims and to be republished in the event of the receipt of amendments.*

(54) Title: **TOKEN BASED SOURCE FILE COMPRESSION/DECOMPRESSION AND ITS APPLICATION**



(57) Abstract

A source file (110) is transformed into a tokenized form (112) with at least language elements present in the source file (110) in an original form being substituted with corresponding tokens to reduce transmission bandwidth required to provide the source file (112) to another computer system (140). In one embodiment, operands present in the source file (110) in the original form are also substituted with corresponding tokens, and entries mapping the operand substituting tokens to the operands are maintained in a symbol table. The symbol table is also provided to the other computer system (140). In one embodiment, the computer system is a web server (100), and the source file (110) is a web page. The other computer system is a client computer system (140) requesting the web page from the web server. The requested web page is provided by the web server (100) to the client computer system (140) in the tokenized form, and the client computer system (140) is equipped with a browser having been enhanced with the ability to restore the provided web page to its original form.

FOR THE PURPOSES OF INFORMATION ONLY

Codes used to identify State party to the PCT on the front pages of pamphlets publishing international applications under the PCT.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LI	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SS	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece	ML	Mali	TR	Turkey
BG	Bulgaria	HU	Hungary	MN	Mongolia	TT	Trinidad and Tobago
BI	Burundi	IE	Ireland	MR	Mauritania	UA	Ukraine
BR	Brazil	IL	Israel	MW	Malawi	UG	Uganda
BY	Belarus	IS	Iceland	NL	Netherlands	US	United States of America
CA	Canada	IT	Italy	MX	Mexico	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NE	Niger	VN	Viet Nam
CG	Congo	KE	Kenya	NL	Netherlands	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	NO	Norway	ZW	Zimbabwe
CI	Côte d'Ivoire	KP	Democratic People's Republic of Korea	NZ	New Zealand		
CM	Cameroon	KR	Republic of Korea	PL	Poland		
CN	China	KZ	Kazakhstan	PT	Portugal		
CU	Cuba	LC	Saint Lucia	RO	Romania		
CZ	Czech Republic	LI	Liechtenstein	RU	Russian Federation		
DE	Germany	LK	Sri Lanka	SD	Sudan		
DK	Denmark	LR	Liberia	SE	Sweden		
EE	Estonia			SG	Singapore		

Token Based Source File Compression/Decompression And Its
Application

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to the field of computer systems. More specifically, the present invention relates to methods and apparatuses associated with the distribution or provision of source files to other computer systems.

2. Background Information

With the recent advances in microprocessor, telecommunication and networking technology, increasing number of computer systems are being networked together. In turn, increasing number of situations arise where source files have to be transferred from one computer system to another computer system. The term "source files" as used herein includes but not limited to compilable or interpretable source files written in machine programming languages such as C, C++, HTML, XML, JAVA™, JAVAScript, and so forth. For example, everyday, millions of users are connected to the Internet downloading web pages from a multitude of web sites. Similarly, millions of users are doing the same within thousands of "corporate" intranets. In a new user centric software distribution paradigm, disclosed in co-pending U.S. patent application number <to be inserted>, filed contemporaneously, and entitled "User centric source control", it is envisioned that software products are distributed to client systems or their proxies in source form. All these activities further exacerbate the well known bandwidth problem confronting private as well as public networks. (Note that the "user centric" approach to source control contributes to the bandwidth problem only in the sense that the

approach is expected to increase the demand for source file transmission. For a given set of source files, its delta feature actually reduces bandwidth demand for maintaining and updating the set of source files.)

Various compression/decompression techniques are known in the art in the data or link layer to reduce the amount of data that needs to be transmitted from a sender to a receiver. For examples, a dictionary based approach replacing previously transmitted character string (e.g. "this string has been sent before") with a code is often employed in modem communication; the run length encoding approach (e.g. encoding a series of 10 0-bit as [0, 10]) is often employed in video signal compression, and a code based approach supplying the identity of a linear excitation code vector is often employed in audio compression. However, notwithstanding these multitude of data or link layer compression/decompression techniques available, as evident by the amount of research and development going into Quality of Service, Bandwidth Reservation, Virtual Private Network, and so forth, the problem of bandwidth in private as well as public networks is expected to remain with the computer and communication industry for years to come.

Thus, further improvement or contribution to alleviating the bandwidth problem, in particular, improvement that further advances the connectivity and exchange of information between computer systems, is desired.

SUMMARY OF THE INVENTION

In accordance with the present invention, a source file is provided from one computer system to another in a tokenized form to reduce transmission bandwidth requirement. In the tokenized form, at least language elements present in the source file in its original form are substituted with corresponding tokens. In one embodiment, operands present in the source file in the original form are also substituted with corresponding tokens, and entries mapping the operand substituting tokens to the operands are maintained in a

symbol table. In this case, the symbol table is also provided to the other computer system.

In one embodiment, the source file is also in either a base or delta form. A new entry is created for the symbol table whenever a new operand is encountered and substituted. In this case, subsequent to the initial provision of the symbol table, new entries associated with a base/delta source file are also provided to the other computer system to update the previously provided symbol table.

In one embodiment, the base/delta source files also have associated versioning control information. In this case, the versioning control information is also provided to the other computer system.

In one embodiment, the computer system is a web server, and the source file is a web page. The other computer system is a client computer system requesting the web page from the web server. The requested web page is provided by the web server to the client computer system in the tokenized form, and the client computer system is equipped with a browser having been enhanced with the ability to restore the provided web page to its original form. In one embodiment, the web page is also in a base/delta form having associated versioning control information, and the client computer system's browser is enhanced with the ability to reconstitute the requested web page using the associated versioning control information.

BRIEF DESCRIPTION OF DRAWINGS

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

Figures 1a-1c illustrate three exemplary embodiments of the token based compressed source file transmission of the present invention;

Figures 2a-2b illustrate one exemplary embodiment each for a collection of tokens and an associated symbol table suitable for use to practice the present invention;

Figure 3a-3b illustrate one exemplary embodiment each of the sender and the receiver's method steps in accordance to the present invention;

Figure 4 illustrates one exemplary application of the present invention to web servers and client systems accessing web servers; and

Figure 5 illustrates one embodiment of an exemplary computer system suitable for use as either a sender or a receiver system to practice the present invention.

DETAILED DESCRIPTION OF THE INVENTION

In the following description, various aspects of the present invention will be described. However, it will be apparent to those skilled in the art that the present invention may be practiced with only some or all aspects of the present invention. For purposes of explanation, specific numbers, materials and configurations are set forth in order to provide a thorough understanding of the present invention. However, it will also be apparent to one skilled in the art that the present invention may be practiced without the specific details. In other instances, well known features are omitted or simplified in order not to obscure the present invention.

Parts of the description will be presented in terms of operations performed by a computer system, using terms such as tables, files, data and the like, consistent with the manner commonly employed by those skilled in the art to convey the substance of their work to others skilled in the art. As well understood by those skilled in the art, these quantities take the form of electrical, magnetic, or optical signals capable of being stored, transferred, combined, and

otherwise manipulated through mechanical and electrical components of a digital system; and the term digital system include general purpose as well as special purpose data processing machines, systems, and the like, that are standalone, adjunct or embedded.

Various operations will be described as multiple discrete steps performed in turn in a manner that is most helpful in understanding the present invention, however, the order of description should not be construed as to imply that these operations are necessarily order dependent, in particular, the order the steps are presented.

Referring now to **Figures 1a-1c**, wherein three exemplary embodiments of the present invention are shown. These three exemplary embodiments will be described in turn, referencing also **Figures 2a-2b**. First, referring to **Fig. 1a**, exemplary sending and receiving systems **100** and **140** are illustrated as being coupled to one another via communication medium **180**. More importantly, in accordance with the present invention, sending system **100** advantageously provides source files to receiving system **140** in a tokenized form, generated from an original form, thereby reducing transmission bandwidth requirement on communication medium **180**. For the illustrated embodiment, sending system **100** includes library **102**, tokenizer **104** and sender **106**, whereas receiving system **140** includes library **142**, de-tokenizer **144** and receiver **146**.

Over in sending system **100**, library **102** is used to store source files **110** in the original form and source files **112** in the tokenized form. Tokenizer **104** is used to transform source files **110** in the original form to source file **112** in the tokenized form. For the illustrated embodiment, tokenizer **104** effectuates the transformation by substituting language elements, such as arithmetic operators, relational operators and so forth, with tokens. The term "token" as used herein is intended to have similar meaning as the term is

commonly used by those skilled in the art of compiler technology, which typically includes a token class designation, e.g. arithmetic operator, and a class value, which may be a value designating a particular operator of the class, e.g. the value designating the "+" operator (see Fig. 2a, wherein an exemplary collection of language element substituting tokens is illustrated). Sender 106 is used to send source files 112 in the tokenized form to receiving system 140, as described earlier. Sender 106 may send source files 112 in the tokenized form to receiving system 140 at its own initiative, at the request of a local requestor (not shown), e.g. an application, or a remote requestor (also not shown), e.g. an application on receiving system 140.

In one embodiment, the programming language a particular source file 110/112 is written in is inferred from the file name of the source file, e.g. the file name including a file extension, such as "htm" for the hypertext markup language (HTML). In another embodiment, the programming language a particular source file 110/112 is written in is determinable from the properties associated with the particular source file 110/112, which is integrally provided along with a particular source file 112, when the particular source file 112 in tokenized form is provided to receiving system 140. In yet another embodiment, sender 106 informs receiver 146, the programming language the particular source file 110/112 is written in.

Still referring to Fig. 1a, over in receiving system 140, receiver 146 is used to receive source files 112 in the tokenized form from sending system 100, including as described earlier, its programming language, either integrally or particularly. Library 142 is used to store received source files 112 in the tokenized form, as well as restored source files 152 in the original form. De-tokenizer 144 is used to restore source files 112 in the tokenized form to source file 152 in the original form. For the illustrated embodiment, de-tokenizer 144 effectuates the transformation by restoring language element substituting tokens with their corresponding language elements. De-tokenizer

144 is equipped with the language element substituting token to language element mappings for a number of programming languages. In one embodiment, source files 110/112/115 may be written in include but not limited to C, C++, HTML, XML, Java™, and JavaScript, and de-tokenizer 144 is accordingly equipped to handle the supported programming languages.

Communication medium 180 is intended to represent a broad range of communication medium known in the art, from local area networks (ethernet, token ring, etc.) to wide area networks (ATM, frame relay, and so forth). Accordingly, communication medium 180 will not be further described. Libraries 102 and 142, sender/receiver 106 and 146, as well as tokenizer/de-tokenizer 104 and 144 are also intended to represent a broad range of these elements known in the art. Thus, except for the manner these elements are employed to practice the present invention, individually, these elements will also not be further described.

Figure 1b illustrates an alternate embodiment. In this embodiment, sending system 100' also advantageously provides source files to receiving system 140' in the tokenized form, thereby reducing the bandwidth requirement on communication medium 180'. Sending system 100' similarly includes library 102', tokenizer 104' and sender 106', while receiving system 140' similarly includes library 142', de-tokenizer 144' and receiver 146'. Each of these elements perform the same functions as the corresponding element described earlier for the embodiment of Fig. 1a. The key differences between these two embodiments are in the manner in which tokenizer 104' transforms source files 110' in the original form to source files 112' in the tokenized form, and de-tokenzier 144' restores source files 112' in the tokenized form to source files 115' in the original form.

More specifically, in addition to substituting language elements with corresponding tokens to reduce transmission bandwidth requirement, as

described earlier, tokenizer 104' further substitutes operands present in source file 110' in the original form with corresponding tokens. Additionally, tokenizer 104' further creates and maintains a symbol table 114' for each group of related source files, e.g. those to be compiled and linked together. In particular, tokenizer 104' creates a mapping entry for symbol table 114' for each new operand it encounters and substitutes with a new token. For the exemplary embodiment of tokens illustrated in Fig. 2a, the class value of the operand class token will be set to a pointer pointing to the appropriate mapping entry in the symbol table (see Fig. 2b, wherein an exemplary embodiment of a symbol table is illustrated).

Symbol tables 114' are also provided to receiving system 140' by sender 106' of sending system 100'. In like manner, receiver 146' stores the received symbol tables 114' in library 142', making them available to de-tokenizer 144' when it restores source files 112' in the tokenized form to source files 152' in the original form. In other words, in addition to the language element substituting token to language mappings de-tokenizer 144' is equipped with, de-tokenizer 144' further uses the operand substituting token to operand mappings in symbol tables 114' to effectuate the restoration.

Figure 1c illustrates yet another alternate embodiment. In this embodiment, sending system 100'' also advantageously provides source files to receiving system 140'' in the tokenized form, thereby reducing the bandwidth requirement on communication medium 180''. Sending system 100'' similarly includes library 102'', tokenizer 104'' and sender 106'', while receiving system 140'' similarly includes library 142'', de-tokenizer 144'' and receiver 146''. Each of these elements perform the same functions as the corresponding element described earlier for the embodiments of Fig. 1a-1b. The key difference between this and the earlier embodiments is the fact that source files 110''/112''/115'' are also in either a base or delta form, having associated versioning control information 116''. Accordingly, sender 106'' also

provides receiving system 140" with new operand substituting token to operand mappings for symbol tables 114", whenever sender 106" provides receiving system 140" with a base/delta source file 112" in the tokenized form involving new operand substituting tokens, not previously employed in base/delta source files 112" earlier provided to receiving system 140". Additionally, for this embodiment, sender 106" also provides versioning control information 116" to receiving system 140".

In one embodiment, each of base/delta source files 110"/112"/115" is identified with a universally unique identifier (UUID), as disclosed in co-pending U.S. patent application number <to be inserted>, filed contemporaneously, entitled "User Centric Source Control", which is hereby fully incorporated by reference (except for the reciprocating incorporation by reference). Each of the UUIDs universally identifies the particular base/delta source file 110"/112"/115" among other base/delta source files of the program product as well as among other base/delta source files of all other program products of all other software vendors. For this embodiment, the versioning control information 116" includes predecessor UUID information and other control information, such as locking and privileges, for the base/delta source files 110"/112"/115", as described in the co-pending application. As described earlier, sender 106" provides these predecessor UUID and other control information to receiving system 140".

Also in like manner, receiver 146" updates symbol tables 114" stored in library 142", whenever it receives new operand substituting token to operand mappings from sending system 100". Receiver 146" also stores versioning control information in library 140", upon receiving them from sending system 100", and making the versioning control information available for use on receiving system 140".

While the present invention is being described with Fig. 1c as an extension of Fig. 1b, those skilled in the art will appreciate that the present invention may also be practiced with Fig. 1a being extended with the additional base/delta and versioning control features of Fig. 1c, but without the additional tokenizing operand feature of Fig. 1b. In fact, those skilled in the art will appreciate that the present invention may be practiced with other additional features, and/or without some of the earlier described features.

Referring now to Figures 3a-3b, wherein one embodiment each of the operational steps of sending and receiving systems 100 and 140 are shown. In the remaining descriptions, when reference is made to an element, such as tokenizer 102, unless specifically noted, the reference is intended to include all embodiments earlier described, i.e. tokenizer 102, 102' as well as 102" of Fig. 1a, 1b and 1c. First, over in sending system 100, as illustrated in Fig. 3a, at step 302, tokenizer 102 is initially employed to transform the source files from the original form to the tokenized form, with at least the language elements being substituted by corresponding tokens. Step 304 is an optional step for those embodiments where at step 302, operands are also substituted by corresponding tokens. Where applicable, tokenizer 102 further creates a symbol table, or new operand substituting token mapping entries for an existing symbol table, depending on whether the source file being processed is a first of a collection of interrelated source files or merely additional ones of the collection. Steps 302 and 304 are presented as two separate discrete steps for ease of understanding. They may be practiced as separate steps as described or as a single combined step.

Upon generating the transformed source files, sending system 100 awaits for requests for the source files, step 306. As described earlier, the requests may be made by a local requestor, such as an application on sending system 100, or by a remote requestor, such as an application on receiving system 140. In any event, upon receipt of a request to provide selected ones

of the source files to receiving system 140, sender 106 provides requested ones of the source files in the tokenized form, reducing transmission bandwidth requirement on communication medium 180. Step 310 is also an optional step for those embodiments where at step 302, operands are also substituted by tokens, and/or the source files are being kept in base/delta form with versioning control information. Where applicable, sender 106 further provides the symbol table, update entries for the symbol table, or versioning control information, as the situation may call for. Similarly, steps 308 and 310 are presented as two discrete steps for ease of understanding. They too may be practiced as separate steps as described or as one single combined step.

Upon providing the requested ones of source files, and other applicable symbol table(s) and/or versioning control information to receiving system 140, sending system 100 returns to step 306, unless sending system 100 is to terminate operation. Steps 308-310 are repeated as many times as necessary to satisfy the various requests received by sending system 100 for receiving system 140 and the likes. Furthermore, while for ease of understanding, Fig. 3a illustrates the process of tokenizing the source files as being performed for a number of source files before requests for selected ones of the source files are received and serviced, those skill in the art will appreciate that in alternate embodiments, the process of tokenizing the source files may be dynamically performed subsequent to receiving a request for the source files instead.

Over in receiving system 140, as illustrated in Fig. 3b, at step 322, receiving system 140 either proceeds with steps 324-326 or step 328 depending on whether it is receiver 146 who has received source files provided by sending system 100 or it is de-tokenizer 144 who has received a request to restore selected ones of the tokenized source files. If it is the former, receiver 146 stores the received source files in tokenized form into library 142 as described earlier. Step 326 is an optional step for those embodiments where

the operands are also substituted by tokens and/or the source files are being kept in base/delta form with versioning control information. Where applicable, receiver 146 also stores the symbol table or versioning control information into library 142 or updates the symbol table, as the situation may call for.

At step 328, de-tokenizer 144 restores the source files from the tokenized form back to the original form, restoring at least the language element substituting tokens to the corresponding tokens. For embodiments where operands are also substituted by tokens, de-tokenizer 144 further restores the operand substituting tokens to the corresponding operands, using the appropriate symbol tables.

Upon responding to the receipt of source files in tokenized form or their associated information, or responding to requests to restore selected ones of the source files, receiving system 140 returns to step 322, unless receiving system 140 is to terminate operation. Steps 324-326 and step 328 are repeated as many times as necessary to service the receipts and various requests received by receiving system 140. Furthermore, while for ease of understanding, Fig. 3b illustrates the process of restoration as being performed "on-demand", those skill in the art will appreciate that in alternate embodiments, the process of restoration may also be performed in batch prior to making the restored source files available for use on receiving system 140.

Referring now to Figure 4, wherein an exemplary application of the present invention to the provision of web pages by web server is illustrated. As shown, web site 400 and an exemplary client system 440 is coupled to one another through Internet 480. Web site 400 provides web pages to client system 440 responsive to requests from client system 440. Incorporated with the teachings of the present invention, web site 400 advantageously provides the requested web pages to client system 440 in the above described tokenized form, reducing the transmission bandwidth requirement on Internet

480, which as those skilled in the art will appreciate, will also likely to result in improving perceived response time to a user of client system 440.

As described earlier for sending system 100" of Fig. 1c, web site 400 includes library 402, tokenizer 404 and HTTP interface 406 (in the role of sender 106). Library 402 is used to store HTML web pages, JAVA scripts and so forth in original as well as tokenized form 410 and 412 (hereinafter simply web page or web pages), including symbol tables 414. For the illustrated embodiment, web pages 410 and 412 are kept in base/delta form having associated versioning control information 416. However, for preferred implementation reasons, versioning control information 416 are stored in a separate repository 418 as opposed to library 402. For alternate embodiments, repository 418 may be implemented as an integral part of library 402. Tokenizer 404 and HTTP interface 406 operate as described earlier for the corresponding elements of sending system 100" to effectuate the desired reduction in bandwidth requirement on Internet 480.

Similarly, as described earlier for receiving system 140" of Fig. 1c, client system 440 includes library 442, de-tokenizer 444 and browser 446 (in the role of receiver 146). Library 442 is used to store web pages in tokenized form 412 and symbol tables 414. Also for preferred implementation reasons, versioning control information 416 are stored in a separate repository 448. Likewise, for alternate embodiments, repository 448 may also be combined with library 442.

Browser 446 includes conventional elements found in many browsers known in the art, HTTP interface 462, HTML web page handler 464, JAVA™ and JavaScript execution engine 466, other script interpreter 468 (e.g. CGI), display interface 470, and a number of "plug-ins", shown as additional Active-X components 472. Included among these Active-X components 472 is a component that interacts with library 442 and repository 448 to store received

web pages 412, symbol tables 414 and versioning control information 416 in library 442 and repository 448, and selectively invoke defokenizer 414 to reconstitute and restore the web pages, as described earlier for corresponding elements of receiving system 140". Except for the teachings of the present invention incorporated in the particular Active-X component, all other elements perform their conventional functions known in the art, and their constitutions are well, accordingly, will not be further described. Additionally, those skilled in the art will also appreciate that the particular Active-X component may be provided integrally with the browser or complementarily as a supplemental function. In fact, the ability might be provided via other "extension" or "plug-in" technology. The browser may also be an integral function of an operating system having other conventional operating system functions such as a file subsystem, task scheduling and so forth.

While the above exemplary application is described in the context of the Internet and World Wide Web, those skilled in the art will appreciate that Internet 480 may be an internal private network of a corporation or an organization, with web site 400 and client system 460 being internal information servers and user computer systems respectively. Furthermore, as described in the incorporated by reference co-pending application, the provision of web pages in the above described base/delta form also has the advantage of enabling a user to selectively roll back to prior versions of the web pages. Those skilled in the art will also appreciate that for certain browsers known in the art, the described manner of provision also has the advantage of making it difficult for the unskilled users to determine the source content of the web pages.

Figure 5 illustrates one embodiment of an exemplary computer system suitable for use to practice the present invention, in particular as a user system. As shown, exemplary computer system 500 includes processor 502 and system memory 504 coupled to each other via system bus 506. Coupled

also system bus 506 are non-volatile storage 508, various user input/output devices 510 and communication interface 520. Each of these elements perform its conventional functions known in the art. In particular, system memory 504 and non-volatile storage 508 are employed to store a working copy and a permanent copy of the programming instructions implementing the teachings of the present invention. The permanent copy of the programming instructions may be loaded into non-volatile storage 508 in the factory, or in the field, through distribution medium 522 or through communication interface 520. As described earlier, any one of a number of recordable medium such as tapes and so forth may be employed. The constitution of these elements 502-520 are also well known, and accordingly will not be further described.

Thus, a novel method and apparatus for token based source file compression/decompression and its application has been described. While the present invention has been described in terms of the above illustrated embodiments, those skilled in the art will recognize that the invention is not limited to the embodiments described. The present invention can be practiced with modification and alteration within the spirit and scope of the appended claims. The description is thus to be regarded as illustrative instead of restrictive on the present invention.

CLAIMS

What is claimed is:

1. In a computer system, a computer implemented source file transmission method comprising:
 - transforming a plurality of source files into a tokenized form, including substituting language elements present in the source files with corresponding tokens to reduce transmission bandwidth required to provide the source files to one or more other computer systems; and
 - providing the source files to the one or more other computer systems by transmitting the transformed source files in said tokenized form to the one or more other computer systems.
2. The method of claim 1, wherein the transformation further includes substituting operands of the source files with corresponding tokens and generating a symbol table with entries mapping the operand substituting tokens to the operands, and the provision further includes transmitting the symbol table to the one or more other computer systems.
3. The method of claim 1, wherein
 - the source files are in a base or a delta form,
 - the transformation further includes substituting operands with corresponding tokens, and generating a new entry for a symbol table for each new operand encountered and substituted by a new token; and
 - the provision further includes transmitting the new entries for the symbol table to the one or other computer systems.
4. The method of claim 1, wherein the source files are in either a base or a delta form having associated versioning control information, and the provision

further includes transmitting the versioning control information to the one or more other computer systems.

5. The method of claim 4, wherein each of the base/delta source files are identified by an universally unique identifier (UUID), and the provision of the versioning control information to the one or more computer systems includes transmission of predecessor UUID information of the base/delta source files.

6. In a computer system, a computer implemented source file reception method comprising:

receiving a plurality of source files in a tokenized form from a second computer system, the source files being provided in the tokenized form to reduce transmission bandwidth requirement;

storing the source file in said tokenized form in the computer system; and

upon request, transforming requested ones of the stored source files back to an original non-tokenized form, including restoring language element substituting tokens of the source files to corresponding language elements.

7. The method of claim 6, wherein said receiving further includes receiving a symbol table having entries that map operand substituting tokens to operands, and the transformation further includes restoring operand substituting tokens of the source files to corresponding operands, using said received symbol table.

8. The method of claim 6, wherein

the source files are in either a base or a delta form,

said receiving further includes receiving from the second computer system new entries mapping new operand substituting tokens to new operands for a symbol table to which the base/delta source files are associated; and

the method further includes updating the symbol table with the received new entries.

9. The method of claim 6, wherein the source files are in either a base or a delta form having associated versioning control information, and said receiving further includes receiving the associated versioning control information from the second computer system.

10. The method of claim 9, wherein each of the base/delta source files is identified by an universally unique identifier (UUID), and the receiving of the associated versioning control information from the second computer system includes receiving predecessor UUID information of the base/delta source files.

11. A computer system comprising:

a tokenizer to transform a plurality of source files into a tokenized form, including substituting language elements present in the source files with corresponding tokens, to reduce transmission bandwidth required to provide the source files to one or more other computer systems; and

a transmitter having access to the transformed source files to transmit the transformed source files in said tokenized form to the one or more other computer systems.

12. The computer system of claim 11, wherein the tokenizer further substitutes operands of the source file with corresponding tokens, and generates a symbol table with entries mapping the operand substituting tokens to the operands, and the transmitter further transmits the symbol table to the one or more other computer systems.

13. The computer system of claim 11, wherein
the source files are in either a base or a delta form,

the tokenizer further substitutes operands with corresponding tokens, and generates a new entry for a symbol table for each new operand encountered and substituted by a new token; and

the transmitter further transmits the new entries for the symbol table to the one or more other computer systems.

14. The computer system of claim 11, wherein the source files are in either a base or a delta form having associated versioning control information, and the transmitter further transmits the versioning control information to the one or more other computer systems.

15. The computer system of claim 14, wherein each of the base/delta source files is identified by an universally unique identifier (UUID), and the transmitter transmits predecessor UUID information of the base/delta source files.

16. A computer system comprising:

a receiver to receive a plurality of source files in a tokenized form from a second computer system, the source files being provided in the tokenized form to reduce transmission bandwidth requirement;

a storage medium to store the source files in said tokenized form; and

a de-tokenizer, to selectively transform, upon request, requested ones of the stored source files back to an original non-tokenized form, including restoring language element substituting tokens of the requested ones of the source files to corresponding language elements.

17. The computer system of claim 16, wherein said receiver further receives a symbol table having entries that map operand substituting tokens to operands, and the de-tokenizer further restores operand substituting tokens of the requested ones of the source files to corresponding operands, using said received symbol table.

18. The computer system of claim 16, wherein the source files are in a base or a delta form, and said receiver further receives from the second computer system new entries mapping new operand substituting tokens to new operands for a symbol table to which the base/delta source files are associated, and the receiver further updates the symbol table with the received new entries.
19. The computer system of claim 16, wherein the source files are in either a base or a delta form having associated versioning control information, and said receiver further receives the associated versioning control information from the second computer system.
20. The computer system of claim 19, wherein each of the base/delta source files is identified by an universally unique identifier (UUID), and the receiver receives predecessor UUID information of the base/delta source files.
21. An article of manufacture comprising:
a recordable medium having recorded thereon a plurality of programming instructions usable to program an apparatus to enable the apparatus to be able to transform a plurality of source files into a tokenized form, including substituting language elements present in the source files with corresponding tokens, to reduce transmission bandwidth required to provide the source files to one or more other apparatuses, and to enable the apparatus to transmit the transformed source files in said tokenized form to the one or more other computer systems.
22. The article of claim 21, wherein the programming instructions further enable the apparatus to substitute operands of the source files with corresponding tokens, generate a symbol table with entries mapping the

operand substituting tokens to the operands, and transmit the symbol table to the one or more other computer systems.

23. The article of claim 21, wherein the programming instructions further enable the apparatus to accommodate the source files being in either a base or a delta form, the programming instructions further enabling the apparatus to substitute operands with corresponding tokens, generate a new entry for a symbol table for each new operand encountered and substituted by a new token, as well as transmit the new entries for the symbol table to the one or more other computer systems.

24. An article of manufacture comprising:

a recordable medium having recorded thereon a plurality of programming instructions useable to program an apparatus to enable the apparatus to be able to receive a plurality of source files in a tokenized form from a second computer system, the source files being provided in the tokenized form to reduce transmission bandwidth requirement, to store the source files in said tokenized form, and to transform, upon request, requested ones of the stored source files back to an original non-tokenized form, including restoring language element substituting tokens of the requested ones of the source files to corresponding language elements.

25. The article of claim 24, wherein the programming instructions further enable the apparatus to be able to receive a symbol table having entries that map operand substituting tokens to operands, and to restore operand substituting tokens of the source files to corresponding operands, using said received symbol table.

26. The article of claim 24, wherein the programming instructions further enable the apparatus to accommodate the source files being in a base or a delta form, to receive from the second computer system new entries mapping

new operand substituting tokens to new operands for a symbol table to which the delta source file is associated, and to update the symbol table with the received new entries.

27. In a web server, a method comprising:
storing a plurality of web pages in a tokenized form with at least language elements present in the web pages in an original form substituted with corresponding tokens; and
upon request from a client computer system, providing requested ones of the stored web pages in the tokenized form to the client computer system.

28. The method of claim 27, wherein the web pages further having operands present in the web pages in the original form substituted with corresponding tokens, and entries mapping the operand substituting tokens to the operands being maintained in a symbol table, and the method further includes transmitting the symbol table to the client computer system.

29. The method of claim 27, wherein
the web pages are also in a base or a delta form, having operands present in the original form substituted with corresponding tokens, and having a new entry created in a symbol table for each new operand encountered and substituted by a new token, and
the method further includes transmitting the new entries for the symbol table to the client computer system.

30. A web server comprising:
a storage medium having stored therein a plurality of web pages that have been transformed into a tokenized form, with at least language elements present in the web pages in an original form having been substituted by corresponding tokens;

an interface to receive a request for one of the web pages from a client computer system; and

a transmitter to transmit the request web page, in the tokenized form, to the requesting client computer system.

31. The web server of claim 30, wherein the web pages further having operands present in the web pages in the original form substituted with corresponding tokens, and entries mapping the operand substituting tokens to the operands being maintained in a symbol table, and the transmitter further transmits the symbol table to the client computer system.

32. The web server of claim 30, wherein the web pages are also in a base or a delta form, having operands present in the original form substituted with corresponding tokens, and having new entries created for a symbol table for each new operand encountered and substituted by a new token, and the transmitter further transmits the new entries for the symbol table to the client computer system.

33. In a computer system, a method comprising:
receiving a web page in a tokenized form from a web server; and
automatically transforming the web page back to an original form, including restoring language element substituting tokens back to corresponding language elements.

34. The method of claim 33, wherein said receiving further includes receiving a symbol table having entries that map operand substituting tokens to operands, and the transformation further includes restoring operand substituting tokens of the web page to corresponding operands, using said received symbol table.

35. The method of claim 33, wherein
the web page file is also in either a base or a delta form,
said receiving further includes receiving from the web server new entries
mapping new operand substituting tokens to new operands for a symbol table
to which the base/delta web page is associated; and
the method further includes updating the symbol table with the received
new entries.
36. A browser comprising:
a first component to receive a web page in a tokenized form from a web
server; and
a second component to automatically transform the web page back to
an original form, including restoring language element substituting tokens back
to corresponding language elements.
37. The browser of claim 36, wherein said first component further receives a
symbol table having entries that map operand substituting tokens to operands,
and said second component further restores operand substituting tokens of the
web page to corresponding operands, using said received symbol table.
38. The browser of claim 36, wherein
the web page file is also in either a base or a delta form,
said first component further receives from the web server new entries
mapping new operand substituting tokens to new operands for a symbol table
to which the delta source file is associated, and updates the symbol table with
the received new entries.
39. An operating system comprising:
a file subsystem; and

a browser having a first component to receive a web page in a tokenized form from a web server, and a second component to automatically transform the web page back to an original form, including restoring language element substituting tokens back to corresponding language elements.

40. The operating system of claim 39, wherein said first component of the browser further receives a symbol table having entries that map operand substituting tokens to operands, and said second component further restores operand substituting tokens of the web page to corresponding operands, using said received symbol table.

41. The operating system of claim 39, wherein
the web page file is also in either a base or a delta form,
said first component of the browser further receives from the web server new entries mapping new operand substituting tokens to new operands for a symbol table to which the delta source file is associated, and updates the symbol table with the received new entries.

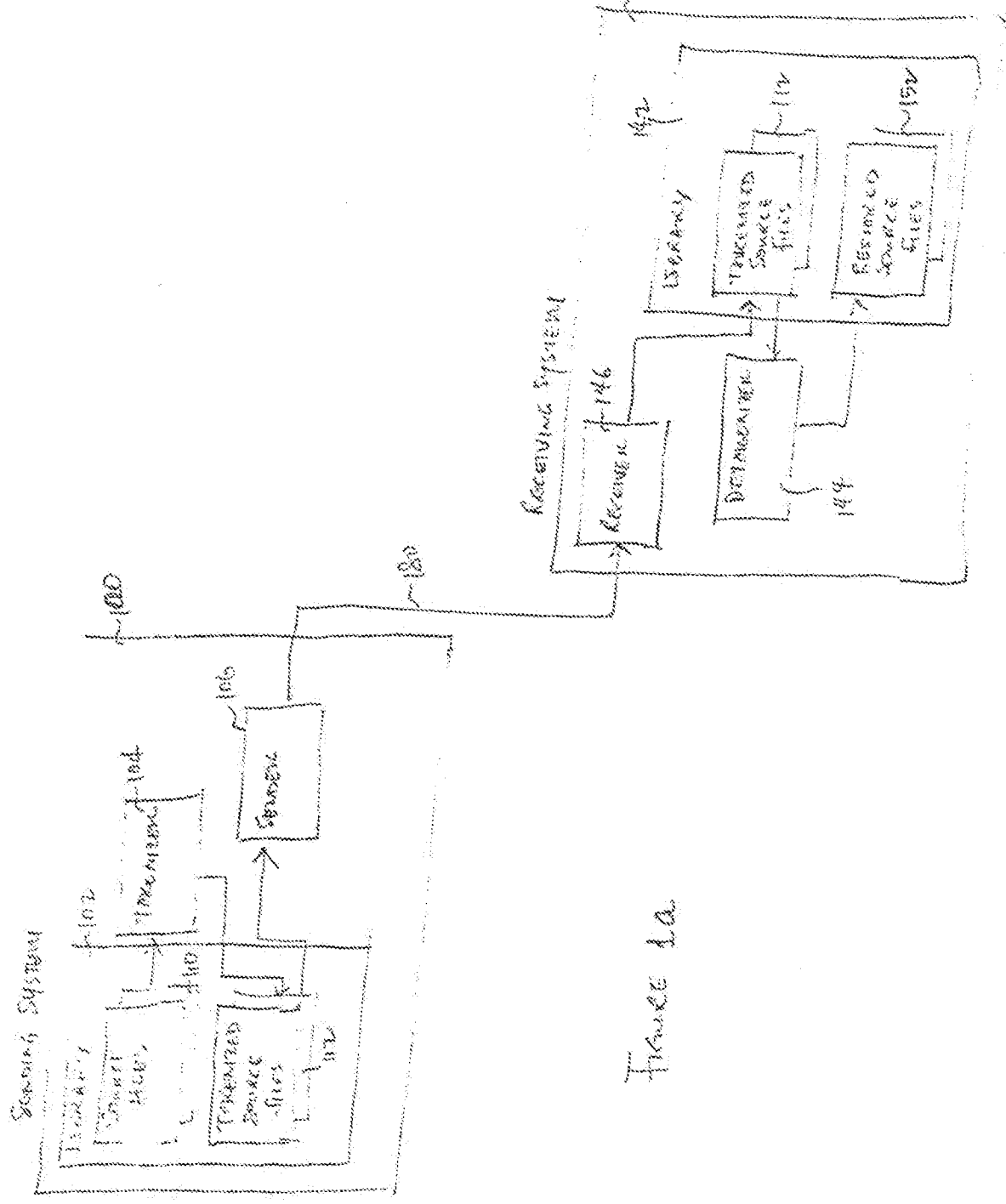


FIGURE 1A

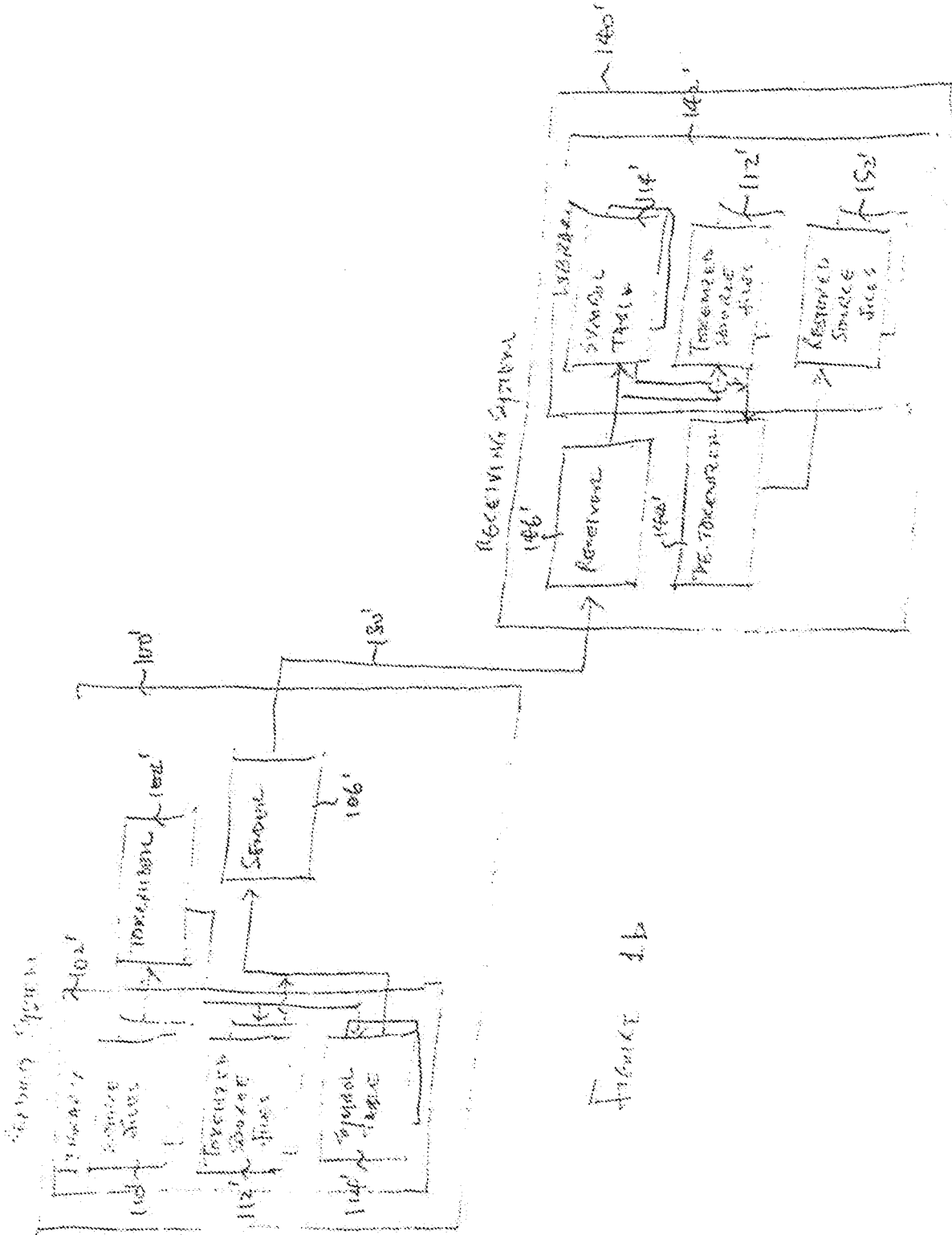


FIGURE 1b

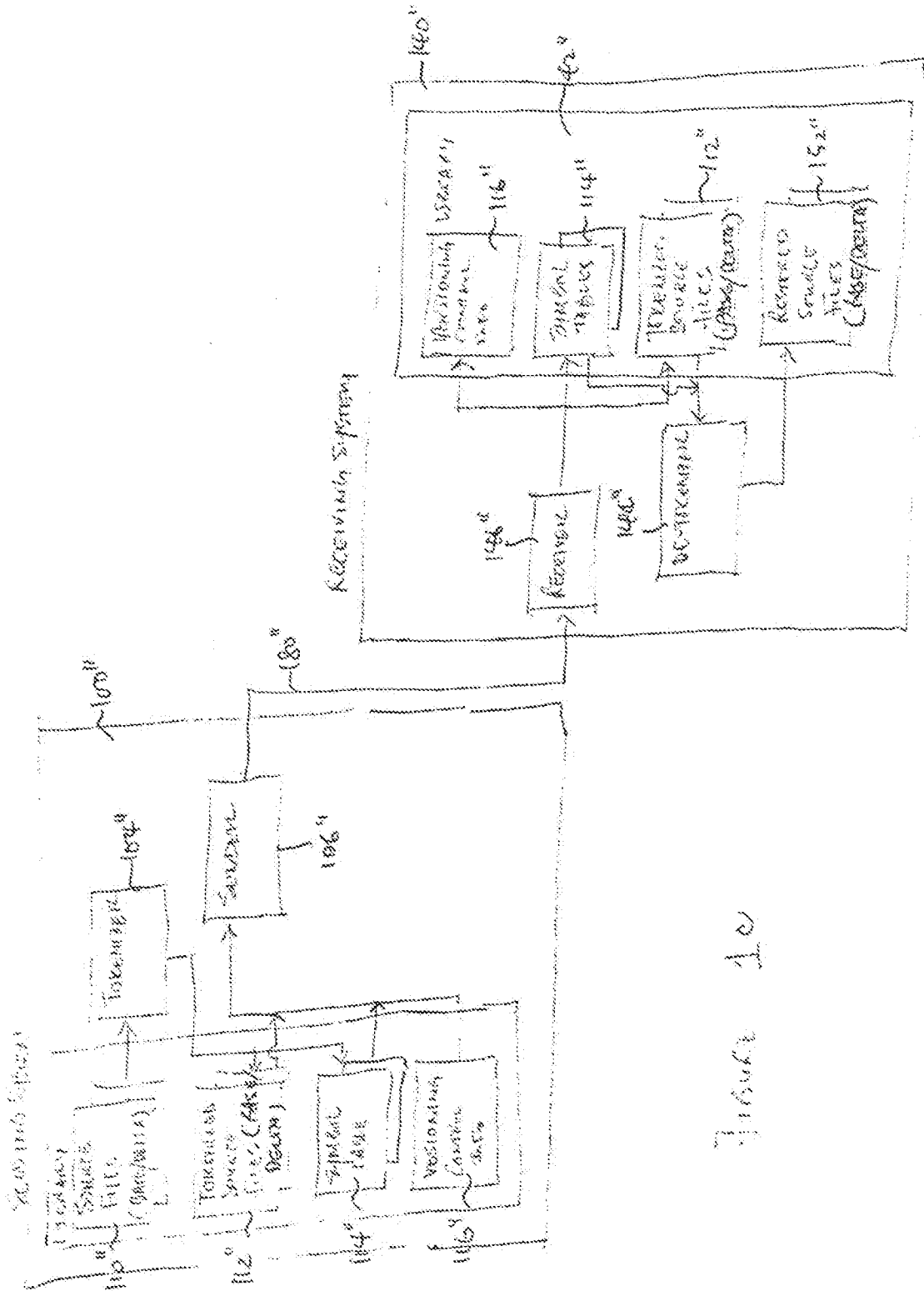


Figure 10

Tokens	
CLASS	VALUE
Arithmetic Operator	+ , * , / , ↑ , -
Relational Operator	= , > , = , < , >
Left Paren	(
Right Paren)
Next	NEXT J
Assign	LET X =
Special	OPENING_BRACKET, ENDING_BRACKET

SYMBOL TABLE

SPECIAL NUMBER	SPECIAL
1	OPENING_BRACKET
2	ENDING_BRACKET
3	

position →

FIG. 2b

FIG. 2a

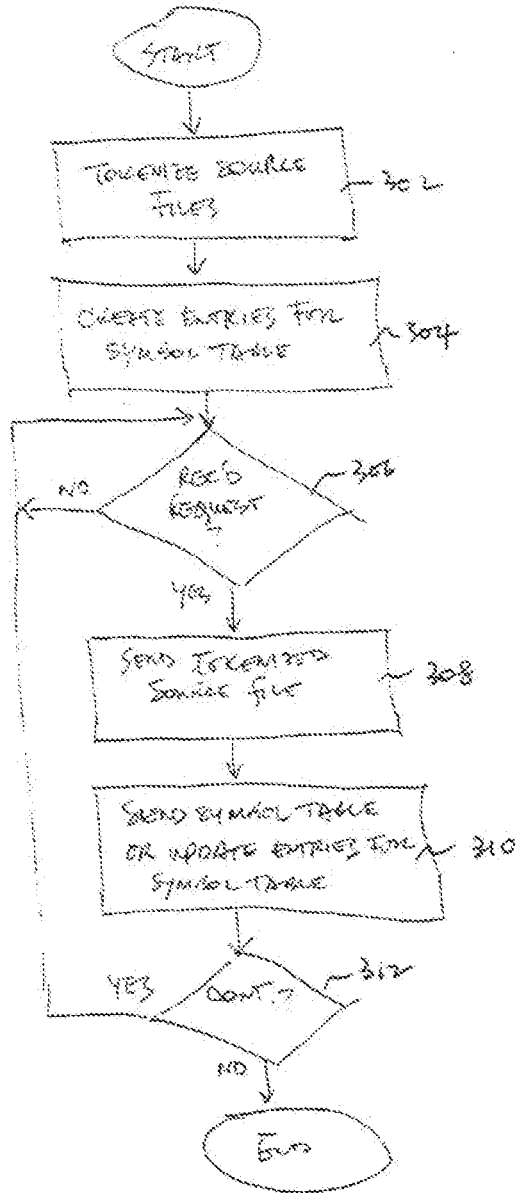


Fig 3a

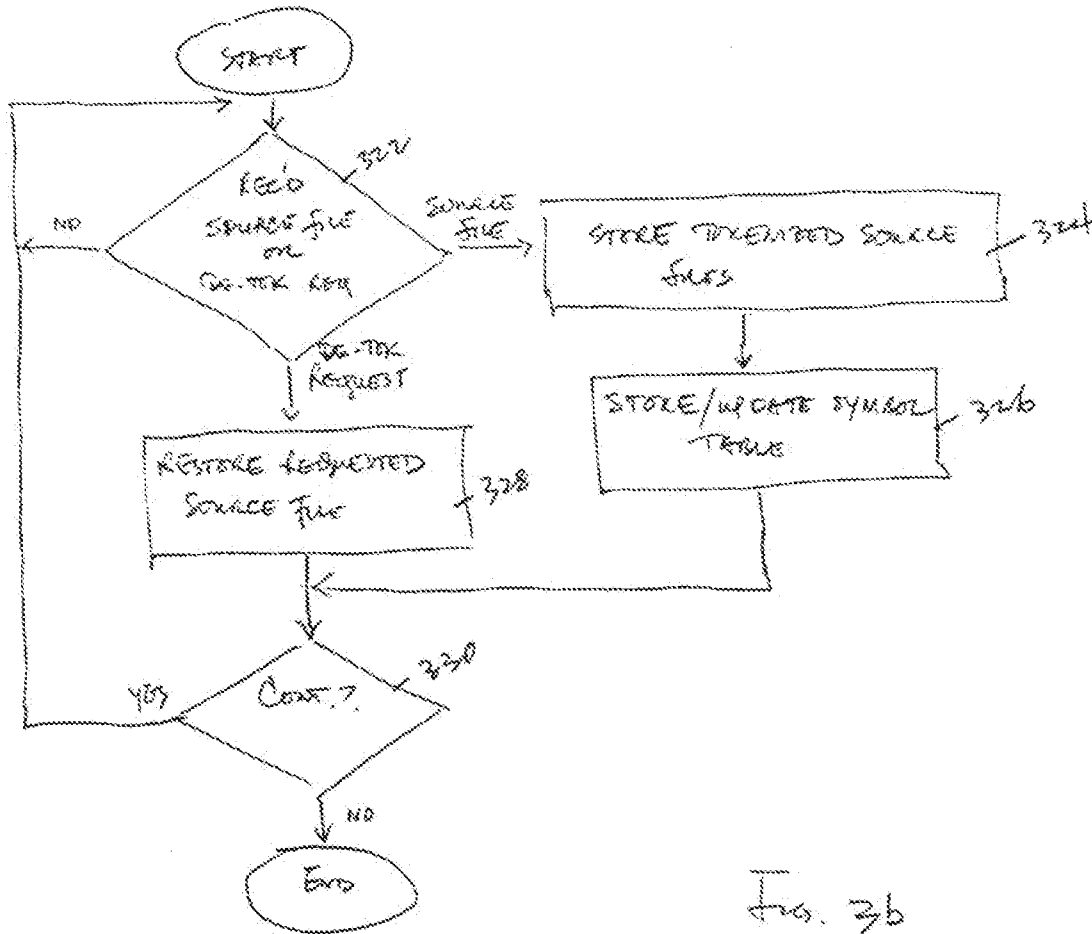


FIG. 3b

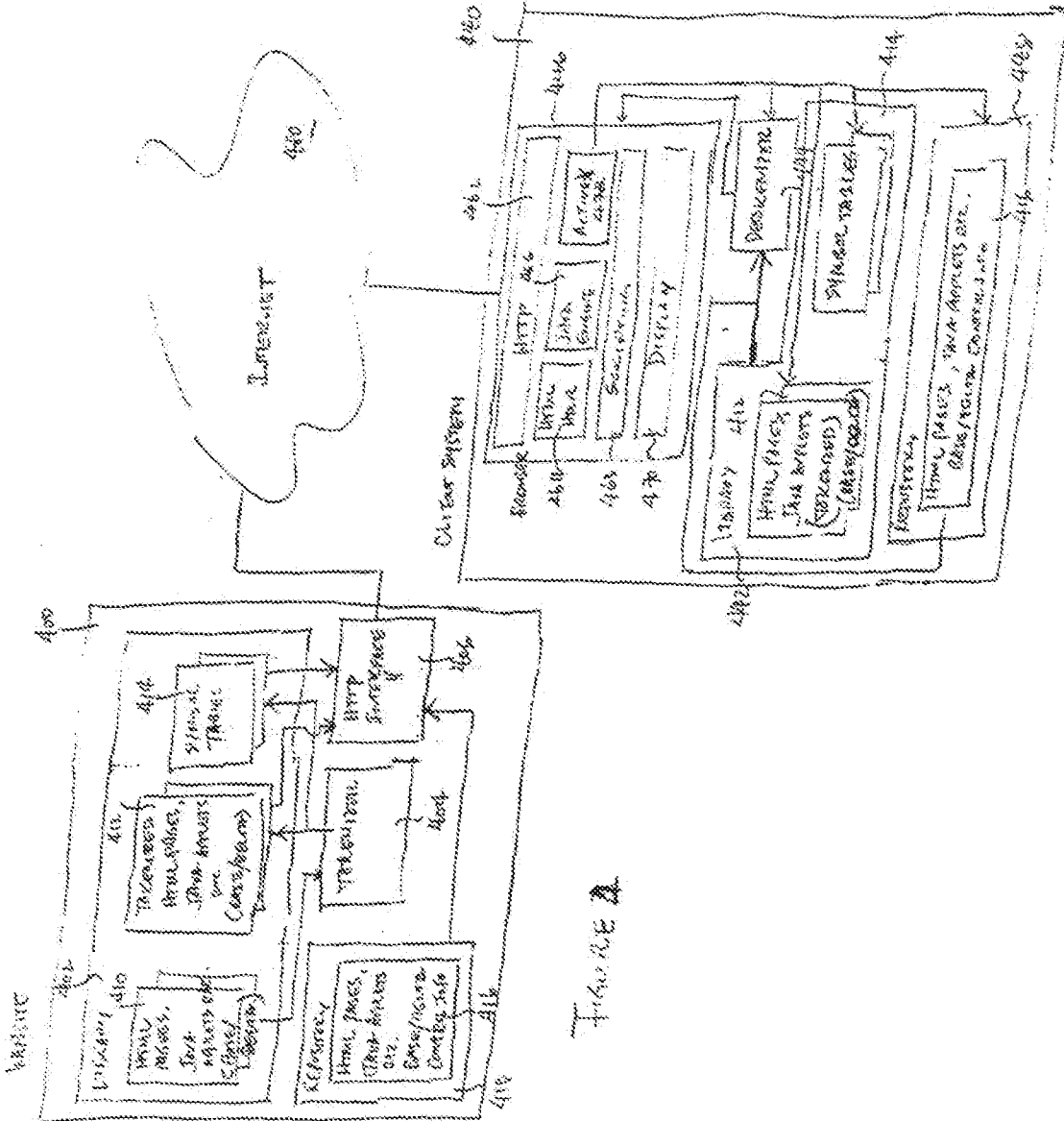


FIGURE 1A

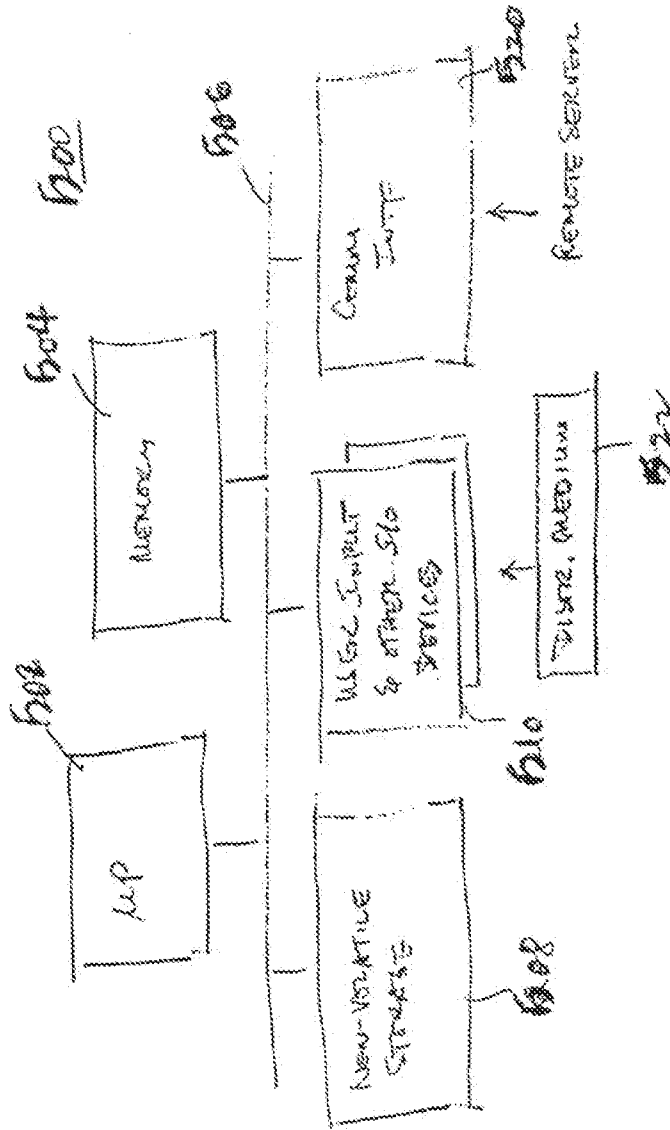


FIGURE 7

INTERNATIONAL SEARCH REPORT

International application No.

PCT/899/24919

A. CLASSIFICATION OF SUBJECT MATTER
 IPC(6) : G06F 15/16, 7/00, 13/38
 US CL : 709/247, 710/68, 707/101
 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)
 U.S. : 709/246, 247, 710/68, 707/101, 540

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
 USPAT, Derwent, EPO, JPO, IEEE, Elsevier
 search terms: differencing, tokens, compression, delta file, difference file, dictionary, parsing

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,E --- Y,E	US 5,991,713 A (UNGER et al.) 23 November 1999, figs. 8, 12, and 13, col. 8 line 54 to col. 16 line 17.	1-2, 6-7, 11-12, 16-17, 21-22, 24-25, 27-28, 30-31, 33-34, 36-37, 39-40 3-5, 8-10, 13-15, 18-20, 23, 26, 29, 32, 35, 38, 41

Further documents are listed in the continuation of Box C. See patent family annex.

* Special categories of cited documents	** later documents published after the international filing date or priority date and not in conflict with the application fee cited to understand the principle or theory underlying the invention
A document defining the general state of the art which is not considered to be of particular relevance	** document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an invention step when the document is taken alone
B earlier document published on or after the international filing date	** document of particular relevance; the claimed invention cannot be considered to involve an invention step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
C document which may throw doubts on priority claims or which is cited to establish the publication date of another document or other special reason (as specified)	** document of particular relevance; the claimed invention cannot be considered to involve an invention step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
D document referring to an oral disclosure, use, exhibition or other means	** document of particular relevance; the claimed invention cannot be considered to involve an invention step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
E document published prior to the international filing date but later than the priority date claimed	** document of particular relevance; the claimed invention cannot be considered to involve an invention step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

Date of the actual completion of the international search: 24 JANUARY 2000
 Date of mailing of the international search report: 17 FEB 2000

Name and mailing address of the ISA/IIS: Commissioner of Patents and Trademarks, Box PCT, Washington, D.C. 20231
 Facsimile No. (703) 305-3230
 Authorized officer: Ahmad Matar
 Telephone No. (703) 305-4731

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US99/24919

C (Continuation) DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X,P --- Y,P	US 5,884,014 A (HUTTENLOCHER et al.) 16 March 1999, figs. 4, 5, 9, and 10, col. 1 line 13 to col. 2 line 64, col. 26 line 52 to col. 28 line 51.	1-2, 6-7, 11-12, 16-17, 21-22, 24- 25, 27-28, 30-31, 33-34, 36-37, 39- 40 ----- 3-5, 8-10, 13-15, 18-20, 23, 26, 29, 32, 35, 38, 41
X --- Y	US 5,530,645 A (CHU) 25 June 1996, col. 2 line 57 to col. 5 line 29.	1-2, 6-7, 11-12, 16-17, 21-22, 24- 25 ----- 3-5, 8-10, 13-15, 18-20, 23, 26
Y	BLACK, A., et al., A Compact Representation for File Versions: A Preliminary Report, Proceedings of the Fifth Int'l. Conf. on Data Engineering, IEEE, May 1989, pages 321-329, particularly pages 321-322.	3-5, 8-10, 13-15, 18-20, 23, 26, 29, 32, 35, 38, 41
A	BELL, T., et al., Modeling for Text Compression, ACM Computing Surveys, December 1989, Vol. 21, No. 4, pages 557-591.	1-41

Form PCT/ISA/210 (continuation of second sheet) July 1992)*



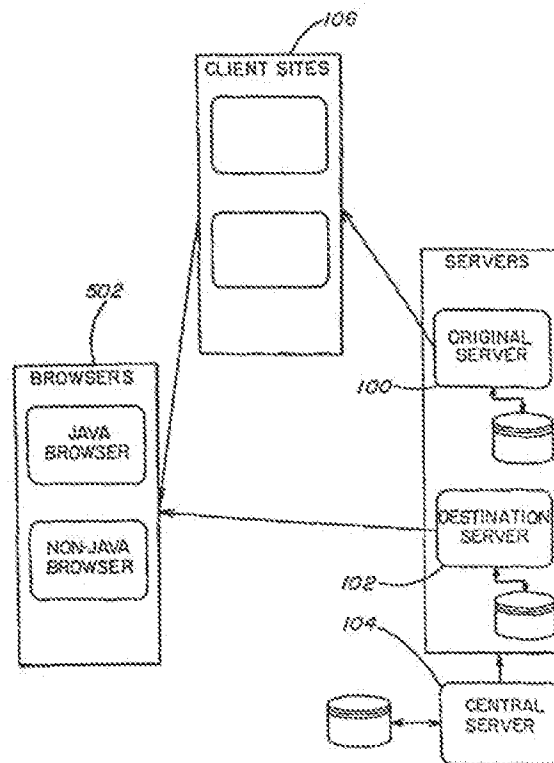
INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

<p>(51) International Patent Classification ⁷ : G06F 17/00</p>	<p>A2</p>	<p>(11) International Publication Number: WO 00/60490 (43) International Publication Date: 12 October 2000 (12.10.00)</p>
<p>(21) International Application Number: PCT/US00/08784 (22) International Filing Date: 3 April 2000 (03.04.00) (30) Priority Data: 09/285,628 3 April 1999 (03.04.99) US (71) Applicant: MUCHOINFO.COM, INC. [US/US]; 4101 East Louisiana Avenue, Denver, CO 80222 (US). (72) Inventor: WEISER, Nicolas, S.; 772 Race Street, Denver, CO 80206 (US). (74) Agents: PARKS, Jonathan, C. et al.; Kirkpatrick & Lockhart LLP, Henry W. Oliver Building, 535 Smithfield Street, Pittsburgh, PA 15222-2312 (US).</p>	<p>(81) Designated States: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, 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, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, 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), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).</p> <p>Published Without international search report and to be republished upon receipt of that report.</p>	

(54) Title: ARCHITECTURE FOR AND METHOD OF COLLECTING SURVEY DATA IN A NETWORK ENVIRONMENT

(57) Abstract

A network based survey system utilizing links from client sites to a set of logical servers to establish communication between users of the client sites and the survey system. The system preferably selects a percentage of the available users for participation in the survey using an adaptive selection process which adjusts to the load on the system. Preferably, users are deterministically mapped to a single one of a set of available servers, allowing user profile information to be stored only on a single server. The profile information can be used to allow a user to continue a survey at a later time or to implement question skip patterns or other traditional survey techniques. Alternatively, a central database is provided to collect, store, and process survey results.



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.

AL	Albania	ES	Spain	LS	Lesotho	SI	Slovenia
AM	Armenia	FI	Finland	LT	Lithuania	SK	Slovakia
AT	Austria	FR	France	LU	Luxembourg	SN	Senegal
AU	Australia	GA	Gabon	LV	Latvia	SZ	Swaziland
AZ	Azerbaijan	GB	United Kingdom	MC	Monaco	TD	Chad
BA	Bosnia and Herzegovina	GE	Georgia	MD	Republic of Moldova	TG	Togo
BB	Barbados	GH	Ghana	MG	Madagascar	TJ	Tajikistan
BE	Belgium	GN	Guinea	MK	The former Yugoslav Republic of Macedonia	TM	Turkmenistan
BF	Burkina Faso	GR	Greece	ML	Mali	TR	Turkey
BG	Bulgaria	HU	Hungary	MN	Mongolia	TT	Trinidad and Tobago
BJ	Benin	IE	Ireland	MX	Mexico	UA	Ukraine
BR	Brazil	IL	Israel	MY	Malawi	UG	Uganda
BY	Belarus	IS	Iceland	MS	Morocco	US	United States of America
CA	Canada	IT	Italy	NL	Netherlands	UZ	Uzbekistan
CF	Central African Republic	JP	Japan	NO	Norway	VN	Viet Nam
CG	Congo	KE	Kenya	NZ	New Zealand	YU	Yugoslavia
CH	Switzerland	KG	Kyrgyzstan	PL	Poland	ZW	Zimbabwe
CI	Côte d'Ivoire	KR	Democratic People's Republic of Korea	PT	Portugal		
CM	Cameroun	KZ	Kazakhstan	RO	Romania		
CN	China	LC	Saint Lucia	RU	Russian Federation		
CU	Cuba	LI	Liechtenstein	SD	Sudan		
CZ	Czech Republic	LK	Sri Lanka	SE	Sweden		
DE	Germany	LR	Liberia	SG	Singapore		
DK	Denmark						
EE	Estonia						

Title: Architecture For And Method Of Collecting Survey Data In A Network Environment

Inventor: Weiser, Nicolas S.

Field of the Invention

The present invention relates to the field of administering surveys electronically and more specifically to doing so over a computer network to users of that network.

Background of the Invention

The Internet is becoming a major force both in society in general and in commerce specifically. It offers unprecedented access to information and, increasingly, to retailers. As use continues to increase, it also represents an enormous population of users about which very little is known.

As is typical when a major new marketplace opens, businesses associated with the Internet want information about the user population. What are their interests, what are the demographics, what can and can't be sold via the Internet? Unfortunately, traditional market or audience survey techniques work poorly in this area. For a physical marketplace, such as a new shopping mall, the customers can be randomly physically contacted as they arrive or as they shop. Potential customers can be assumed to come from the surrounding locale, and randomly contacted by telephone or mail. With the Internet, geographic boundaries are irrelevant. Of more concern is the ease of finding the site and the language used. This can make it extremely difficult to identify the potential customers of a web site.

While actual visitors to a web site can be contacted as they enter the site, difficulties still arise. Many of the users of the Internet are anonymous to some degree. Seldom are actual names used, a login ID being the norm. These are often cryptic, either by choice or necessity. Some users will take further steps to intentionally hide their identity. Further, a single user may have multiple login IDs.

In this environment, traditional techniques can be difficult to apply. Selection of the survey respondent to fit certain criteria is difficult where there is no interviewer to see, hear, or question the respondent. The control provided by the human interviewer in preventing a single user from responding many times is not available.

The logistics of administering surveys on the Internet can be daunting. Users are in the millions; web sites in the hundreds of thousands and both numbers are growing rapidly. Limiting the web sites to only those relevant to a particular content area may still leave thousands of sites. How can all of the users to all of these sites be contacted? How to select a
5 sub-population of these visitors to provide a statistically valid sample?

Clearly, a survey program could be installed at each of the sites. However this would take resources away from the web site itself, in terms of processing time, storage, and network bandwidth. Few web site owners or administrators would willingly give up such resources. In addition, it would take a significant amount of time and cost to install and test
10 thousands of survey systems. This could easily consume many weeks and tens of thousands of dollars before the first survey could be presented. Similar costs may be incurred to remove the survey program at the completion of the survey.

Compounding this is the pace at which the Internet grows and changes. A period of months can be a very long time in the Internet marketplace. A retailer trying to position a
15 web site, or trying to identify a cause for dropping sales cannot wait months for a survey to be complete. Weeks may even be too long. They may need to see at least preliminary information within days. Traditional techniques can not hope to meet these timelines.

Traditional techniques are also labor intensive. In-person interviews and telephone interviews are slow, one on one, processes. Data entry of the responses can also be labor
20 intensive and error prone. All of these increase the cost of a survey.

While many web site administrators may be interested in performing their own surveys, they likely lack the knowledge, experience, and resources to do so properly. Those who do, may lack the time or interest to develop a valid survey. Further, each web site only has access to the users which visit that site. While it may provide an accurate picture of their
25 population it may not be valid for the general web population. Where multiple sites combine their data for a larger picture, the problem of duplicate responses arises. If the sites are all in the same content area, the likelihood of web users visiting more than one of the sites is very high.

The above and other problems make it difficult to administer a useful, statistically
30 valid survey on the Internet within reasonable time and budget constraints and relatively few such surveys are conducted. However, the information is desperately needed and that need will continue to grow as the Internet user population and web site base grows. If the

techniques could be developed, the Internet user population also provides a rich resource for surveys of non-Internet topics. The instant access to a diverse worldwide population is appealing even to those companies who do not deal directly with the Internet. In this time of a growing global marketplace, access to international users is highly desirable.

5 There is a need for a method of conducting surveys on the Internet, or other network, which is statistically valid, addressing the problem of duplicate responses by anonymous users, does not allow the users to self select, and provides the necessary control of quotas. The system should be able to adapt to variations in such factors as the number of visitors and the completion percentage of users presented a survey. Ideally, this adaptation would occur
10 continuously during the administration of the survey, maintaining smooth progress towards the survey goals. The survey method should be scaleable from a single site to integrating the responses of thousands, or even hundreds of thousands, of sites. At the same time, it should be possible to quickly define and initiate the survey without significant impact on the web sites from which the sample population is drawn. Upon completion of the survey, it should
15 be possible to easily disconnect the survey from those web sites. When drawing from multiple sites, the system should be able to eliminate duplicate users. Ideally, the system should allow a user to partially complete a survey during one session and return to complete it at a later time, even if connecting from a different site. Traditional techniques such as question skip patterns should also be supported. The survey results should be available
20 promptly upon completion of the survey and the availability of intermediate results would be highly desirable. It should be possible to configure the system to either focus on specific user groups or like web sites or to sample a diverse population across the Internet.

Summary of the Invention

The present invention is directed to an apparatus and method for surveying users of a network, such as the Internet, which makes use of links from the pages of independent content providers to provide the initial contact with the system. A set of logical servers
5 administers surveys to a subset of the available users by selecting them randomly using an adaptive process which continuously adjusts the percentage of users selected in response to the number of users visiting the client sites and the percentage of those who are presented a survey that actually complete it.

According to the invention there are provided plural logical servers connected to a
10 network which also serves one or more content provider sites. A link is embedded in a page of the provider sites which connects to a logical server. When a link is activated, communications is established between the user and the survey system. The user is then presented a survey to be completed.

According to an aspect of the invention there may be both original servers which
15 handle the initial communication with and identification of the user and destination servers to which the user is then connected to complete the survey. Preferably, the user is mapped to a destination server by a process which results in the same user always being connected to the same destination server without regard to the client site or original server used to initially access the system. The original and destination servers may be roles played by a single type
20 of logical server which can serve as either type for different connections or at different times.

According to another aspect of the invention a central database server may be provided which collects, stores, and processes the survey results and makes them available to customers of the survey system.

Further in accordance with the invention user profile information may be maintained
25 by the destination server(s) which includes information about the user which allows completion of a survey at a later time, implementation of question skip patterns, and other techniques to improve the validity of the survey.

Still further in accordance with the invention, the survey system may restrict access to
30 only a percentage of those users visiting the client site and may adaptively adjust this response in response to factors such as the number of hits on the client sites and the percentage of users completing a survey which is presented to them.

The advantages of such a system and method are that the population of Web users who visit a site or collection of sites can be surveyed in a statistically valid and timely manner at a minimum cost. A survey can be quickly defined, set up, administered, and the results obtained with little impact on the participating client sites. The system is self regulating in terms of sampling quotas and adjusts to load changes to maintain desired sampling levels. The system can be quickly reconfigured to focus on the visitors to a single site, a collection of sites in a single category, or to a diverse set of sites representative of the Web as a whole. Survey results are available shortly after completion of the survey in a variety of formats including electronically downloadable.

The above and other features and advantages of the present invention will become more clear from the detailed description of a specific illustrative embodiment thereof, presented below in conjunction with the accompanying drawings.

Brief Description of the Drawings

- FIG. 1 - provides a block diagram of the Internet.
- FIG. 2 - illustrates an abstract HTML web page.
- FIG. 3 - illustrates a hypertext link from one web page to another.
- 5 FIG. 4 - provides a high level block diagram of the inventive system architecture.
- FIG. 5 - illustrates the creation of links to the original and destination servers.
- FIG. 6 - illustrates the sequence of messages involved in establishing a connection to the system and completing the survey.
- FIG. 7 - provides a block diagram of the major components of a logical server.
- 10 FIG. 8 - graphically represents the selection of participating users from the available population and the reduced number who complete the survey.
- FIG. 9 - is a flowchart of the sampling adaptation process.
- FIGs. 10 A & B - are a data flow diagram illustrating the determination of new sampling parameters.
- 15 FIGs. 11 A - K - are pseudo code of an illustrative implementation of the sampling adaptation process.
- FIG. 12 - illustrates the logical loop arrangement of servers used to collect statistics.
- FIG. 13 - illustrates the merging of template and form to create a survey questionnaire.

Detailed Description of the Invention

The following discussion focuses on the preferred embodiment of the invention, in which the disclosed architecture is used in conjunction with the Internet to collect survey information. However, as will be recognized by those skilled in the art, the disclosed method
5 and apparatus are applicable to a wide variety of situations in which the collection of statistical information using a network is desired.

The following is a brief glossary of terms used herein. The supplied definitions are applicable throughout this specification and the claims unless the term is clearly used in another manner.

10 Applet - a special form of computer program designed to be downloaded from a host in conjunction with a web page. Typically written in the JAVA language, an applet is unique in that it can be executed on any hardware platform which includes a JAVA engine. This differs significantly from normal programs which are built for a specific computer. Applets are usually restricted in the access which they are allowed to the resources of the computer on
15 which they are executed and the type of network communications they are allowed to perform.

Browser Software - generally a computer program executing on the users local computer which is designed to navigate and display (browse) WWW documents but which includes any software program which provide an interface between a computer network and a
20 user of that network. Examples include NCSA's Mosaic, Netscape's Navigator and Microsoft's Internet Explorer.

Central Server - in the present invention, a data storage server which collects, stores, and merges survey results.

25 CGI (Common Gateway Interface) - a protocol for how a web server communicates with another program executing on the same computer. Any program can be a CGI application if it handles input and output according to the CGI standard. CGI applications differ from applets in that they run on a specific server and must be compatible with the hardware and operating system provided by that server.

30 Client Site - in the present invention, a client site is a content provider which has been modified to provide a link to the inventive system. The user first makes contact with the survey system through a link incorporated into one or more pages of the client site.

Destination Server - in the present invention, the server which handles the communication with the user to present and collect the survey information. After the identity of the user is established, the Original Server passes the user on to the Destination Server to handle the remainder of the transaction.

5 Form - in the present invention, a document which contains the survey text in a generic format. This text will be merged with a Template to create the final survey document presented to the user.

HTML (Hypertext Markup Language) - a hypertext document specification language used primarily for the creation of WWW documents. It is a block oriented language which
10 utilizes tags to define formats and features which are then interpreted by browser software.

Hypertext - a method of constructing documents such that there are multiple pathways through the contents that the user can select and follow, rather than only providing sequential access from beginning to end. The pathways are provided by hypertext links which can lead to other documents, other sections of the same document, or to alternate views. The link,
15 (sometimes referred to as a hyperlink) is often embedded in the text of the document and distinguished by the use of a different color, font, style, or any combination of these. This type of link is typically activated by the user selecting, or clicking on, the link. Links may also be hidden from the user and activated automatically by the browser.

ISP (Internet Service Provider) - a company which provides its clients with a presence
20 on the Internet. This may include hosting of the client's web pages and/or access to the Internet via either a dial-up or dedicated connection. An ISP which provides only access may be referred to as an Internet Access Provider (IAP).

Load - generally the utilization of a resource on a computer. It may be expressed either as an absolute number, such as the number of users connected, or as a percentage, such
25 as the ratio of the portion of the disk or CPU capacity being used to that available.

Look and Feel - a broad term encompassing most, if not all, aesthetic and some functional elements of how a computer program interacts with the user of that program. This includes, but is not limited to, color and font choices, the types of interactive controls used, and the general layout of visual elements on a screen or page.

30 Original Server - in the present invention, the server with which the user makes initial contact as the result of a link from the client site. The original server establishes the identity

of the user and then passes the user on to the Destination Server to handle the remainder of the transaction.

Server - generally, a computer or program, in a distributed environment which provides a specialized service such as data storage, printing, or communications. In the Internet environment the service is more likely to be specific to supporting the Internet such as a Web server that provides WWW pages to a browser program or a Domain Name Server (DNS) that translates logical network names into numeric addresses.

Template - in the present invention, a document which captures the appearance of the pages on a client site and which includes one or more tokens which will be replaced by the text of the survey.

Transaction Monitor - in the present invention, an application running on a server which gathers the statistics used by the adaptive sampling algorithms.

User - the human user of a computer or software program. With respect to the Internet, the person who is using a browser to access the web. In the present invention, the user is that subset of Internet users who are interacting with the inventive system in some way. The same term is often used to refer to the browser software being used by the human user. Often, the distinction between the user and the user's browser is not important.

URL (Uniform Resource Locator) - one form of a logical link which specifies the location of an object on the Internet, such as a file or another web page. URLs are commonly embedded in HTML web pages to specify the target of a hypertext link. A URL consists of multiple fields containing information about the target of the link. This information includes the access method, or format, of the target; the address of the server on which the target resides; and the path to the target in the server's file system. Other information may be included as necessary.

Web Page - a logical page of HTML text which forms the basic medium of the World Wide Web (WWW) protocol. The page can also include images, sounds, embedded programs (such as applets), and other data types.

WWW (World Wide Web) - a particular protocol used for the Internet and intranets which specifies a graphical, hypertext format which provides a point and click interface to distributed documents via browser software. Often, that portion of the Internet which supports the WWW protocol is loosely referred to as the "Web."

Preferred Embodiment

The disclosed invention is described below with reference to the accompanying figures in which like reference numbers designate like parts.

Internet Overview

5 As the preferred embodiment of the present invention is of use primarily in the context of the Internet, that environment will be briefly described. However, the present invention is not restricted to the present day Internet. It is equally applicable to other network architectures, both present and future. The network provides a communications medium between the various components of the system and any network or communications grid
10 which serves this role is considered equivalent. An abstract representation of the Internet architecture is shown in FIG. 1. While the Internet itself, 500, is often thought of, and even described as a single "backbone" to which all of the systems are connected, this is incorrect. The Internet is a computer network which, by design, has no centralized control. It is a loose agglomeration of a very large number of computers and sub-networks which cooperate to
15 provide the services which are viewed as the Internet. Its model of distributed control sometimes borders on anarchy. No single entity, computer, or communications link is critical to the Internet. Services are duplicated, data storage is mirrored, and communications paths are redundant. This results in a system which is very resistant to failures, or attack, but which can be daunting to use.

20 The cooperation of the various systems involved in the Internet is regulated by a set of communications protocols and interface standards which simplify the system interactions. Chief among these are the protocols and standards which comprise the World Wide Web (WWW or Web). The WWW is a subset of the Internet which provides a user friendly, largely graphical, point and click view of the Web. Access to the Web is typically through
25 the use of browser software. This is a computer program which resides on the Web user's local computer and which interprets and presents information received from the Web. It understands the relevant protocols and assists the user in navigating the Web. It is also instrumental in directing requests to the various servers (such as search engines) and displaying the results.

30 The web user's computer, or more specifically the browser software executing on it, 502 and 504, may be connected to the Internet via a dedicated connection, 502, more common in the workplace, or may connect as-needed via a dial-up connection, 504, through

an ISP, 506. For most purposes these connections are equivalent, differing only in speed or bandwidth and will generally be referred to throughout the specification as a browser, 502. Further the concept of computer is very broad with respect to the user. Any device which provides network browsing capability is contemplated including laptops, personal
5 information managers, and even telephones.

Various types of servers, 508, are accessible via the Web. Most common, of course, are WWW servers which provide web pages conforming to the Web protocols. Many of these WWW servers are what are referred to as "content providers." These are servers which provide information or data (the "content") which is of interest to some or all of the Web
10 users. Other servers include search engines, which help search for content providers of interest, and providers of various support services, such as dynamic name translation, needed by the infrastructure of the Internet itself.

FIG. 2 provides an illustration of a generic web page as displayed by a browser. The browser will typically present the web page information, 511, in a large window, and will
15 provide it's own information in a separate area, 510. The browsers information area often includes a title bar, 514, a set of menus, 516, and a set of buttons, 518. The menus and buttons provide access to the commands supported by the browser. A status line, 520, is also typically provided for the display of messages to the user. Within the web page area a variety of material may be presented to the user. This includes text, 522, image, 524, and graphical,
20 526, information. The types of information which can be presented are expanding rapidly and include sound and full motion video. Within the text area certain segments of the text, 530, may be designated as hypertext links. In a similar manner, the photos, graphics, and embedded controls, such as buttons, 532, can also be used as links.

The hypertext links are one of the chief mechanisms used in navigating the Web.
25 Web pages are almost universally written in HTML which provides for both formatting of the page content and the specification of links to other pages. From within a browser, the user can easily elect to follow any link presented in the page or can choose to continue with the present page. This process is shown in FIG. 3. If a link, 536, from the page currently being displayed, 534, is followed, the browser, 502, then loads a new page, 538, which may be on
30 the same server, 540, as the present page or may be on another server, 542, anywhere on the Web. Following links, a user can easily retrieve pages from dozens of servers scattered

around the globe in just a matter of minutes. Web pages may also contain links which are automatically activated by the browser when a page is loaded.

Supplementing the HTML pages are a variety of scripts and programs which provide more tailored and powerful services. While HTML is a very flexible language for the presentation of documents, it is limited in the complexity and power of the tasks it can perform. When a more involved manipulation is required, such as retrieving data from a database or generating moving graphics, a program will be activated by the browser, often as a result of a link. These programs may be interpreted scripts executed by an extension to the browser, small applications, often called applets, downloaded to the user's computer and then executed, or larger programs executed on remote server which then supplies its results to the browser for display.

Architecture

It is within the Internet environment that the preferred form of the present invention is preferably used. Making use of the distributed server concept of the Web and interfacing with Web servers, the present system surveys users of the Internet, adapts to the use patterns of those users, merges the results, and presents statistically reliable information for use by its clients.

The general architecture of the present invention is shown in FIG. 4. The major components of the survey system are the original, 100, and destination, 102, servers and the central server, 104. A lesser role is served by the client sites, 106, which are modified to link to the system. While the browser software, 502, is significant in that it provides the user with access to the system and presents surveys to the user, it is not part of the inventive system.

The client sites, 106, are generally content providers as they normally exist in the Web. One or more pages on each of the relevant client sites has been modified with a link to an original server, 100. This link may be either a visible link, which the user elects to follow, or may be a hidden link activated by the browser. For the purposes of the present invention, either type will work. The sole purpose of the link is to establish the initial communications between the user and a pre-selected original server. The selection of server is made by the address specified in the link.

Note that in the following discussion of original and destination servers, each of these is a logical server. One or more logical servers may be hosted by a single physical computer. As the load on the servers varies, the logical servers can be moved between computers. This will be entirely transparent to the users, and to the system itself, as it deals with the logical entities. This flexibility also allows the system to adapt to hardware failures by moving the servers off of the faulty computer.

Referring to FIGs. 5 and 6 the process of establishing communication between the user and the inventive system will be outlined. FIG. 5 provides a graphical depiction of the connections and FIG. 6 captures the sequence of messages which occur. As discussed above, the user, 140 in FIG. 5, first requests a page from a client site, message 110 in FIG. 6, and is provided with that page, message 112. The user then clicks on the link to the survey system (or it is automatically activated) resulting in a request to the original server, message 114. This establishes connection, 142 in FIG. 5, to a server, 144, which acts as an original server.

Alternatively, an applet could be associated with the page. The applet could contain one or more links or could retrieve link information such as from a logical server. The applet would then select one of the links and activate it, establishing the connection. From the users perspective, this approach would be equivalent to the selection and use of the single embedded link, and likely indistinguishable. From the system side, the use of the applet provides more flexibility by making available more decision making capability prior to establishing the link to the system. This enables such capability as randomizing the selection of the original server; connecting directly to the destination server as described below; or merely storing the current set of links in a centralized location to remove the dependency of the client sites on a specific set of link addresses. These and other means of selecting the initial link are interchangeable with respect to their ability to determine the link itself and activate the connection.

The first page presented by the original server, message 116, will ask the user for identifying information such as name, age, and birth date. Additional, or alternative, fields could be used as necessary to uniquely identify each user. Clearly social security numbers or drivers license numbers could be used where appropriate. Each of the original servers is provided with a copy of an identical algorithm which maps a user to a single specific destination server, using the data supplied by the user, message 118, in response to the first page. In the preferred embodiment, this is implemented as a hashing algorithm where each

resultant hash code matches the address of a destination server. This address is provided to the user in a second page, message 120, which also contains privacy notices, links to survey rules, and such other "housekeeping" information as may be desirable to present to the user. Completing this screen, message 122, links the user, connection 148 FIG. 5, to the selected destination server, 150, and starts the actual survey process. In alternative embodiments, the original server could cause a page to be transferred from the destination server to the user's browser without first providing the second, housekeeping page. In a further alternative, the mapping algorithm could be implemented in either an applet which executes locally on the users computer, or in a CGI, or other, program executing on the original server. Either of these approaches would also eliminate the second page. These approaches reduce the number of preliminary pages seen by the user, but incur a cost in requiring multiple implementation of the mapping algorithm, one for each type of computer to be supported. The use of Java as the implementation language could alleviate some or all of these problems. The core component of all of these approaches is that, based on the user provided information, each user is always mapped to the same, specific destination server.

Once the user is connected to a destination server, 150, that server has access to local storage, 152, and can retrieve, messages 124 and 126 in FIG. 6, containing information about the user including a history of previous responses. In this way, the destination server can provide survey pages, message 128, which are tailored to the specific user. The simplest application of this is to begin the survey at the point where the user previously left off. This option is available even if the user connects from a different local computer or through a different client site. When the user completes the survey, message 130, the destination server stores the survey results and updates the user's profile, message 132. If the user responds to the first page and then declines to complete the survey when presented the second page, the database may be updated by a message from the destination server to the database, in place of message 124, and the sequence will terminate.

The ability to track users greatly increases the reliability of the information obtained by the survey system. The approach of mapping users to destination servers also decreases the storage needs for the servers. Without this approach, every server would need to have a duplicate copy of every user's information, or there would have to be a single centralized database which all servers would access for the user information. Either approach would incur a severe performance penalty. With the present architecture, a user's information need

only be stored in a single location and the load can be adjusted by varying both the number of logical servers and the number of physical computers on which they are hosted.

In the preferred embodiment, which utilizes a hashing algorithm, a certain amount of growth can be handled by increasing the number of computers being used and distributing the logical servers across these computers. This will work until each server is hosted on its own
5 computer. Up to this point, these changes can be made with little impact on the system as it continues to run. After this point, if the load continues to increase, the number of logical servers will have to be increased. This will require an increase in the number of "bins" used by the hashing algorithm and distribution of a new hashing algorithm to the original servers.
10 The user data will also have to be re-distributed across the new set of logical servers to match the mapping of the new algorithm. This would require a short down time for the system as the changes are implemented.

FIG. 5 also illustrates the connection of a second user, 154, to the system. In this case, server 150, which was the destination server for the first user, acts as the original server
15 in response to the initial connection, 156, and then directs the user to another server, 160, as the destination server. Note that while different users may be mapped to different servers, as illustrated, the same user will always be connected to the same destination server, as discussed above.

It should be noted that in the preceding discussion the "original server" and
20 "destination server" are roles played by the pool of logical servers, 146 in FIG. 5, within the system. Any server can serve as both an original server and a destination server. Alternatively, these roles could be separated and be supported by distinct logical servers.

The central server, 104 in FIG. 4, serves as the central database for the survey system. The database provides a central location for compiling survey responses and generating
25 results. Any of the analysis techniques or tools well known in the art can be applied to the responses once they are gathered together in the database. The results can then be supplied to the survey customer in any desired form: printed hardcopy, magnetic media, electronic download, etc. At the option of the survey system administrator, a variety of data can be supplied, from the raw responses to the final statistical analysis. Because of the
30 computerized, networked architecture of the system, these responses can be made available almost immediately upon completion of the survey. Alternatively, intermediate results can also be provided as they are compiled by the system. This fast response time is one of the

benefits of an on-line survey system which can not be provided by traditional in-person, mail or telephone interview approaches. The central server is a different logical entity than the other servers in the system. If desired, it can be co-located on a physical computer which also hosts one or more of the original, 100, or destination, 102, servers. In the preferred
5 embodiment, the central server is hosted on a separate physical server. This provides an additional level of security because it makes it more difficult to locate, and allows flexibility in terms of shutting down the central server or disconnecting it from the network, whether for maintenance or security reasons. Performance can also be an issue where significant analysis of the results is performed on the central server. This processing would potentially be slowed
10 by the server(s) responding to survey requests, and, in turn, could slow the server responses.

In the present architecture, the survey system is connected to a large number of client sites, but remains largely independent of them. All of the processing associated with gathering and compiling survey data is performed on the original and destination servers and on the central server. The servers also store all of the forms, survey results, and user profiles.
15 A first advantage of this approach is that the system has very little impact on the client sites. Only the inclusion of a link on the client sites web page is required. This makes the inventive survey system more attractive to the client sites because they will incur no performance or storage cost by allowing the connection. A second advantage is that the servers remain completely under the control of the owner of the survey system. Logistics are greatly
20 simplified because there is no need to consult with the client sites prior to making a system change. Decisions as to increasing or decreasing the number of logical or physical servers can be based solely on performance and cost factors affecting the survey system. Security issues are more easily handled since there is no need to share the server systems, which contain the most sensitive data, with other users such as the client sites. In the preferred
25 embodiment, the various servers are not encrypted. However, in an alternative embodiment, the central server would be encrypted to allow access only by the owner and the destination servers would implement an encryption scheme which would allow access only by those users who access the systems through the normal login path via the original server. This would preferably be implemented via a two key encryption system in which the users key is
30 derived from the hash key used in the user to destination server mapping as described above.

Note also that this implementation enables literally unlimited client sites. Because the initial link is created and stored on the client site, the number of such sites has no impact on

the survey system itself. The more critical factor is the number, or more accurately, the frequency with which the links are activated and the surveys presented. The survey system is capable of handling millions of client sites which experience only light traffic as easily as it handles hundreds of sites with very heavy traffic. This range of application greatly increases the utility of the system.

The high level architecture of a logical server is presented in FIG. 7. The majority of the processing is performed by the transaction monitor, 136, which receives all incoming messages. This includes satisfying requests for survey pages, storing survey results when the user completes a survey, and compiling the local site statistics, 138, as a survey progresses.

The transaction monitor also handles the compilation of the system wide statistics and the adaptive processing as described below. Where the user's browser, 244, is Java enabled, it communicates directly with the transaction monitor via the Internet, 210. If the browser, 242, does not handle Java, the use of a local applet is not available. In this case, a CGI script, 134, or program will execute on the server as a front end to the transaction monitor to handle the details of communication with the browser. In this way, the transaction monitor need only support a single interface. The transaction monitor maintains the user data, 140, which includes identifying information as well as historical information, such as which surveys, or portions of surveys, the user has responded to. This allows the user to complete a survey at a later time and enables such survey techniques as question skip patterns. The survey data, 142, includes both the forms and templates used to present the survey and the survey responses.

Features and Functionality

Within the above architecture, each of the logical servers operates with a fairly high degree of autonomy while presenting surveys. At the start of a survey, each server is configured with a set of control parameters. As the survey progresses, each logical server updates its local statistics and the logical servers periodically communicate to update global statistics. The control parameters, local statistics and global statistics are then used by an adaptive selection algorithm, replicated on each server, to actively control the sampling process in order to achieve the goals of the survey. The efforts of the logical servers combine to present a single survey to a distributed user population in a statistically valid manner.

The overall goal is to randomly select from among the available user population a subset to whom surveys will be presented and to then collect and compile the survey results

from those who complete the survey. This process is presented graphically in FIG. 8. Users are initially represented as "hits" on a client site. A hit is essentially one occurrence of a user requesting a page from a server. The hits of concern to the survey system, 144, are those requests for pages which have had a link to the survey system embedded in them. When such a page is requested, a set of code associated with the link executes to determine whether to present a survey to this user. The details of this process are described below. Of those users presented with a survey, 146, some will decline to participate, some will start the survey but not finish, and some will complete the survey. The completed surveys, 148, are collected and compiled to generate the survey results. Note that this process is not self selecting. The visitors to a client site can not participate in the survey unless invited to do so by the system. This helps avoid the bias associated with convenience sampling and redundant user entries, for example.

The decision process as to which user hits to select for survey participation is handled by the original servers. A variety of appropriate techniques are well known in the field of statistical surveys. In the preferred embodiment, the system identifies a fixed ratio of initial hits to completed surveys and then attempts to accomplish that goal. Two simple approaches are available for reaching that goal. Assume that N out of 100 hits will need to be presented a survey in order to meet the completion percentage. The first approach is to present a survey to every N th hit reported to an original server (several client sites may report hits to the same original server). A second approach is to use a probabilistic or pseudo random process which selects $N/100$ of the hits with a reasonably random distribution over the visits.

The sampling process periodically adjusts the parameters of the user selection decision process in an attempt to keep the number of survey completions at or near the goal for the survey. While each original server may use a different set of tailored parameters, the same algorithm is used by each server. One purpose of the use of different parameter by each server is to allow a server which handles smaller sites to sample a larger percentage of the users while a server handling a very large site samples a smaller percentage. This would keep the absolute numbers in more equal proportion.

The primary output of the adaptation algorithm is the number of surveys to be presented during the next period. In generating this value, the algorithm also updates predictions for the number of hits expected during the period and the probability that the

presented surveys will be completed. The algorithm is depicted graphically in FIGs. 9 and 10 and presented as pseudo-code in FIGs. 11A & B.

The flow chart of FIG. 9 provides a high level view of the sequence of events. The processing for each iteration occurs at or near the end of the period. Two sets of data are gathered independently. The branch, 162-166, waits for time $(i * \Delta t - T_i)$ where $(i * \Delta t)$ represent the end of period "i" and T_i represents the mean time to complete a survey. At this time system wide (global) statistics are gathered on the number of hits occurring during the latest period and the number surveys completed during the period. Since any survey started after this point would not (on average) be completed within the period, it can be ignored.

Branch, 168-170, waits for $(i * \Delta t)$, the actual end of the period, and gathers the number of actual completions system wide. In the preferred embodiment, these collections are asynchronous and can be carried on concurrently. Upon completion of both gathering steps, the process synchronizes, 172, and begins the adaptation calculations based on these statistics. First, the number of Survey desired to be collected by the end of the upcoming (next) interval is determined, 174. From this, the number of desired collections during the next period can be determined, 176. The estimated probability of completion for the next period, 178, is combined with the desired number of completions to calculate the number of surveys that will have to be presented during the next period to achieve the goal for completions, 180. This value is provided to the survey processing portion of the system, 182, and the next period starts. This process continues iteratively until the desired number of collections have been made, 184.

This adaptive process periodically adjusts the survey system parameters, primarily, to allow for variations in the number of hits on the client pages, and the probability of completion of those surveys presented. In this way the system actively works to achieve its goal in the estimated amount of time and smoothes out variations in collection performance to maintain an even distribution across the duration of the survey. Other adaptive processes, considering other parameters are clearly possible. Using the present process, any form of adaptation used in conventional surveys can be used in a distributed environment. If desired, the frequency of the local adjustments can be higher than that of the collection of global statistics. For example: if the global statistics are compiled every 30 minutes, the servers can adjust their collection parameters at 15 minute or 10 minute intervals if desired.

In the preferred embodiment of the survey system, the above adaptation process makes use of both global and local statistics in adapting to changes. This allows individual servers to adapt to local changes and allows them to utilize individualized control parameters, such as the percentage of hits to which to present a survey. The process is illustrated in more
5 detail by the data flow diagram of FIGs. 10A & B. Further details are available by reference to FIGs. 11A - K. Referring to FIG. 10a, the steps of the adaptation process are illustrated starting at the end of the period and assuming that all system wide statistics have been gathered and stored. Details of this process are discussed below. First, the number of hits to be experienced by the system during the next period are calculated, 188, by retrieving the
10 number of hits during the last period from the stored global statistics, 186, and applying a predictive algorithm. This result is combined with the cumulative total number of hits received by the end of the last period to estimate the cumulative hits by the end of the next period, 190. This result is then multiplied by the constant completion ratio (determined at the start of the survey), available from the survey control parameters, 192, to calculate the desired
15 number of completions by the end of the next period, 194. Subtracting out the number of actual completions achieved by the end of the last period provides the number of completions needed during the next period, 196, in order to achieve the goal for the end of the period. This value is then divided among the available servers to determine the number of completions each servers must achieve, 198. In the preferred embodiment, this series of
20 calculations utilizes global data but is separately calculate by each server, using a common algorithm. In an alternative embodiment this calculation could be performed once, at least to the point of calculating the system wide desired calculations, and the result distributed.

Referring to FIG. 10B, the remainder of the adaptation process is illustrated. This portion of the process starts with the number of desired completions which has been allocated
25 to a particular server and utilizes local statistics and control parameters to develop the parameters for the presentation of surveys during the next period. A calculation of a local probability of completion, 204, generates a value which is combined with the allocated number of desired completions to generate the number of surveys which must be presented for this server to achieve its goal, 206. This value is used during the next period to determine
30 to which users surveys are presented, 208. During the process of presenting and collecting surveys, local statistics on the number of hits, presented surveys, and completions are maintained in local storage, 200. Note that this process has been presented as two separate

logical steps for illustration purposes only. The process can, and preferably is, implemented as a single combined process.

Referring to FIG. 12, the process of gathering the system wide statistics is illustrated. For purposes of statistics collection, the original servers, 210-216, are logically organized as a ring. When it is time to gather the statistics, a designated one of the servers, 210, retrieves its local use statistics and sends them to the next server, 212, in the ring. This server adds in its local statistics and forwards the message to the next server, 214. This process continues until the last server in the ring, 216, forwards the message to the first server, 210. At this point, the completion of the first circuit, the message contains the totals for all statistics being gathered. The message is then forwarded again, following the same path, to allow each server to make a copy of the total for its own use in its adaptation calculations. In the preferred embodiment the number of hits, h , and surveys presented, s , are compiled with one pair of message circuits and the number of collections, c , is compiled with a second independent sequence. In alternative embodiments other statistics can be compiled in the same manner and using any desired number of message circuit pairs. Where an error occurs, such as the failure of a host computer, interrupting the transmission of the statistics, any of several well known recovery techniques can be used, including a time-out followed by retransmission, skipping a server, or reversing the direction of flow.

It is important to note that this ring arrangement is only a logical arrangement of the servers and has no impact on their physical connections or on any other logical arrangement. In an alternative embodiment of the system a different type of logical arrangement could be used. One such anticipated alternative is a binary tree structure which would provide the well known $O(\log_2(n))$ performance improvement over the $O(n)$ performance of the ring structure described above.

In an alternative embodiment, the original servers are not informed of survey completion. This information is maintained on the destination servers. The collection of system side statistics then incorporates both original and destination servers to generate a complete set of statistics.

The JAVA code in FIG. 11 is presented as pseudo code to illustrate a particular implementation of segments of the adaptive process. Other, more complex, implementations are anticipated. As an example, the estimation routines illustrated utilize a simple weighted

average approach. The use of more accurate approaches, such least squares fit, will be used in alternative embodiments.

One feature of the present system is that the survey forms which are used to interface with the users will have a look and feel similar to that of the pages from which they entered the system. The same user, entering the system from two different client sites, would see the same content, but the presentation style would differ. This is accomplished by using a combination of forms and templates as illustrated in FIG. 13. The templates, 218, are essentially web pages created in the style of the client site. They may use colors, icons, background images, etc. which are available from the client site. Embedded in the template are one or more tokens which identify the location at which to insert survey text. In the preferred embodiment, these templates are created manually. In an alternative embodiment, they could be created automatically by scanning the client web page and extracting design elements. The forms, 220, contain the text which comprises the survey itself. The text is separated into one or more sections which correspond to the tokens embedded in the templates. When a survey page is requested, a script, 222, or program running on the server retrieves the template corresponding to the client site associated with the requesting user, and the form appropriate to the user (dependent on the survey to which responding and historical data such as how much of the survey has been completed) and merges them to create the survey page, 224, which will be presented to the user. The merge process includes the steps of replacing the tokens embedded in the template with the corresponding sections of text from the form. In this manner, a single set of survey forms can be presented to the users of many client sites, in a familiar style. Having only a single set of forms significantly reduces the overhead of creating and maintaining the forms. Utilizing the style of the client site makes the survey look more integrated and more acceptable to both the user and the owner of the client site.

In the preferred embodiment, the forms are created by the administrators of the survey system, based on the requirements of the person, or organization, requesting the survey (the survey customer). In an alternative embodiment, the survey customer could create their own forms and templates, possibly with the assistance of a menu driven interface. When authorized by the survey system, the customer could create and update the forms as desired. This ability would allow them to adapt the survey to the responses being received, changing information needs, or other factors important to them. With a knowledgeable customer, there

would be no need for human intervention by the survey system administrator. In a further alternative, survey results could be compiled automatically, or on request of the survey customer, and provided to the customer directly from the survey system.

When these alternatives are combined, a customer could develop the survey, supply
5 the forms and templates to the survey system, provide the needed configuration values (such as sample size desired, skip patterns to use, number and type of sites to survey, etc.), activate the survey collection process, and retrieve the survey results without involving a human from the survey system administrator. This approach would provide significant benefits in terms of turn-around time and responsiveness for the customers.

10 The design of the inventive survey system is such that it offers several advantages for collecting survey data. As discussed above, rapid turn-around of results is possible. The architecture is scaleable from a single host with a single logical server to a large number of hosts and logical servers. This allows the system to be adapted to surveying the users of a single site or the users of thousands of sites. Where a web site owner desires audience
15 research focused on that site, the system can be configured with link only from that site, with the option of differentiating different pages or entry points into the web site. The results will consist solely of user responses originating from that site. The system can also be configured to sample a large number of sites, all related to a particular content area (such as snow sports, or gardening) for audience research of that content area with a diverse sample population.

20 The system can, of course, also connect to diverse types of web sites to collect information on the general Web population. Additional capability can be enabled by recording with each response the user and the client site through which they entered. This would allow post processing of the responses to extract data specific to a single site or content area.

25 While the preferred form of the invention has been disclosed above, alternative methods of practicing the invention are readily apparent to the skilled practitioner. The above description of the preferred embodiment is intended to be illustrative only and not to limit the scope of the invention.

Claims

I/We claim:

1. Where one or more users are accessing a network via respective browser software, the network including one or more content providers, a system for gathering survey
5 information comprising:

one or more logical servers connected to the network and deployed independently of

operational infrastructure for each of said one or more content providers,

wherein at least one of said one or more logical servers stores one or more

survey questionnaires; and

10 an interface on a page on at least one of said one or more content providers, wherein

the interface connects said respective browser software to one of said one or

more logical servers when said respective browser software accesses said

interface, thereby allowing said respective browser software to communicate

with said one of said one or more logical servers over the network.
- 15 2. The survey system of claim 1 wherein said logical servers comprise at least one original

server and one destination server.
3. The survey system of claim 2 wherein said interface includes a link that connects to said

original server, and wherein said one or more survey questionnaires are stored on said

destination server.
- 20 4. The survey system of claim 3 wherein said original server and said destination server are

each able to provide the services of the other.
5. The survey system of claim 1 further comprising means for selecting a specific one of

said one or more logical servers to provide at least one of said one or more survey

questionnaires to a corresponding one of said one or more users.

6. The survey system of claim 5 wherein said means for selecting includes a computer program transmitted by said one of said one or more logical servers over the network to the respective browser software for said corresponding one of said one or more users for execution.
- 5 7. The survey system of claim 5 wherein said means for selecting is responsive to information provided by the corresponding one of said one or more users.
8. The survey system of claim 7 wherein said means for selecting deterministically selects said specific one of said one or more logical servers for all transactions with said corresponding one of said one or more users.
- 10 9. The survey system of claim 1 further comprising a central database connected to the network and logically distinct from said one or more logical servers, wherein the central database comprising non-volatile storage for survey results transmitted from said one or more logical servers over the network.
10. The survey system of claim 1 wherein at least one of said one or more logical servers
15 comprises non-volatile storage for profile information about at least one of the one or more users.
11. The survey system of claim 10 wherein said profile information for a specific one of said one or more users is stored on only one of said one or more logical servers.
12. The survey system of claim 10 wherein said profile information about a user comprises
20 data specifying which of said one or more questionnaires is to be presented to the user.
13. The survey system of claim 1 wherein said survey system restricts access to said one or more logical servers to a percentage of those of said one or more users who access said page on said at least one of said one or more content providers, and wherein said survey

system further comprises means to adaptively adjust said percentage while a survey is being presented.

- 14. The survey system of claim 13 wherein said adaptive adjustment means is responsive to a load on more than one of said one or more logical servers.
- 5 15. The survey system of claim 13 wherein said adaptive adjustment means adjusts said percentage for each of said one or more logical servers individually and is responsive to at least one of the following:
 - a first performance value which is specific for each individual server in said one or more logical servers, and
 - 10 a second performance value encompassing more than one of said one or more logical servers.
- 16. The survey system of claim 1 wherein said one or more survey questionnaires comprise at least one template specifying an aesthetic element and at least one form comprising one or more questions to be presented to a user from said one or more users and wherein said 15 template and said form are combined when presented to said user.
- 17. The survey system of claim 1 wherein said survey system is used by customers of the survey system to administer one or more surveys to the one or more users and wherein said survey system further comprises means for on-demand creation of said one or more surveys by said customers.
- 20 18. Where one or more users are accessing a network via respective browser software, the network including one or more content providers at least one of which makes available one or more pages for access by the respective browser software, a method of surveying the one or more users comprising:
 - establishing a connection between the respective browser software and a first logical

server when a link to said first logical server is activated, wherein at least one first logical server is provided to store and present one or more survey questionnaires, wherein each first logical server is deployed independently of operational infrastructure for each of said one or more content providers, and
5 wherein said link is provided on at least one of said one or more pages; and said first logical server transmitting at least one of said one or more survey questionnaires to the respective browser software over the network in response to the activation of said link.

19. The method of surveying of claim 18 further comprising providing a second logical server
10 that is deployed independently of operational infrastructure for each of said one or more content providers, wherein said link connects to said second logical server when activated.

20. The method of surveying of claim 18 wherein a plurality of said one or more pages contains corresponding links and wherein said method further comprises:
15 identifying a user from said one or more users prior to establishing said connection; and
deterministically connecting the user to the same first logical server independently of which of said corresponding links was initially activated.

21. The method of surveying of claim 18 further comprising selecting, from among a portion
20 of said one or more users activating links available on said one or more pages, a percentage of said portion of said one or more users to be connected to said first logical server with essentially equal probability of connection for each user in said portion of said one or more users.

22. The method of surveying of claim 21 wherein selecting periodically adapts to a load on said first logical server by adjusting the percentage of said portion of said one or more users.
23. The method of surveying of claim 18 further comprising uniquely identifying each of said one or more users connected to said first logical server so as to continue presentation of a survey to said each of said one or more users at a corresponding point where said each of said one or more users previously stopped.
24. The method of surveying of claim 18 wherein said method further comprises uniquely identifying each of said one or more users connected to said first logical server so as to implement a survey technique requiring selection of a subset of questions from said at least one of said one or more survey questionnaires for presentation to the each of said one or more users.
25. The method of surveying of claim 18, further comprising providing means to select said link only in those of said one or more pages that are associated with a common category of goods or services.
26. The method of surveying of claim 18 further comprising providing intermediate survey results while survey administration continues.
27. The method of surveying of claim 18 further comprising providing survey results in an electronic format over the network.
28. Where one or more users are accessing a network via respective browser software, the network including one or more content providers at least one of which makes available one or more pages for access by the respective browser software, wherein at least one of said one or more pages contains a link to a survey system, said survey system comprising: at least one original logical server connected to the network and serving as a target of

the link;

at least one destination logical server comprising non-volatile storage for one or more survey questionnaires and profile information about at least one of the one or more users, wherein said profile information for a specific one of said one or more users is stored on only one destination logical server, and wherein said one or more survey questionnaires comprising at least one template specifying an aesthetic element and at least one form comprising one or more questions to be presented to a user from said one or more users and wherein said template and said form are combined when presented to said user;

10 means for selecting a specific destination logical server to provide said one or more survey questionnaires to the user, wherein said selecting means deterministically selects the same destination logical server for all transactions with said user, said selecting means being responsive to an activation of the link;

15 means for restricting access to said specific destination logical server to a percentage of those of said one or more users who access a link-containing page in said one or more pages; and

means to adaptively adjust said percentage while a survey is being presented, wherein said adaptive adjustment means is responsive to a load on more than one of said original and destination logical servers.

20

FIG. 1

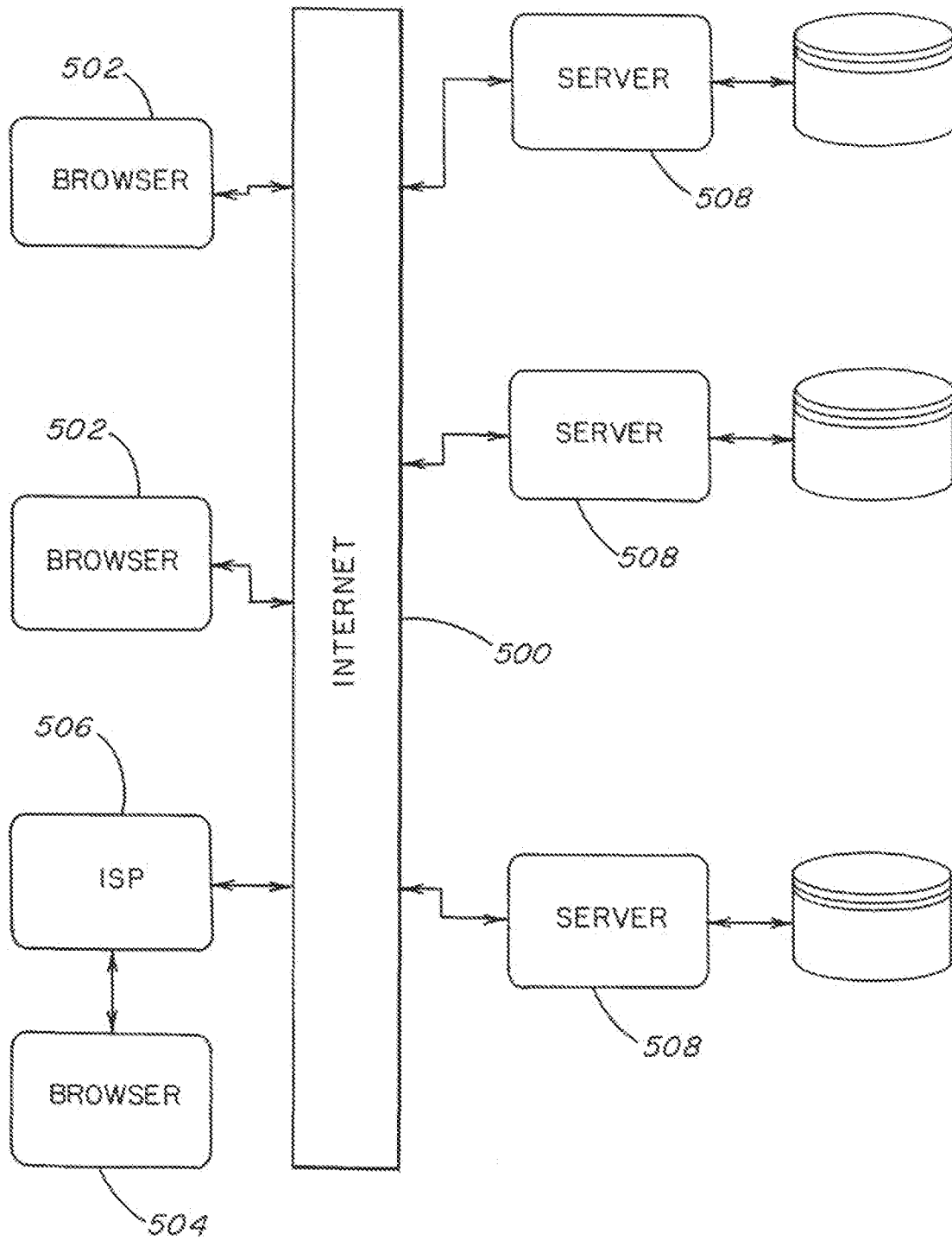


FIG. 2

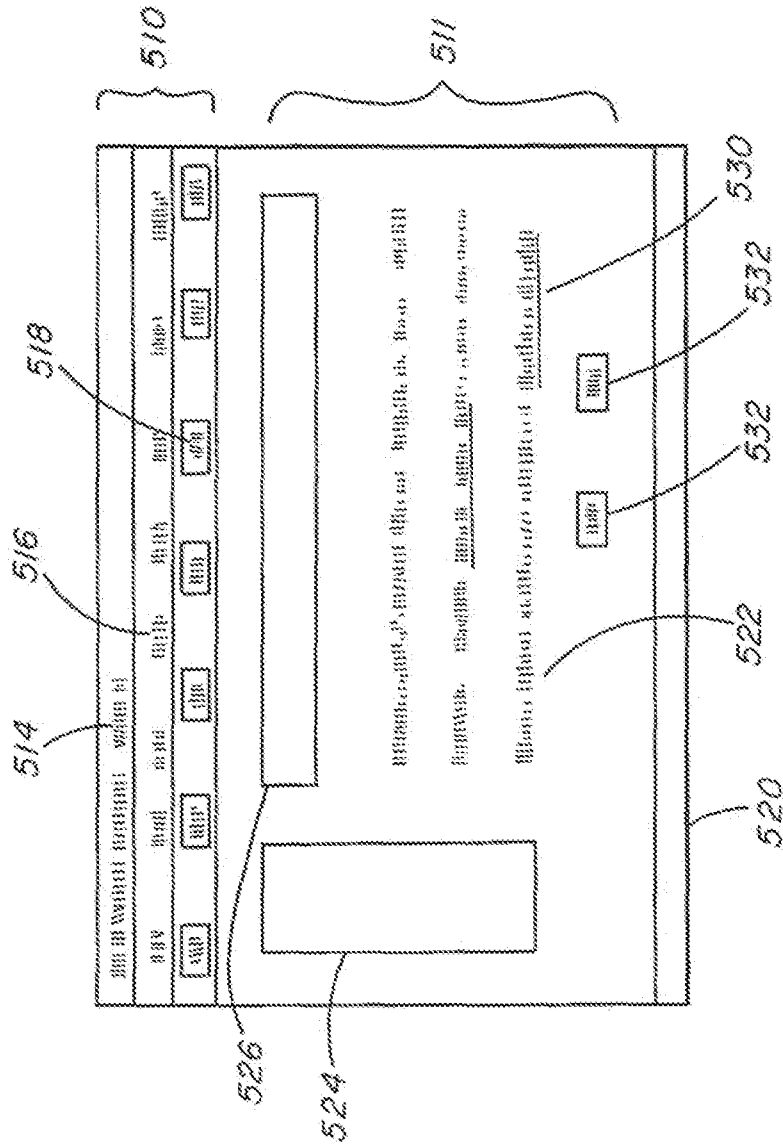


FIG. 3

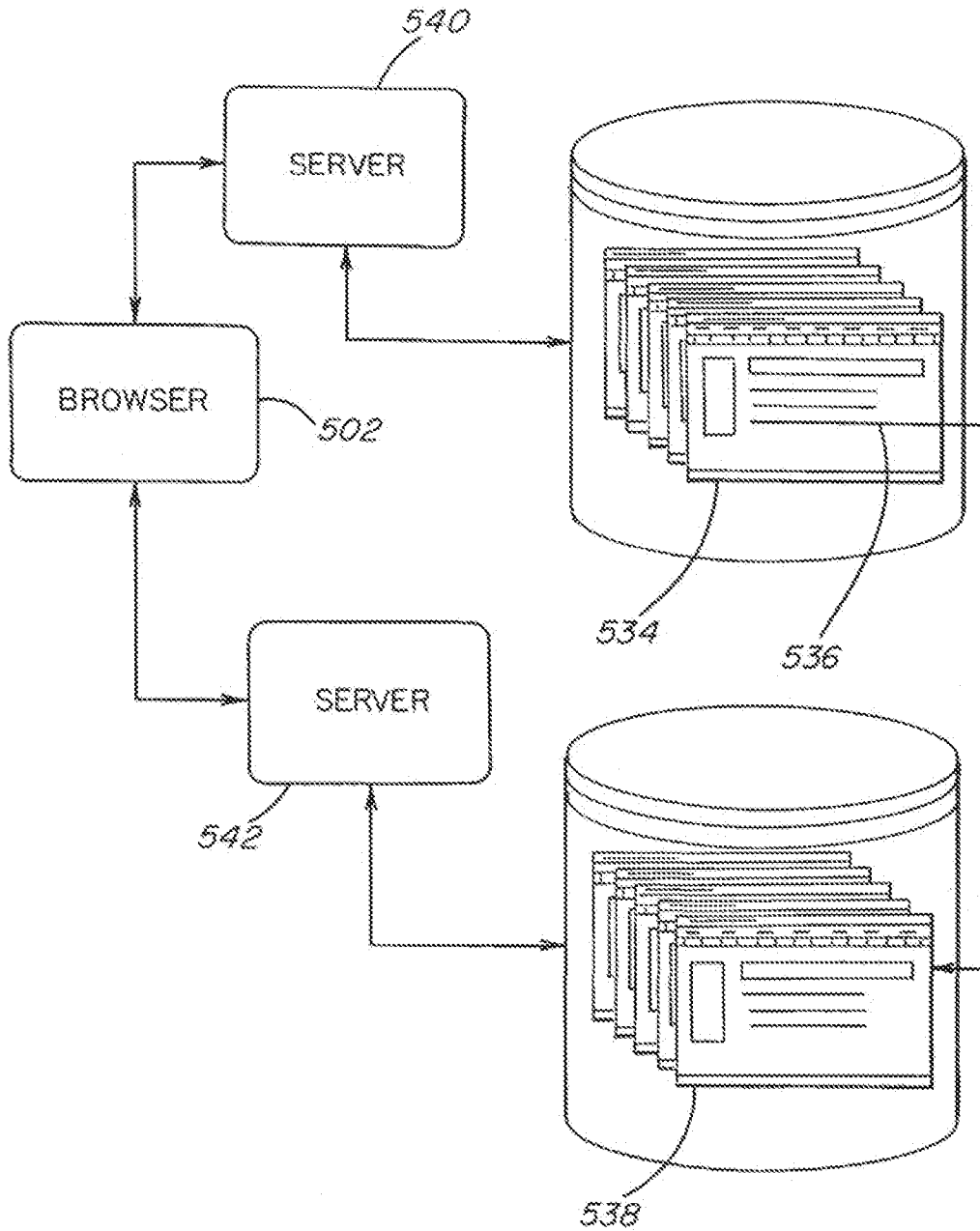


FIG. 4

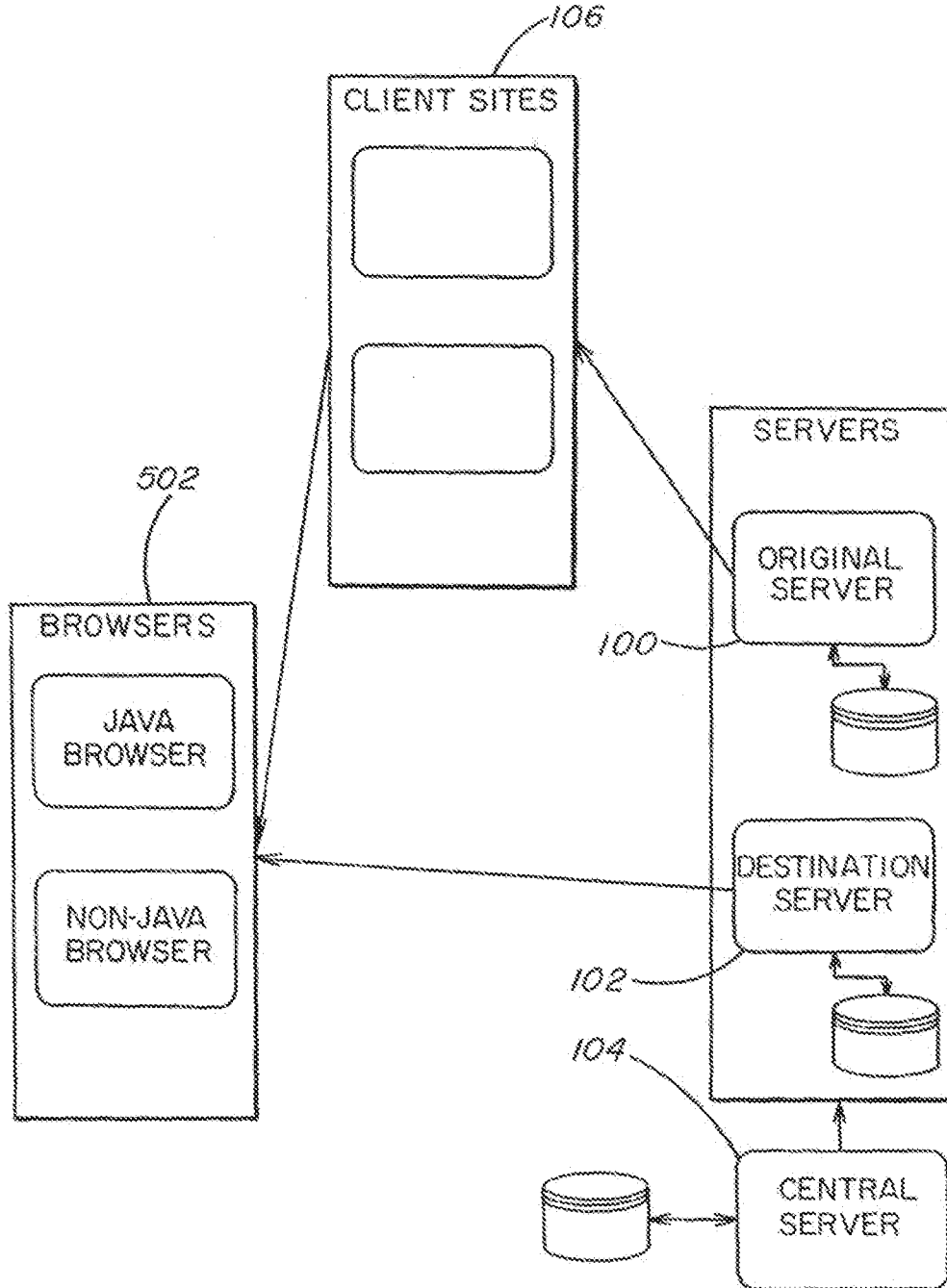


FIG. 5

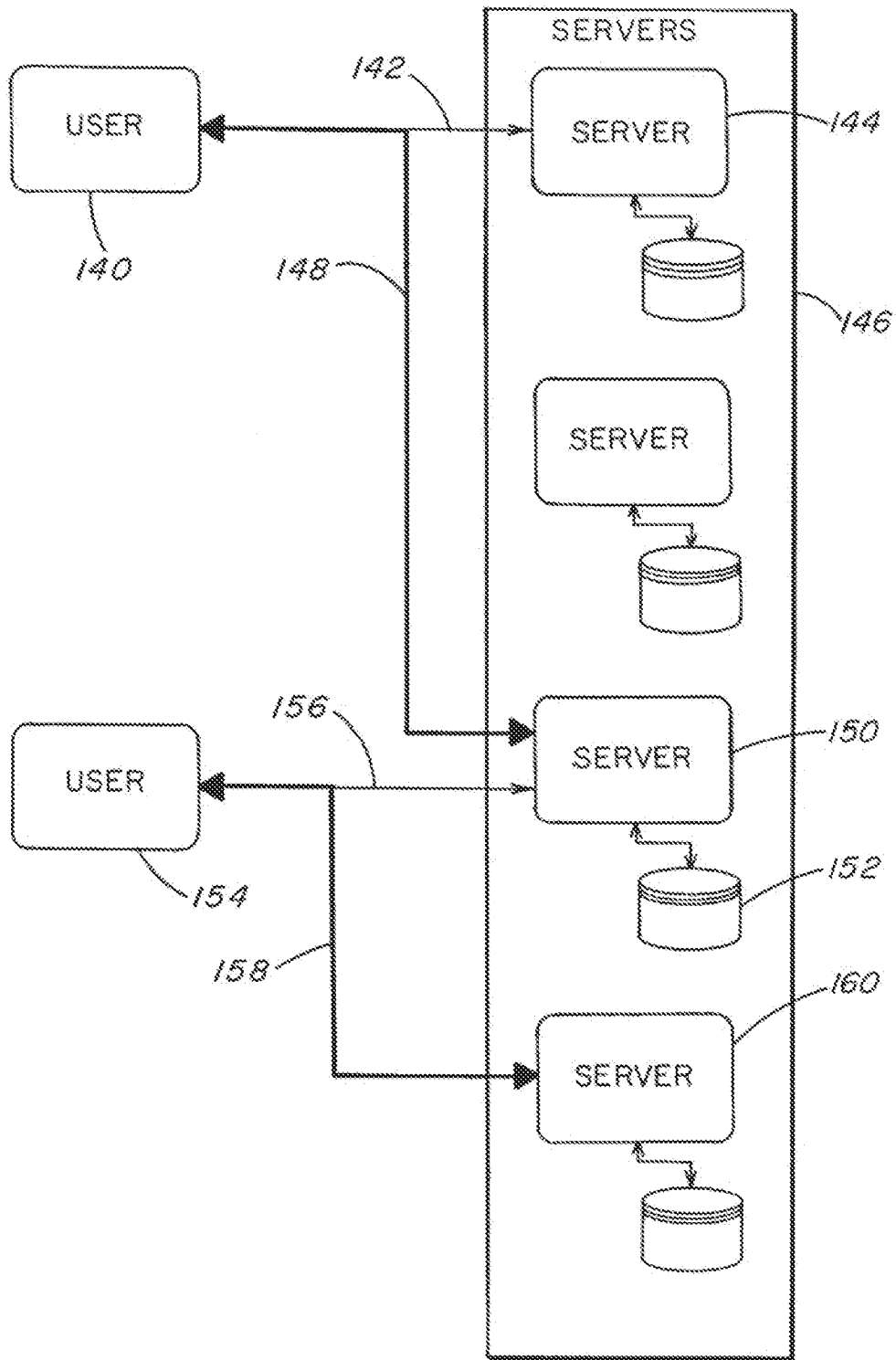
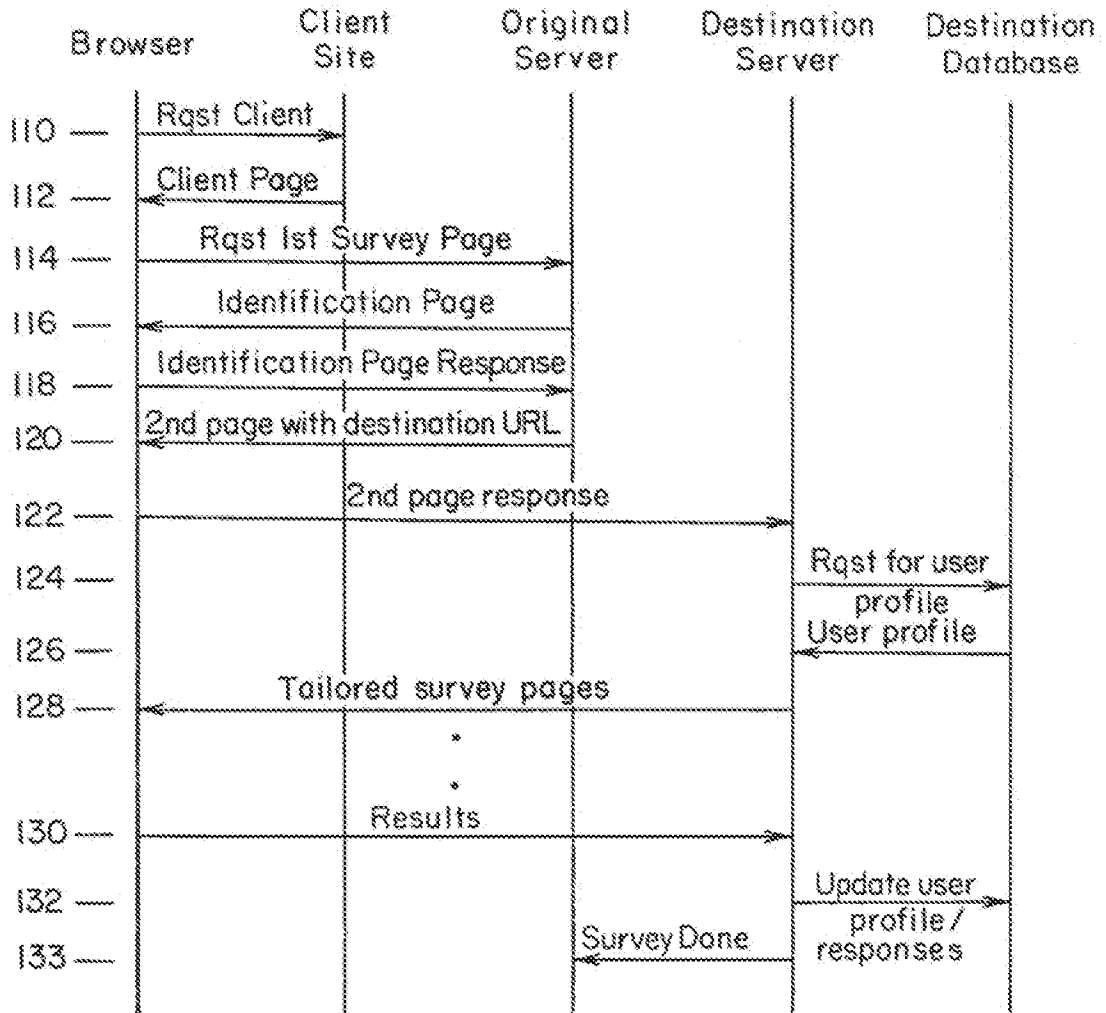


FIG. 6



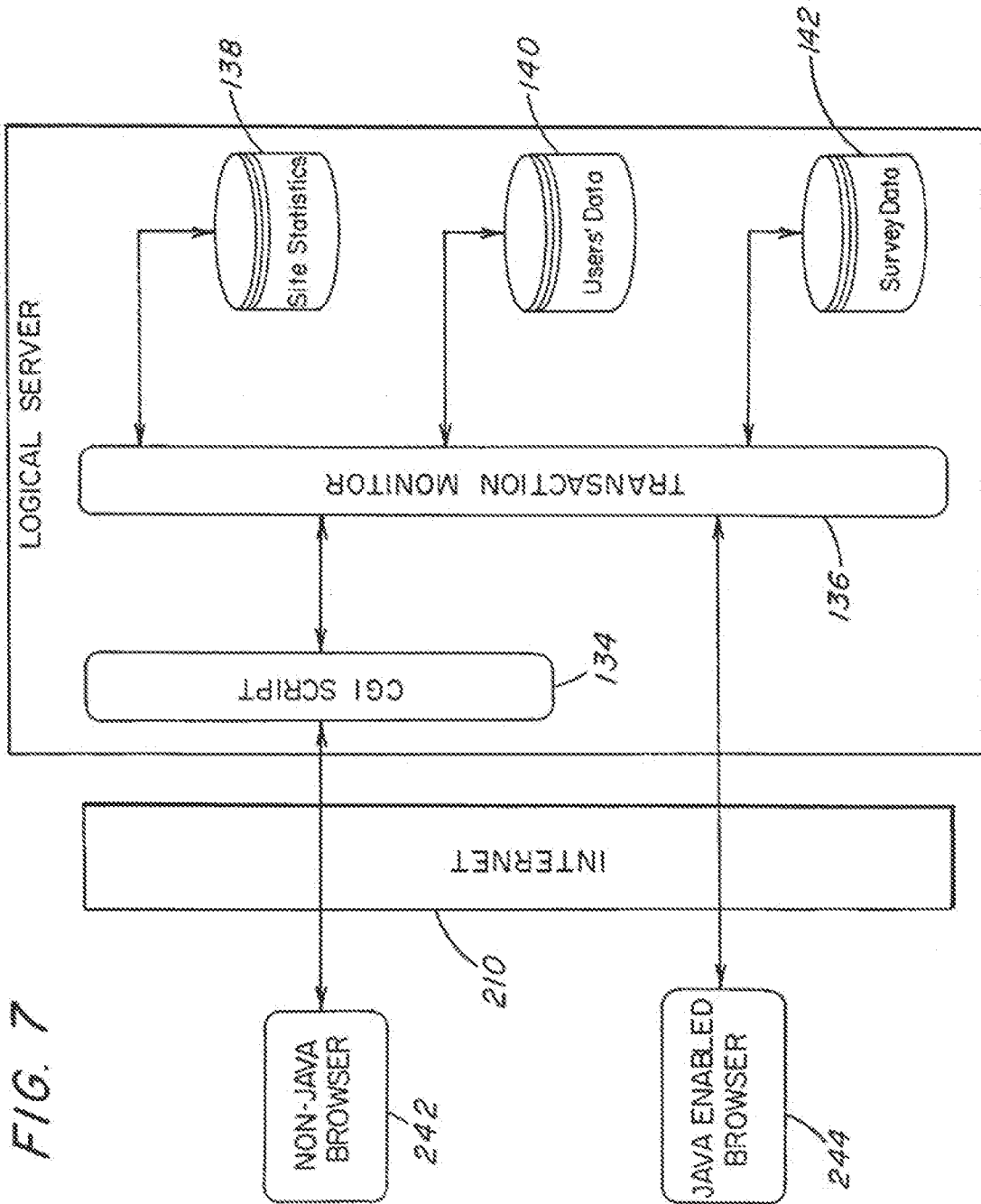


FIG. 7

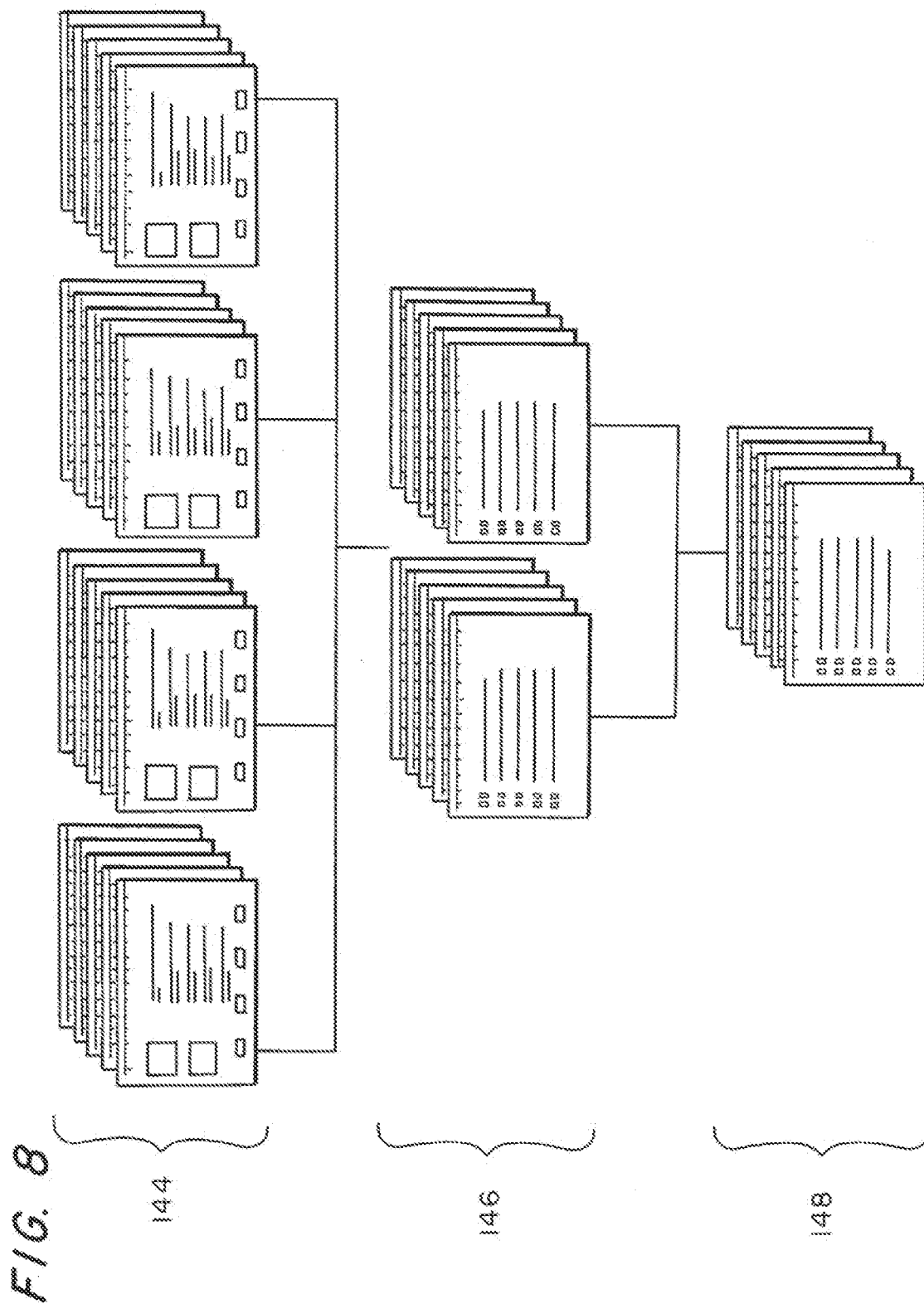
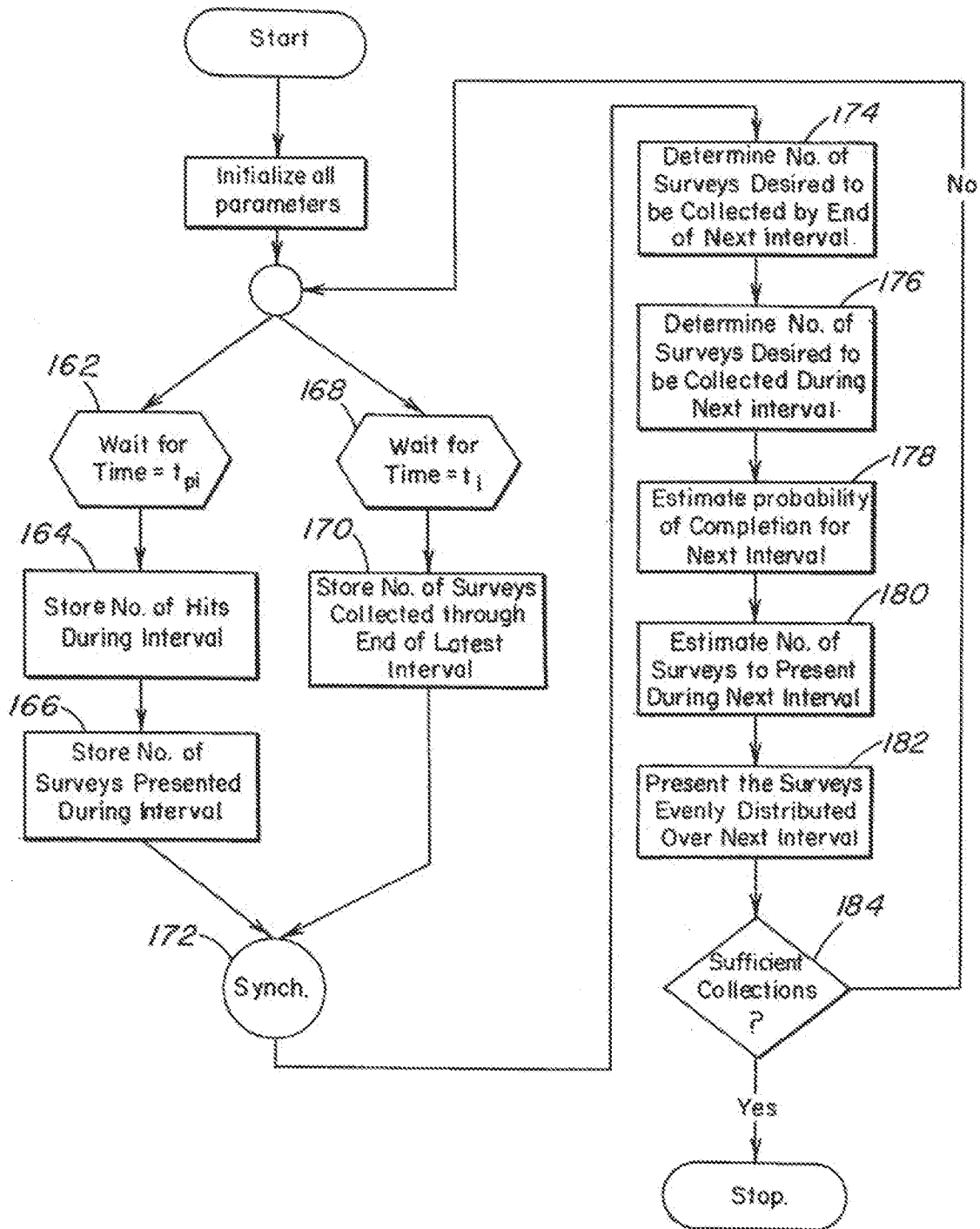
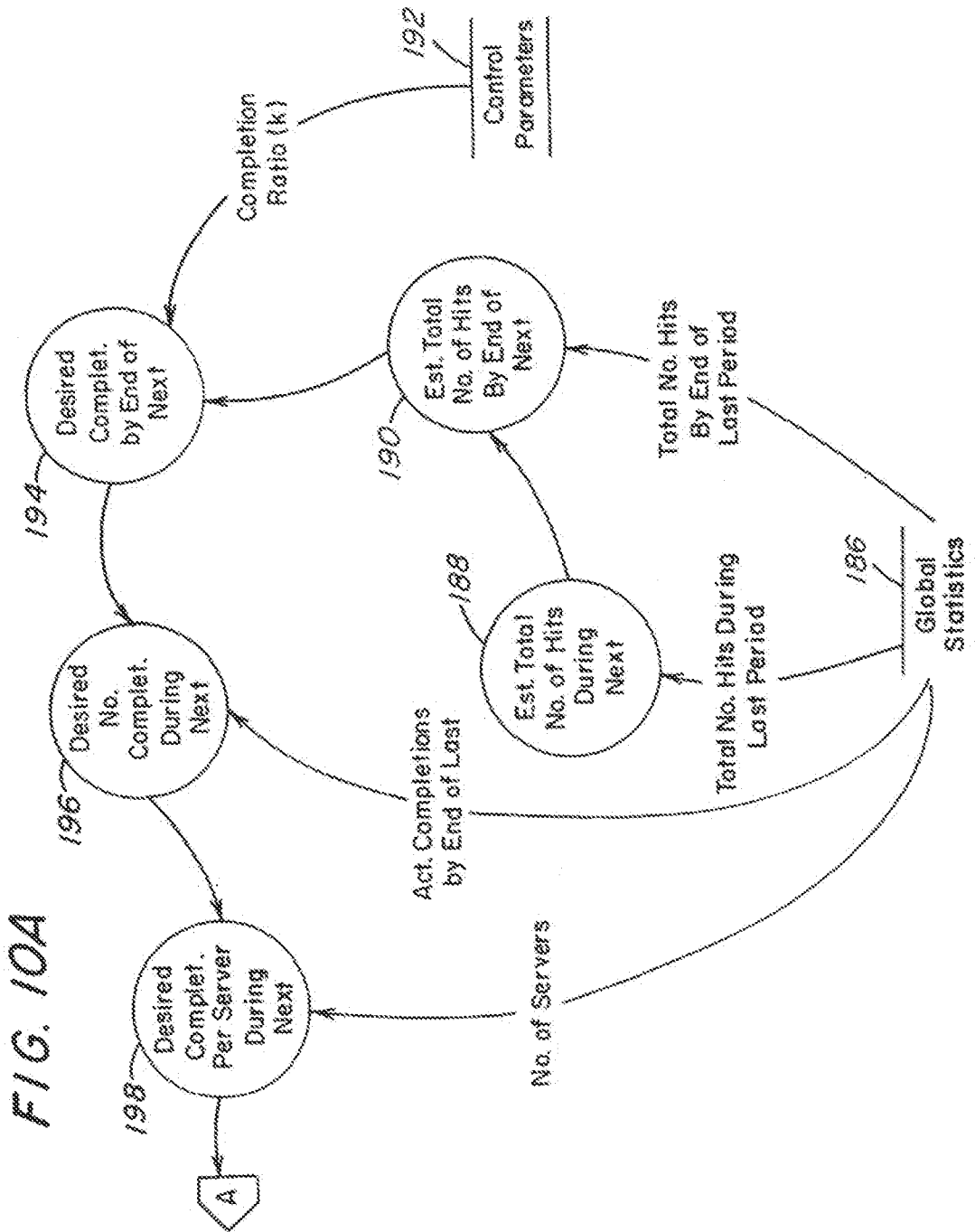


FIG. 8

FIG. 9





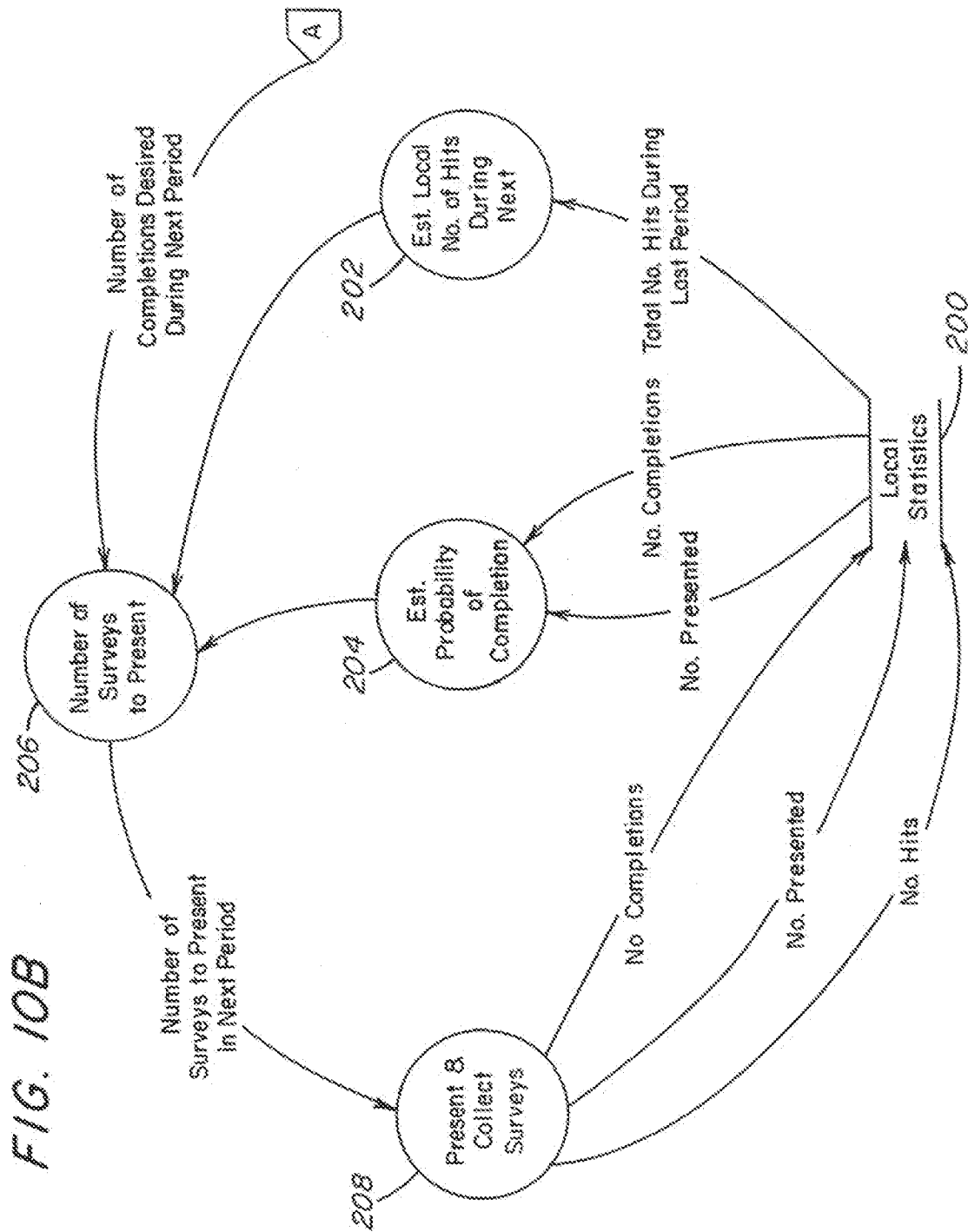


FIG. 10B

12 / 24

```

import java.util.*;
import java.io.*;

/*****
// Class Tm - the Transaction Monitor
*****/
public class Tm {

// CONSTANTS
private int dtMinute=30;
private int tcMinute=2;
private double pc=0.1;
private double include=1;

// VARIABLES

// Default control parameters if none specified by clients
public int periodDays=7; // Number of days
public int cFinal=2000; // Number of surveys to collect
public int hFinal=10000; // Minimum number of hits

/* Number of hits, shown surveys and collected surveys */
public int h{};
public int s{};
public int c{};

// RING BUFFERS
private int nfit1=5; // Size of ring buffer 1
private int nfit2=4; // Size of ring buffer 2
private int nfit3=4; // Size of ring buffer 3

private int rpointer1; // Pointers to current location in
each of the ring buffers
private int rpointer2;
private int rpointer3;

private double[] ring1; // h, number of hits
private double[] ring2; // c, number of collections
private double[] ring3; // s, number shown

private final double a=-1.78; //Weights used in error calculation
for ds
private final double b=0.36;

/* Variables used to convert to seconds */
private int period;
private int dt;
private int T0; // Initial value for T

```

FIG. 11A

13 / 24

```

private double k; // Percentage of hits to be maintained

/* Values for a certain period */
private double hNext;
private double yNext;

/* Values used to calculate frequency */
private double dc;
private double dh;
private double ds;
private double old_dc;
private double dif_dc;
private double error;
private double old_error;

private double cperiod;
private double speriod;

// other variables
private int i=0;
private int bye=0;
private long my_time;
private long next_time;

// FILES: HOST SITE STATISTICS, SURVEYS AND DATA
// WE'LL ONLY USE HOST SITE STATISTICS

public File host_site;
public FileOutputStream file_writer3;
public BufferedOutputStream file_writer2;
public PrintStream file_writer;

/* MAIN LOOP, WAITING FOR APPLETS TO CONTACT, AND
UPDATING VALUES EVERY 30 MINUTES */

public static void main(String args[]) {

    // temporary values for h,c and s (in every iteration).
    // To not have to write every time h[indax]
    int temp_h;
    int temp_c;
    int temp_s;

    // These variables indicate the number of filled elements in
    the ring buffers.
    int full1=0;

```

FIG. 11B

14 / 24

```

int full2=0;
int full3=0;

/* FIRST INITIALIZATIONS */
Tm my_tm=new Tm();
my_tm.establish_values(0,2100,15000);

// INITIALIZE RING BUFFERS

my_tm.ring1=new double[my_tm.nfit1]; //Allocate storage for
the ring buffers
my_tm.ring2=new double[my_tm.nfit2];
my_tm.ring3=new double[my_tm.nfit3];

my_tm.rpointer1=my_tm.nfit1; //Init all ring pointers
to last location in buffer
my_tm.rpointer2=my_tm.nfit2;
my_tm.rpointer3=my_tm.nfit3;

// Convert time periods to seconds
my_tm.period=my_tm.periodDays * 60 * 60 * 24;
my_tm.dt=my_tm.dtMinute * 60;
my_tm.T0=my_tm.tcMinute * 60;
my_tm.k= (double) (my_tm.cFinal/my_tm.hFinal);

/* Establishing values for a period */
my_tm.hNext= my_tm.hFinal * my_tm.dt / my_tm.period;
my_tm.yNext= (my_tm.cFinal * (my_tm.dt - my_tm.T0))/
my_tm.period;

// ESTABLISHING NEW DATE
Date my_date= new Date();
my_tm.my_time=my_date.getTime();
my_tm.next_time=my_tm.my_time+300000; //Interval of 30
minutes

/* Alternative: Implement a thread running background to
receive applet's requests
and update c[],h[] and s[] asynchronously */

/* Initializing for first sample period */
/* These arrays are defined for one year */
my_tm.h=new int[17520];
my_tm.s=new int[17520];
my_tm.c=new int[17520];

my_tm.h[0]=0;
my_tm.s[0]=0;

```

FIG. 11C

15 / 24

```
my_tm.c[0]=0;

Recv receive_thread= new Recv();
receive_thread.start();

// temporary
my_tm.make_files();

// MAIN LOOP
do {
    /* Get current time */
    Date next_date=new Date();
    my_tm.my_time=next_date.getTime();

    /* If 30 minutes are gone... */
    if (my_tm.my_time>=my_tm.next_time){

        /// Updating next time
        my_tm.next_time=my_tm.next_time+300000;

        // GET VALUES FOR CURRENT PERIOD
        temp_h=receive_thread.get_h();
        temp_c=receive_thread.get_c();
        temp_s=receive_thread.get_s();

        my_tm.pc=temp_c/temp_s;

        my_tm.h[my_tm.i]=temp_h;
        my_tm.c[my_tm.i]=temp_c;
        my_tm.s[my_tm.i]=temp_s;

        my_tm.error=my_tm.yNext-temp_c;

        /* CALL TO UPDATING */
        /* To update values to calculate frequency */

        my_tm.updating(my_tm.hNext,my_tm.yNext,my_tm.k,my_tm.error,my_
tm.incl,my_tm.pc);

        /* UPDATE VALUES IN HOST SITE FILES */

        my_tm.file_updating(my_tm.h[my_tm.i],my_tm.c[my_tm.i],my_tm.s[
my_tm.i],my_tm.i);

        /* Call contact other servers */
        my_tm.coherency();
    }
}
```

FIG. 11D

16 / 24

```

    my_tm.i++;
    // RING BUFFERS !!! -> Update values for all 3 ring
buffers
    // WE'LL USE THE SAME FUNCTIONS TO WORK WITH RING BUFFERS

    // Ring buffer for h
    if (full1<my_tm.nfit1)
        full1++;

    my_tm.rpointer1=my_tm.new_ring_pointer(my_tm.rpointer1,my_tm.n
fit1);
    my_tm.ring1=my_tm.update_ring(my_tm.ring1, temp_h,
my_tm.rpointer1);

    // Ring buffer for pc
    if (full2<my_tm.nfit2)
        full2++;
    my_tm.rpointer2=my_tm.new_ring_pointer(my_tm.rpointer2,
my_tm.nfit2);
    my_tm.ring2=my_tm.update_ring(my_tm.ring2, temp_c,
my_tm.rpointer2);

    // Ring buffer for incl
    if (full3<my_tm.nfit3)
        full3++;
    my_tm.rpointer3=my_tm.new_ring_pointer(my_tm.rpointer3,
my_tm.nfit3);
    my_tm.ring3=my_tm.update_ring(my_tm.ring3, temp_s,
my_tm.rpointer3);

    // Reset local values for next period:
    my_tm.h[my_tm.i]=0;
    my_tm.c[my_tm.i]=0;
    my_tm.s[my_tm.i]=0;

    // Reset h,c,s values in the receive thread for the next
period
    receive_thread.set_h();
    receive_thread.set_c();
    receive_thread.set_s();

    // UPDATE YNEXT, PC, INCL

    // YNEXT
    // First, we update

```

FIG. 11E

17 / 24

```

    my_tm.hNext=my_tm.approxhNext(my_tm.hNext,my_tm.rpointer1,my_t
m.i,my_tm.ring1,full1,my_tm.nfit1);
    // LAST PARAMETER IS USED FOR THE 6 FIRST ITERATIONS,
    // WHEN THE RING BUFFER IS STILL NOT FULL !!!
    my_tm.yNext= my_tm.k * my_tm.hNext;

    // PC
    my_tm.pc=my_tm.approxProb(my_tm.pc, my_tm.rpointer2, my_tm.i,
my_tm.ring2, full2, my_tm.nfit2);
    // INCL
    my_tm.incl=my_tm.approxincl(my_tm.incl, my_tm.rpointer3,
my_tm.i, my_tm.ring3, full3, my_tm.nfit3);
}

}while (c[i]<final); //end of MAIN LOOP
}

// CODE FOR RING BUFFER

// Update ring pointer to next location, moves backwards, at
zero
// it resets to end
int new_ring_pointer(int ring_pointer, int Size){

    if (ring_pointer<=0)
        ring_pointer=ring_pointer + Size - 1;
    else
        ring_pointer--;

    return ring_pointer;
}

//Insert data into next location in ring
double[] update_ring(double[] RingArr, double Val, int
ring_pointer){

    RingArr[ring_pointer]=Val;

    return RingArr;
}

// CODE TO CALCULATE NEXT HNEXT
double approxhNext(double h, int ring_pointer, int I, double[]
h_ring, int full1, int tamany){

```

FIG. 11F

18 / 24

```

double[] c;
double hacu=0;

c=new double[5]; //Weights to be used for estimation
c[0]=0.3;
c[1]=0.25;
c[2]=0.20;
c[3]=0.15;
c[4]=0.1;

for (int j=0; j<full1; j++)
    hacu=hacu + (c[j] * h_ring[tamany-j-1]);
hacu=hacu/full1;

return hacu;
}

// CODE TO CALCULATE NEXT PC
double approxProb(double pc,int ring_pointer,int i,double[]
ring,int full2,int tamany){

double[] c;
double pcacu=0;

c=new double[4]; //Weights to be used for estimation
c[0]=0.35;
c[1]=0.30;
c[2]=0.20;
c[3]=0.15;

for (int j=0;j<full2;j++){
    pcacu=pcacu+(c[j]*ring[tamany-j-1]);
    file_writer.println("--> "+ring[tamany-j-1]);
}
pcacu=pcacu/full2;
file_writer.print("PCACU:#+pcacu+#");

return pcacu;
}

// CODE TO CALCULATE NEXT INCLINATION (SLOPE)
double approxIncl(double incl,int ring_pointer,int i,double[]
ring,int full3,int tamany){

double[] c;
double inclacu=0;

```

FIG. 119

19 / 24

```

c=new double[4];    //Weights to be used for estimation
c[0]=0.35;
c[1]=0.30;
c[2]=0.20;
c[3]=0.15;

for (int j=0;j<full3;j++)
    inclacu=inclacu+(c[j]*ring[tamany-j-1]);
inclacu=inclacu/full3;

return inclacu;
}

public void updating(double hNext,double yNext,double k,double
error,double incl,double pc){

/* MAIN LOOP */
if (i!=0)
    old_dc=dc;
else
    old_dc=yNext-c[1];    // First value

dc = yNext - c[1];    // Number to collect in next period
dh = hNext-h[1];    // Number of hits in next period
dif_dc = dc - old_dc; //Unused?

if (i==0) {
    if (pc>0.0)
        //ds is lesser of expected hits or dc/pc
        ds=Math.min(dh, (dc/pc));
    else
        ds=dh;
}
else { // i!=0
    if (pc>0.0)
        // ds is lesser of expected hits or previous ds corrected
        // for error
        ds=Math.min(dh, (((ds+dc/pc)/2) + ((error * a)*(1-
(b*incl))))));
    else
        ds=dh;
}
}

// INTERFACE FUNCTION TO CHANGE CONTROL PARAMETERS
// TO BE USED BY CLIENTS, BEFORE STARTING SURVEY

// To allow for changing only certain values:

```

FIG. 11H

20 / 24

```
// If a parameter=0 -> this value will remain unchanged

public void establish_values(int Duration, int Collections, int
Hits) {

    if (Duration!=0)
        periodDays=Duration;
    if (Collections!=0)
        cFinal=Collections;
    if (Hits!=0)
        hFinal=Hits;
}

// INTERFACE FUNCTION -> This function creates the three files
// in the server (if they didn't exist)

public void make_files() {

try{
    host_site= new File("hostsite");
    file_writer3= new FileOutputStream(host_site);
    file_writer2= new BufferedOutputStream(file_writer3);
    file_writer= new PrintStream(file_writer2,true);

} catch(IOException e) {
    System.out.println(" PROBLEM CREATING FILES");
}

}

/// FUNCTION TO UPDATE FILE HOST SITE STATISTICS

public void file_updating(double h,double c,double s,int i){

try{
    file_writer.print(i+" "+h+" "+c+" "+s+"\n");
} catch (IOException e){
    System.out.println("PROBLEM WRITING TO HOST SITE STATISTICS");
}

}

// FUNCTION TO MANTAIN COHERENCY BETWEEN SERVERS !!!!

public void coherency(){

// Currently, being implemented.
}

}
```

FIG. 11I

21 / 24

```
} // End of class Tm

/* CODE WAITING TO RECEIVE APPLET'S REQUESTS, AS WELL AS
WHEN A SURVEY IS SHOWN OR FINISHED */

//*****
// Class Recv - thread for receiving statistics
//*****
class Recv extends Thread{

// We define temporary variables, h,s,c
// Number of hits, shown and collected surveys

public int h;
public int s;
public int c;

public int exit=0;

// INTERFACE: Methods to access these variables

public int get_h(){
    return h;
}

public int get_s(){
    return s;
}

public int get_c(){
    return c;
}

// Methods to reset variables

public void set_h(){
    h=0;
}

public void set_c(){
    c=0;
}

public void set_s(){
    s=0;
}
}
```

FIG. 11J

22 / 2A

```
public void run(){
    do{
        yield();
        // Code not yet implemented.
        // Will receive data, identify the value, and update the
        // corresponding variable
        }while(exit!=1);
    }
} //End of class Recv
```

FIG. 11K

FIG. 12

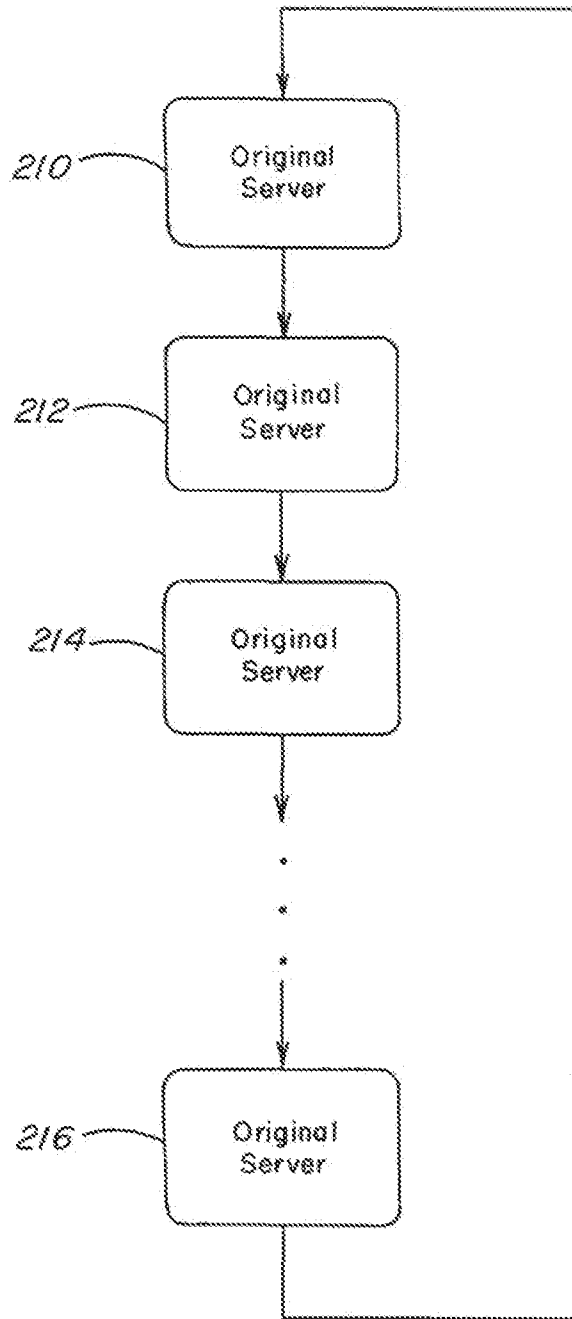
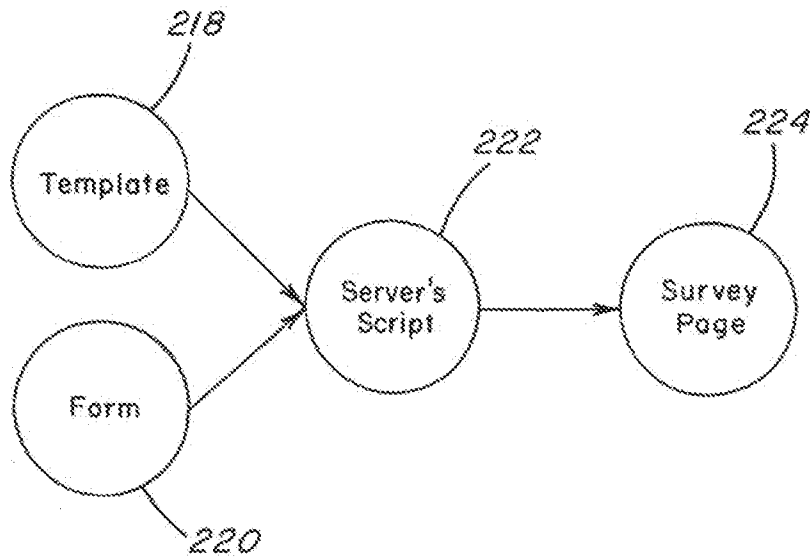


FIG. 13



(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
14 December 2000 (14.12.2000)

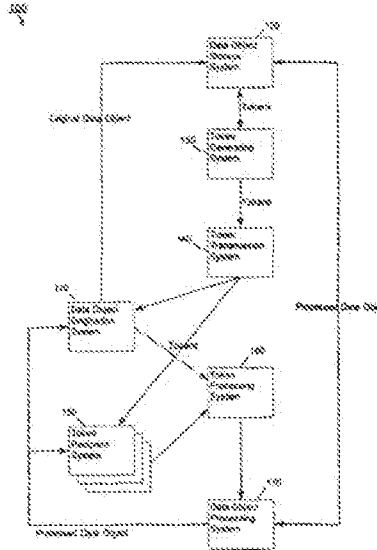
PCT

(10) International Publication Number
WO 00/75779 A2

- (51) International Patent Classification⁷: G06F 9/46 (US). ARCHER, Emory, Scott [US/US]; 4799 White Rock Circle, #E, Boulder, CO 80301 (US).
- (21) International Application Number: PCT/US00/15224
- (22) International Filing Date: 2 June 2000 (02.06.2000)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data: 60/137,568 4 June 1999 (04.06.1999) US
- (74) Agents: SABETT, Randy, V. et al.; Coolby Godward LLP, One Freedom Square-Reston Town Center, 11951 Freedom Drive, Reston, VA 20190-5601 (US).
- (81) Designated States (national): AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, 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, EG, KZ, MD, RU, TI, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE,
- (71) Applicant (for all designated States except US): TWITNESS, INC. [US/US]; Suite 2N, 2995 Wilderness Place, Boulder, CO 80301 (US).
- (72) Inventors; and
- (75) Inventors/Applicants (for US only): LAMBERT, Francis, T. [US/US]; 1901 Spruce Street, Boulder, CO 80302

[Continued on next page]

(54) Title: TOKEN BASED DATA PROCESSING SYSTEMS AND METHODS



WO 00/75779 A2

(57) Abstract: A token based data processing system is disclosed that uses data object tokens (DOTs) to manage the associated data objects within the data processing system. The numerous advantages of a token based data processing system stem from the ability to utilize a DOT separate from its associated data object. A data processing system according to the present invention can include a data object origination system, a data object storage system, a token generating system, a token transmission system, one or more token recipient systems, a token processing system, and a data object processing system. The token generating system can generate a DOT that contains metadata corresponding to the information that the token generating system receives regarding the data object, including its attributes, the parameters stored with it, and other information regarding its originating system and its recipient system. After being distributed by a token transmission system, token recipient systems can analyze and process the DOTs in lieu of the associated data objects.



IT, LU, MC, NL, PT, SE), OAPI patent (BF, BJ, 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:

----- *Without international search report and to be republished upon receipt of that report.*

TOKEN BASED DATA PROCESSING SYSTEMS AND METHODS

RELATED APPLICATION

The present application claims priority to Provisional Application No.
 5 60/137,568, filed June 4, 1999, which is incorporated herein by reference in its entirety.

BACKGROUND

Field of the Invention

This invention relates to a data processing system containing diverse data
 10 objects and to methods for using those data objects. More specifically, the invention
 relates to a data processing system where the system utilizes data objects called
 "tokens" to perform processes related to, and upon, other data objects to which the
 tokens refer.

15 **Description of the Related Art**

The advent of electronic data interchange, the growth of electronic
 communications, the growth of the number of electronic communication users, the
 growth of the availability of information, and the growth in the size, number and types
 of data objects has greatly increased the requirements of both data storage and data
 20 communication bandwidth. Typical use of these data objects requires processing and
 transfer of the data object itself, often due to the complexity or sensitivity of the data
 content of the object, or to the limitations of the data systems used to manipulate the
 data object. These systems generally require transfer or transmission of the entire data
 object for a user to be able to edit, preserve, alter, or "own" the data on the person's

local computing system. Examples of these systems can include any that use e-mail attachments, ftp file transfer, and http file transfer.

Unfortunately, a modern data object storage event or a transmission event, such as an electronic document transfer via e-mail, may contain a large volume of data that one entity wishes to transfer to another entity. This type of transfer forces both the sending and receiving parties to expend substantial resources in terms of CPU usage, data storage, and transmission bandwidth at both ends of the transmission. The cost and lost productivity from these types of transmissions will increase as the number and size of these transmissions increase, as is the case in modern data communications.

Additionally, the originator of the transmitted data may wish to keep the transfer of the data secure and private. This requires sending the data using encryption, which requires additional CPU usage. The originator may also wish to guarantee that the transmitted data arrived exactly as sent, which requires that all of the data be sent, verified, and the verification returned to the originating system each time the data is transmitted. These security and authentication technologies must also be present on the receiving system. Prior to the data being released to the transmission network, the originator must have knowledge that the receiving system is capable of reliably authenticating, decrypting, and controlling access to the recipient's copy of the data object.

In many cases, the receiver of the data in this type of transmission is forced to receive all of the transmitted data whenever the originator sends it, regardless of whether or not it is convenient for the receiver at that time. An example of this is a mobile computer user being forced to download a large e-mail attachment over an expensive hotel data line so that they can retrieve all of their e-mail. Another example

is a wireless data transceiver being forced to download unnecessary data when only limited bandwidth is available.

In the case where a data object is transmitted to multiple recipients, the originator is required to transmit the data object multiple times, thereby expending
5 valuable CPU and bandwidth resources in a the concentrated time period of transmission. Also, each recipient is required to receive and store its copy of the transmitted object, thereby using bandwidth and data storage resources at each recipient. If the recipients are receiving diverse variations on a data object, the
10 originator is required to create, store, and transmit the separate variations of the data object.

It is also possible that the volume of data in the transmitted data object is too large for the receiving system to efficiently store, or that numerous accumulated objects are too large for the system to efficiently store, and that it would be desirable to have another system with greater storage capacity hold the object and provide authorized
15 access to the object only as needed.

It is also possible that the originator of the data would want to control the access and use of the object by the receiver, or create an audit trail of the receiver's use of the data object. The originator may wish that various receiving parties have access only to portions of the data object specific to the nature of the receiver. The originator or the
20 receiver may desire to provide authentication relating to a data object without transmitting or storing the object itself.

Mechanisms have existed in the past for providing information about stored data objects. For example, a computer file (one type of data object) under the Disk Operating System (DOS) or the Windows operating system has certain flags associated

with it to indicate information about that file. The flags indicate, for example, whether the file is "read only", whether it has been archived, and whether it can be written. The UNIX operating system utilizes a similar approach of storing flags with files.

However, the flags in all of these systems are rudimentary, and they are not easily
5 extensible. The flags also tend to be stored as part of the object to which they refer and are therefore cumbersome to process without access to the file itself.

Other types of data objects contain elements that refer to other data objects within the context of the containing object. An example of this could be a filename combined with disk sector locations in a storage directory object, such as a file
10 allocation table. Another example could be a hyperlink with addressing information contained within an HTML object. These object elements act as "pointers" to an object, but do not function outside of the context of the containing object. They also require a process to access the referenced object to perform processes or analysis on the referenced object. They do not act as a "surrogate" for the object, allowing processes
15 to perform functions on an object without accessing the object itself.

Unfortunately, current data network transmission and storage systems fail to provide a robust, extensible, and universally compatible means for those users or systems transmitting and receiving data to efficiently use data objects in such as way as to conserve CPU usage, bandwidth usage, and storage through the use of smaller, more
20 efficient objects containing metadata about the data object that can be processed separately from the data object itself. This smaller, more efficient object can contain metadata for use in access control, use auditing, redactability (or the filtering of content in an object), recipient and sender authentication, workflow logic, data integrity, and

timing of data delivery, among others. Consequently, a need exists for a data processing and transfer system that overcomes the foregoing drawbacks.

SUMMARY OF THE INVENTION

5 The present invention provides a data processing system that can allow a user to manipulate a data object through the use of one or more separate data objects comprised of metadata about the original data object. The metadata can represent many attributes of a data object, including, but not limited to, information about the content of the data object and the context of its occurrence. It can also represent information
10 about searchable elements in the data object or information on the use history of the object. Some specific examples of metadata can include object identification, access parameters, authentication requirements and values (for both originator and user), encryption methods and keys, access history, integrity checking values, redaction values (or restricting access to certain elements or properties of the original data
15 object), and token validity expiration period. In an embodiment, the DOT can also contain references (such as hyperlinks) to other locations where other metadata values can be found. Systems holding metadata values can also control access to and log use of those metadata values in other locations according to the contents of the token. A data processing system according to the present invention can combine diverse
20 metadata into a discrete and processable data object called a Data Object Token (DOT) that refers to one or more data objects. The DOT includes metadata that provides information about a data object.

In one embodiment, a data object originator can submit a data object to a token based data processing system, which can optionally store the data object and generate

one or more tokens containing metadata of varying complexity referring to the stored data object. The one or more tokens can then be returned to the originator, or sent directly to token recipients, or stored separately in the data object storage system for later use. This Data Object Token, the DOT, can be used by the originator or by token recipients to interact with the data object that is optionally stored on a system storage device such as a central data depository attached to a network, or in another location specified in the DOT according to the protocol used to locate a data object, including a URL, a storage device identifier, or other data object locating protocols. The DOT user can submit the DOT to the token based data processing system to retrieve, view, alter, delete, or otherwise interact with the data object. The token based data processing system can analyze the metadata values contained in the DOT to determine use of the data object, including allowing access, processing authentication values, performing redaction, transmitting the object or portions thereof, destroying the data object, or otherwise manipulating the data object. It can also store and transmit metadata about data object as a smaller, discrete object that can be managed using native and pre-existent file and transmission processes.

One advantage of a token based data processing system is that it allows users of data objects to have access to and control over data objects without being required to expend local storage or bandwidth resources in the process. A DOT can be of very small size and still refer to a much larger data object in a remote location, while providing many capabilities to the local processes that would otherwise require the presence of the data object on the local system. A DOT can be very efficiently transmitted to multiple users, requiring much less storage and bandwidth than multiple transmissions of the referenced data object.

Another advantage is that the user of the DOT can retrieve, download, view, or otherwise process the referenced data object at their convenience by submitting the DOT to the token based data processing system. Another advantage is that the user can receive a DOT that allows them to decide upon the usefulness of, and securely retrieve, 5 the referenced data object without requiring them to download the data object at time of originator transmission to them.

Furthermore, a token based data processing system can contain the authentication values and methods for both the originator and the DOT user that validate access and other rights to the data object. The token based data processing 10 system can validate the DOT as it contains the originator authentication, as well as validate other parameters of the DOT based on the user authentication, before access, processing, or retrieval of the original data object is allowed.

The DOT can also contain data redaction information tailored to the nature of the DOT user. The DOT can provide the system information as to which portions of 15 the data object the user will be allowed access. This relieves the data object storage system from the requirement of having to store and process redaction information for every potential user of the data object.

A further advantage is that the DOT can contain data object routing information for workflow and rules processing of data objects. The process can be more efficient 20 and less bandwidth intensive if a token representing the workflow data object is passed around instead of the object itself.

Another advantage is that the DOT can append data object usage information to itself whenever it is used, and thereby become a self-contained data object usage audit

trail. This usage information could contain user identity and authentication, and time and type of use of use of both the token and the referenced object.

Yet another advantage is that the data object originator can stipulate in the DOT an expiration or starting time for processing of the referenced data object. This allows
5 an originator to control the timing of various aspects of processing the stored data object, as well as avoid the need to provide a local copy of the data object to the DOT user.

Another advantage to using a DOT as a surrogate for a data object is that it avoids exposure of the data object to potential security and privacy problems inherent
10 in the transmission of data over public networks such as the Internet. For example, even if a DOT is sent to the wrong recipient, or "sniffed" from the public network, the referenced object cannot be processed or retrieved since the unauthorized token possessor will not have the correct authentication to access the object.

Another advantage to using the DOT as a surrogate to the referenced data object
15 is that the DOT can contain searchable elements of the original data object, such as keywords or text, and allow searching of the original data object, or a set of data objects, without requiring access to them. This allows searching to be distributed to the token user machine, thus conserving bandwidth and CPU usage at the object storage system.

20 Another advantage of using a DOT to retrieve an object is that an integrity value, such as a hash digest, of the retrieved data object can be compared to an integrity value in the token user's DOT to determine if the data object was received correctly.

Another advantage of using a DOT approach is that a token generating system can be embedded in an e-mail server, allowing the e-mail system to embed a DOT in

the place of large e-mail attachments. This conserves bandwidth and storage for both the sender and receiver.

Another advantage of DOTs is that they can be used as an electronic payment system due to their high level of security and ability to provide an audit trail. For example, the DOT can represent a portion of a pre-paid monetary amount represented by the referenced data object, such as a bank account. The payment DOT can be submitted to a token processing system that reduces the bank account balance by the amount embedded in the DOT, and then expires the DOT by placing its unique identifier on an expired DOT list. This would allow cashless payment in electronic business transactions, without exposing any sensitive financial information to a public network.

Another advantage of using DOTs is that they decouple the object storage technology from the object access technology, allowing object storage technologies to change without changing the way the stored objects are accessed by the users, and allowing the access objects, or DOTs, to remain valid and functional over long periods of time, even if the storage systems that they access change.

Another advantage of using DOTs is they allow users to maintain representations of libraries of objects on a local system for off-line browsing, management, and processing.

Additional aspects and embodiments of the present invention will become apparent upon a review of the following detailed description and accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

Fig. 1 is a functional block diagram of an embodiment of the present invention.

Fig. 2 depicts the process flow of a system according to the present invention.

Fig. 3 depicts the basic structure of a data object token.

5 Fig. 4 is a notational representation of the structure of one embodiment of a data object token.

Fig. 5 is a notational representation of additional elements in the structure of one embodiment of a data object token.

Fig. 6 is a command table for use in an embodiment of the present invention.

10 Fig. 7 is a flow chart of an embodiment of a token generating system.

Fig. 8 is a flow chart of an embodiment of a token processing system.

Fig. 9 is a flow chart of an embodiment of a token recipient system.

DETAILED DESCRIPTION

15 Fig. 1 depicts a block diagram of an embodiment of a token based data processing system 100 according to the present invention. Token based data processing system 100 can contain data object origination system 110, data object storage system 120, token generating system 130, token transmission system 140, token recipient system 150, token processing system 160, and data object processing system 170.

20 Data object origination system 110 can be any device or system that can be utilized to process information and produce data objects, including but not limited to such things as a personal computer, a mainframe computer, or an embedded computing system connected to a network. Data object origination system 110 can further include numerous processes that utilize data objects. For example, data object origination

system 110 could be utilized to create a data object and store that data object in data object storage system 120. Similarly, data object origination system 110 could also be utilized to download and store information, in the form of one or more data objects, from a data network. In particular, data object origination system 110 can utilize and
5 store various data objects including, but not limited to, database files, electronic documents, e-commerce transactions (such as stock trades or merchandise purchases), electronic records (such as medical records or personal histories), multi-part records from document imaging systems, Web sites, sequences of HTTP data, or digitized video and audio data.

10 Following the creation or reception of one or more data objects in data object origination system 110, the data objects can be stored in data object storage system 120, which can be a local data storage system or a remote data storage system. For example, in the case of a personal computer, a local data object storage system 120 can include a hard drive, or a combination of a floppy disk drive and a floppy disk. Alternatively, a
15 remote version of data object storage system 120 can include storage on a Local Area Network (LAN) or on a Wide Area Network (WAN) using such mechanisms as file servers or digital tape libraries.

In addition to receiving and storing the data objects, data object storage system 120 can expose the data objects to token generating system 130. Token generating
20 system 130 can analyze and process the information about the data object, such as its attributes, the parameters stored with it, and other information regarding its originating system and its recipient system. Token generating system 130 can then generate metadata corresponding to that information, and create a data object called the Data Object Token (DOT) that contains the metadata associated with the data object. Token

generating system 130 can further process the DOT to embed data values that provide or ensure such things as security, privacy, reliability, or efficiency. Numerous means exist for providing these attributes, including encryption, digital signatures, hash digests, other authenticating technologies, compression, or other processing technologies. As one particular example, token generating system 130 can provide authentication, integrity, and non-repudiation information about the data objects using a public key digital signature. Upon successfully verifying the digital signature, the recipient of the digital signature can have the additional assurance of the authenticity of the originator, of the integrity of the data object, and that the data object creation cannot be repudiated.

Once processed, token generating system 130 can further present the DOT to data object storage system 120 and to token transmission system 140 for transmission to one or more token recipient systems 150. Data object storage system 120 or token transmission system 140 can store the DOT or a portion of the DOT for later use, thereby avoiding later reprocessing of the data object.

Token generating system 130 can read the object to be tokenized from data object storage system 120. Token generating system 130 can then determine from a token reference file the types and fields of metadata desired to be embedded in the token. A token reference file can be viewed as a template that contains rules governing the creation or transformation of tokens, including but not limited to encoding of the DOT (e.g. using XML, ASN.1, or comma-separated); required data and metadata fields; organizing structure of metadata fields; and mappings of data types and structures from the source data object to a token. Token generating system 130 can then read the metadata from the object and write it into a file that will become the DOT.

This metadata can include a unique identifier for the object, object location information, object hash digest values, or other metadata fields as required. Token generating system 130 can then determine from the token reference file if it is required to generate metadata from processing the referenced data object's content or metadata fields. Token generating system 130 can then read the content or metadata from the object, process it as required (such as, for example extracting keywords from content text, or reducing images for more efficient searching), and then embed the processed metadata into the token extension fields (see Fig. 3) in the DOT file. The metadata can be written in popular data formats, such as XML or comma delimited fields.

Token generating system 130 can then determine from the token reference file whether token generating system 130 is required to retrieve metadata external to the object, such as the date, a stock price, a digital certificate, encryption keys, other authentication data, or other data as is available to token generating system 130 and required by the token reference file. Token generating system 130 can then embed the externally retrieved data into the appropriate fields in the DOT file. It can also use predetermined time stamp and authentication fields, as well as token extension fields as required. Token generating system 130 can also embed additional metadata into the token file extension fields, such as token validity period, redaction values, or workflow logic to be used when referencing the original data object. Token generating system 130 then can embed its version number in the token file to allow later processing by token processing system 160 and, particularly, to differentiate among different versions of tokens.

Token generating system 130 can then digitally sign the DOT file by methods well known in the art of digital signing, including without limitation, encrypting, with a

private key from a public key infrastructure (PKI) key pair, a combination of a hash digest of the token with a digital certificate, and embedding the digital signature data into the token file. Token generating system 130 can then encrypt the entire token file using well known encryption algorithms, such as triple DES or the RC5 encryption
5 methods.

The DOT file generated by token generating system 130 can then be stored in the data object storage system 120, in the data object generation system 110, or in token generating system 130 itself to await further processing.

Once token generating system 130 creates a DOT, token transmission system
10 140 can read the DOT from token generating system 130 and determine from a token transmission reference file whether or not the DOT is required to be transmitted to a token recipient system 150, or stored for later use. A token transmission reference file, similar to a token reference file, can contain template rules, such as recipient types, required transmission metadata, and required authentication, that would govern the
15 transmission of DOTs. If the DOT is to be transmitted, the token transmission system 140 can embed information on accessors to the data object in the DOT as required. This information can be used by token processing system 160 to determine the authorization and access rights of the submitted DOT. The token transmission system 140 can embed multiple object accessors values to allow use of the DOT by multiple
20 users or systems. The DOT can then be encrypted in the public key of the transmission destination system, and can then be securely and reliably transmitted to one or more token recipient systems 150 using either a standard communications protocol, such as FTP, e-mail attachment, or HTTP, or by using a proprietary communications protocol.

A token recipient system 150 system can include a system capable of receiving, storing, or processing a DOT. The token transmission system 140 can determine addressing and routing information for the destination token recipient systems 150 from a token transmission reference file, or from the token reference file contained in token
5 generating system 130. In one embodiment, data object origination system 110 can also be a token recipient system. In an embodiment, token transmission system 140 can further verify and log reception of the DOT by one or more token recipient systems 150.

Token recipient system 150 can receive, store, and optionally process or
10 otherwise condition a DOT. Processing on a DOT could include searching on metadata or content from the object embedded in the DOT. This searchable metadata or content could include, for example, text, keywords, or images from the original data object. Other types of processing that could be performed on a DOT by the token recipient system include embedding recipient reception information, such as time stamp and
15 digital signature, appending additional accessors to the referenced object if the recipient system is so authorized, or forwarding the DOT to another token recipient system. The optional processing performed by token recipient system 150 can include other types of analysis and processing of the information contained in the DOT, including its metadata. Since token recipient system 150 can perform its analysis and processing on
20 the metadata contained within the DOT (which is acting as a surrogate for the data object itself), no resources within data object storage system 120 would need to be used.

For example, in one embodiment, token recipient system 150 can contain an e-mail computer application that receives DOTs associated with e-mail messages. In this

case, each message could be a data object. The metadata contained within a particular DOT that is associated with a specific e-mail message can contain information about the e-mail message, including but not limited to such things as the date, time, subject, sender, and, in particular, any attachments associated with the e-mail message. The

5 DOTs can be used by a token recipient system to search for a particular e-mail message or for a group of messages. However, this search could be done solely within the token recipient system using metadata embedded in the DOTs associated with the data objects (i.e. the e-mail messages), not on the messages themselves, which can be stored on a remote data object storage system 120. Also, using this approach, any attachments to

10 the e-mail message would only need to be stored in the data object storage system, not sent to a token recipient system nor stored at the data object origination system.

Furthermore, in one particular embodiment related to e-mail, a token generating system can be embedded in an e-mail server to allow automated storage and tokenization of e-mail, including, without limitation, any of its components such as

15 header, body, or attachments. Token generating system 130 embedded in the e-mail server can read the e-mail objects controlled by the server, generate a DOT for the e-mail object, replace the original e-mail object with the DOT, and store the e-mail object in a more efficient data object storage system location, such as a digital tape library. The e-mail user can then use the DOT from within the e-mail client application to

20 access or process the e-mail data object. Searching for e-mail content within a group of e-mails can be accomplished by searching the search metadata embedded in the DOTs in the e-mail client application, thereby avoiding the CPU and bandwidth costs associated with searching the group of e-mails on the e-mail server or at the data object storage system. This process of substituting DOTs for data objects such as e-mails can

be applied to many types of data objects, including without limitation, scanned or imaged documents, electronic documents, database tables, electronic spreadsheets, web sites, and other types of data objects specified by the user of a token based data object management system.

5 Upon the completion of a search and a determination that one or more specific e-mail messages, or other type of data objects, need retrieval or processing, token recipient system 150 can submit one or more DOTs to token processing system 160 which can then cause data object storage system 120 to be accessed and the referenced data objects to be processed according to the instructions in the DOT, or transmitted to
10 a data object processing system 170 as requested in the submitted DOT.

 After token recipient system 150 submits a DOT to token processing system 160 to retrieve a data object from data object storage system 120, token processing system 160 can interpret the metadata contained in the DOT. Using the metadata, token processing system 160 can submit processing commands to data object
15 processing system 170 according to the metadata contained in the DOT. These commands could include, for example, transmit, redact, alter, or destroy a data object as shown in Fig. 6.

 Alternatively, a token recipient system can submit a DOT to a token processing system to retrieve a set of DOTs associated with the submitted DOT. This referenced
20 set of DOTs can reference data objects or other DOTs, or a combination thereof, as required by the system.

 Data object processing system 170 can interact with the data object stored in data object storage system 120 according to the commands submitted by token processing system 160. For example and without limitation, in response to a particular

command, data object processing system 170 can transmit, redact, alter, or destroy the data object stored on the data object storage system, according to the authorizations embedded in the submitted DOT, and as interpreted by the token processing system 160, which can pass the commands to the data object storage system 120.

5 In another embodiment where data object storage system 120 contains data object processing system 170, data object storage system 120 can process the referenced data object according to the instructions embedded in the DOT. For example, if data object storage system 120 is instructed to transmit the object, it can place the object into a data transmission process as is commonly known to the art, or it
10 can encrypt the object using an encryption value submitted with the token, such as the destination system public key, or it can transmit the encrypted object to a system different from the system that submitted the token, or it can transmit the object to multiple systems. Similarly, data object storage system 120 can redact the stored object by recomposing the object prior to transmission, according to redaction rules embedded
15 in the token. An example without limitation of this can be the removal of personal information from a data object for transmission of the data object from within the European Community to a destination outside of the European Community, thereby complying with the European Privacy Act, or other regulations that require redaction of information in transmission or storage.

20 Data object storage system 120 can additionally alter the data object by changing its retention information, its authorized accessor list, its storage format or location, or even its content if required by the system. Data object storage system 120 can even destroy the data object according to the processing instructions embedded in the submitted token.

Fig. 2 depicts a process flow diagram of an embodiment of the present invention. In data object origination system 210, a data object can be created. Once created, that data object can be transmitted to data object storage system 220. Once received, data object storage system 220 can process the data object so as to enhance the security and reliability of the data object storage process by, for example, performing encryption, compression, hashing, or mirroring, or by digitally signing the data object. For example, the well known DES encryption algorithm can be used to encrypt the data object. In addition, the well known RSA public key digital signature algorithm can be used to digitally sign the data object in data object storage system 220. Following the processing performed on the data object by data object storage system 220, the data object can be stored. The data object storage system can then place the processed object where it will be read by token generating system 230.

Token generating system 230 can generate tokens in standard data formats. For example, token generating system 230 can generate XML-based tokens based on the content of the data objects. To do so, token generating system 230 first reads the stored data object in data object storage system 220. It then analyzes the data object and the associated token reference files. The token reference files can contain instructions and information that can allow the token generating system to determine how the DOT will be constructed from the data object. Once analyzed, token generating system 230 can generate the necessary XML tags to be placed in the DOT for the particular system. In other embodiments, a token generating system can also, for example, generate fixed length fields, comma delimited fields, and other data formats that are commonly used by computer applications. In general, a token generating system can produce any type of DOT that may be needed for a particular data processing system.

In addition, token generating system 230 can encrypt, compress, hash and otherwise process one or more DOTs so as to enhance the security and reliability of the DOTs. Similar to the processing that can be done on the data object itself, the well known DES encryption algorithm can be used to encrypt the DOT. In addition, the well known RSA public key digital signature algorithm can be used to digitally sign the DOT in token generating system 230.

Furthermore, in an embodiment of the present invention, token generating system 230 can embed in the DOT time relevant attributes of the DOT, the data object, or the token recipient system's 250 ability to manipulate it. These time relevant attributes could include, for example, a time stamp of the DOT generation time, expiration of the DOT, the data object, or authentication metadata as determined by the contents of the token reference file. These time relevant attributes could also include "no access until" information that could not allow the token submitting entity to manipulate the data object until a certain time or event.

In an embodiment, once token generating system 230 has generated a DOT, token transmission system 240 can sense the DOT and can cause token generating system 230 to pass the DOT to token transmission system 240. To do so, token transmission system 240 can poll the system holding the DOT and retrieve the DOT in a data "pull" operation once it is available. Token transmission system 240 can then determine the destination for the DOT, followed by determining the methods to be used for the delivery of the DOT, according to mechanisms such as the transmission reference file in the token transmission system, or from the token reference file in the token generating system, or from a workflow logic file. Token transmission system 240 can then transmit DOTs using any well known data transmission technologies,

including, without limitation, the FTP protocol, the HTTP protocol, the SSL protocol, a virtual private network (VPN), or Secure Shell tunneling in the Unix operating system. In another embodiment, a token transmission system can send the DOT as an attachment to an e-mail message. Alternatively, a token transmission system can send
5 a notification to a token recipient system of a token pending transmission to which token recipient system 250 must respond to release or access the token, and a token transmission system 240 can then issue a receipt describing the token delivery event.

After receipt of a DOT from a token transmission system, another token recipient system, or any other means of transmitting a token, such as in an e-mail as an
10 attachment, token recipient system 250 can authenticate the DOT using any authentication techniques that may have been applied by token generating system 230. Once authenticated, token recipient system 250 can store the DOT and subsequently submit the DOT to token processing system 260 for data object manipulation.

In addition to submitting the DOT for data object manipulation, a token
15 recipient system can also add new metadata to the received DOT, or alter the DOT according to the token recipient system's authority to do so as determined by the authorization metadata embedded in the DOT, and optionally transmit that updated DOT to another token recipient system or to the token processing system 260. This can allow metadata within a DOT to accumulate without having to access the associated
20 data object from storage. A data processing system utilizing this type of DOT approach could undergo a change of the native platform and data formats of data object storage system 220 without impacting the capabilities of pre-existing DOTs used to manage and process the data objects within the system, provided the new native platform and data formats remained compatible with the required versions of the DOT structure.

In one embodiment, workflow logic can be embedded within the token or can be captured in an external file. Workflow logic refers to workflow processing rules stored and interpreted both inside and outside of a DOT, and that can affect both the internal operations of the token based data processing system and the overall workflow distribution patterns of tokens and data objects within an organization.

For example, workflow logic could be used by an organization to manage the workflow associated with a particular project. Workflow routing instructions could be embedded in the DOT if it is advantageous for the logic to be contained in the DOT, or the DOT could reference an external workflow logic file that could be dynamically updated. The DOT can reference an object or a set of objects intended to be the subject of a workflow process. This can allow the workflow system to distribute metadata about workflow-related data objects without expending the CPU and bandwidth to distribute the actual data object(s) themselves. The token recipient system 250 itself can execute the workflow logic embedded in the DOT, such as forwarding a DOT automatically to another token recipient system if the referenced data object is not accessed within a certain period of time, or other similar workflow logic.

Upon receiving the DOT from token recipient system 250, token processing system 260 can perform on the DOT any type of processing that might be necessary. For example, the DOT can be decrypted, uncompressed, or otherwise processed to restore it to its original form in response to security and reliability processing performed by the token generating system. The type of processing required can be determined by token processing system 260 by reading the metadata embedded in the DOT, such as the version number, or other metadata indicating how the DOT has been previously processed.

As another example, token processing system 260 can authenticate the entity submitting the DOT using digital certificates, recipient supplied passwords, biometric measurements, or any combination thereof, according to values contained in the token. Alternatively, the entity originating the DOT can be authenticated using digital
5 certificates, recipient supplied passwords, biometric measurements, or any combination thereof, according to values contained in the token.

In yet another embodiment, token processing system 260 can create and append metadata to the DOT that can describe the use of the DOT and the data object. This metadata can be used create a self-contained audit trail within the token. For example,
10 this self-contained audit trail can be comprised of use metadata and authentication values embedded into the token during a workflow execution by the associated token recipient systems.

Token processing system 260 can, in another embodiment, process metadata that describes the token submitting entity's access rights to the referenced data object.
15 Those access rights can be used by data object processing system 270 to control access to all or portions of the data object. Metadata can also be used to describe the time attributes of a DOT, the referenced data object or the authenticating metadata. Token processing system 260 can interpret this time-based metadata to determine proper usage of the token and the associated data object.

20 The results of the token processing done by token processing system 260 can be passed as commands to data object processing system 270. Upon receiving one or more of these commands, data object processing system 270 can authenticate the command using any well known authentication technique already described herein. Upon successful authentication, data object processing system 270 can determine what

action needs to be taken to respond to the command. For example, destroying or deleting the referenced object from the object store, or appending the audit trail information embedded in the submitted DOT are actions that can be taken in response to a command.

5 One frequent action by data object processing system 270 can be the accessing of one or more data objects in data object storage system 220. If data object processing system 270 determines that the proper conditions have been met, including, for example, appropriate authentication information, one or more data objects can then be accessed in data object storage system 220. Once accessed, the requested actions
10 can be performed on the accessed data objects by data object processing system 270 and those actions can also be logged by data object processing system 270. Finally, any results of the processing performed by data object processing system 270 can be passed back to token processing system 260. In another embodiment, data object processing system 270 can transmit one or more data objects to the token submitting
15 entity. Prior to doing so, data object processing system 270 can optionally encrypt, compress, hash, sign, or otherwise process the data object to enhance security and reliability in transmission and delivery.

Figure 3 depicts a DOT 300 in accordance with the present invention. The basic DOT structure consists of a version field 304, token information 308, object
20 accessors 328, token and referenced object use information 352, and optional token extensions 368.

Version field 304 indicates the version of the data object token protocol to which DOT 300 conforms. The inclusion of version field 304 permits a token processing system to properly parse the DOT.

Token information 308 contains the fields necessary for a token processing system to assess the validity of the DOT and the data object to which the DOT is associated. Data object identifier 312 can contain a unique value or character string that identifies the data object to which it refers. Validity information 316 allows a system to make the determination of whether the DOT is valid. For example, validity information 316 can contain date and time information, such that a validity period can be established. Alternatively, validity information 316 can contain a hash digest of the object and the DOT that can be used for purposes of determining data integrity of a received object or DOT.

Data object and DOT authentication information 320 can contain any type of data that can be used to authenticate the data object or the DOT itself. In an embodiment, data object authentication information can contain a public key from a PKI key pair. In another embodiment, data object authentication information 320 can contain a digital signature. The authentication information can be used to authenticate the system creating the referenced data object, the system generating the DOT, or the system submitting the DOT to a token processing system or token recipient system. Timestamp 324 can provide information to an entity about the time of creation of the token. A data object originator could use the timestamp, for example, to prove that a data object representing a transaction was created at a specific time on a specific date.

The next main component of the DOT is object accessor 328. This can provide information about the entity or entities permitted to access and use the data object referenced by the DOT. In an embodiment, there may be multiple instantiations of object accessor 328 for each permitted accessor of the data object. Each object accessor 328 can contain information about the accessor 332, access rights 336,

authentication requirements 340, accessor authentication information 344, and timestamp 348. In an embodiment, the object accessor metadata can contain a list of workflow destinations upon which workflow distribution processing can be based.

5 Accessor 332 can contain information that identifies the entity being given rights to access the data object. This entity identifier could identify a system, a process, a class of processes, a specific user, a group of users, a reference to a dynamically alterable accessor list, or the general public. For example, accessor 332 can contain a name, an e-mail address, and a unique identifier. In another embodiment, accessor 332 can contain a digital certificate.

10 Access rights 336 can provide information to the token processing system regarding the rights that the accessor identified by accessor 332 has to the data object to which the DOT refers. Access rights 336 can include, for example, general rights, such as those used to access, download, modify, or destroy a data object. In addition, access rights 336 can include other, more specific rights that are unique to the particular data processing system. For example, a DOT used in a stock trading system might contain 15 rights particular to a stock trader, such as buy unrestricted, buy limited, sell unrestricted, or sell limited. In another example, access rights can also stipulate authorization of the token recipient system to alter the metadata in the token, or the authorization to append audit trail metadata, as required by the system. Authentication requirements 340 provide information about the type of authentication required to be 20 used by the token user to prove that it is the entity identified by accessor 332.

Numerous, well understood, technologies exist to provide authentication in several different formats. For example, a particular data processing system may require something to be known by the token user, such as a password or personal identification

number (PIN). Alternatively, a data processing system may require the user to have an object, such as a smart card or a hardware "dongle". Finally, a data processing system might require physical information from the token user (i.e. a biometric), such as a fingerprint, a retinal scan, or a voice print.

5 Accessor authentication information 344 provides information about the entity that created the object accessor entry. This can be used to audit the granting of rights to access the referenced data object.

 Timestamp 348 can provide information to an entity about the time of creation of the object accessor 328 metadata. A token processing system could use the
10 timestamp, for example, to prove that the authentication information 344 was historically valid at the time of its creation. For example, if a digital certificate is used as authentication information 344, time stamp 348 can be used at a later time to determine if the digital certificate had expired or been revoked at the time of object accessor 328 metadata creation.

15 The next component of DOT 300 in an embodiment of the present invention, token use information 352, contains information about the use of the DOT and use of the data object to which the DOT refers. Token use information 352 can include information on the type of access requested 356 by the token user. Once authenticated using access authentication information 360, a token processing system can utilize
20 token use information 352 to determine how it should process the object referenced by the DOT. This could include uploading the object to the token submitter, changing data in the object, or destroying the object.

 One or more token extensions 368 allow the DOT to provide additional information beyond the standard DOT including, but not limited to, such things as

redaction instructions and audit trail data related to the associated data object. The format of each token extension 368 can consist of extension type 372, extension data 376, and extension authentication information 380. Within token extension 368, extensible metadata can be stored that can then be interpreted by the token recipient 5 system or the token processing system. Some examples of the use of token extension 368 include DOT expiration values for a particular system, searchable content metadata values, workflow logic, additional (system specific) access rights, externally retrieved metadata, and any other fields that may be added by a token recipient system.

Fig. 4 and Fig. 5 provide a description of one embodiment of a DOT that can be 10 associated with a Verified Data Object (VDO), as implemented by iWitness, Inc. of Boulder, Colorado in its electronic records management products. The descriptions in Fig. 4 and Fig. 5 use the well known Abstract Syntax Notation 1 (ASN.1) that is defined in "CCITT Recommendation X.208: Specification of Abstract Syntax Notation One (ASN.1)", published in 1988. The ASN.1 descriptions in Fig. 4 and Fig. 5 depict a 15 Verified Data Object Token (VDOT).

Fig. 4 depicts the basic VDOT structure 410, which includes the two main components of tokenInfo and accessors. The tokenInfo component can provide information specific to the VDO, while the accessors component can provide information about the entities that may access the VDO. As shown in 420, the 20 tokenInfo component can include globally unique object identifier vdo_id, the interval of time during which the token is valid, a digital signature and certificate identifying the creation event, and a timestamp. The vdo_id element can be used to uniquely identify the particular VDO to which the VDOT refers and can contain a randomly generated ASCII character string that had been verified as being unique. In a different

embodiment, the vdo_id could be a consecutive number applied to a particular VDO amongst a large set of other VDOs. The validFrom and validTo components together provide the validity period for the VDOT user to access the particular VDO. In an embodiment, if the UniversalTime value within the data processing system utilizing the VDO is less than the validFrom time or greater than the validTo time, the entity submitting the token in an attempt to access the associated VDO would be denied that access.

The signerInfo component provides the digital signature and information about the signer who applied the digital signature used to verify the VDO. In an embodiment, signerInfo can be the signer information defined in "PKCS #7: Cryptographic Message Syntax Standard: An RSA Laboratories Technical Note", Version 1.5, Revised November 1, 1993, which specifies, amongst other things, the signer's digital certificate, the algorithm used to sign the VDO, and the digital signature itself.

The accessors component within VDOT basic structure 410 specifies a collection of entities along with their access rights and authentication requirements for the specified VDO. As shown in accessor information 430, those entities can be identified in a table consisting of a set of DataObjectTokenAccessor elements, wherein each DataObjectTokenAccessor element can contain a PKCS #6 certificate, the access rights needed to access the VDO, the authentication requirements for the particular VDO, the PKCS #7 signer information, the PKCS #6 certificate of the signer of the VDOT, and a timestamp.

Access rights listing 440 for a particular VDOT can provide information on how a particular entity may use the VDOT in an embodiment of the invention. Only entities identified by the distinguished name and serial number from their enclosed PKCS #6

certificate can make use of the VDOT, including, for example, the ability to retrieve the corresponding VDO. Those entities are limited to their corresponding access rights, which are signed by the entity responsible for their assignment.

Authentication requirements 450 for a particular VDOT provides information
5 on how a particular entity must authenticate itself prior to its use of the VDOT. Any use of the token, whether for VDO retrieval or communication to another entity, may require the using entity to satisfy one or more authentication requirements, such as knowledge of a password, a physical match of biometric data, or an external key. Additionally, each entry in the accessor table can be signed by its creator.

10 Fig. 5 depicts an ASN.1 description of the token use information 510 element and the token extension 520 element in accordance with an embodiment of the present invention. Token use information 510 can contain request information that indicates how a VDOT user would like to use the corresponding VDO and authentication information that allows the token processing system to determine the validity of the
15 request. Token use information 510 can be included upon initial generation of the VDOT or can be appended at some later time after VDOT creation.

In an embodiment of the invention, one or more instantiations of the optional token extension 520 element can be included in a VDOT. In a VDOT, extType can specify the extension type of this particular extension.

20 Fig. 6 shows a command table implemented in one embodiment of the present invention that contains commands for performing various operations on a data object. A data object processing system can use the information shown in 610 to execute the commands on the specified data object. Transmit object command 620 can result in a data object being transmitted to the destination specified by the destination argument,

and can include information on authorization, transmission time, recipient requirements, and transport mechanism. Destroy object command 630 can result in the destruction of a data object, and can include information on authorization, destruction time, and disposition rules. Copy object command 640 can result in the copying of a data object from one location to the location specified by the destination argument. Alter object command 650 can result in one or more modifications to the data object, and can include information on the module to perform the operation, a reference to the data object, the actual alteration to be performed on the data object, and any necessary parameters. Redact object command 660 can result in certain information within a data object being removed while the rest of the data remains, and can include the module to perform the redaction, a reference to the data object, the redaction to be performed on the data object, and any necessary parameters. Finally, object encryption command 670 can result in the encryption or digital signing of a data object, and can include the operation to be performed on the data object, the method or algorithm to be used, the key to be used to perform the cryptographic operation, and whether to just perform a hash on the data object for data integrity purposes.

Figure 7 depicts a flow chart of the processing that can occur in a token generating system utilized for workflow processing. In step 704, the token generating system can receive a DOT generation request, which contains request attributes specifying the data object to be analyzed, the token reference file that defines the structure and required contents of the generated DOT, and possibly other control data. In order to ensure the security of the token generating system, and thus the security of the DOT and its future use in other systems, the DOT generation request is validated in step 712 through one or more of the various authentication means available including,

but not limited to, public key digital signatures. Once the DOT generation request is validated, the data object to which it refers is analyzed in step 716 to determine the viability of the request. In particular, as shown in the substeps in 720, the DOT generation request is checked against workflow logic (if any), the data object is then
5 checked against workflow logic carried in the DOT generation request (if any), and finally, the data object is assessed to determine if it contains the necessary data/metadata to fulfill the request according to the token reference file. If the request cannot be fulfilled, a result indicating this can be returned in step 736.

If the authentication and subsequent analysis succeed, the token generation
10 system can proceed to generate the requested DOT in step 724. As shown in substeps 728, all requested metadata that is derived directly from the data object can first be retrieved and attached. Then, all requested metadata that is derived external to the data object (if any) can be attached, possibly including workflow logic defined in the token request file. Finally, any requested DOT access rights can be constructed and attached.
15 The newly formed DOT can then be digitally signed in step 732 by the token generation system, and possibly encrypted and enveloped following the Cryptographic Message Syntax as defined in PKCS #7. As shown in step 736, the DOT can then be returned synchronously to the originator of the DOT request and/or asynchronously transmitted through a token transmission system (as shown in the substeps in 740).
20 Control could then be returned to the token transmission system in step 744.

Figure 8 depicts a flow chart of the processing that can occur in a token processing system utilized for workflow processing. In step 804, the token recipient system can send a processing request. In step 808, the token processing system can receive a processing request, which can contain, as shown in 812, a request DOT and

request attributes specifying one or more operations to be performed on the data object that the DOT references. Before processing, the token processing system can validate the public-key digital signatures on the DOT in step 816, to ensure the authenticity of the DOT and the authenticity of the requestor. Once validated, the token recipient
5 system can open the cryptographic envelope of the DOT in step 816 (if it is encrypted) with the private-key that corresponds to the public-key used in securing the DOT for transmission. This ensures that the token can only be used by the intended token processing system.

Once validated and decrypted, the token processing system can check the
10 token-embedded or external file workflow logic in step 820 and in the substeps shown in 824. The token processing system can then interpret the processing request according to the workflow logic of the token in step 828 in order to cause the data object processing system in 836 to perform the requested operations in step 832, such as fetch data/metadata, update metadata, destroy object, and transmit object/token, all
15 of which can be performed on the data object within the data object processing system.

Results of the data object operation can be validated by the token processing system in step 840 to ensure their authenticity and conformance to the requirements of the request. The results can then be returned in step 844, and in particular, as shown in substeps 848, the results can be returned synchronously to the originator of the
20 processing request and/or asynchronously transmitted as an updated token through a token transmission system. Once the results have been returned, control can return to the token transmission system in step 852.

Figure 9 depicts a flow chart of the processing that can occur in a token recipient system utilized for workflow processing. The token transmission system can

send a DOT in step 904 and token recipient system can receive a DOT from the token transmission system in step 908. Before processing, the token recipient system can validate the public-key digital signatures on the DOT in step 912, to ensure the authenticity of the DOT and the authenticity of the sender. Once validated, the token recipient system can open the cryptographic envelope of the token in step 912 (if it is encrypted) with the private-key that corresponds to the public-key used in securing the token for transmission. This ensures that the token can only be used by the intended token recipient system.

Once validated and decrypted, the token recipient system checks both the token-embedded and external file workflow logic for conflicts in steps 916 and the substeps in 920. If non-critical conflicts are found, resolution may require priority-based assessment or possibly user intervention. The token recipient system can then process the token according to the workflow logic in step 924. As shown in the substeps in 928, processing can include presentation to the user, storage for archival or for later use. At some point, the user of the token recipient system or the system itself can use the received (and possibly stored) token to request one or more operations on the data object to which it refers, as shown in step 932. This processing can be done through the token processing system as shown in step 936.

While the invention has been described in detail, including references to specific embodiments, it will be apparent to one skilled in the art that changes and modifications can be made to the invention without departing from the spirit and scope thereof. Thus, it is intended that the present invention cover the modifications and variations of this invention provided they come within the scope of the appended claims and their equivalents.

What is claimed is:

1. A method of managing one or more data objects within a data processing system using one or more data object tokens, the method comprising the steps of:
 - generating one or more data objects within a data object origination system;
 - 5 storing said data objects in a data object storage system;
 - analyzing said data objects to determine information to be placed in one or more data object tokens;
 - generating said one or more data object tokens;
 - transmitting said one or more data object tokens to one or more token recipient
 - 10 systems;
 - processing said one or more data object tokens; and
 - processing said one or more data objects based upon said data object tokens.
2. A method as in claim 1, wherein said storing step further comprises storing said
- 15 data objects in a local storage system.
3. A method as in claim 1, wherein said storing step further comprises storing said data objects in a remote storage system over a network.
- 20 4. A method for generating one or more data object tokens associated with a data object, the method comprising:
 - receiving a token generation request;
 - validating said token generation request;

- analyzing said data object to determine information to be placed in said one or more data object tokens;
- populating said data object token with metadata;
- creating validation information related to said data object token; and
- 5 returning a completed data object token.
5. A method as in claim 4, wherein said validating step further comprises the step of verifying a digital signature using a public key.
- 10 6. A method as in claim 4, wherein said analyzing step further comprises the step of determining information to be redacted in said data object.
7. A method as in claim 4, wherein said analyzing step further comprises the step of determining information to be altered in said data object.
- 15 8. A method as in claim 4, wherein said populating step further comprises the step of appending one or more optional token extensions.
9. A method for processing one or more data object tokens associated with a data
- 20 object, the method comprising:
- receiving a token processing request;
- validating said token processing request;
- interpreting said token processing request to determine the operations to be performed on one or more data objects;

commanding said operations to be performed on said one or more data objects
by a data object processing system;
validating the results of said operations; and
returning the resulting data object.

5

10. A method for limiting access by a user to data objects in a data processing
system, the method comprising:

storing said data object securely within said data processing system;

creating a data object token corresponding to said data object, said data object

10 containing authentication requirements;

distributing said data object token;

receiving an access request from a data object token user, said request

containing data object token user authentication information and access authentication
information;

15 authenticating said access request;

processing the data object to secure it for use by the data object token user; and

transmitting said processed data object to said data object token user.

11. A method as in claim 10, wherein said storing step further comprises

20 performing local encryption on said data object.

12. A method as in claim 10, wherein said storing step further comprises locating
said data object behind a firewall.

13. A method as in claim 10, wherein said first validation step further comprises verification of a first public key digital signature, and wherein said second validation step further comprises verification of a second public key digital signature.
- 5 14. A method for validating the use of a data object using a data object token in a token based data processing system, the method comprising:
- performing a first validation on data object originator authentication information contained within said data object token; and
 - performing a second validation on data object token user authentication
- 10 information contained within said data object token.
15. A method as in claim 14, wherein said first validation step further comprises verification of a first public key digital signature, and wherein said second validation step further comprises verification of a second public key digital signature.
- 15
16. A method of retrieving a data object in a token-based data processing system, the method comprising the steps of:
- receiving a data object token from a token generating system;
 - storing said data object token;
 - 20 appending data object token user authentication information;
 - submitting said data object token to a token processing system; and
 - receiving said data object.

17. A method as in claim 16, wherein said data object token user information comprises a public key digital signature.
18. A method for distributing analysis and processing activities related to one or
5 more data objects stored in a data object storage system within a token based data processing system, comprising the steps of:
- receiving one or more data object tokens in a token recipient system;
 - storing said data object tokens; and
 - processing said data object tokens within said token recipient system.
- 10
19. A method as in claim 18, wherein said processing step further comprises the steps of:
- extracting metadata associated with said one or more data objects from said one
or more data object tokens;
 - 15 performing an analysis on said metadata; and
 - accessing one or more data objects within said data object storage system based
upon said analysis.
20. A method as in claim 18, wherein:
- 20 said processing step further comprises a search through data object tokens containing electronic mail header information based on a set of search criteria; and
said accessing step further comprises accessing e-mail messages meeting said search criteria.

21. A token based data processing system that manages one or more data objects within a data processing system using one or more data object tokens, comprising:
- a data object origination system for generating one or more data objects;
 - a data object storage system for storing said one or more data objects;
 - 5 a token generating system for generating one or more data object tokens;
 - a token transmission system for transmitting said one or more data object tokens to one or more token recipient systems;
 - a token processing system for processing said one or more data object tokens;
- and
- 10 a data object processing system for processing said one or more data objects based upon the processing of said one or more data object tokens.
22. A token based data processing system as in claim 21, wherein said data object storage system is contained within said data object origination system.
- 15 23. A token based data processing system as in claim 21, wherein said data object storage system is located over a network remotely from said data object origination system.
- 20 24. A token based data processing system as in claim 21, wherein said data objects further comprise scanned or imaged documents.
25. A token generating system for generating one or more data object tokens that refer to one or more data objects, comprising:

- a means for receiving a token generation request;
- a means for validating said token generation request;
- a means for analyzing said data object to determine information to be placed in said one or more data object tokens;
- 5 a means for populating said data object token with metadata;
- a means for creating validation information related to said data object token;
- and
- a means for returning a completed data object token.

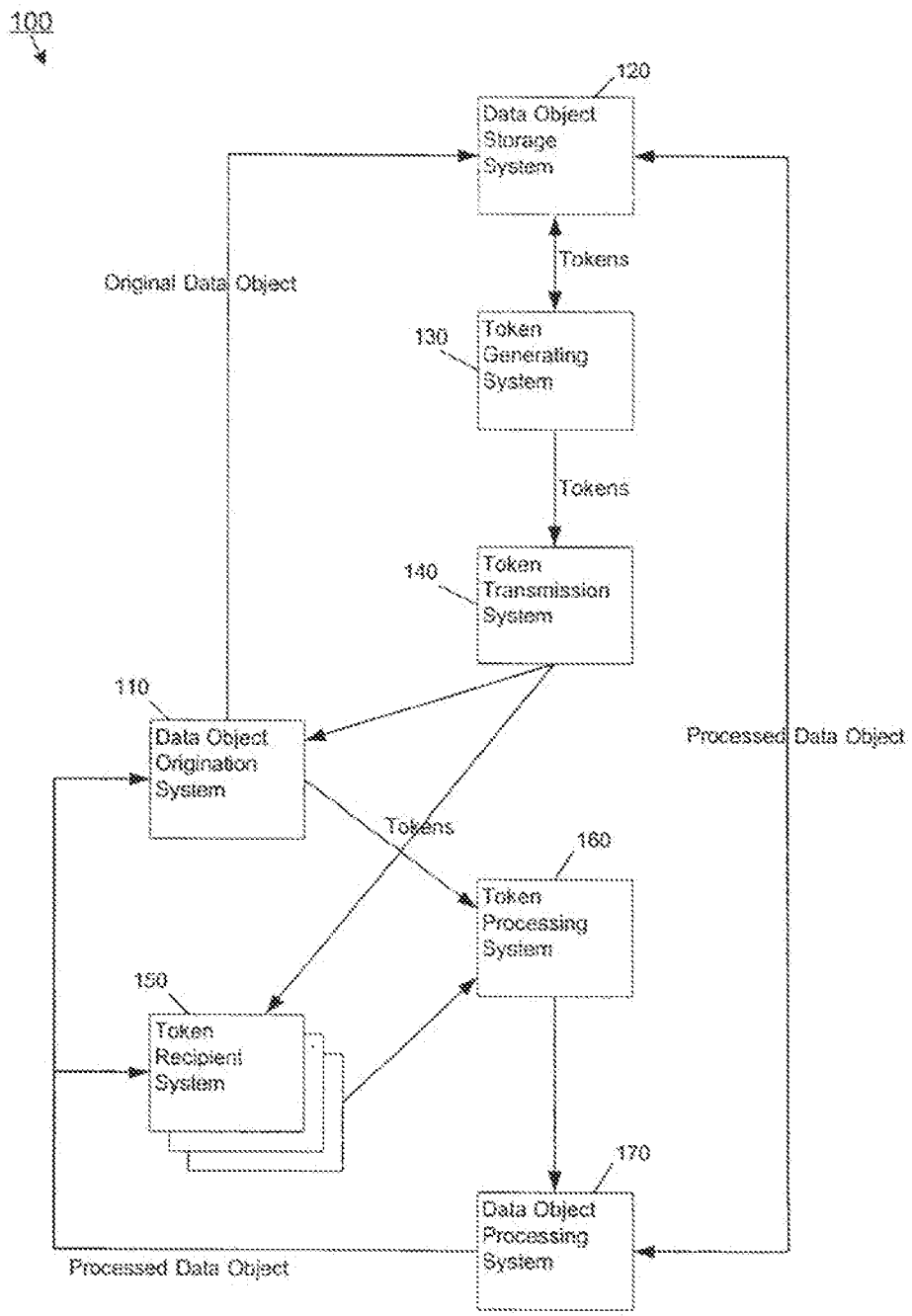


Figure 1

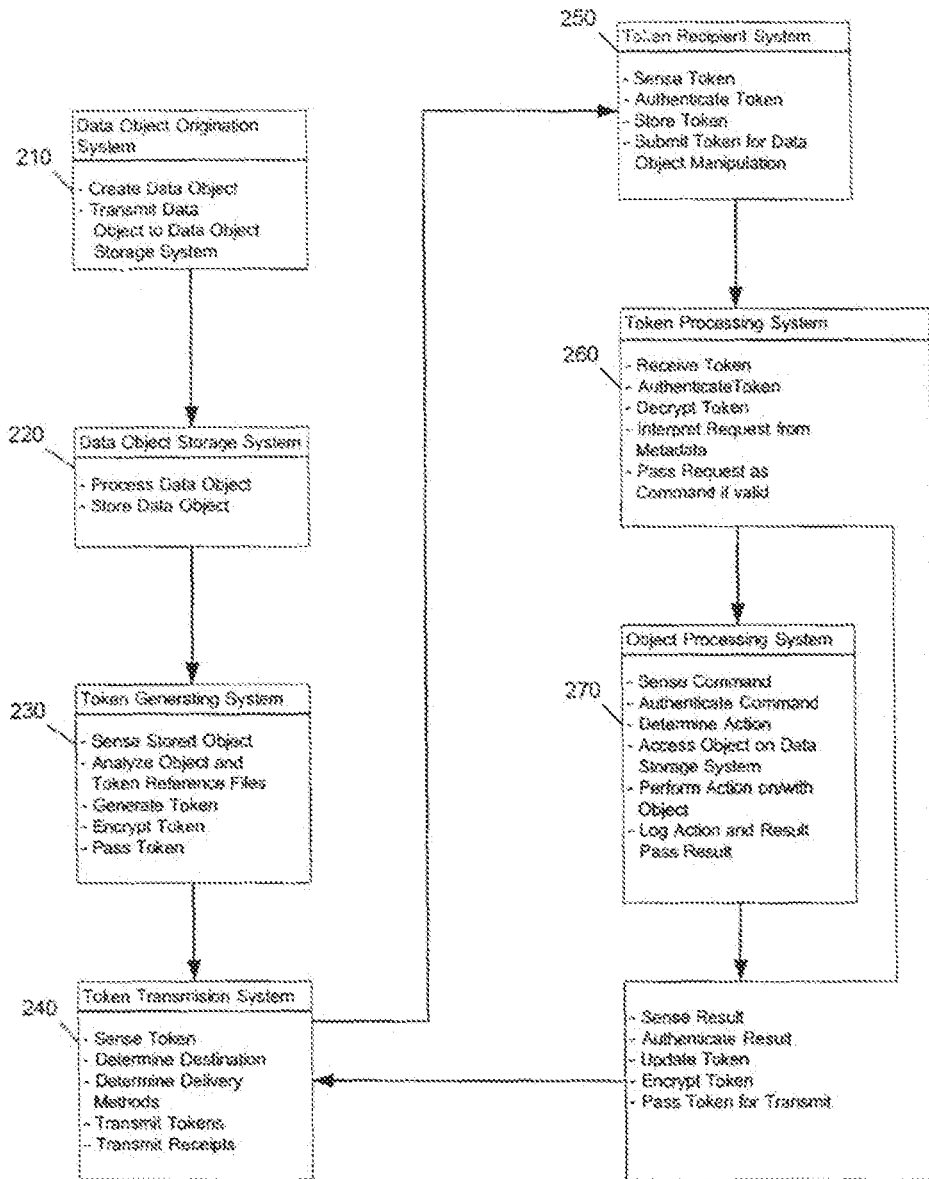


Figure 2

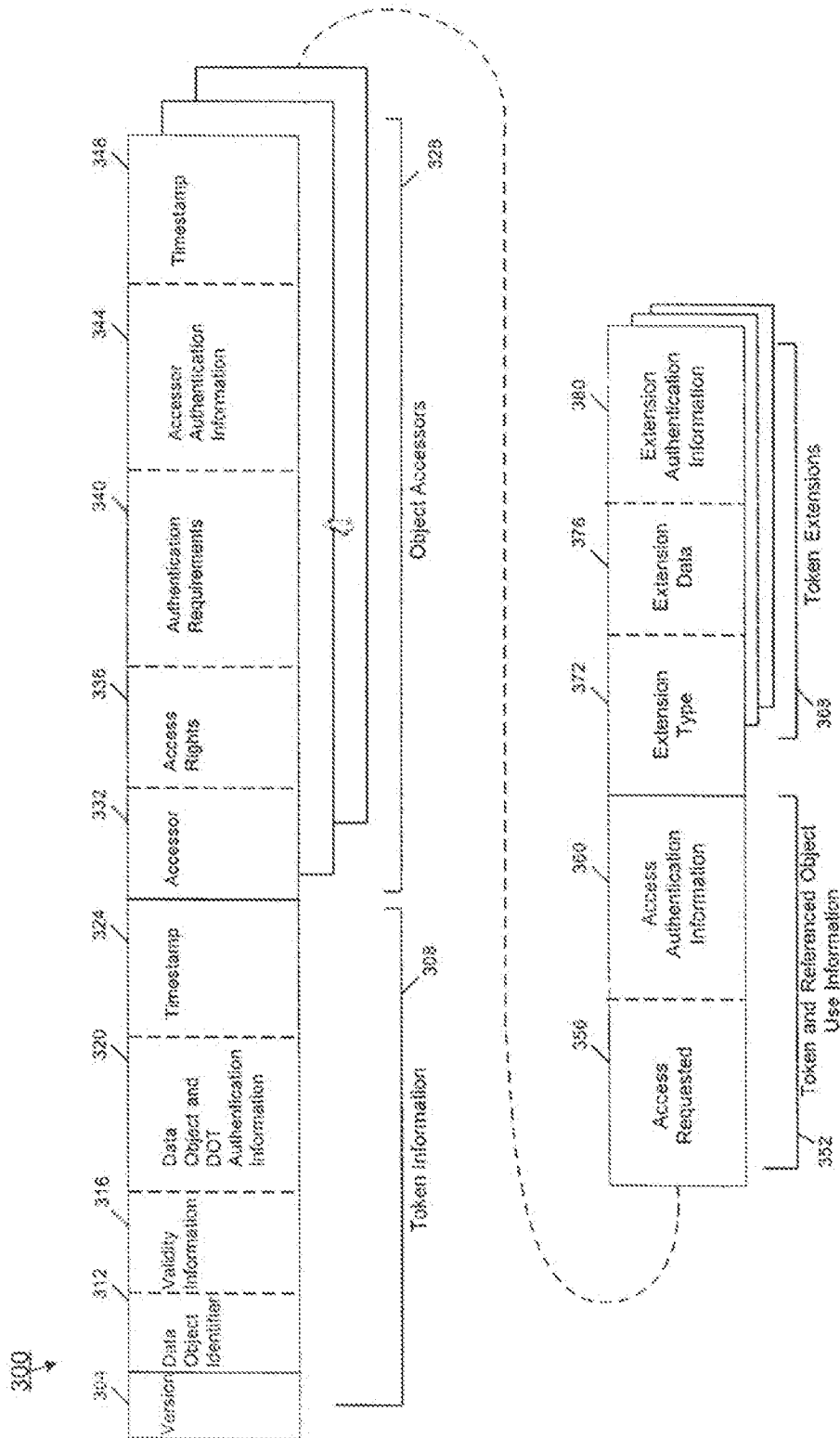


Figure 3

```

DataObjectToken ::= SEQUENCE {
  version OCTET STRING,
  tokenInfo DataObjectTokenInfo,
  accessors DataObjectTokenAccessors,
  userInfo DataObjectTokenUserInfo OPTIONAL,
  subjectExt DataObjectTokenExtensions OPTIONAL
}

```

410

```

DataObjectTokenInfo ::= SEQUENCE {
  vdo_id OCTET STRING,
  validator UniversalName,
  validTo UniversalTime,
  extAttributes InfoExtAttributes,
  signerInfo PKCS_7_SignerInfo,
  attributes // sig on previous
  signerCert PKCS_6_Certificate,
  timeStamp VerifiedTimeStamp,
  infoExtAttributes ::= Attributes
}
Attributes ::= SET OF Attributes
Attribute ::= SEQUENCE {
  key OCTET STRING,
  value OCTET STRING
}

```

420

```

DataObjectTokenAccessors ::= SET OF
DataObjectTokenAccesser
DataObjectTokenAccesser ::= SEQUENCE {
  accessor PKCS_6_Certificate,
  accessRights AccessRights,
  authRequirements AuthRequirements,
  signerInfo PKCS_7_SignerInfo,
  signerCert PKCS_6_Certificate,
  timeStamp VerifiedTimeStamp,
}

```

430

```

AccessRights ::= SET OF AccessRight
AccessRight ::= SEQUENCE {
  type ENUMERATED {
    read_whole,
    read_partial,
    update_whole,
    update_partial,
    propagate_access_same,
    propagate_access_limited
  },
  extended UserObject // extended info
  // regarding
  // partial/limited
}

```

440

```

AuthRequirements ::= SET OF AuthRequirement
AuthRequirement ::= SEQUENCE {
  type ENUMERATED {
    basic, // certificate signature
    // always at least
    // this
    password,
    biometric,
    external_key,
    other
  },
  extended UserObject // extended info
  // regarding other
}

```

450

Figure 4

510

```

DataObjectTokenInfo ::= SEQUENCE {
  accessRequested      Access,
  authentication       UserObject,
  accessInfo           PKCS_7_SignerInfo,
  accessCert           PKCS_6_Certificate
}

Access ::= SEQUENCE {
  type ENUMERATED {
    read_whole,
    read_partial,
    update_whole,
    update_partial
  }
}
extended UserObject

```

520

```

DataObjectTokenExtension ::= SET OF DataObjectTokenExtension
DataObjectTokenExtension ::= SEQUENCE {
  entryType           OCTET STRING,
  userData            UserObject,
  signerInfos         PKCS_7_SignerInfos,
  signerCerts         PKCS_6_Certificates
}

PKCS_7_SignerInfos ::= SET OF PKCS_7_SignerInfo
PKCS_6_Certificates ::= SET OF PKCS_6_Certificate

```

Figure 5

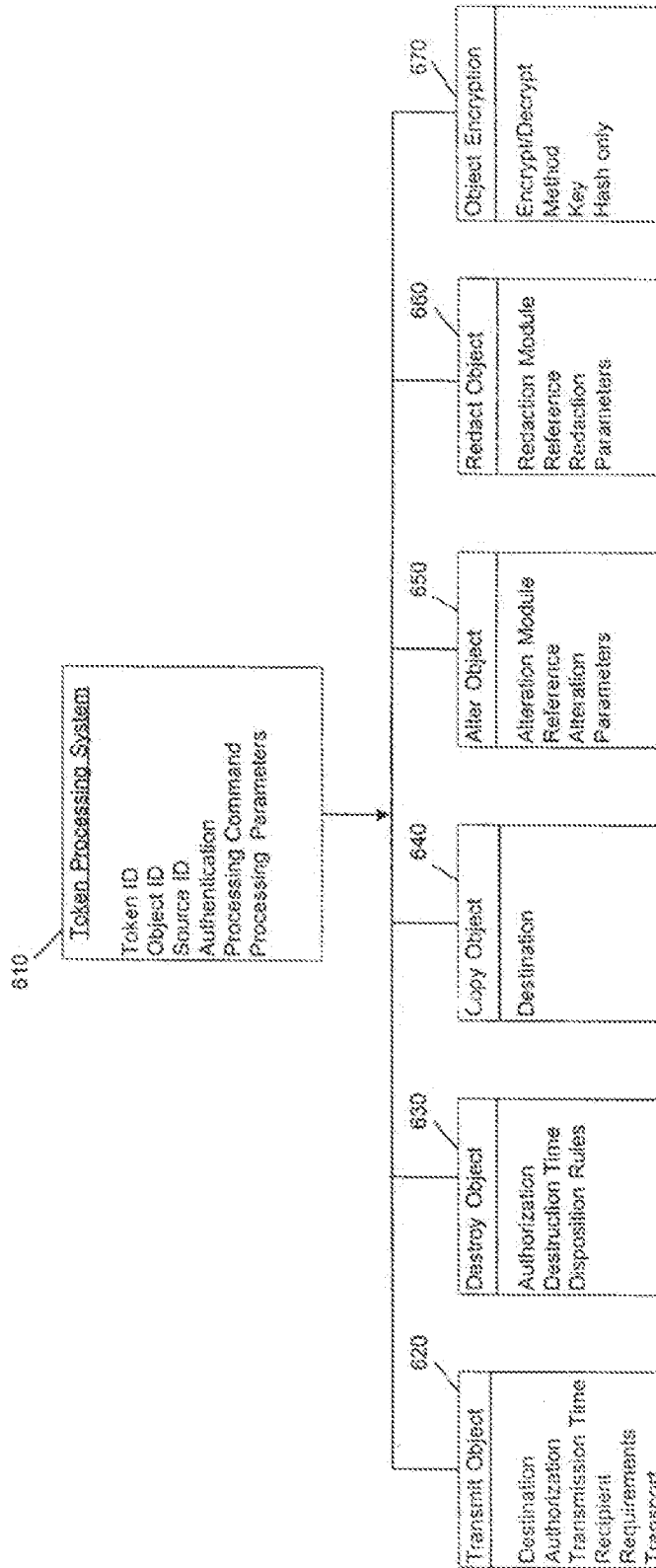


Figure 6
Partial Token Based Data Processing System Command Table

719

Token Generating System

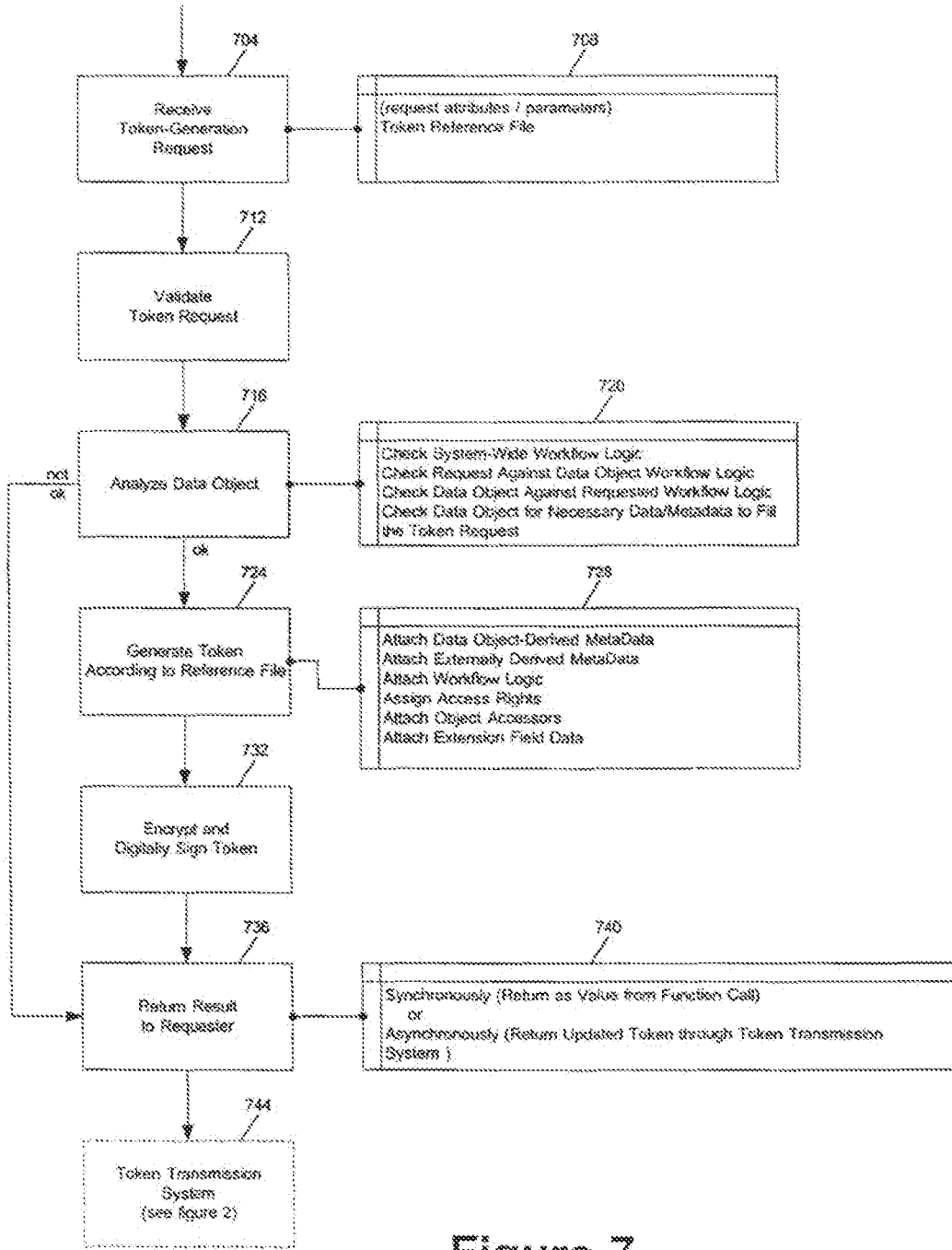


Figure 7

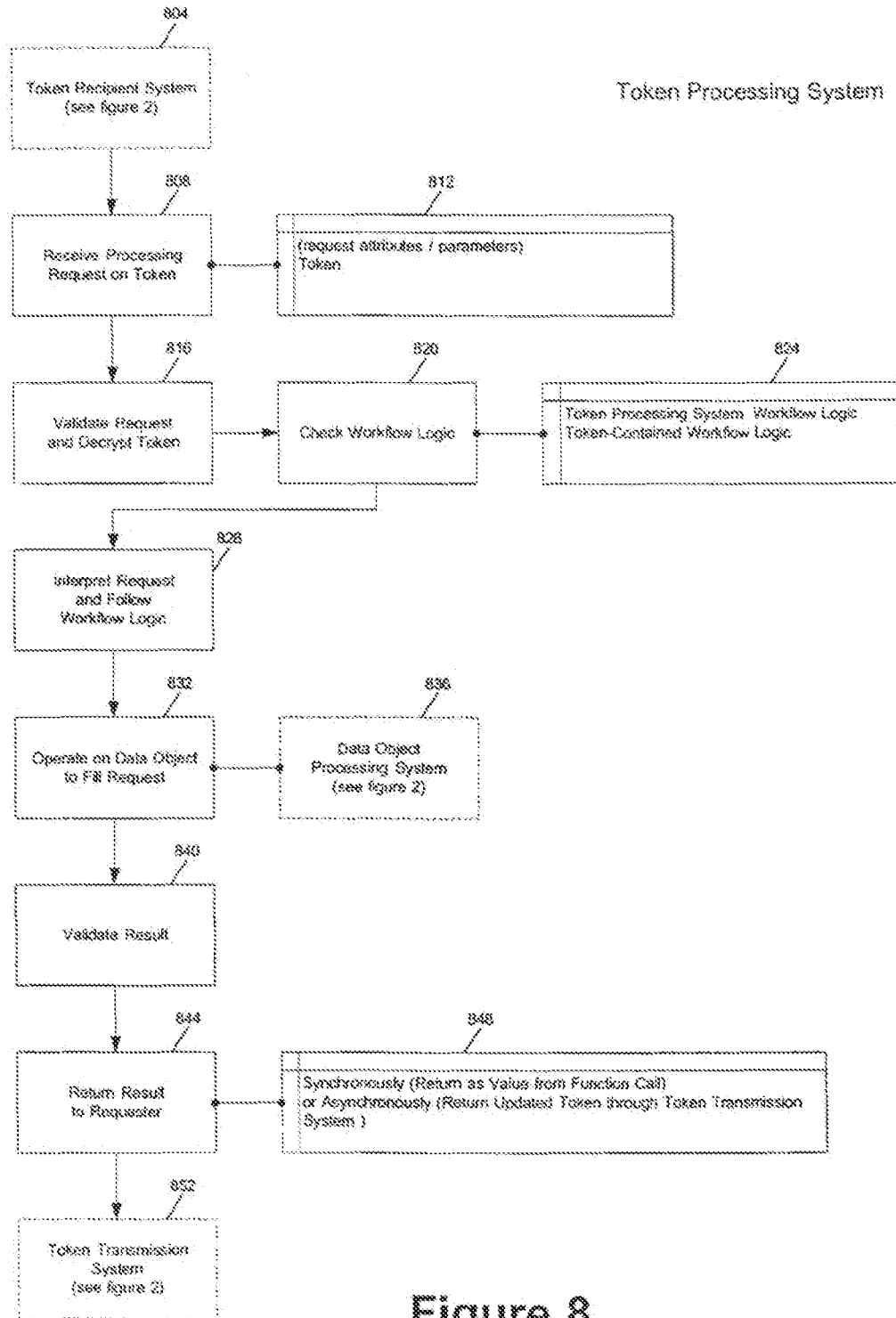


Figure 8

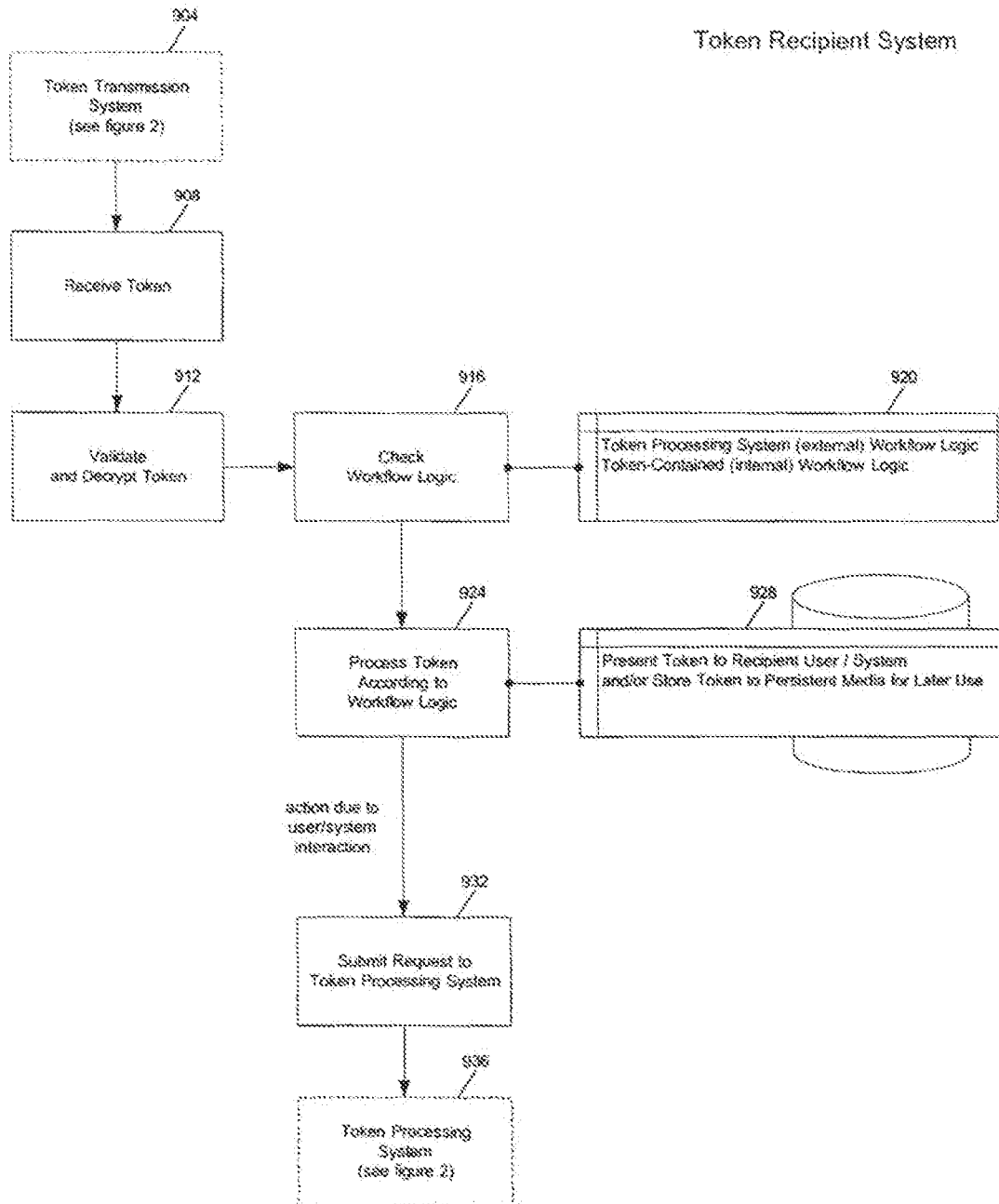


Figure 9

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
29 March 2001 (29.03.2001)

PCT

(10) International Publication Number
WO 01/22308 A2

(51) International Patent Classification: G06F 17/60

(21) International Application Number: PCT/IL/00/00576

(22) International Filing Date:
18 September 2000 (18.09.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
131991 22 September 1999 (22.09.1999) IL

(71) Applicant (for all designated states except US):
STEP2BUY INC. [US/US]; 55 Northern Blvd., Suite #
304, Great Neck, NY 11021 (US).

(72) Inventors; and

(75) Inventors/Applicants (for US only): BROOK-LEVIN-
SON, Edward [IL/IL]; Mana Semesh Street 11, 49507

Petah Tikva (IL), HERMAN, Abraham [IL/IL]; Hagilgal
Street 45, 52392 Ramat Gan (IL), GRABELKOVSKY,
Alexander [IL/IL]; Bar Yehuda Street 12, 42344 Netanya
(IL), GRABELKOVSKY, Michael [IL/IL]; Michael
Ankave Street 16, 42344 Netanya (IL), LAN, Felix
[IL/IL]; Haviva Reik Street 5, 58000 Holon (IL).

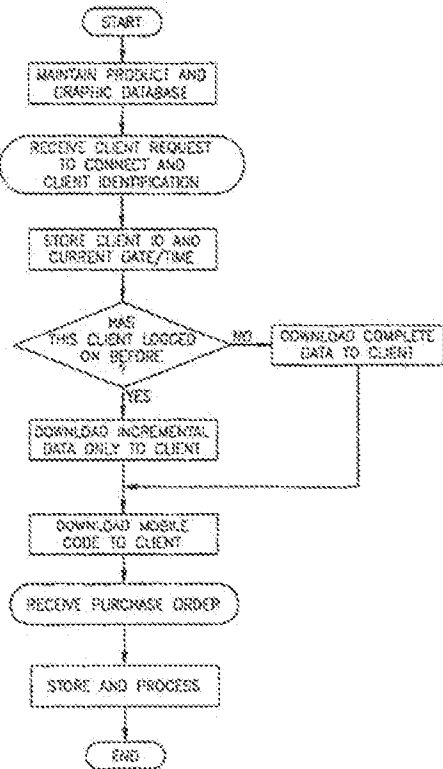
(74) Agent: REINHOLD COHN AND PARTNERS; P.O.
Box 4060, 61040 Tel-Aviv (IL).

(81) Designated States (national): AE, AG, AL, AM, AT, AU,
AZ, BA, BB, BG, BE, BY, BZ, CA, CH, CN, 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, LR,
LS, LI, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ,
NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, 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

[Continued on next page]

(54) Title: COMPUTER-IMPLEMENTED METHOD AND SYSTEM FOR SELECTING ONE OR MORE REQUIRED ITEMS FROM A VIRTUAL STORE



(57) Abstract: A computer-implemented method and system for selecting one or more required items from a virtual store and stored in a storage medium thereof, wherein a client displays a 3-dimensional image of at least part of a store showing one or more items for sale, freely and continuously navigates through the image so as to display further parts of the store each showing respective items until one of the required items is displayed and selects the required items for further investigation. This cycle is repeated as necessary until all of the required items have been selected. Data relating to the virtual store may be downloaded from a central database, typically from a website of the virtual store.

WO 01/22308 A2



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, LI, MC, NL, PT, SE), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MB, 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:

----- *Without international search report and to be republished upon receipt of that report.*

**Computer-implemented method and system for selecting
one or more required items from a virtual store**

FIELD OF THE INVENTION

This invention relates to interactive e-commerce methods and system for allowing purchase of one or more items from a virtual store. In particular, it relates to a 3-dimensional interactive shopping method and system.

5 BACKGROUND OF THE INVENTION

Interactive shopping systems are known for allowing a user to purchase items from a virtual store over the Internet. US Patent No. 5,796,351 (Tsutomu) discloses a system for providing a user with proper information about a target exhibition object among many exhibition objects in exhibition facilities according
10 to a user's request. The system has a database for storing information about exhibition objects, a host computer, and at least one terminal. The host computer retrieves information from the database according to the identification code of a given exhibition object. Each of the exhibition objects has a communication unit that returns an identification code in response to a request signal issued by the user.

15 US Patent No. 5,721,832 (Westrope *et al.*) discloses a method and apparatus for an interactive, computerized catalog system in which a customer can selectively access video and audio catalog data from a computerized catalog memory that permits a customer to peruse an entire catalog of products or services or select specific portions from specific catalogs or services and if desired place an order
20 which is processed electronically and from which customer profile marketing data is selectively generated.

US Patent No. 5,053,956 (Donald *et al.*) discloses an interactive computer system for retail transactions including a video display connected to a computer for displaying a composite display image including an object-image of a selected saleable object superimposed on a background-image of a selected background scene. An object-image storage device is connected to the computer for storing a plurality of object-image frames each of which contains a photographic or video image of a respective salable object, wherein each object-image frame is categorized with other related object-image frames according to at least one category of sales characteristic. The object-image storage means under control of said computer provides an output of a selected object-image frame to the video display in response to a corresponding object-image address input generated by the computer. A background-image storage device is connected to the computer for storing a plurality of background-image frames each of which contains an image representation of a respective background scene, wherein each background-image frame is selectable by an individual background-image address, and wherein the background-image storage device under control of the computer provides an output of a selected background-image frame to the video display in response to a corresponding background-image address input generated by the computer.

US Patent No. 5,664,111 (Nahan *et al.*) discloses a computerized, interactive system and method of electronically executing transactions with a preprogrammed main computer having data and image storage and retrieval equipment. A plurality of electronic images of works of art which are for sale are created by at least one listing dealer and stored on the storage equipment associated with the main computer. Data is input about each stored image and input data is associated with each corresponding stored image. A plurality of preprogrammed intelligent terminals each having data storage and retrieval equipment, at least one display screen and at least one input device, located at at least one listing dealer location and at at least one buying dealer location communicate with the main computer. Search criteria are input through the intelligent terminals for selecting at least one of the stored electronic images for review. Selected images and corresponding data

are communicated to the intelligent terminals and at least a portion of the selected electronic images are displayed. A reservation on at least one of the displayed electronic images can be made to prevent the completion of a sale transaction involving the artwork corresponding to the selected reserved electronic image. An indication of the reserve status of the work is displayed in conjunction with the display of the reserved work on any of the intelligent terminals. A purchase order can be input on the intelligent terminals to transact a purchase of the artwork corresponding to the electronic image subject to the purchase order. Instructions to complete the purchase are automatically generated and communicated to the intelligent terminals corresponding to the appropriate listing dealer and the appropriate buying dealer.

US Patent No. 5,825,881 (Colvin) discloses a public network merchandising system for conducting commerce over a large public network such as the Internet. The system facilitates communications between a merchant, a customer, and a bank or credit card processor.

US Patent No. 5,717,923 (Dedrick) discloses a method and apparatus for dynamically customizing electronic information to individual end users

US Patent No. 5,383,111 (Homma *et al.*) discloses a visual merchandizing (VMD) control method and system. A merchandising system for conducting a control of article display positions that are generally changed by the customers includes a method of obtaining article display positions based on the present display state. In this system, when selecting coordinate items to be suitably combined with an article, the display positions thereof can be easily determined, which advantageously helps the customer to search the store or floor for the desired items.

WO 9804984 (Efrat *et al.*) discloses a system for linking information to and accessing information from a video. A hotspot can be defined in a frame of a video. The hotspot can be tracked in other frames of the video. Also, the hotspot can be linked to a target. When the video is played, the hotspot can be actuated and the corresponding target executed.

- 4 -

This technology is used by Veon, Inc. in their virtual store shown on their Website <http://www.veon.com/solutions/step1.html> and subsequent pages in the same site demonstrating a virtual shopping experience where a customer can browse products in the aisle of a store, allowing the viewer to start, stop and interrupt the tour at any time and to navigate the store for particular products by pointing and clicking with a mouse or remote control device. Product information is displayed and the user can explore URLs or database information to learn more about the product functionality, assembly and so on. On-the-fly promotions are presented via a promotional incentive banner or special offer to influence the purchase decision. Finally, the system allows for product purchase using secure e-transactions in known manner.

In such a system, the customer navigation is limited by a predetermined linear scenario, which precludes the possibility for arbitrary customer displacements relative to the displayed products. As a result, the experience of the virtual customer in the Veon e-store differs significantly from that of a real customer in a non-virtual store. Specifically, whilst in a regular store, the customer can freely navigate her or his shopping trolley along the aisles in a completely non-determined and arbitrary fashion, this is not possible in the Veon store. Rather, there is presented a 3-d image of a store and the customer clicks on a subject of interest, such as a particular shelf, product and so on. The displayed image may possibly include a film sequence of photographic images. An enlarged image is now displayed showing the selected subject allowing selection of a product. Thus, when it is required now to go to a different part of the store which is not currently displayed, the customer must first go back to the complete image in order to be able to click on the desired store section. This is inconvenient and results in time-consuming processing which merely serves to present what amounts to an intermediate image of the complete store. By the same token, it is not possible for the customer to navigate to a specific location of the entire store since all coordinates are relative to the displayed image only. Furthermore, the viewing

- 5 -

angle of the customer relative to products in the Veon store is limited, thus not permitting the customer to look at and approach the products at arbitrary angles.

Yet a further problem associated with prior art systems is that the bandwidth of networks available at present limits the quantity of graphic resources which can be provided for delivering the visual presentation to the customer's terminal. In a typical scenario, where a customer accesses the virtual store through the Internet, most of the graphic and database resources are remote to the customer and frames of graphic data are downloaded from a web server to the client machine on-the-fly, as required. Such graphic data is, of course, normally transferred in compressed format and is decompressed by the web browser resident on the client machine. This notwithstanding, the rate at which data can be transferred to the client from the web server is limited and the processing required to allow free navigation in real time is prohibitive. Much effort is being made in Internet-related graphic applications to transfer the graphic data ahead of time to the client so that the bulk of the processing is, in fact, performed locally by the client machine. This, of course, overcomes bandwidth limitations and gives the client the impression that the overall system response is higher than it actually is. Such approaches, utilizing pre-fetching of data, are based on prediction of a client's action and are therefore not properly commensurate with absolute free navigation. Put another way, free navigation by the customer will frequently require the downloading of graphic data different to that which was pre-fetched from the web server and, in this case, the problems associated with bandwidth restriction and limited processing remain unresolved. It will further be appreciated that limited bandwidth capacity militates against any possibility of downloading large quantities of graphic, or other, data thus again militating against the possibility of moving the processing burden from the web server to the client machine.

Free navigation *per se* is available commercially in a large number of games programs: both in commercially obtainable games and in software designed for the graphics programmer to develop his or her own applications. For example, US Patent No. 5,368,484 entitled "Vehicle simulator with realistic operating feedback"

- 6 -

(Copperman *et al.*) and assigned to Atari Games Corporation discloses a driver training system for a user of a simulated vehicle. The system includes input devices for controlling the simulated vehicle, a video display having three dimensional graphics, a computer, modeling software for determining position information based on the input devices, atmospheric effects software to simulate time-of-day and weather conditions, and realistic operating feedback software for simulating on the input devices the feedback normally experienced with operating the vehicle. The system includes a low frequency speaker mounted on an enclosure adjacent to the simulation user's seat through which road feel cues such as hitting an object are transmitted to the user in response to signals received from the computer. Further included is a system for simulating the feel to the user of anti-lock brakes on a brake pedal in response to signals received by the computer.

A web site providing reference to commercially available 3-dimensional graphics software engines and books supplying details thereof may be found at http://cg.cs.tu-berlin.de/~ki/3del_26203_spec_books.html. A good example of suitable software is provided by is Conitec (<http://www.conitec.com/vgtinfo.htm>) who offer a Virtual Game Toolkit that is a commercial Doom engine and VR toolkit for PCs that allows a programmer to develop adventure, action or role playing games based on an event controlled script language. A demonstration may be downloaded from <ftp://x2ftp.culu.fi/pub/msdos/programming/ack/acknex.zip>. The graphics includes:

- Non-orthogonal walls, variable level floors/ceilings, tilted floors and ceilings.
- Rooms above rooms.
- Movable and rotatable room parts.
- Light source plus ambient shading, distance cueing, fogging.
- Looking up/down via y-shearing.
- Transparent and animated textures.
- Animated 2D sprites.

- 7 -

- 320x200, 320x240 or 320x400 Pixels in 256 Colors on standard VGA and SVGA support.
- Sound and music support, distance-dependent volume control.
- Each wall, region, thing or actor is a state machine with event-controlled behavior.
- Support of 3D polygonal actors.
- Import of Quake MDL 3D objects.
- Landscape editor WED with integrated game compiler, WRUN interpreter and runtime module.
- Stereo sound support and CD-Audio support.

No attempt appears to have been made to exploit such techniques for an on-line virtual store, notwithstanding the fact that such 3-d graphics techniques have been available for several years: at least as long as virtual stores on the Internet. In particular, no such attempt has been made to display an on-line virtual store such as that used by Veon, Inc using a 3-d graphics engine. The reason for will be clear from the foregoing discussion of the Veon store, which as noted above is based on the technology described in WO 9804984 (Efrat *et al.*). Such an approach is based on partitioning the store into a sub-image centered around a selected "hot spot" which is clicked by the user. Thus, movement from one location to another is necessarily discontinuous and "jerky" and precludes direct selection of an area that is currently off-screen, for the reasons noted above. There is certainly no motivation to one of average skill in the art reading either the disclosure of WO 9804984 or accessing the Veon, Inc. web site to consider substituting the technology described in WO 9804984 by a 3-d graphics engine, since the two technologies are mutually exclusive. Thus it is that in spite of the proliferation of 3-d graphics simulation software and the ever-increasing interest in e-commerce, there has been no suggestion to combine the two.

SUMMARY OF THE INVENTION

It is an object of the present invention to provide an improved interactive 3-D e-shopping system allowing for free navigation by the customer through a virtual store. It is a particular object of the invention to provide such a system suitable for
5 operation by a user with minimal computer skills.

These objects are realized in accordance with a broad aspect of the invention by a computer-implemented method for selecting one or more required items from a virtual store and stored in a storage medium thereof, the method comprising the following steps carried out by a client:

- 10 (a) displaying a 3-dimensional image of at least part of a store showing one or more items for sale;
- characterized by:
- (b) freely and continuously navigating through said image so as to display further parts of the store each showing respective items until one of the
15 required items is displayed,
- (c) selecting the required items for further investigation, and
- (d) repeating steps (b) and (c) as necessary until all of the required items have been selected.

Thus, in accordance with the invention there is generated at a client work
20 station a 3-D image of a virtual store comprising trading halls with walls, doors, floor and ceiling; shelves for placing products thereon and forming aisles for client navigation; images of store attendants, advertisements in the form of transparent and opaque posters; animations; and visible and invisible light sources for illuminating the store.

25 Also provided are simple means for allowing navigation through the aisles of the virtual store along the shelves at a pace comparable to that of a real shopping experience. Such navigation may be effected via a joystick, keyboard or by means of verbal instructions.

The system allows the client to stop navigation at any point in the virtual
30 store, to change his or her orientation relative to the virtual store aisles, shelves and

products, both vertically and horizontally and with unrestricted angular deviation, to approach or withdraw from any graphic object, be it an aisle, door, wall, shelf or product and to allow selection of a product for placement into a shopping trolley.

The system further allows a plan of the virtual store to be presented displaying thereon a marker showing the client's current location and allowing the client to jump from one department of the virtual store to another.

The shopping trolley is embedded within the virtual store image, thus giving the client the impression of actually being inside the virtual store.

Preferably, data relating to a selected item is extracted from a product database and displayed in association with the shopping trolley. Such information may be displayed in a distinct area of the display screen or in a special display unit mounted on and associated with the shopping trolley. For example, the display may be mounted on the handlebar of the shopping trolley and, in addition to displaying product information, may also be used to show other information such as advertisements, discounts, bonuses and so on.

In order to allow for faster processing and to render the system independent of bandwidth limitations, the store database is preferably stored locally at the client as is the navigation program so that, in effect, the only on-line communication required between the client and the virtual store is when one or more selected items are to be purchased. Once such a purchase has been made, client details as well as an inventory of the purchased items are stored for future reference in association with the virtual store and this information may be used during subsequent sessions to download from the web server to the client data that is of particular interest to the client based on the client's profile and other personal preferences derived from preceding sessions. Likewise, any changes to the store database may be made to a local copy thereof stored on the client machine by transferring to the client incremental data only based on the date of a previous session of the same client recorded at the virtual store. By such means, updates to the product database since a previous session may be quickly reflected at the client machine well within bandwidth limitations of current communication channels.

- 10 -

Other objects and features of the invention will become apparent from the detailed description of some preferred embodiments, which now follows.

BRIEF DESCRIPTION OF THE DRAWINGS

In order to understand the invention and to see how it may be carried out in practice, a preferred embodiment will now be described, by way of non-limiting example only, with reference to the accompanying drawings, in which:

Fig. 1 is a pictorial representation of a computer-implemented virtual shopping trolley for selecting one or more required items from a virtual store in a virtual shopping mall;

Fig. 2 is a pictorial representation of the shopping trolley shown in Fig. 1 when selecting one or more required items from a specific virtual store in the virtual shopping mall;

Fig. 3 is a block diagram showing functionally a system according to the invention including a virtual store and a virtual shopping trolley;

Fig. 4 is a block diagram showing functionally the principal components in the virtual store of Fig. 3;

Fig. 5 is a flow diagram summarizing operation of the virtual store;

Fig. 6 is a block diagram showing functionally the principal components in the virtual shopping trolley of Fig. 3; and

Fig. 7 is a flow diagram summarizing operation of the virtual shopping trolley.

DETAILED DESCRIPTION OF THE INVENTION

Fig. 1 shows a computer-implemented virtual shopping mall depicted generally as 10 including a plurality of virtual stores, three of which are referenced 11, 12 and 13 all displayed in 3-d on a display device 14. A shopper 15 operates a computer having a keyboard 16 and a pointing device, such as a mouse 17 or joystick, constituting selection devices, (not shown) for selecting one of the stores for subsequent selection of specific items therein. Selection of a desired store may

- 11 -

be effected either by typing at the keyboard a unique identifier of the required store, or alternatively and more simply by pointing the mouse to the required store and clicking on the left mouse button. The displayed image of the virtual shopping mall and stores as well as specific items in a particular store are managed by a graphics engine residing in the shopper's computer and responsive to a 3-d graphics database stored therein.

Fig. 2 shows part of a computer-implemented virtual store depicted generally as 20 including a plurality of shelves 21 displayed in 3-d on the display device 14 and carrying items 22. The shopper 15 selects an item 22 for possible purchase, using either the keyboard 16 or the mouse. The selected item then appears inside a shopping trolley 23 having associated therewith a handlebar 24 on which there is mounted a display panel 25. Product data pertaining to the selected item is extracted from a database for display in the display panel 25. It will be understood that the display panel 25 is optional and the product data can instead be displayed on an area of the display device 14 overlying the representation of the store.

The displayed product data includes descriptive information about the product as well as unit cost, weight (where applicable) and cost. The descriptive information may also include ingredients as well as health warnings and other notifications, e.g. unsuitable for diabetics, suitable for vegetarians and the like. If the selected item addresses the shopper's requirements, she can verify selection by clicking the left mouse button. Alternatively, she can reject it by clicking the right mouse button.

It will be understood that this mode of operation is representative only and other equivalent approaches are contemplated: these being merely design choices well within the competence of one of average skill in the computer programming arts. For example, merely directing the mouse towards an item displayed on the shelf may be adapted to display product data without, at this stage, placing the item in the shopping trolley, this being done only when the shopper confirms the selection. Other modifications are also feasible.

- 12 -

Other items may be similarly selected by controlling the joystick so as to allow totally free and unrestricted navigation through the store, so as to present different store sections. Successive store sections may be contiguous but do not have to be since the graphics engine is responsive to the joystick control for
5 extracting display data from the 3-dimensional graphics database for presentation on the display device 14. Thus, a plan of the virtual store may be presented on the display device 14 having displayed thereon a marker showing the client's current location and allowing the shopper to jump from one department of the virtual store to another. This allows a section of the store to be selected, which is "off-screen".
10 Alternatively, the shopper 15 can direct the marker using the pointing device to a displayed section of the virtual store; but in either case, the selected section of the store clearly need not be contiguous with the previously displayed section. Such an approach allows genuine free and unrestricted navigation by the shopper through the store according to her individual fancy and avoids the need for pre-program-
15 ming the shopper's route through the store, so common with hitherto proposed on-line stores. This feature affords the shopper a much more realistic shopping experience since it allows reproduction of a typical scenario where the shopper moves from one section to a remote section and suddenly remembers that she has forgotten something in a previous section, thus requiring her to go back to the
20 earlier section. Such a scenario clearly cannot be programmed ahead of time and can only be faithfully reproduced if free navigation through the store by the shopper is provided.

Once all items have been selected, they may be purchased by sending an order to the virtual store in a manner similar to that disclosed in above-mentioned
25 US Patent No.5,721,832 (Westrope *et al.*). The manner in which items are actually purchased and paid for is not itself a feature of the present invention, which is concerned only with the interaction between a virtual store and shopping trolley so as to provide the shopper with a realistic shopping experience.

Fig. 3 shows functionally a system 30 according to the invention including a
30 virtual store 31 connected via a data communication channel 32 to a virtual

-- 13 --

shopping trolley 33. The communication channel 32 can be a telephone line allowing for modem communication between the virtual store 31 and the shopping trolley 33. Most typically, communication is made through the Internet.

Fig. 4 shows functionally a detail of the virtual store 31, whilst Fig. 5 is a flow diagram summarizing operation thereof. The virtual store 31 comprises a processor 34 coupled to a storage medium 35 having stored therein a product database of all items currently stocked in the virtual store. The product database includes data representative of an identity and unit price of each item, as well as other descriptive information, as explained above.

A communications port 36 is coupled to the processor 34 and is capable of coupling to the virtual shopping trolley 33 via the communication channel 32 for receiving from the virtual shopping trolley 33 a request to connect and an identification of the virtual shopping trolley. A data transfer unit 37 is coupled to the processor 34 and to the storage medium 35 for extracting therefrom data representative of a respective 3-dimensional image of at least a part of the store showing one or more items for sale for adding to the virtual shopping trolley 33 and downloading via the communication channel to the virtual shopping trolley.

The data transfer unit 37 is adapted to download from the storage medium 35 to the virtual shopping trolley 33 a mobile program 38 for permitting unrestricted navigation by the virtual shopping trolley 33 through the image of the store. The mobile program may be a Java applet or an Active X module, operating in conjunction with a net browser on the shopper's computer in known manner. Java is a trademark of Sun Microsystems, Inc. This may be downloaded each time the virtual shopping trolley connects to the virtual store, so as to allow updates to the mobile program to be correctly reflected at the virtual shopping trolley. In order to allow for fast response time, the graphics database and the product database should be stored locally on the shopper's computer. Owing to bandwidth limitations, it is impractical to download these databases from the virtual store to the client. It is equally impractical to require remote real-time access by the shopping trolley 33 to the storage medium 35 owing to the volume of graphics data which would need to

-- 14 --

be downloaded and the associated processing overhead. The processor 34 is therefore responsive to information characterizing the shopper and is programmed to:

- 5 (a) store in the storage medium the identification of the virtual shopping trolley 33 and data representative of a current date, and
- (b) check whether the virtual shopping trolley 33 has made a previous connection and, if so, download to the virtual shopping trolley via the data communication channel 32 incremental data relating to changes in the database since the previous connection by the virtual shopping
10 trolley to the virtual store.

In order further to minimize processing and data transfer times, the processor 34 may be further adapted to download to the virtual shopping trolley 33 a previous purchase order to serve as a basis for selection of items, and store the purchase order in the storage medium 35. Thus, if the shopper has indicated, for
15 example, that she has certain food restrictions, this may be used to prevent irrelevant data from being transferred. In order to allow the virtual store to take the shopper's preferences into account, the processor is adapted to:

- 20 (a) check whether the virtual shopping trolley has made a previous connection and, if not download to the virtual shopping trolley a questionnaire for determining at least one client preference,
- (b) receive from the virtual shopping trolley data representative of at least one client preference,
- (c) store the at least one client preference in the storage medium, and
- 25 (d) download a reduced inventory of items according to the at least one client preference.

Although as noted above, selection and processing are preferably performed using a local version of the product and graphics databases stored on the shopper's computer, the invention contemplates the possibility of remote data extraction where bandwidth restrictions allow for this. The processor may then be adapted to:

- 15 -

- (a) receive data from the shopper relating to a required item to be selected, and
- (b) extract from the database the respective 3-dimensional image of the store showing the selected item for downloading to the virtual shopping trolley.

5

In order to afford an even more realistic shopping experience, the processor is further adapted to:

- (a) associate a respective event with different locations in the virtual store, and
- (b) carry out the respective event if the client navigates to a location having the event associated therewith.

10

For example, at least one of the events may be downloading information to the virtual shopping trolley. Such information may include data representative of a vocalized message, or it may relate to a promotion. By such means, when the shopper moves the shopping trolley to a specific part of the store showing, for example, dairy products, there may be vocalized a special offer relating to purchase of milk or cheese. Such promotions may be vocal and may also include downloading graphic notices showing special offers.

15

In order to allow for the purchase of selected items, the processor is further adapted to receive and process a purchase order of the selected items from the virtual shopping trolley. The purchase order thus received may be stored in the storage medium and serve as a template for subsequent orders.

20

Fig. 6 shows functionally a detail of the virtual shopping trolley 33, and Fig. 7 is a flow diagram summarizing operation thereof. The virtual shopping trolley 33 comprises a processor 40, and a display device 41 coupled to the processor 40 for displaying a 3-dimensional image of at least part of a store showing one or more items for sale. A storage medium 42 is coupled to the processor 40 and stores a program for freely and continuously navigating through the displayed image so as to display further parts of the store each showing respective items until one of the

25

– 16 –

required items is displayed. A selection device 43 is coupled to the processor 40 for selecting the required items for further investigation.

Stored within the storage medium 42 is a database of all items currently stocked in the virtual store, the database including data identifying each item and a unit price thereof. The processor 40 is adapted to extract from the database at least some of the data relating to the selected item for display on the display device 41 and, as noted above, is adapted to allow de-selection of the item.

As explained above, it is preferable for the database to be stored locally so as to speed data extraction and processing. To this end, a previous version of the database is stored in the storage medium 42, and the processor 40 is adapted to:

- (a) connect to a remote storage medium of the virtual store via a communication channel,
- (b) upload to the remote storage medium an identification for identifying the virtual shopping trolley,
- (c) download via the data communication channel incremental data relating to changes in the database since a previous connection by the virtual shopping trolley to the virtual store, and
- (d) update the previous version of the database according to the downloaded incremental data.

The storage medium 42 may be a hard disk to which an initial copy of the product and graphics databases are downloaded through the Internet. Alternatively, it may be a portable data carrier such as a CD-ROM. In such case, incremental downloaded from the virtual store is stored on the hard disk and serves as the reference point for the next session in order to determine what incremental data must be downloaded.

When the virtual shopping trolley 33 is connected to the virtual store through the Internet, the virtual shopping trolley 33 includes a browser 44 for downloading the navigation program 46 from the storage medium 35 in the virtual store (constituting a remote storage medium). The navigation program 46 is a mobile code processor for enabling unrestricted navigation by the client through the

- 17 -

image of the store section. In such an embodiment, the virtual shopping trolley is a program run on a client connected to a web server and the virtual store is a program run on the web server. In the appended claims, the term "client" is used in preference to "shopping trolley" in accordance with the normal distinction in Internet literature between a machine and the user at the machine.

The processor 40 is further adapted to upload to the remote storage medium data characterizing at least one client preference. This can relate to specific dietary restrictions, for example, and can be used to ensure that only food products conforming to these restrictions are accessible to the shopper.

The shopping trolley 33 may itself serve as the selection device so that by directing the shopping trolley to a required item, data relating to the item is extracted from the product database and displayed. The processor 40 is thus adapted to:

- (a) restrictions display an image of a shopping trolley for pushing by the client through the displayed image of the store,
- (b) identify an item pointed to by the trolley,
- (c) access the database to extract the data relating to the identified item, and
- (d) display the data relating to the item pointed to by the trolley.

Alternatively or additionally, the keyboard 16 may serve as an input device coupled to the computer for inputting data relating to a required item to be selected. In this case, the processor 40 is adapted to:

- (a) access the database so as to determine in which part of the store the selected item is stored, and
- (b) display the respective image of the appropriate part of the store showing the required item so as to allow selection thereof.

The processor 40 may be further adapted to:

- (a) associate a respective event with different locations in the virtual store, and
- (b) carry out the respective event if the shopper navigates to a location having the event associated therewith.

-- 18 --

The event may be a store promotion, for example, and information relating to the event may be displayed to the shopper or vocalized and played by a loudspeaker 47 coupled to the processor 40 and being responsive to the event for vocalizing information to the shopper.

5 Upon selection and authorization of items, the processor may be further adapted to compile a purchase order of the selected items for submission to the virtual store. The purchase order may be stored in the storage medium 42 to serve as a template for subsequent orders. It will be appreciated, however, that the purchasing of selected items is neither mandatory nor a feature of the invention.
10 Specifically, just as in real life shoppers frequently browse through a store merely to gauge product availability and prices, so too the invention allows selection of items for this purpose alone, without necessarily requiring that they be purchased.

It will be appreciated that modifications may be effected to the manner in which data is presented to the virtual shopping trolley whilst remaining true to the
15 essence of the invention, which resides in the ability to freely navigate through the store image. Thus, for example, de-selection of items may be done after all items have been selected; information may be displayed graphically and, in addition, or instead textually using modern GUI tools such as combo and list boxes. Other such changes will be apparent to those of average skill in the art without departing from
20 the scope of the invention as defined in the appended claims.

Likewise, whilst the system hardware has been described with particular reference to a computer-implemented system, it should be understood that the term "computer" is not restricted to any one type of client machine. Within the context of the description and appended claims, the terms "computer" and "client machine"
25 embrace any suitable client machine having at least data processing and display facility such as a PC, hand-held computer, mobile telephone, TV set top box having processing capability, and so on.

In the case that the client shops via a TV set top box, access to the virtual store can be via CATV. Much effort is being directed to the use of CATV for
30 downloading data of a non-TV related matter, in particular for providing Internet

- 19 -

users to the large population of potential surfers who are neither computer nor Internet literate. One problem is doing so relates to the limited bandwidth of cable TV, which is frequently used to near capacity in the effort to carry as many channels as possible, particularly in view of the increasing competition between CATV providers. Such bandwidth-related problems can be eliminated by using satellite communication. In either case, television techniques allows the incremental database data to be broadcast on a dedicated channel at predetermined and advertised times of day.

Thus, within the context of the description and appended claims, the term communication channel can be any channel allowing data communication and is certainly not intended to encompass only conventional Internet channels. Indeed, as has been noted above, any distinction between Internet communication channels and other types of data communication channels is becoming increasingly fuzzy and is likely to become even more so, as conventional telephone and entertainment channels are used to provide Internet access. Similarly, whilst in the preferred embodiment, the virtual store is an Internet store, it is clear that it need not be connected to the Internet. Thus, in the case where data is broadcast via CATV, for example, a client machine can tune in to the dedicated channel at the appropriate time to download the incremental database data. In the case where bandwidth considerations are not important, the complete database can be broadcast and stored at the client machine. Such broadcasts can be done in the middle of the night, when speed is of no importance to the client and can, of course, be activated by means of a preset clock similar to that used in VCR machines, for example. Once the data is stored, communication between the client and the virtual store may be made through the appropriate TV channel via CATV or satellite.

It will also be understood that both the virtual store and the virtual shopping trolley according to the invention may be a suitably programmed computer. Likewise, the invention contemplates a computer program being readable by a computer for executing the method of the invention. The invention further

- 20 -

contemplates a machine-readable memory tangibly embodying a program of instructions executable by the machine for executing the method of the invention.

In the method claims that follow, alphabetic characters used to designate claim steps are provided for convenience only and do not imply any particular order
5 of performing the steps.

- 21 -

CLAIMS:

1. A computer-implemented method for selecting one or more required items from a virtual store and stored in a storage medium thereof, the method comprising the following steps carried out by a client:

- 5 (a) displaying a 3-dimensional image of at least part of a store showing one or more items for sale;

characterized by:

- (b) freely and continuously navigating through said image so as to display further parts of the store each showing respective items until one of the
10 required items is displayed,
- (c) selecting the required items for further investigation, and
- (d) repeating steps (b) and (c) as necessary until all of the required items have been selected.

2. The method according to Claim 1, further including after step (c):

- 15 i) accessing a database of all items currently stocked in the virtual store, said database including data identifying each item and unit price thereof,
- ii) extracting from the database at least some of said data relating to the selected item,
- 20 iii) displaying at least some of said data; and
- iv) allowing de-selection of the item.

3. The method according to Claim 1 or 2, further including:

- (e) storing a basic version of the database locally at the client,
- (f) connecting via a data communication channel to the remote storage
25 medium of the virtual store,
- (h) uploading via the data communication channel to the remote storage medium an identification for identifying the client,

-- 22 --

- (i) downloading via the data communication channel incremental data relating to changes in the database since a previous connection by the client to the virtual store, and
- (j) updating the basic version of the database according to the downloaded incremental data.
- 5 4. The method according to Claim 1 to 3, further including:
- (k) downloading from the remote storage medium a mobile code program for enabling unrestricted navigation by the client through the image of the store section.
- 10 5. The method according to any one of Claims 2 to 4, further including:
- (l) uploading to the remote storage medium data characterizing at least one client preference.
6. The method according to any one of Claims 2 to 5, further including:
- (m) upon selection of an item in step (c), displaying data related to the selected item.
- 15 7. The method according to Claim 2 to 6, further including:
- (n) displaying an image of a shopping trolley for pushing by the client through the displayed image of the store, and
- (o) displaying in association with the shopping trolley said data relating to the selected item.
- 20 8. The method according to any one of Claims 2 to 7, further including:
- (p) displaying an image of a shopping trolley for pushing by the client through the displayed image of the store,
- (q) identifying an item pointed to by the trolley,
- 25 (r) accessing the database to extract the data relating to said item, and
- (s) displaying said data relating to the item pointed to by the trolley.
9. The method according to any one of the Claims 2 to 7, further including:
- (s) inputting data relating to a required item to be selected,

-- 23 --

- (t) accessing the database so as to determine in which part of the store the selected is stored, and
 - (u) displaying the respective image of said part of the store showing the required item so as to allow selection thereof.
- 5 10. The method according to any one of the preceding claims, further including:
- (q) associating a respective event with different locations in the virtual store, and
 - (r) carrying out the respective event if the client navigates to a location
- 10 having the event associated therewith.
11. The method according to Claim 10, wherein at least one of the events is displaying information to the client.
12. The method according to Claim 10, wherein at least one of the events is vocalizing information to the client.
- 15 13. The method according Claim 11 or 12, wherein the information relates to a promotion.
14. The method according to any one of the preceding claims, further including:
- (t) displaying an image of a shopping trolley for pushing by the client
- 20 through the displayed image of the store, and
- (u) upon selection of an item in step (c), displaying an image of the selected item in the shopping trolley.
15. The method according to any one of the preceding claims, further including:
- (v) submitting a purchase order to the virtual store of the selected items.
- 25 16. The method according to Claim 15, further including:
- (w) storing the purchase order to serve as a template for subsequent orders.
17. The method according to any one of the preceding claims, wherein the virtual store is part of a shopping complex allowing remote unrestricted navigation
- 30 through constituent stores therein.

-- 24 --

18. The method according to any one of the preceding claims, wherein:
in step (a) a plan of the virtual store and a marker showing a current location of the client are displayed, and
in step (b) the client jumps from one part of the virtual store to another.

5 19. A computer-implemented method for providing remote selection of one or more required items from a virtual store over a data communication channel, the method comprising the following steps carried out via a storage medium associated with the virtual store:

- 10 (a) maintaining a database of all items currently stocked in the virtual store, said database including data representative of an identity and unit price of each item,
- (b) receiving from the client a request to connect and a client identification, and
- (c) downloading to the client via the data communication channel data
15 representative of a respective 3-dimensional image of at least a part of the store showing one or more items for sale for selection by the client.

20. The method according to Claim 19, further including:

- (d) downloading to the client a mobile program for permitting unrestricted navigation by the client through the image of the store.

20 21. The method according to Claim 19 or 20, wherein following step (b) there are further included the steps of:

- i) storing the client identification and data representative of a current date, and
- ii) checking whether the client has made a previous connection and,
25 if so, downloading to the client via the data communication channel incremental data relating to changes in the database since the previous connection by the client to the virtual store.

22. The method according to Claim 21, further including:

- 30 iii) downloading to the client a previous purchase order to serve as a basis for selection of items, and

-- 25 --

iv) storing the purchase order.

23. The method according to Claim 19, wherein following step (b) there are further included:

- 5
- i) checking whether the client has made a previous connection and, if not downloading to the client a questionnaire for determining at least one client preference,
 - ii) receiving from the client data representative of at least one client preference,
 - iii) storing the at least one client preference, and
 - 10 iv) downloading a reduced inventory of items according to the at least one client preference.

24. The method according Claim 19, further including:

- (s) receiving data relating to a required item to be purchased, and
 - (t) extracting from the database the respective 3-dimensional image of the store showing the required item for downloading to the client.
- 15

25. The method according to any one of Claims 19 to 24, further including:

- (u) associating a respective event with different locations in the virtual store, and
 - (v) carrying out the respective event if the client navigates to a location having the event associated therewith.
- 20

26. The method according to Claim 25, wherein at least one of the events is downloading information to the client.

27. The method according to Claim 26, wherein the information includes data representative of a vocalized message.

28. The method according Claim 26 or 27, wherein the information relates to a promotion.

25

29. The method according to any one of Claims 19 to 28, further including:

- (w) receiving and processing a purchase order of the selected items from the client.

-- 26 --

30. The method according to Claim 29, further including:

- (s) storing the purchase order to serve as a template for subsequent orders.

31. The method according to any one of Claims 19 to 30, wherein the virtual store is part of a shopping complex allowing remote unrestricted navigation
5 through constituent stores therein.

32. A computer-implemented virtual shopping trolley (33) for selecting one or more required items from a virtual store (31), said virtual shopping trolley comprising:

- a processor (40),

10 a display device (41) coupled to the processor for displaying a 3-dimensional image of at least part of a store showing one or more items for sale,

- a storage medium (42) coupled to the processor,

a program stored (46) in said storage medium for freely and continuously navigating through said image so as to display further parts of the
15 store each showing respective items until one of the required items is displayed, and

- a selection device (43) coupled to the processor for selecting the required items for further investigation.

33. The virtual shopping trolley according to Claim 32, further including:

20 a database stored within said storage medium of all items currently stocked in the virtual store, said database including data identifying each item and a unit price thereof;

the processor being adapted to extract from the database at least some of said data relating to the selected item for display on the display device and being
25 adapted to allow de-selection of the item.

34. The virtual shopping trolley according to Claim 32 or 33, wherein a previous version of the database is stored in said storage medium, and the processor is adapted to:

30 (a) connect to a remote storage medium of the virtual store via a communication channel (32),

– 27 –

- (b) upload to the remote storage medium an identification for identifying the virtual shopping trolley,
- (c) download via the data communication channel incremental data relating to changes in the database since a previous connection by the virtual shopping trolley to the virtual store, and
- (d) update the previous version of the database according to the downloaded incremental data.

35. The virtual shopping trolley according to any one of Claims 32 to 34, wherein the storage medium is a portable data carrier.

10 36. The virtual shopping trolley according to any of Claims 32 to 35, further including:

a browser (44) for downloading from a remote storage medium a mobile code processor for enabling unrestricted navigation by the client through the image of the store section.

15 37. The virtual shopping trolley according to Claim 34 or 36, wherein the processor is further adapted to upload to the remote storage medium data characterizing at least one client preference.

38. The virtual shopping trolley according to any one of Claims 33 to 37, wherein the processor is responsive to selection of an item for displaying data related to the selected item.

39. The virtual shopping trolley according to any one of Claims 33 to 38, wherein the processor is further adapted to display an image of a shopping trolley for pushing by the client through the displayed image of the store, and to display in association with the shopping trolley said data relating to the selected item.

25 40. The virtual shopping trolley according to any one of Claims 33 to 39, wherein the processor is further adapted to:

- (e) display an image of a shopping trolley for pushing by the client through the displayed image of the store,
- (f) identify an item pointed to by the trolley,
- (g) access the database to extract the data relating to said item, and

30

- 28 -

(h) display said data relating to the item pointed to by the trolley.

41. The virtual shopping trolley according to any one of the Claims 33 to 39, further including an input device (47) coupled to the computer for inputting data relating to a required item to be selected;

5 the processor being further adapted to:

(i) access the database so as to determine in which part of the store the selected item is stored, and

(j) display the respective image of said part of the store showing the required item so as to allow selection thereof.

10 42. The virtual shopping trolley according to any one of Claims 32 to 41, wherein the processor is further adapted to:

(k) associate a respective event with different locations in the virtual store, and

15 (l) carry out the respective event if the client navigates to a location having the event associated therewith.

43. The virtual shopping trolley according to Claim 42, wherein the display device is responsive to at least one of the events for displaying information to the client.

44. The virtual shopping trolley according to Claim 42, further including a 20 loudspeaker which is responsive to at least one of the events for vocalizing information to the client.

45. The virtual shopping trolley according Claim 43 or 44, wherein the information relates to a promotion.

46. The virtual shopping trolley according to any one of Claims 32 to 45, 25 wherein the processor is further adapted to:

(m) displaying an image of a shopping trolley for pushing through the displayed image of the store, and

(n) upon selection of an item, displaying an image of the selected item in the shopping trolley.

-- 29 --

47. The virtual shopping trolley according to any one of the Claims 32 to 46, wherein the processor is further adapted to compile a purchase order of the selected items for submission to the virtual store.
48. The virtual shopping trolley according to Claim 47, wherein the
5 processor is further adapted to store the purchase order in the storage medium to serve as a template for subsequent orders.
49. The virtual shopping trolley according to any one of Claims 32 to 48, wherein the virtual store is part of a shopping complex allowing remote unrestricted navigation through constituent stores therein.
- 10 50. The virtual shopping trolley according to any one of Claims 32 to 49, wherein the processor is adapted to display a plan of the virtual store and a marker showing a current location of the client, and is further responsive to movement of the marker for allowing the client jumps from one part of the virtual store to another.
- 15 51. A computer-implemented virtual store for providing remote selection of one or more required items, said virtual store comprising:
- a processor,
 - a storage medium coupled to the processor and having stored therein a database of all items currently stocked in the virtual store, said database including
20 data representative of an identity and unit price of each item,
 - a communications port coupled to the processor and being capable of coupling to a remote virtual shopping trolley via a data communication channel for receiving from the virtual shopping trolley a request to connect and an identification of the virtual shopping trolley, and
 - 25 a data transfer unit coupled to the processor and to the storage medium for extracting from the storage medium data representative of a respective 3-dimensional image of at least a part of the store showing one or more items for sale for adding to the virtual shopping trolley and downloading via the communication channel to the virtual shopping trolley.

-- 30 --

52. The virtual store according to Claim 51, wherein the data transfer unit is further adapted to download from the storage medium to the virtual shopping trolley a mobile program for permitting unrestricted navigation by the virtual shopping trolley through the image of the store.

53. The virtual store according to Claim 51 or 52, wherein the processor is adapted to:

- (a) store in the storage medium the identification of the virtual shopping trolley and data representative of a current date, and
- (b) check whether the virtual shopping trolley has made a previous connection and, if so, download to the virtual shopping trolley via the data communication channel incremental data relating to changes in the database since the previous connection by the virtual shopping trolley to the virtual store.

54. The virtual store according to Claim 53, wherein the processor is further adapted to:

- (c) download to the virtual shopping trolley a previous purchase order to serve as a basis for selection of items, and
- (d) store the purchase order in the storage medium.

55. The virtual store according to Claim 51, wherein the processor is adapted to:

- (a) check whether the virtual shopping trolley has made a previous connection and, if not download to the virtual shopping trolley a questionnaire for determining at least one client preference,
- (b) receive from the virtual shopping trolley data representative of at least one client preference,
- (c) store the at least one client preference in the storage medium, and
- (d) download a reduced inventory of items according to the at least one client preference.

- 31 -

56. The virtual store according Claim 51, wherein the processor is adapted to:
- (a) receive data relating to a required item to be selected, and
 - (b) extract from the database the respective 3-dimensional image of the store showing the selected item for downloading to the virtual shopping trolley.
57. The virtual store according to any one of Claims 51 to 56, wherein the processor is further adapted to:
- (e) associate a respective event with different locations in the virtual store, and
 - (f) carry out the respective event if the client navigates to a location having the event associated therewith.
58. The virtual store according to Claim 57, wherein at least one of the events is downloading information to the virtual shopping trolley.
59. The virtual store according to Claim 58, wherein the information includes data representative of a vocalized message.
60. The virtual store according Claim 58 or 59, wherein the information relates to a promotion.
61. The virtual store according to any one of Claims 51 to 60, wherein the processor is further adapted to receive and process a purchase order of the selected items from the virtual shopping trolley.
62. The virtual store according to Claim 61, wherein the processor is further adapted to store the purchase order to serve as a template for subsequent orders.
63. The virtual store according to any one of Claims 51 to 62, wherein the virtual store is part of a shopping complex allowing remote unrestricted navigation through constituent stores therein.

-- 32 --

64. A computer program product comprising a computer useable medium having computer readable program code embodied therein for selecting one or more required items from a virtual store and stored in a storage medium thereof, the computer readable program code comprising:

- 5 (a) computer readable program code for causing a computer to display a 3-dimensional image of at least part of a store showing one or more items for sale,
- (b) computer readable program code for causing a computer to freely and continuously navigate through said image so as to display further parts
10 of the store each showing respective items until one of the required items is displayed,
- (c) computer readable program code for causing a computer to select the required items for further investigation, and
- (d) computer readable program code for causing a computer to repeat steps
15 (b) and (c) as necessary until all of the required items have been selected.

65. A program storage device readable by machine, tangibly embodying a program of instructions executable by the machine to perform method steps for selecting one or more required items from a virtual store and stored in a storage
20 medium thereof, the method steps comprising at least the steps of:

- (a) displaying a 3-dimensional image of at least part of a store showing one or more items for sale,
- (b) freely and continuously navigating through said image so as to display further parts of the store each showing respective items until one of the
25 required items is displayed,
- (c) selecting the required items for further investigation, and
- (d) repeating steps (b) and (c) as necessary until all of the required items have been selected.

- 33 -

66. A computer program product comprising a computer useable medium having computer readable program code embodied therein for providing remote selection of one or more required items from a virtual store over a data communication channel, the computer readable program code comprising:

- 5 (a) computer readable program code for causing a computer to maintain a database of all items currently stocked in the virtual store, said database including data representative of an identity and unit price of each item,
- (b) computer readable program code for causing a computer to receive from the client a request to connect and a client identification, and
- 10 (c) computer readable program code for causing a computer to download to the client via the data communication channel data representative of a respective 3-dimensional image of at least a part of the store showing one or more items for sale for selection by the client.

67. A program storage device readable by machine, tangibly embodying a
15 program of instructions executable by the machine to perform method steps for providing remote selection of one or more required items from a virtual store over a data communication channel, the method steps comprising at least the steps of:

- (a) maintaining a database of all items currently stocked in the virtual store, said database including data representative of an identity and unit price
20 of each item,
- (b) receiving from the client a request to connect and a client identification, and
- (c) downloading to the client via the data communication channel data representative of a respective 3-dimensional image of at least a part of
25 the store showing one or more items for sale for selection by the client.

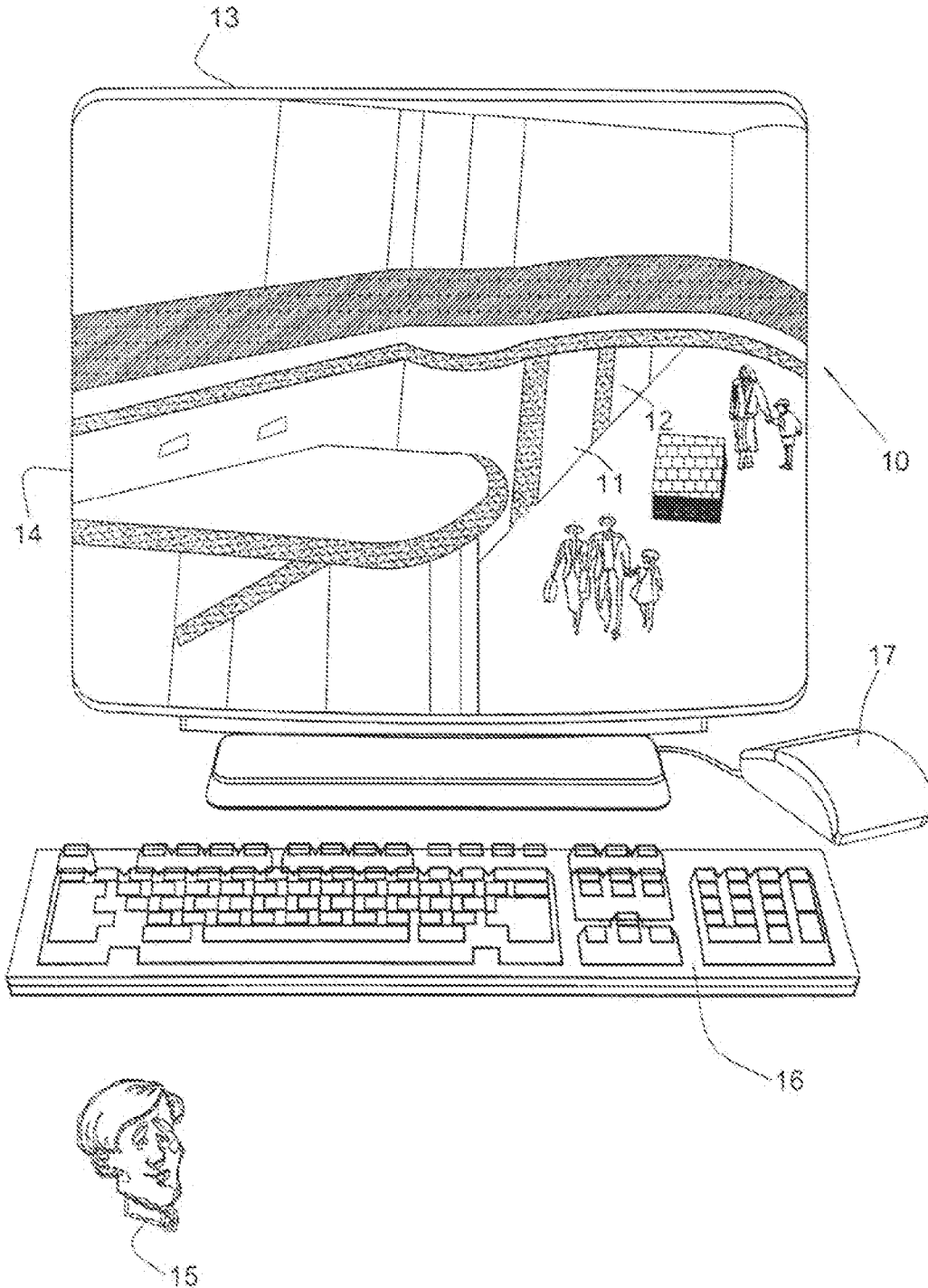


FIG. 1

2/6

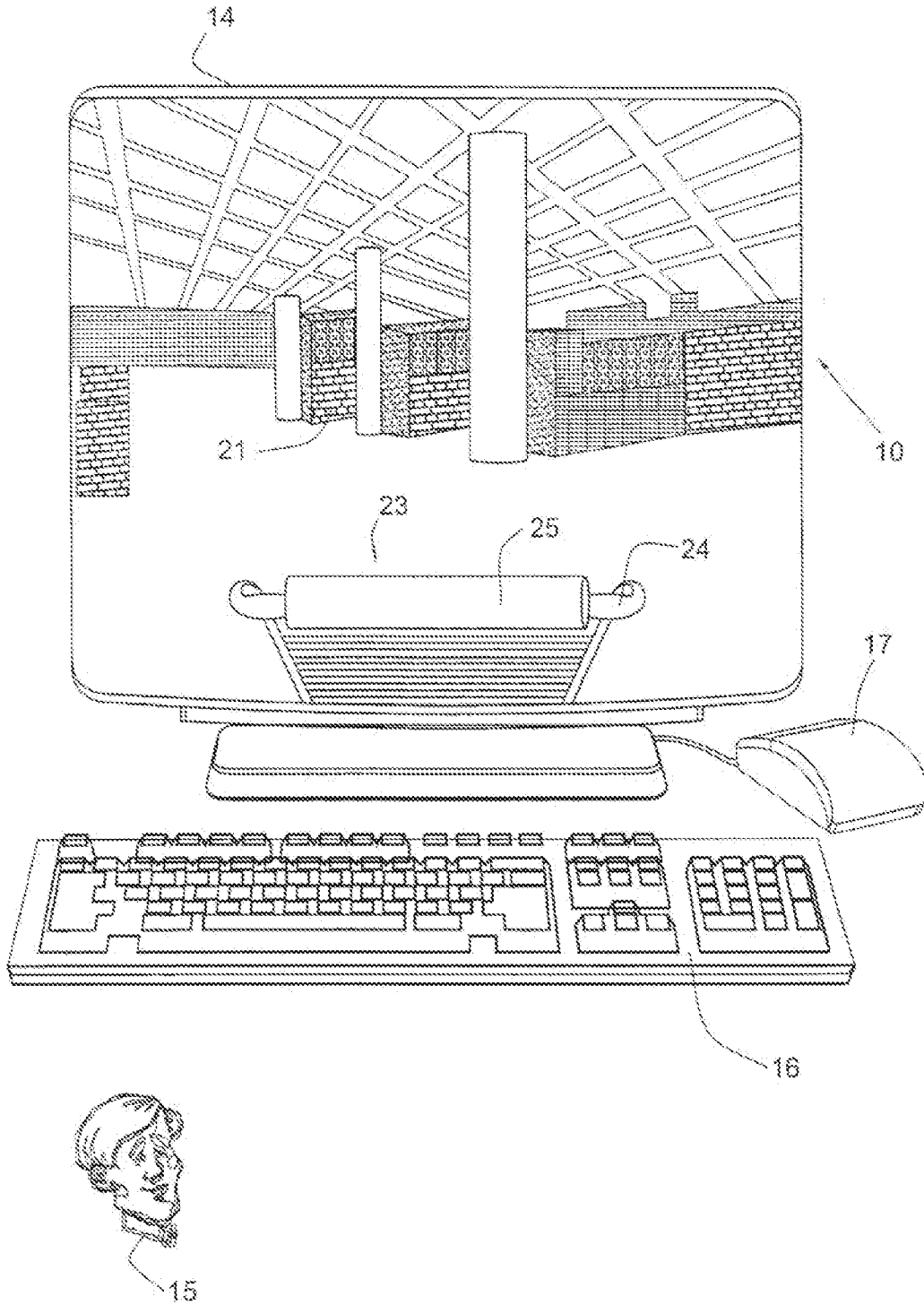


FIG. 2

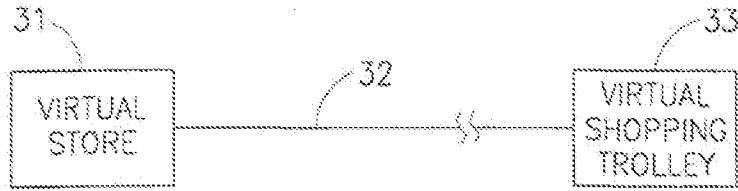


FIG. 3

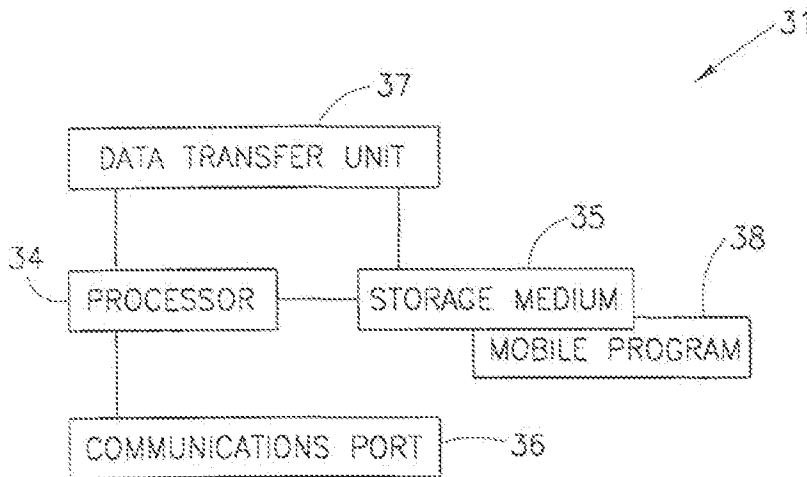


FIG. 4

4/6

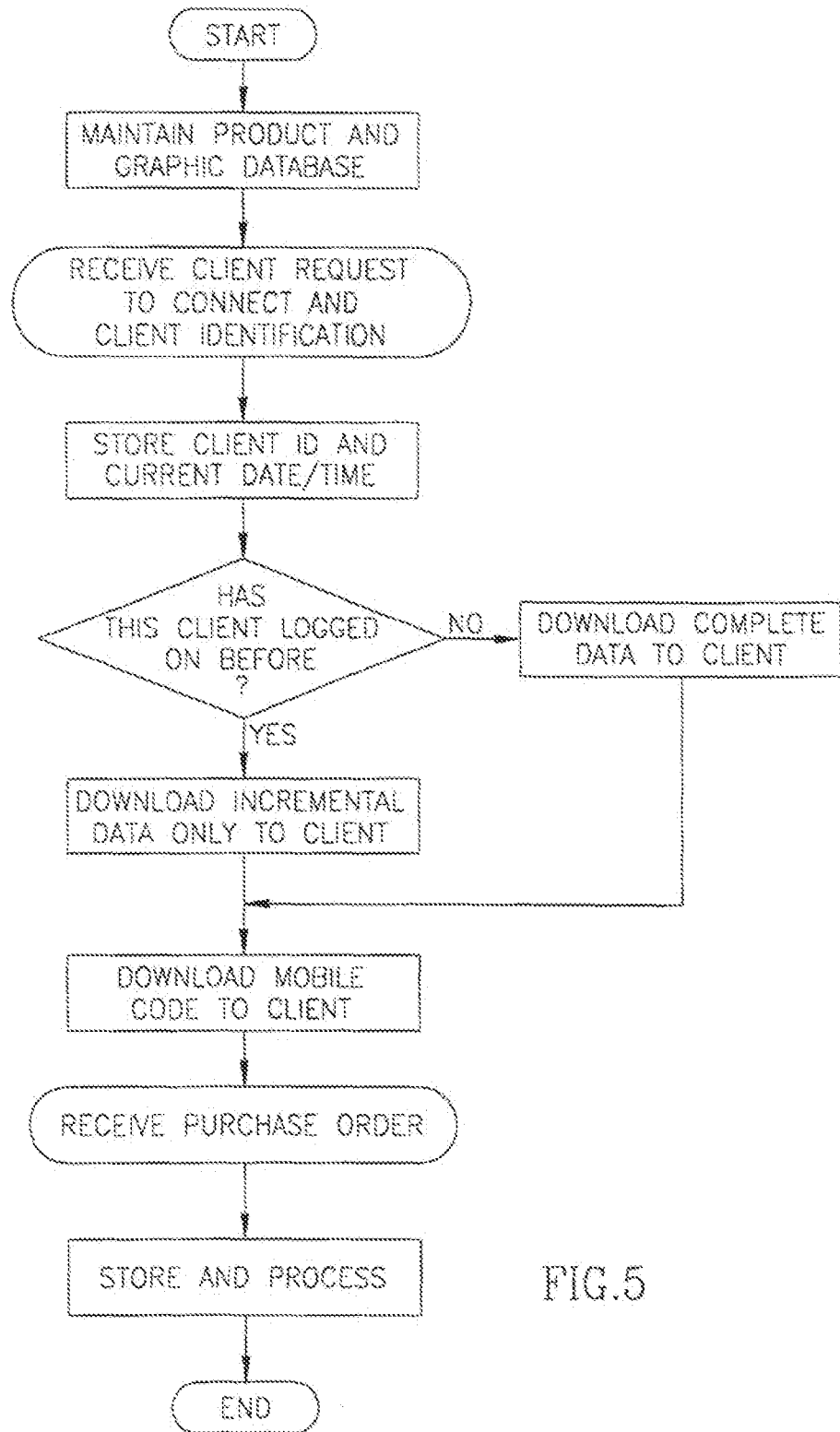


FIG.5

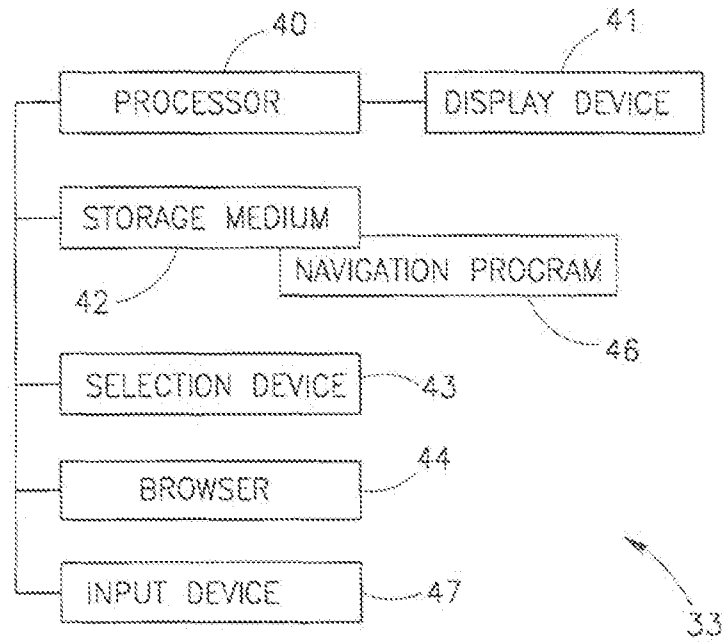


FIG. 6

6/6

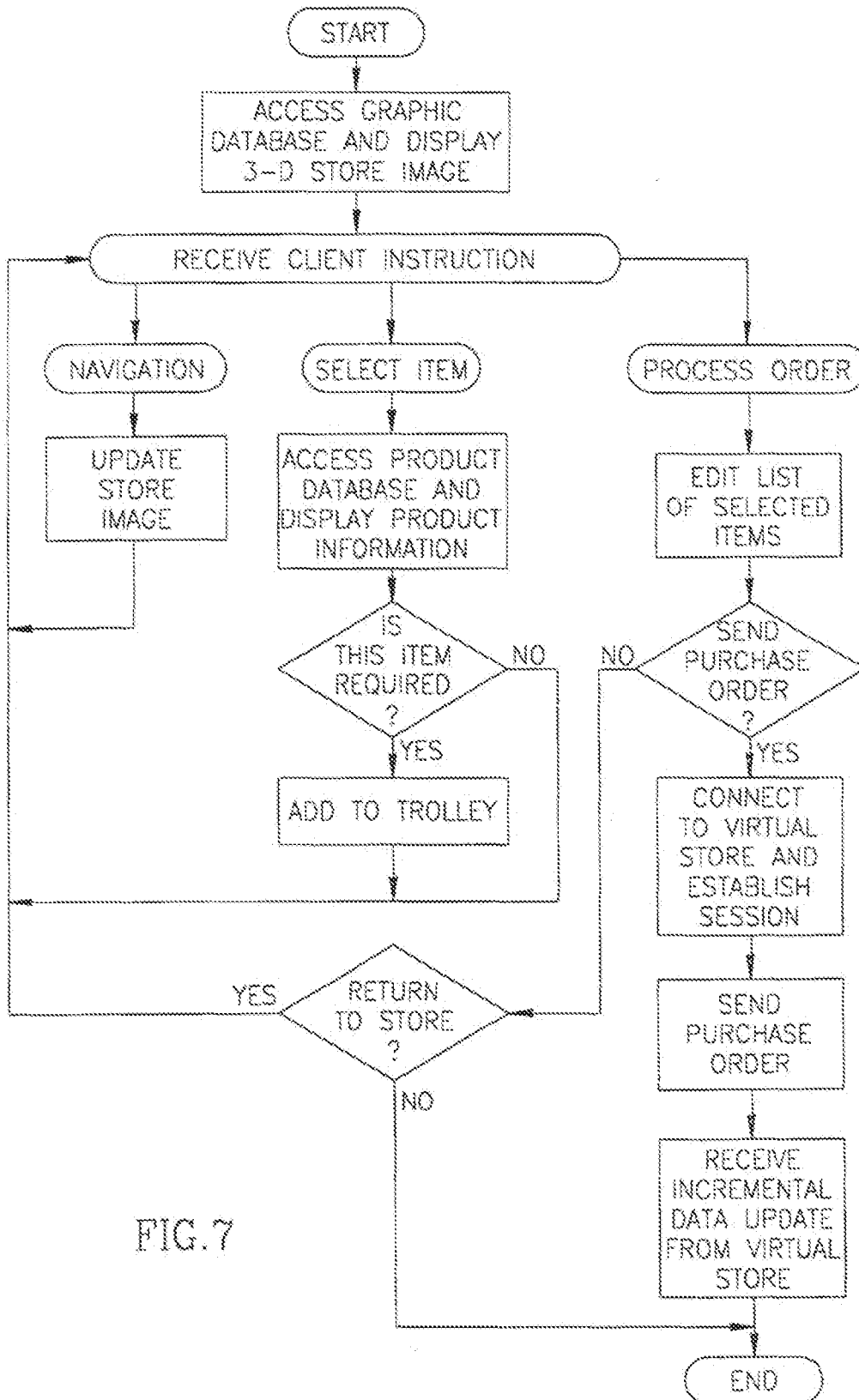


FIG. 7

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
7 June 2001 (07.06.2001)

PCT

(10) International Publication Number
WO 01/40899 A2

(51) International Patent Classification: G06F

(21) International Application Number: PCT/US99/42089

(22) International Filing Date:
10 November 2000 (10.11.2000)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/438,739 11 November 1999 (11.11.1999) US

(71) Applicant:

(72) Inventors: MASTER, Sydni, Beth [US/US]; Suite 510,
1995 University Avenue, Berkeley, CA 94704-1058 (US);
SYMONS, Timothy, J., M. [GB/US]; 4 Greenwood Com-
mon, Berkeley, CA 94708 (US).

(74) Agents: MALLIE, Michael, J. et al.; Blakely, Sokoloff,
Taylor & Zafman LLP, 7th floor, 12400 Wilshire Boul-
vard, Los Angeles, CA 90025 (US).

(81) Designated States (national): AE, AG, AL, AM, AT, AU,
AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, 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, LR,
LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ,
NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM,
TR, TT, TZ, UA, UG, 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, EG, KZ, MD, RU, TJ, TM); European
patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GR, GB, 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).

Published:

Without international search report and to be republished
upon receipt of that report.

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



WO 01/40899 A2

(54) Title: A COMPUTER IMPLEMENTED METHOD OF IMPLEMENTING, MAINTAINING, AND UPDATING A BRAND ARCHITECTURE SYSTEM

(57) Abstract: A method of implementing an automated brand architecture system is described. The method comprises displaying a navigable brand-architecture chart. The method further comprises receiving a nomination for a new name including a location in the brand-architecture chart. The method further includes testing the name against rules, to see whether the new name fits criteria. If the new name fits criteria, the brand architecture chart is updated with the new name.

A COMPUTER IMPLEMENTED METHOD OF IMPLEMENTING,
MAINTAINING, AND UPDATING A BRAND ARCHITECTURE SYSTEM
FIELD OF THE INVENTION

The present invention relates to brand architectures, and more specifically, to an automated brand architecture administration system.

BACKGROUND

A brand name is a name or symbol used to identify a seller's goods or services, and to differentiate them from those of competitors. Because a brand name identifies a product's or service's source -- protecting against competitors who may attempt to market similar goods or services -- companies have an incentive to invest in the quality, consistency, and imagery of their brands.

In the prior art, often there was no unified method to manage brand names within a company. In one prior art system, trademark counsel kept a list of trademarks and trademark filings for the purposes of keeping trademarks up-to-date and renewing them. Product managers in the organization created new products and potential names for them. The product managers then requested a search for trademark availability of the names. Other companies did not do any searching, causing the companies to risk being sued for trademark infringement.

Generally, in the prior art, organizations had no marketing or strategic central determiner for whether a product needed a name or whether the name chosen by the product manager was appropriate. Furthermore, each time an additional product or service was created the trademark counsel and product manager had to individually process the proposed name.

Therefore, it would be advantageous to reduce the complexity of administering a company's brand names.

SUMMARY OF THE INVENTION

A method of automatically managing and maintaining a brand architecture is described. The method comprises displaying a navigable brand-

architecture chart. The method further comprises helping users identify the appropriate, approved location of the desired name on the brand architecture chart. The method further includes testing the name against rules, to see whether the name fits the criteria. If the new name fits criteria, for one embodiment the proposed name is automatically sent to an approver, or to a legal department for approval. For another embodiment, the brand architecture chart is automatically updated with the new name.

For one embodiment, if the name does not fit the criteria, an appeal process to higher tier managers may be automatically triggered. For one embodiment a request for additional conforming name candidates is generated and sent to the appropriate party or parties.

BRIEF DESCRIPTION OF THE DRAWINGS

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:

Figure 1 is a block diagram of one embodiment of a network on which the present invention may be used.

Figure 2 is one embodiment of computer system on which the present invention may be implemented.

Figure 3 is a flowchart of one embodiment of the method of editing the brand architecture chart.

Figure 4 is a block diagram of one embodiment of the branding software.

Figure 5 is a sample filled-out questionnaire.

Figure 6A is an example of a possible layout for a brand architecture chart.

Figure 6B is an example of a possible layout for displaying a portion of the brand architecture chart of Figure 6A.

Figure 6C is another example of a possible layout for a brand architecture chart.

Figure 6D is an example of a possible layout for displaying a portion of the brand architecture chart of Figure 6C.

Figure 7 is an example of the rules implemented by the sample brand architecture charts of Figures 6A-D.

DETAILED DESCRIPTION

In the present invention, a brand architecture system is described. A brand architecture specifies and organizes the naming relationships among a company's brands, products, services, divisions, subsidiaries, etc. A well-conceived brand architecture will accommodate company and product line growth, and provide guidance for future product and service names. Some brand architectures are comprised of many tiers, with specific naming guidelines for each.

The brand architecture system described includes a navigable brand-architecture chart. A user is permitted to nominate a new name including a desired location on the brand-architecture chart to the system. For one embodiment, the user may be anyone with authority to access the brand architecture chart. For one embodiment, employees at different tiers have different tiers of access, from viewing access only, to name proposal access, to alteration access.

The proposed new name is tested against the naming rules at that location, to see whether the name fits criteria. If the new name fits the criteria, for one embodiment upon the approval of the legal department and/or a designated approving authority, the brand architecture chart is automatically updated with the new name. Alternatively, the brand architecture chart may be updated without the specific approval, based on the fit of the proposed name into the chart. For an alternative embodiment, the new name is added to a list of approved names, and the brand architecture chart is updated by an individual with alteration access.

For one embodiment, if the name does not fit the criteria, alternative allowable naming parameters are suggested. For one embodiment, if the name

does not fit the criteria, an appeal process to higher tier managers is automatically triggered. For one embodiment, request for additional name candidates is also sent to the appropriate parties.

Figure 1 is a block diagram of one embodiment of a network on which the present invention may be used. An internal network server 110 may be set up on a client's system. The internal network server 110 is accessed by a plurality of clients.

The internal network server 110 accesses a system 130 on which the brand architecture software is implemented through a network 120. For one embodiment, this system 130 may be referred to as the brand architecture system 130, although the server may not be a dedicated server, and the brand architecture software may be on an existing LAN/Intranet. For one embodiment, the internal network server 110 may be the same computer as the brand architecture server 130. For another embodiment, the internal network server 110 may be a separate computer, and the network 120 may be an internal large area network (LAN), wide area network (WAN), the Internet or other network. For yet another embodiment, the brand architecture server 130 may be outside the company, and in fact may be administered by someone outside the company. In that case, the network 120 may be the Internet. For one embodiment, if the network 120 is the Internet, communications between the internal network server 110 and the brand architecture server 130 may be secured by using secure hypertext transmission protocol (https), by encryption, or through some other means of securing the communication. The brand architecture server 130 includes brand architecture software 140, which is described in more detail below.

Figure 2 is one embodiment of computer system on which the present invention may be implemented. Figure 2 illustrates a typical data processing system upon which one embodiment of the present invention is implemented. It will be apparent to those of ordinary skill in the art, however that other alternative systems of various system architectures may also be used.

The data processing system illustrated in Figure 2 includes a bus or other internal communication means 245 for communicating information, and a processor 240 coupled to the bus 245 for processing information. The system further comprises a random access memory (RAM) or other volatile storage device 250 (referred to as memory), coupled to bus 245 for storing information and instructions to be executed by processor 240. Main memory 250 also may be used for storing temporary variables or other intermediate information during execution of instructions by processor 240. The system also comprises a read only memory (ROM) and/or static storage device 220 coupled to bus 240 for storing static information and instructions for processor 240, and a data storage device 225 such as a magnetic disk or optical disk and its corresponding disk drive. Data storage device 225 is coupled to bus 245 for storing information and instructions.

The system may further be coupled to a display device 270, such as a cathode ray tube (CRT) or a liquid crystal display (LCD) coupled to bus 245 through bus 265 for displaying information to a computer user. An alphanumeric input device 275, including alphanumeric and other keys, may also be coupled to bus 245 through bus 265 for communicating information and command selections to processor 240. An additional user input device is cursor control device 280, such as a mouse, a trackball, stylus, or cursor direction keys coupled to bus 245 through bus 265 for communicating direction information and command selections to processor 240, and for controlling cursor movement on display device 270.

Another device that may optionally be coupled to computer system 230 is a communication device 290 for accessing other nodes of a distributed system via a network. The communication device 290 may include any of a number of commercially available networking peripheral devices such as those used for coupling to an Ethernet, token ring, Internet, or wide area network. Note that any or all of the components of this system illustrated in Figure 2 and associated hardware may be used in various embodiments of the present invention.

It will be appreciated by those of ordinary skill in the art that any configuration of the system may be used for various purposes according to the particular implementation. The control logic or software implementing the present invention can be stored in main memory 250, mass storage device 225, or other storage medium locally or remotely accessible to processor 240. Other storage media may include floppy disks, memory cards, flash memory, or CD-ROM drives.

It will be apparent to those of ordinary skill in the art that the methods and processes described herein can be implemented as software stored in main memory 250 or read only memory 220 and executed by processor 240. This control logic or software may also be resident on an article of manufacture comprising a computer readable medium having computer readable program code embodied therein and being readable by the mass storage device 225 and for causing the processor 240 to operate in accordance with the methods and teachings herein.

The software of the present invention may also be embodied in a handheld or portable device containing a subset of the computer hardware components described above. For example, the handheld device may be configured to contain only the bus 245, the processor 240, and memory 250 and/or 225. The handheld device may also be configured to include a set of buttons or input signaling components with which a user may select from a set of available options. The handheld device may also be configured to include an output apparatus such as a liquid crystal display (LCD) or display element matrix for displaying information to a user of the handheld device. Conventional methods may be used to implement such a handheld device. The implementation of the present invention for such a device would be apparent to one of ordinary skill in the art given the disclosure of the present invention as provided herein.

Figure 3 is a flowchart of one embodiment of the method of accessing and querying the brand architecture chart. At block 300, the process starts. The process starts when an authorized user accesses the software, and indicates that

he or she wishes to access the brand architecture, to learn about it, to test a name candidate against it, or to add a new item to the architecture.

At block 305, a tutorial is displayed. For one embodiment, the tutorial may be optionally skipped. For one embodiment, this step may be omitted entirely. The tutorial may teach a branding philosophy and/or strategy of the company, the objectives of the branding architecture. The tutorial may include the company's brand architecture policies, methodologies, and rules. For one embodiment, the tutorial may further teach how the branding structure of the company works. For one embodiment, the tutorial may be interactive. Alternatively, the tutorial may be text based, video based, audio based, or in another format.

At block 310, a navigable brand architecture chart is displayed to the user. Sample charts are shown in Figures 6A-D. For one embodiment, this chart may be large enough to cover multiple pages or screens.

For one embodiment, the chart is displayed hierarchically, i.e. only the product family names are displayed, and the user can select to display the lower tier information for each family. For another embodiment, another method of displaying the chart may be used. Figures 6B and 6D illustrate another embodiment of the chart, permitting access to all tiers of the chart, while displaying only the relevant portions of the chart.

For one embodiment, the user may access the brand architecture chart without initiating the editing process. In that instance, the process stops here, and the user can view the navigable brand architecture chart. If the user initiates the editing process, the process continues to block 315.

At block 315, the user is prompted to choose a specific tier of the chart they wish to view. For one embodiment, this is a selection from a pull-down menu or by clicking the tier in the chart. For another embodiment, the user types in the tier name. For yet another embodiment, the user is not directly prompted for this information, but is rather asked to fill in a questionnaire from which this information is deduced. A sample of the questionnaire is included in Figure 5. For another embodiment, another means of selecting the tier may be

used. For one embodiment, this tier may be a "product family" as shown in Figures 6A-D. Alternative organizations of brand architecture charts may have alternative tiers, and selections.

At block 320, the navigable chart of the selected tier is displayed to the user. This is a subset of the brand architecture chart displayed at block 310. For one embodiment, this is the same chart, with the colors altered, to indicate the selected tier.

At block 325, the user is prompted to choose a product tier. As above, this choice may be made in a variety of ways, including automatically based on the questionnaire.

At block 330, the tier and product tier chosen is displayed to the user.

At block 335, the user is prompted to enter a proposed name.

At block 340, it is determined whether the proposed name fits into the rules of the selected family and tier. As will be discussed in more detail below, the rules restrict the types of names that will be acceptable at each tier. The purpose of the rules is to ensure the implementation of the company's adopted brand architecture, and simplify naming options. These rules are discussed in more detail below, with respect to Figure 7. If the proposed name fits into the rules, the process continues to block 370. At block 370, the user is informed that the proposed name will be added to the chart.

For one embodiment, the chart is automatically updated at this stage. If, for example, only those with naming authority within the company have access to this process, automating the updating of the chart may be advantageous. The process then ends at block 375. For another embodiment, the approved name is added to a list of proposed names. An authorized person with authority to update the chart then can update the actual chart. For another embodiment, the proposed name may be sent to the legal department or other appropriate person for approval. For example, the company may wish to review a name, even if it fits the rules, to verify that the name is appropriate, that there are no trademark issues, and/or that there are no other criteria that are not embodied in the brand architecture chart.

If the proposed name did not fit into the rules, at block 340, the process continues to block 345. At block 345, it is determined whether the user wishes to proceed with the name in spite of this. For one embodiment, this is determined by prompting the user. For one embodiment, the user should meet certain criteria in order to proceed at this stage. For example, the user may be asked to commit a certain amount of money from his or her budget to search and/or provide marketing support for the proposed name. Other indications of the user's commitment to the name may also be requested. If the user does not wish to go ahead with the name, the process returns to block 335, and the user is again prompted to enter a proposed name. If the user wishes to go ahead with the name, the process continues to block 350.

At block 350, the user is prompted to enter reasons why the non-conforming name should be approved. For one embodiment, this may be done in the form of a questionnaire, or a free-form data entry. For one embodiment, the user provides projections for the product that is being named, including potential sales, etc. For one embodiment, the user's answers to the questions will determine whether the name is likely to be approved although it is outside the approved architecture.

At block 355, the user's entry is forwarded to a designated individual or group who have authority to override the rules embodied in the brand architecture. For one embodiment, this step automatically generates an e-mail sent to the authorized person or persons, including the user's name suggestion, why it does not conform to the branding architecture, whether override criteria have been met, and any other relevant information. Designated persons are authorized to approve the override. For one embodiment, this step automatically connects to a organization chart of the company, and determines the correct person or persons to receive this e-mail. For one embodiment, the correct person may be a branding executive, a marketing executive and/or the intellectual property legal department.

For one embodiment, the user is disconnected from this process at this stage. For one embodiment, the user is informed that the request is being

forwarded to the person or persons authorized to approve the override request. For one embodiment, the time for approval may be days or weeks.

At block 360, it is determined whether the name has been approved. For one embodiment, the authorized person may return an affirmative or negative statement directly to the brand architecture software. The brand architecture software then analyzes the response, and determines whether it was positive or negative. For one embodiment, if multiple people have authorization to override, either a subgroup of or all of the group must return an affirmative answer in order to end with an approval.

For another embodiment, if the system does not receive a response, either affirmative or negative, within a preset period, the system sends a reminder request to the authorized persons.

If an affirmative response is received, the process continues to block 370, and the user is informed that the requested name has been approved and the proposed name will be added to the chart, and the chart is updated.

If a negative response is received, the process continues to block 365, and the user is informed of the unfavorable response. For one embodiment, the user is also provided a pointer to follow, to reapply for a new name. For one embodiment, the user may follow this pointer, and not have to select the product family and product tier, but rather go directly to block 335, to propose a new name. For one embodiment, the user can request help developing alternative names. A request is then sent to the appropriate name development person or outside agency.

The process terminates at block 375.

Figure 4 is a block diagram of one embodiment of the brand architecture software. The brand architecture system 410 includes a brand architecture chart display unit 420. The brand architecture chart display unit 420, for one embodiment, is capable of displaying a multi-page brand architecture chart, in various formats. For one embodiment, the format may be a hypertext markup language (HTML) format, a JavaScript implemented format, or another display

format. For one embodiment, the chart display unit 420 permits navigation along a family, tier, or arbitrary axis.

The chart configuration unit 430 controls which parts of the chart are displayed by the display unit 420. For one embodiment, as a user navigates through the chart, a narrower and narrower slice of the chart may be visible. The chart configuration unit 430 controls this, based on data from the user interface. If, for example, as described above, the user selects a particular product family, the chart configuration unit 430 instructs the chart display unit 420 to display only the selected product family.

The brand architecture system 410 further includes a user interface 440. The user interface 440 allows a variety of users to access the chart. An access control unit 450 is coupled to the user interface. The access control unit 450 determines the access tier of each user. For one embodiment, the access control unit 450 includes a plurality of classifications for potential users of the system 410. For example, there may be trusted users, those with authority to actually update the system, authorized users, those with authority to propose new names for inclusion in the chart, and other users, those who may look at the system, but may not change it. For another embodiment, the access control unit 450 may use some other method, such as a list of names, to determine access tiers.

For one embodiment, the user interface 440 is configured to the appropriate tier for a user, once that user accesses the system 410. For one embodiment, the user interface 440 displays only those options that are available to the user.

A comparison unit 460 is coupled to the user interface 440. When the user interface 440 receives a proposed name, for a selected product family and product tier, this information is passed to the comparison unit 460. The comparison unit 460 accesses the rules for this particular product family and product tier in the rule storage unit 460. If the name passes the rules, the user is informed of this. For one embodiment, the rules are displayed to the user by the comparison unit 440. If the proposed name meets rules, the name is submitted for approval and entry into the brand architecture chart. If the proposed name

is a non-conforming name, the proposed name is forwarded to the override facility 480, if the user so requests.

For one embodiment, the comparison unit 460 automatically determines whether the name as proposed passes certain rules. Certain of the rules may be automatically tested by comparison unit 460. For example, if the rule states that all product names must start with the letter "N," the comparison unit 460 may test whether the proposed name passes this rule. Similarly, for rules such as incorporating certain words, or family brand names, into the proposed name, having a certain number of letters in a name, etc. the comparison unit 460 may automatically determine whether the rule is met by the proposed name.

For one embodiment, for certain rules, the user is merely made aware of the rule and the user has to determine whether the rule is met by the proposed name. For example, for a rule such as "the name must be descriptive" the comparison unit 460 may not test the proposed name, for one embodiment. Rather, the user determines whether the proposed name meets the rule, or whether the override facility should be invoked.

For one embodiment, if the comparison unit 460 determines that the name does not meet the rules, the user is informed of this. For another embodiment, if the comparison unit 460 determines that the name does not meet rules, this is highlighted for the approving authority.

If the name does not pass the rules, the user may ask for assistance for generating alternate names. The a request may automatically be sent to a naming company, or other agency internal to or external to the company, to assist the user in generating an appropriate name. For one embodiment, the comparison unit 460 permits the user to generate this message, and passes the message to the appropriate organizations and/or individuals.

If the name did not pass the rules, but the user confirms that he or she wishes to go ahead with the name, the comparison unit 460 passes the information to the override facility 480.

The override facility 480 automatically forwards the override request to an authorized approver. For one embodiment, the override request is e-mailed

to the authorized approver. If the authorized approver is not within the secure network on which the brand architecture system 410 resides, the e-mail or other notification may be encrypted or otherwise secured.

The override facility 480 receives the response (s) from the authorized approver(s). For one embodiment, the override facility 480 sends a reminder or other notification if no response is received within a set period. When the response, or all of the responses, are received, the override facility determines whether the response(s) are positive or negative. For one embodiment, if there are multiple responses, the system can be set to require positive responses from all approvers, or from only a certain number of approvers, or from only a certain approver. The override facility 480 returns the affirmative or negative indication to the comparison unit 460.

The comparison unit 460 forwards this information to the user, and to the chart updating facility 490. If the name was approved, the chart updating facility 490 updates the chart displayed by the display unit 420. For one embodiment, the chart updating facility 490 can automatically update the chart if the nominated name either meets the rules, meets the rules and has been approved, or if the override request is approved. For another embodiment, the chart updating facility 490 maintains approval information. Then, the next time a user with authority to alter the chart logs in, this information is made accessible to the user, and the user can add the approved names.

Figure 5 is a sample questionnaire. The sample questionnaire is drawn up to correspond to the structure illustrated in Figures 6A-D. Of course, as the brand architecture changes for each company, the questionnaire would change correspondingly. It is to be understood that Figure 5 is exemplary, and should not be interpreted to be a complete or necessary format for such a questionnaire.

A similar type of questionnaire may be displayed to a user attempting to nominate a new name for a new product, service, feature, or technology. The flowchart described in Figure 3 may be automatically processed -- the selection of the appropriate category for the name, the name selection, as well as the

method of going outside the rules -- based on the information filled into the questionnaire.

The general information 510 that is requested from the user may include the user's name, title, group, and contact information. For one embodiment, the general information 510 may be automatically filled in based on the user's log-in into the system. However, the user is permitted, in one embodiment, to alter the data in the general information category 510.

The first tier selection 520 permits the user to select a brand, product, or category for the new name, in the example chart of Figures 6A-D. The user can select whether the new item will be a product family brand, a product, or an ingredient or feature.

The second tier selection 530 permits the user to select a subcategory for the new name. For example, in the brand architecture of Figures 6A-D, if the user selected a Product as the first tier selection for the proposed name, the user at this point can select whether the proposed name should be added into the Cheerios Cereal product family, the Super Moist Cake Mix family, etc. After this selection is complete, the system knows what the category and subcategory location is for the new name. Of course, this type of categorization may continue. For example, included in the Cheerios Cereal product family, there may be further sub-categories, such as Children's Cheerios and Adult Cheerios or Low Salt Cheerios and Standard Cheerios. Any such sub-categorizations could be consecutively selected by the user. The user may select either to the lowest tier, i.e. the lowest subcategory available, or the user may suggest the name for a higher tier. For example, in the brand architecture of Figures 6A-D, the user may propose a new product family, such as Baking Ingredients.

The product/service definition category selection 540 permits the user to enter a description. This may be optional, unless the user wishes to select a name outside the rules for the selected location.

The proposed name 550 is the new name the user wishes to add to the brand architecture. For one embodiment, the user may review the rules for the proposed location for the name. For one embodiment, the user may be able to

view the list of current names within the selected location, in order to simplify the name selection. For another embodiment, the user may see examples of approved names.

For one embodiment, a secondary questionnaire page may be shown to the user if the user's first proposed name has been found to be non-conforming name. For one embodiment, these questions are always shown to the user, but the user is only specifically prompted to fill in the answers if the name is a non-conforming name.

The user's commitment to the product 560 may include a certain amount of money from a budget to research and/or provide marketing support for the name. Other indications of commitment, such as manpower, advertising budget may also be requested. For one embodiment, the commitment tier 560 is automatically requested for product family brands. For one embodiment, this question is only displayed for lower categories if the name was found to be a non-conforming name and the user wishes to submit the name for extraordinary authorization.

The projected income from the product 570 is similar to the commitment tier. For one embodiment, the projected income is automatically requested for certain tiers, such as product family brands and special ingredients, but is not requested for lower tiers, unless the proposed name is a non-conforming name.

The request for help naming the product 580 may also be displayed. For one embodiment, the request for help is displayed only when the user's initial name is rejected. For another embodiment, the request for help may also be displayed if the proposed name is a non-conforming name and is likely to be rejected, or if a user is attempting to overcome a rejection. For one embodiment, the request for help 580 sends an e-mail or other communication to an outside agency or to a naming group within the corporation, to assist the user in selecting an appropriate name. For one embodiment, this is a discontinuous process, such that the user is disconnected from the brand architecture system, and is separately contacted with naming help.

Figure 5 illustrates one example a questionnaire that may be used to guide a user through the naming process. For an alternative embodiment, the user may make all of the above selections through navigating web pages or computer display screens and the brand architecture chart, and using menus. These types of selections among choices and entry of data may be used, and are well known in the art.

Figure 6A is an example of a brand architecture chart that may be displayed by the present invention. The brand architecture chart of Figure 6A is only a sample chart, implementing sample rules. It is to be understood that the methods used to generate this chart may be used to generate more or less complex charts, and implement dissimilar rules. The format of the chart may also vary from the format shown in Figure 6A. Figure 6A is merely one possible display of one sample chart. Figure 6C illustrates an alternative display for the brand architecture chart. It is to be understood Figures 6A-D illustrate exemplary displays of the brand architecture chart.

This brand architecture chart is an example of a chart that may be viewed, queried, manipulated, and updated by the invention, as described above. The General Mills product line is shown as an example. No affiliation of this patent is meant or implied by using the General Mills product line as an example. All trademarks mentioned in the chart and in the text below are the property of their respective owners.

The first tier 610 is the corporate name 610, which is General Mills.

The second tier 620, are product family names. In this example, product family names 620 include a number of different brands, for example Cheerios cereals, Betty Crocker's Supermoist Cake Mix, Hamburger Helper, Stir N Bake, and Gold Medal Flour. In this example, the product family names 620 may be fanciful names, such as Cheerios, or descriptive names, such as Super Moist Cake Mix. In other examples, a wide variety of other types of names may be allowed.

For a different company, the second tier 620 may not be product family names, but rather, for example, service types, or targeted customers, or another

type of category. The selection of the actual categories for the brand architecture chart is not made by the present system, but rather implemented by a person or persons who are familiar with the brand architecture of the actual corporation.

The third tier 630, in this case, is specific products. Thus, in each product family brand 620 there are a number of specific products 630. Thus, for example, the Stir N Bake product family brand 620 includes Brownies, Carrot Cake, and other products. In this example, the products 630 generally have descriptive names. For some product family brands 620, e.g. Cheerios, the product names 630 each include the product family brand name.

Again, in a different corporation, this tier may have a different name, and different rules, and different types of names or name relationships may be allowed.

The fourth tier 640, in this case, is special ingredients or features. Certain products 630 may further include special ingredients 640. For one embodiment, the special ingredients 640 may be items are named on the package, or otherwise highlighted. As can be seen, the special ingredients 640 in this instance are descriptive names of the actual additional ingredient.

As discussed above, these tiers and the items within each tier are entirely dependent on the actual structure of the corporation. For example, in a service corporation, the categories would be different, as would the types of items within each category. Similarly, the rules implemented at each tier may differ substantially depending on the actual corporation. The above list of products and product organizations has been derived independently based on products currently available for a company. It is not to be taken as an actual representation of the products or brand architecture of the company in question. Furthermore, this does not imply any connection between the patentee and the company.

Figure 6B illustrates a portion of the brand architecture chart of Figure 6A. Figure 6B shows a sample display format for displaying a large brand architecture chart that would generally not fit on a single page. The user can select one of the arrows 650, for example, selecting the Super Moist Cake Mix

product family brand 620. In that instance, the products listed within this product family brand 620 would be displayed. For one embodiment, the higher categories are not shown, and only the current category, i.e. product family brand, product, or special ingredients is displayed. For another embodiment, once a product family brand 620 is selected, the lower and higher tiers associated with the selected product family brand 620 are displayed. Thus, for example, if the user selects the Cheerios Cereal product family brand 620, the display would include the Corporate name, the Cheerios Cereal product family brand 620, and the products 630 (not shown) that belong in the Cheerios Cereal Product Family brand 620. In this way, the user can easily navigate the brand architecture chart, while only seeing the relevant category at any time.

Figure 6C illustrates another alternative method of displaying a brand architecture chart. The brand architecture chart is shown as a circular structure, with the corporate name at the center, product family brands in the first ring, products in the second ring, and special ingredients in the third ring. It is to be understood that other brand architecture charts, not having similar components, could similarly be shown in a circular structure.

Figure 6D illustrates a portion of the brand architecture chart of Figure 6C. Figure 6D shows a sample display format for displaying a large brand architecture chart that would generally not fit on a single page. The user can select one of the arrows 650, for example selecting the Cheerios Cereal product family brand 620. In that instance, the products 630 listed within this product family brand 620 would be displayed. For one embodiment, the higher categories are not shown, and only the current category, i.e. product family brand, product, or special ingredients is displayed. For another embodiment, once a product family brand 620 is selected, the lower and higher tiers associated with the selected product family brand 620 are displayed. Thus, for example, if the user selects the Cheerios Cereal product family brand 620, the display would include the Corporate name, the Cheerios Cereal product family brand 620, and the products 630 that belong in the Cheerios Cereal Product Family brand 620.

In this way, the user can easily navigate the brand architecture chart, while only seeing the relevant category at any time.

It is to be understood that Figures 6A and 6C are exemplary displays of brand architecture charts. Alternative display formats, such as three dimensional, triangular, diamond shaped, or having some other shape suited to the particular brand architecture may be used.

Figure 7 shows a sample set of rules that are implemented in the brand architecture chart of Figures 6A-D. The rules may include naming conventions, such as the form of the proposed name, the syllables in the name, the length of the name, the distinctiveness of the name, and any other rules the corporation may wish to implement. The rules may further include information and/or requirements regarding the use of logos, colors, typefaces, and graphics.

The rules may include such rules as the number of products in a product family at launch, the budget, etc. For one embodiment, certain categories, such as product family and proprietary product names may automatically require the approval of someone. In these instances, the flowchart automatically follows the "name rejected" route, described above.

It is to be understood that these rules are only exemplary rules, and different rules may be implemented. Additional rules may be implemented as well, including such rules as requiring a trademark search for a new name, requiring certain persons to support a new name, etc. Other rules may be specific to the corporation -- for example, a rule may be that new product names must start with the letter T. Any rules that can be imagined may be implemented in the rules for the brand architecture chart. In this manner, the approval of new names and the adding of new names to the brand architecture chart to form a coherent branding strategy may be automated.

In the foregoing specification, the invention has been described with reference to specific exemplary 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 as set forth in the appended claims. Each brand architecture solution will vary more or less from

these examples. Similarly, the display formats and the rules that are implemented vary based on the actual structure of the corporation and the brand strategy of the corporation. The focus of the present invention is the automatic requests, updating, and referrals, as described above. The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense.

CLAIMS

What is claimed is:

1. A computer-implemented brand architecture system comprising:
a hierarchical brand-architecture chart comprising:
a plurality of groups displayed at first tier; and
a plurality of items displayed at a second tier;
an updating facility for updating the navigable brand-architecture chart,
such that a new entry is entered in a proper group of the plurality of groups, and
at a proper tier.
2. The system of claim 1, wherein the plurality of tiers comprise:
a first tier including a plurality of product families;
a second tier including a plurality of individual products, wherein each of
the individual products is categorized in one of the product families.
3. The system of claim 2, wherein the second tier includes a product
group having multiple individual products within the product group.
4. The system of claim 1, further comprising:
a chart configuration unit for controlling the amount of the chart that is
displayed.
5. The system of claim 1, further comprising:
a comparison unit, for receiving a proposed name, and for comparing the
proposed name against a set of rules, and determining if the proposed name fits
within the rules.
6. The system of claim 5, further comprising:
an override facility, the override facility for requesting permission to
override the rules, and permit the proposed name that failed the rules.

7. The system of claim 6, wherein requesting permission comprises automatically sending a request to authorized personnel, and determining whether the response was affirmative or negative.
8. The system of claim 1, further comprising:
an access control unit for limiting access to the brand architecture chart, the access control unit permitting only certain users to alter the chart.
9. The system of claim 8, wherein the access control unit permits certain users to alter the chart, and certain other users to request a new name.
10. A method of implementing an automated brand architecture management system, comprising:
displaying a hierarchical brand-architecture;
receiving a nomination for a new name including information indicative of a position in the hierarchy of the brand-architecture;
comparing the nomination against rules to determine whether the new name fits criteria; and
if the new name fits criteria submitting the nomination for approval and automatically updating the brand-architecture with the new name upon receipt of approval.
11. The method of claim 10, wherein if the new name does not fit the criteria,
determining if there is an exception procedure; and
if there is an exception procedure, implementing the exception procedure
12. The method of claim 11, wherein the exception procedure comprises:

automatically electronically sending the nomination and accompanying information to an authority that can approve the exception to the rules.

13. The method of claim 12, further comprising:
receiving a response from the authority; and
if the response is affirmative, automatically updating the brand architecture chart, and informing a nominator of the affirmative response.

14. The method of claim 13, further comprising:
if the response is negative, informing a nominator of the negative response, and permitting the nominator to nominate a new proposed name for the product.

15. The method of claim 13, further comprising:
if the response is negative, informing a nominator of the negative response and permitting the nominator to request help generating a proposed name that would result in a positive response.

16. The method of claim 13, wherein the authority is a plurality of persons, and wherein the response comprises a plurality of responses, and an affirmative response comprises affirmative responses from each of the plurality of persons.

17. The method of claim 11, wherein the exception procedure comprises:
requiring a commitment from the nominating group to support the new name.

18. The method of claim 10, wherein the rules comprise a limitation on a type of name being used.

19. The method of claim 10, wherein the step of receiving a nomination for a new name including a location in the brand-architecture chart, comprises:

receiving a filled-in questionnaire, indicating a tier and a category, and a proposed name.

20. The method of claim 10, wherein the step of receiving a nomination for a new name comprises:

prompting a user to select a product family, after reviewing the entire brand chart;

prompting the user to select a location, after reviewing the entire product family; and

prompting the user to enter the nomination for a proposed product name.

21. The method of claim 19, wherein after the user selects a product family and a product tier, the rules for naming within that product family and product tier are displayed to the user, prior to allowing the user to enter the nomination.

22. The method of claim 21, further comprising indicating to the user that the name may be outside of the rules, if certain conditions are met.

23. A method of maintaining a brand architecture management system to simplify maintaining coherent branding strategy, the method comprising:

defining a brand architecture chart including current and proposed names;

defining a rule set for the brand architecture chart to maintain the coherent branding strategy;

incorporating the rule set into an automatic nomination system that receives new name nominations from users; and

prompting a user to compare a new name nomination against the rule set, and accepting the new name nomination if the new name fits within the rule set.

24. The method of claim 23, wherein if the new name nomination does not fit within the rule set, the method further comprises:

 permitting the user to request help in selecting a different name; and
 permitting the user to appeal the decision to an authority that may override the rule set.

25. The method of claim 23, further comprising automatically adding the new name to the brand architecture chart if the new name is accepted.

26. An automated brand architecture management system comprising:
a rule storage unit for storing pre-determined rules;
a display mechanism for displaying a hierarchical brand-architecture;
a user interface for receiving a nomination for a new name from a user, the nomination including information indicative of a position in the hierarchy of the brand-architecture;

 a comparison unit for comparing the nomination against the pre-determined rules from the rule storage unit, to determine whether the new name fits criteria and submitting the nomination for approval if the new name meets the pre-determined rules; and

 a chart updating facility for automatically updating the brand-architecture with the new name upon receipt of approval.

27. The automated brand architecture management system of claim 26, further comprising, if the new name does not fit the criteria:

 an override facility for implementing an exception procedure.

28. The automated brand architecture management system of claim 27, wherein the exception procedure comprises the override facility automatically

electronically send's the nomination and accompanying information to an authority who can approve the exception to the rules, if the new name.

29. The automated brand architecture management system of claim 28, further comprising:

the override facility for receiving a response from the authority; and
if the response is affirmative,

a chart updating facility for automatically updating the brand architecture chart, and informing a nominator of the affirmative response.

30. The automated brand architecture management system of claim 29, further comprising:

if the response is negative,

the user interface for informing a nominator of the negative response, and permitting the nominator to nominate a new proposed name for the product.

31. The automated brand architecture management system of claim 29, further comprising:

if the response is negative,

the user interface for informing a nominator of the negative response; and
an access control unit for permitting the nominator to request help generating a proposed name that would result in a positive response.

32. The automated brand architecture management system of claim 28, wherein the authority is a plurality of persons, and wherein the response comprises a plurality of responses, and an affirmative response comprises affirmative responses from each of the plurality of persons.

33. The automated brand architecture management system of claim 26, wherein the exception procedure comprises:

the override facility requiring a commitment from the nominating group to support the new name.

34. The automated brand architecture management system of claim 26, wherein the rules comprise a limitation on a type of name being used.

35. The automated brand architecture management system of claim 34, wherein after the user selects a tier, the user interface displays rules for naming within that tier, prior to allowing the user to enter the nomination.

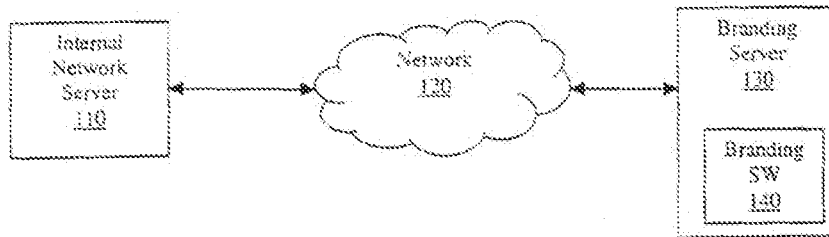


Fig. 1

2,10

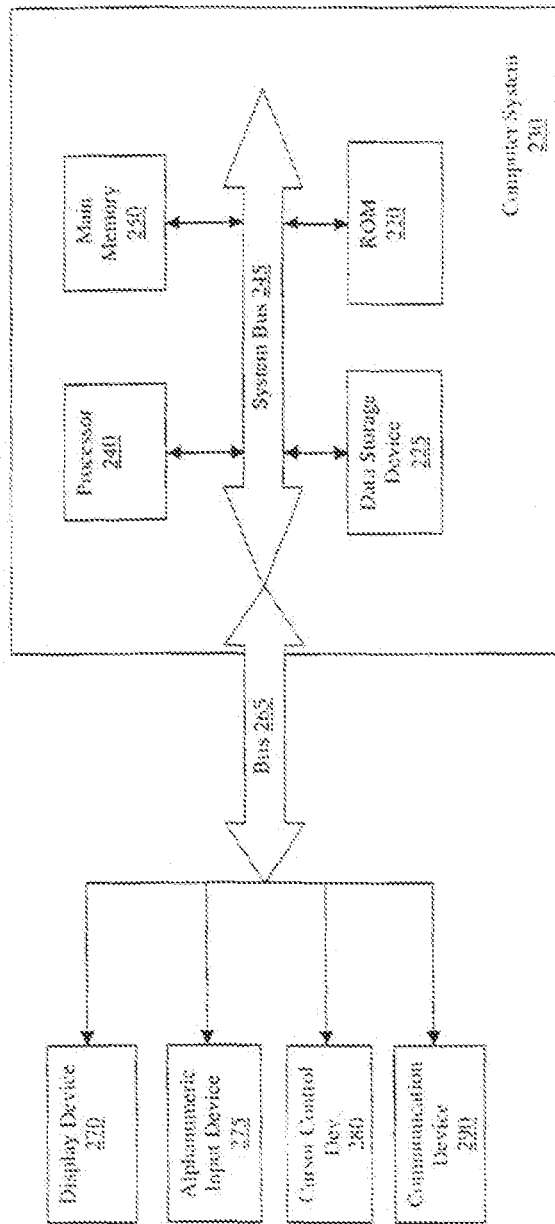


Fig. 2

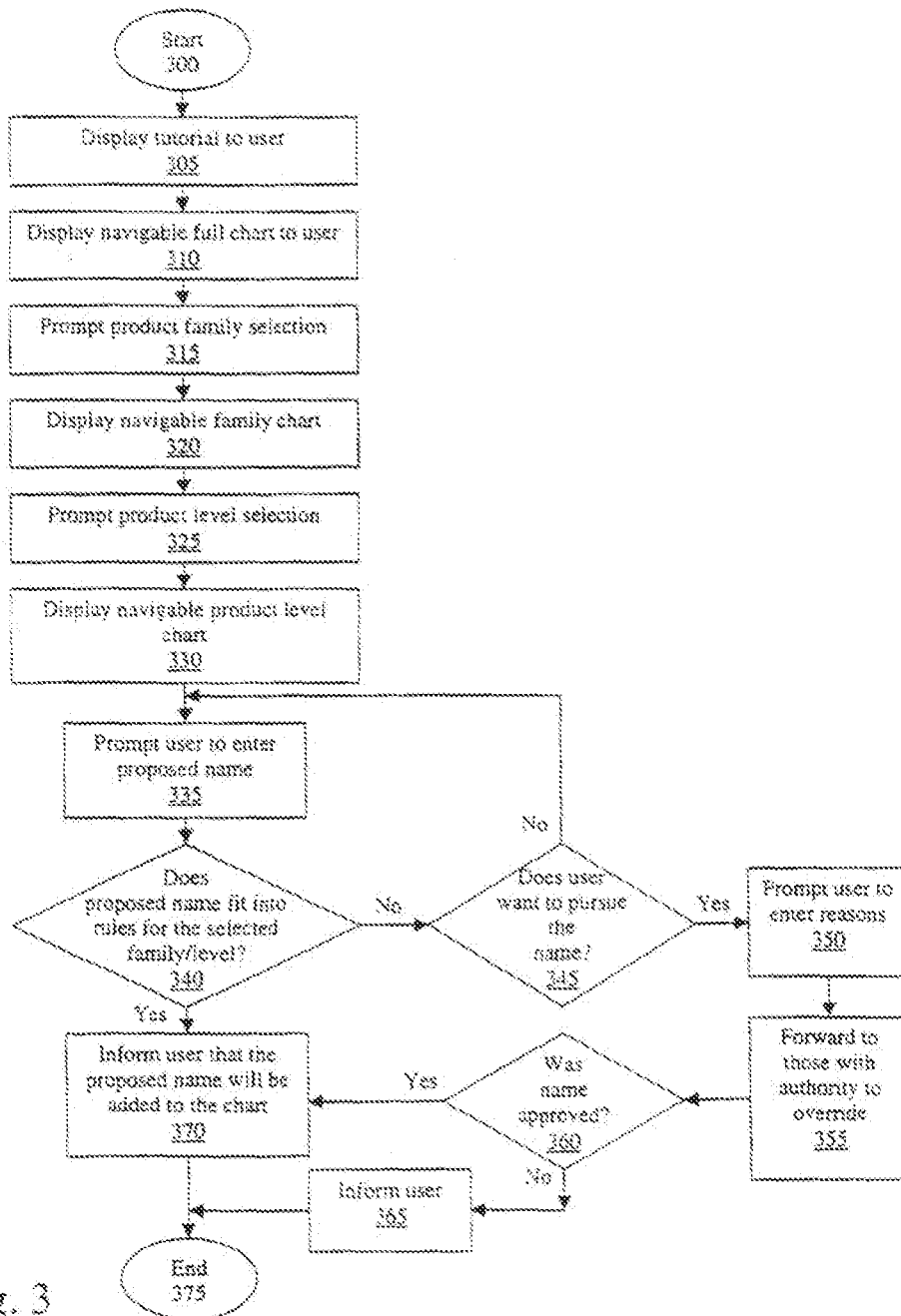


Fig. 3

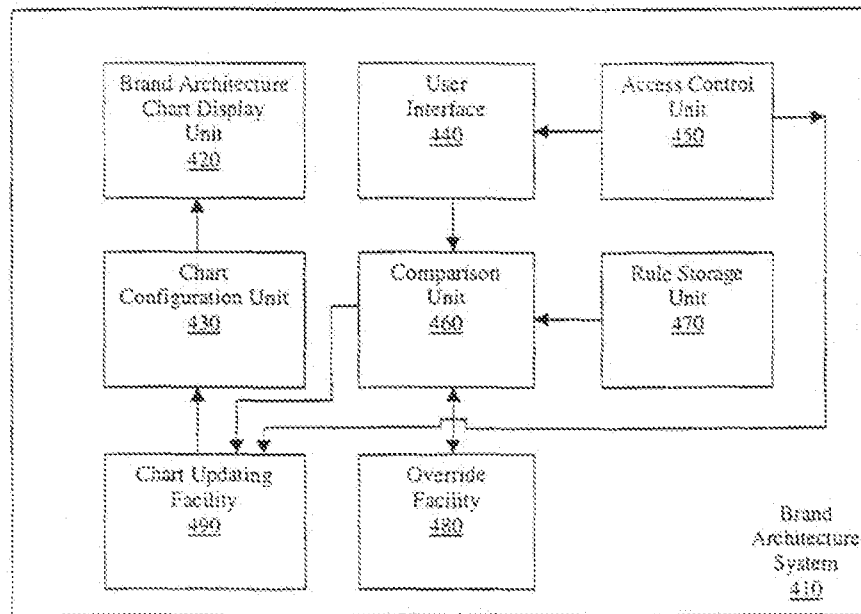


Fig. 4

510	Name	
	Title	
	Group	
	Contact information	
520 *	First Level	Brand/Product or Ingredient?
530 *	Second Level	Within what Brand/Product/or Ingredient list?
540 *	Definition of Product	
550 *	Proposed Product Name	
560 *	Proposed Support level for product	
570 *	Projected income from Product	
580 *	Does user request help with naming?	

Fig. 5

GENERAL MILLS			
Corporate Name Product Family Brands	Cheerios Cereal	Betty Crocker	SHR N Bake Instant Cake Mix
Products	Super Moist Cake Mix	Hamburger Helper Main Meal Dish	All Purpose Organic All Purpose Softasilk Unbleached Whole Wheat Wondra
	Carrot Chocolate Fudge Devils Food German Chocolate Lemon Marble Sour Cream White Strawberry White Yellow	Beef Pasta Cheddar Cheese Melt Cheddar N Bacon Cheesy Shells Chili Macaroni Creamy Stroganoff Fettuccine Alfredo Homestyle Beef Stew Italian Herb Lasagna Meatloaf Pizza Pasta Pasta Stroganoff Ravioli Rice Oriental Salsbury Spaghetti Zesty Mexican	
Special Ingredients	Apple Cinnamon Cheerios Frosted Cheerios Heavy Nut Cheerios Multi Grain Cheerios Plus Team Cheerios	With Hershey's Fudge	Better for Better for Biscuits Better for Bread Unbleached Better for Bread Wheat Blend
			With Hershey's Mini Kisses

FIGURE 6A

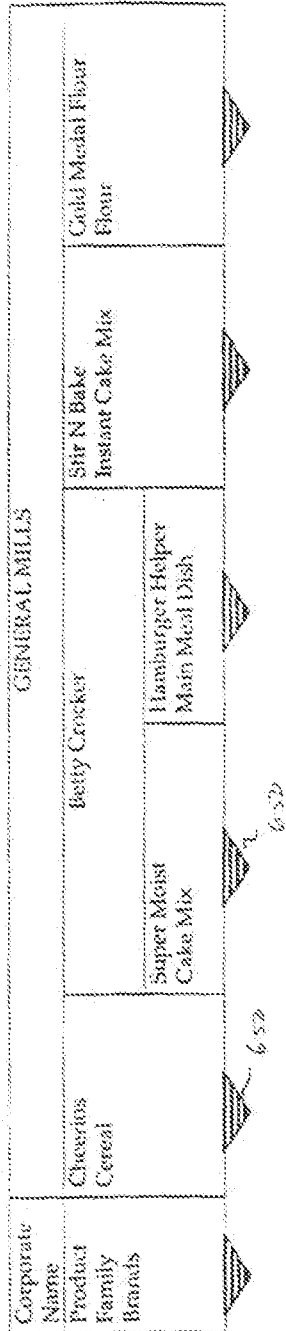


FIGURE 6B

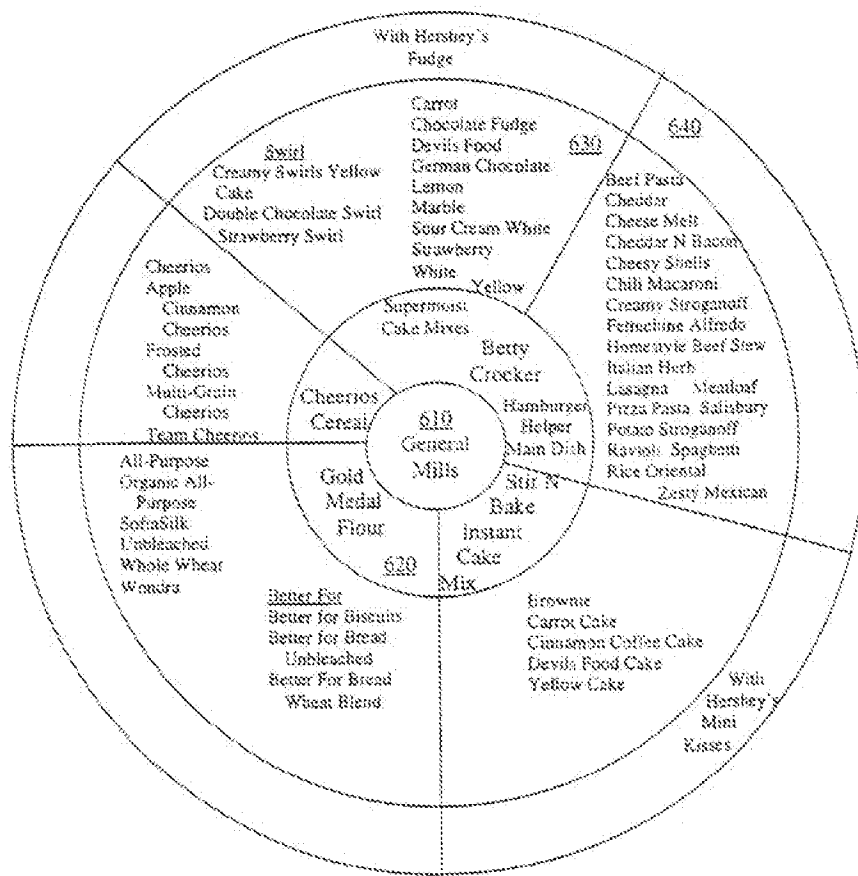


Fig. 6C

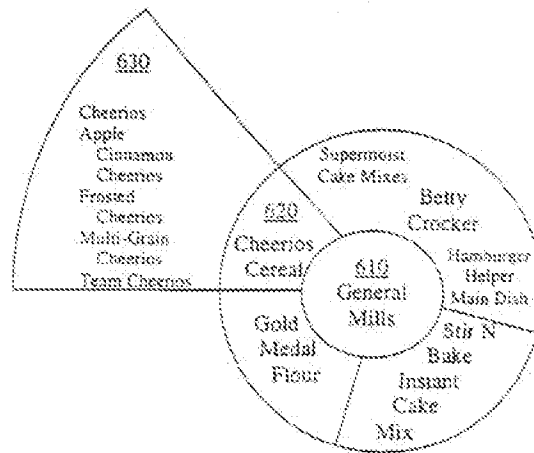


Fig. 6D

10/10

Sample Rules for The Brand Architecture Chart of Figure 6A

Product Family Brands

- Multiple products at launch
- Minimum \$50 million launch budget in year 1
- Approval: Executive VP of Marketing and CEO

Products

- Appropriate for flavors, form factors, or other extensions of Product Family brands
- Descriptive terminology only
- Approval: automatic if descriptive terminology employed
- May only consider proprietary (trademarked) sub-brand name (e.g., Softastik) if
 - Minimum \$20 million launch budget in year 1
 - Potential to line extend sub-brand, such that it could eventually grow into a Product Family brand
 - Approval: Executive VP of Marketing

Special Ingredients

- Must employ the following format: "with [ingredient]"
- Use of ingredient name must be expected to add at least 30% to sales of relevant Product Family brand
- Ingredient must not be greater in prominence than General Mills or Product Family brand, as measured by annual advertising budgets
- Approval: Executive VP of Marketing

Figure 7

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
14 June 2001 (14.06.2001)

PCT

(10) International Publication Number
WO 01/42873 A2

- (51) International Patent Classification: G06F
- (21) International Application Number: PCT/US00/33328
- (22) International Filing Date: 7 December 2000 (07.12.2000)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
09/458,621 10 December 1999 (10.12.1999) US
- (71) Applicant: TOUCHPAK, INC [US/US]; 4th Floor, 609 Greenwich Street, New York, NY 10014 (US).
- (72) Inventor: CHEN, Chia-Wei; Apartment #1, 110 Jefferson Street, Hoboken, NJ 07030 (US).
- (74) Agent: FLYNN, Kevin, E., Daniel & Daniels, P.A., P.C. Drawer 12218, Research Triangle Park, NC 27709 (US).

(81) Designated States (national): AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, 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, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MY, NG, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YD, 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 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).

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: METHOD AND SYSTEM FOR ELECTRONIC DISTRIBUTION AND COLLECTION OF SURVEY INFORMATION

(57) Abstract: A method and apparatus for providing a flexible means to distribute electronic surveys (3000) (including both traditional information gathering events typified by marketing surveys, political polling surveys, and sessions designed to distribute rather than collect information such as distribution of promotional material or general information) to various queuing locations (3400) such as retail establishments with waiting patrons; efficiently present the information contained in the survey (3000); and collect information from respondents (3550) that meet the qualification criteria established by the marketer (3500). The invention allows for imposition of qualifications not only on the respondent (3500), but on the type of queuing location (3000) qualified to distribute the survey (3000), and even the time, place, and manner of survey presentation conditions (3026). The invention allows for a variety of incentives to be offered to the respondent (3550) including incentives for completing a survey (3000) that can be used and enjoyed during the same visit to the queuing location (3400). The use of information accumulated in a respondent profile, cross connections to a Point of Sale computer system (3480), short-circuiting the qualification process based on information available or implied about the respondent (3550), and alterations to the survey presentation to adjust to information known about the respondent (3550) makes the survey process more efficient and less tedious to the respondent (3550). The preferred mode uses mobile survey devices (3450) in communication directly or indirectly with a Central Computer System (3600) remote from the queuing location (3400).

WO 01/42873 A2

METHOD AND SYSTEM FOR ELECTRONIC DISTRIBUTION AND COLLECTION OF SURVEY INFORMATION

BACKGROUND ART

The invention relates to the electronic distribution of information in the field of direct marketing typified by surveys, advertisements and other promotions in retail establishments. The invention also provides a means for recording data regarding an individual person across interactions that specific person has with the Digital Survey System. The invention also allows for: protection of an individual's privacy and anonymity; selective presentation of choices of surveys, advertisements and other information; customization of the surveys, advertisements and other information. These can be accomplished based on the individual demographic, locale demographic, and other relevant information about the target audience and environment.

DISCLOSURE OF THE INVENTION

Advertising in previous years has become a rapidly growing, well-developed field. Whether implicit or not, all advertisers have centered on capturing the attention of the consumer. However, an advertisement that interrupts a form of entertainment tends to be regarded as an unwanted obstruction to one's enjoyment of a television show or a song on the radio. To overcome consumer reticence and capture their open attention, advertisers are by no means lacking in schemes. They readily rely upon entertainment principles based on demographics of particular target groups to win the consumer's interest. Many advertisers have imbedded their advertisement in the entertainment content itself, so as to make the advertisement part of the entertainment rather than an interruption of the entertainment. Moreover, in more recent years many advertisers have effectively won the interest of the consumers by giving the consumers a greater incentive to read their ads; that is, offering direct compensation to the consumer to read their ads. For example, Cybergold, an internet marketer, grants credit to consumers to read ads demographically suited to the consumer's needs and interests.

The present invention relies primarily on two principles to attract the attention of the consumer: (a) waiting for a service is often boring to consumers and (b) the consumers are more likely to fill out a survey during this otherwise unused time than the consumer would be when confronted with a survey at home or while walking in a shopping mall. Hence, the present invention targets consumers that must wait for a service; such as those found in the restaurants. Under this scenario, restaurants will be able to provide a portable computer to their waiting consumers that contains various entertaining programs geared towards one or more age groups. Therefore, both the consumer and the restaurant are able to benefit from this service. The consumer is able to wait without becoming bored or irritated or worse yet leaving when the wait seems too long. The

restaurant is able to benefit from an increase of sales as the consumers patronizing that particular restaurant will find the services more enjoyable. Moreover, the advertisers are also able to benefit from this process. For inextricably embedded in this entertainment service are advertisements demographically geared towards the restaurant's clientele. As an incentive to read the ads or fill out short surveys, the advertisers will offer incentives such as coupons to the consumer that will reduce their bill or perhaps give them a free drink or appetizer. Through the present invention, the survey information can be transferred electronically to the marketer, and the restaurant can be compensated for the rebate (for example, a free appetizer).

The primary tool of effective marketing is current and accurate information. The ability to know more about the consumers and their demographics is what allows the marketers to tailor their message for their audience. The more they know about their audience, the more persuasive they become. To gain this information, the marketers use promotions and incentives to persuade consumers to cooperate. They offer rebates, coupons, gifts, prizes and even cash.

For the purposes of this document and the claims that follow, the present invention will be referred to as the Digital Survey System.

For the purposes of this document and the claims that follow, a "survey" will be taken to mean any survey, advertisement or promotion, including but not limited to (a) information requested from consumers such as customer response cards, market research surveys, polls and academic research studies, and (b) information supplied to consumers such as promotional offers of services or goods, advertisements and informational/promotional articles.

A "marketer" will be taken to mean any entity requesting responses from the public including but not limited to pollsters, academics doing research companies performing, traditional market research surveys, advertisers or marketers. For the purposes of this document and the claims that follow, the delegation of certain marketing acts of a company to an advertising agency can be ignored and all of the acts relating to a survey from the decisions within a company to the actions of the marketing professionals (both in-house and external) can be lumped together.

For the purposes of this document and in the claims that follow, a "credit" will be taken to mean any type or form of compensation offered to a survey respondent in exchange for using the system. A credit includes but is not limited to immediate discounts or rebates on purchases at the locale, the selection of a service such as a jukebox song, the provision of a promotional item such as a complimentary pair of sunglasses, and immediate discounts on purchases of items made electronically such as the purchase of a book through the Digital Survey System.

Surveying consists of three fundamental activities: getting the surveys to the right people, getting the people to fill them out, and gathering all of the survey information to be analyzed. The innovations presented by the invention address these three areas.

A) Targeting Survey Respondents Based on Various Demographics.

Misleading information is worse than no information at all. Marketers want to have information about certain types of people so that they can target those types of people more effectively. The present invention allows the marketer to target specific audiences based on the combination of one or more of the following: survey respondent demographics, locale demographics and Time, Place, and Manner demographics.

B) Survey Respondent Demographics

Survey respondent demographics refer to information known about a specific survey respondent. With traditional survey methods, this is often little or no information. These qualifications include any parameter that could be used to separate a group of people into two or more groups. Examples start with gender, age, marital status, income level, home ownership status, hobbies, education level, favorite magazines, favorite television shows, amount that is spent annually on buying computer equipment, average monthly phone bill, whether the person currently has a particular medical condition, or whether the person intends to purchase a certain type of item in the next three months; and whether one has viewed a particular movie.

Possible qualifications might include attributes of one's children, attributes of one's spouse, attributes of one's parents (for example -- the medical condition of one's elderly parent), attributes of one's job responsibilities -- do you have authority to select computer printers for use in your work area; attributes of one's pet -- does your dog have fleas?

The reasons for using the various qualification criteria vary between marketers. Examples include:

Some marketers may want to find a survey applicant that has the general characteristics of one particular demographic segment and then attempt to gather information about how to most effectively reach that segment;

Some marketers may seek to find people that have recently purchased a specific item and then find out how the person feels about the quality, price, and support that came from the product;

Some marketers may be trying to see if executives that are subscribers to particular magazine have noticed an advertising campaign that has been running.

Some marketers may be political pollsters who are looking for information from groups that are difficult to survey using telephone surveys;

Some marketers may be academics or researchers who are seeking answers about drug use or sexual practices in young adults and who feel that this format is more likely to provide truthful answers than a face-to-face interview;

Some marketers may be seeking to record reactions to people who are waiting out a lengthy delay

at an airport in order to gage the impact of the delay on travelers and on their patience with additional stressors. To these marketers, the qualification may be that the person flies more than 50,000 miles a year and has been traveling today for more than 5 hours.

5 The marketers might deem several different litmus tests as suitable to satisfy a particular condition. Thus a marketer for minivans might seek someone with several children, or someone with two dogs, or someone who regularly shops for antique furniture as candidates for a minivan.

C) Locale Demographics

10 Locale demographics refer to information known about the consumers based on the type of locale (sports bar, comedy show, family restaurant), the area of town, nearby attractions. These demographics might include information about the town or area of town such as residences, average level of education, average income of residents, average income of commuters into the location which can be represented and collected from such sources as the visitor's bureau, local and state chambers of commerce. An example of using a locale demographics qualification is to limit the distribution of a snow tire survey to locales that average more than two feet of snowfall per winter. This use of a locale demographic differs from a Time, Place, and Manner demographic in that the snowfall limitation is one of many limitations that apply to a place on a map -- a location. It does not specify whether the survey is given at a bus station, a hair styling saloon, at a bowling alley, or at a jazz club after midnight. The locale demographics do not check the local temperature or the amount of precipitation that fell in the last three days. The fine tuning required for such questions would come from a Time Place and Manner set of qualifications.

D) Time, Place, and Manner Demographics

25 Time, Place, and Manner demographics refer to information inferred about consumers based on conditions such as time of day, days with baseball games, festival days or other special events. Time, Place, and Manner restrictions can be useful in several ways. Time, Place, and Manner demographics can provide an inference about the likes and dislikes of the people presented with the survey so as to reduce the need to ask qualifying questions. Time, Place, and Manner can also be used as a rough filter to reduce the likelihood that surveys on controversial or "adult" topics will be presented to someone who will be offended. The risk may not be reduced to zero, but the survey system may want to reduce offended respondents since many respondents are also customers of the queuing locations that are retail establishments. For example, a survey that was limited to retail establishments that have a bar is less likely to be presented to someone who has a religious objection against alcohol. A survey about the effectiveness of various advertisements on the importance of using condoms might be allowed within a particular queuing location. An option to reduce the likelihood that a child would see the ads on a survey device being used by an appropriate respondent is to limit the survey to after 11 p.m. The limitation could be placed by

the marketer or by the queuing location. If placed by the queuing location then it would be a second Time, Place, and Manner qualification file for that survey, one for the queuing location and one for the marketer. As a practical matter both the Time, Place, and Manner qualification sets would both have to be satisfied so that the marketer's Time, Place, and Manner qualifications are not bypassed. To the extent that the queuing location Time, Place, and Manner qualifications cannot be met while at the same time meeting the marketer's Time, Place, and Manner qualifications, then the survey becomes unavailable. The queuing location would not have an incentive to remove surveys it had decided to allow from availability. When the response rate for the survey burdened with mutually exclusive Time, Place, and Manner qualification files stays at zero, then someone should notice and remedy the situation. This particular conflict at one queuing location is not a particular problem for the marketer since other queuing locations will be providing responses.

Another advantage of Time, Place, and Manner qualifications to limit the survey distribution is to find respondents with an active interest in a topic rather than a passing or mild interest. A survey on music preferences may find many qualified respondents at a music store. A marketer wanting information about preferences for tanning products among people engaged in water sports may want to limit the survey to people who respond at a facility with an outdoor pool and then may want to further limit the response to a survey given in December to those states with warm weather at that time of year.

E) Use Time, Place, and Manner Information to Assist Qualification of Respondent

Process of qualification may be assisted, reinforced, or streamlined by using qualifications about the valid time place and manner of providing the survey. Since there is no point in asking everyone if they are waiting for a plane since the only people likely to be waiting for a plane are people at an airport, the requirement that a particular survey be given only at an airport and at various facilities within an airport will streamline that process.

Depending on the amount of confidence the sponsoring marketer wants in the answer, the Time, Place, and Manner qualification can be used in addition to rather in place of a respondent qualification question on interest in a given topic. Returning to the airport delay survey, limiting the distribution to an airport and then asking if the respondent is waiting for a plane provides an additional filter against responses from people at a shopping mall or a restaurant who answer a question incorrectly as a joke or respond with any answer that they believe will lead to them obtaining the specific survey premium.

As another example, a marketer wanting a survey set from those people likely to spend money on jazz recordings may chose to use a Time, Place, and Manner restriction that limits offering the music survey to night clubs shortly before and during a jazz performance. Note that the Time, Place, and Manner qualifications may require a two-step screening. The first step limits

distribution of the surveys to queuing locations with the potential to meet the Time, Place, and Manner criteria in addition to the locale criteria. For example, the nightclub might have a location profile that indicates that it has performances on a regular basis from several categories of entertainment on the profile qualification questionnaire. One of the indicated entertainment types would have been jazz in this example. The Time, Place, and Manner profile is updated more frequently, from daily to several times a day. When the Time, Place, and Manner profile was updated to show a Jazz performance starting at 10:00 p.m. the intelligence sent with the survey would be able to close a qualification switch after 9 p.m. for the Time, Place, and Manner qualification that the survey be given during a Jazz performance or within an hour of the start time of a Jazz performance.

Time, Place, and Manner qualifications could be any measurable feature that is useful for the marketer to target the audience and environment that the marketer seeks. Because the marketer wants responses and wants to rely on the accuracy of the Time, Place, and Manner status files of various queuing locations, the marketer is discouraged from seeking bizarre or difficult to monitor parameters unless these are particularly important to the survey. Thus a Time, Place, and Manner parameter that the bulk of the current set of patrons be dressed casually, would be hard to judge and would be difficult to keep current in the Time, Place, and Manner file. It is unlikely that a queuing location would run over and update that Time, Place, and Manner file when a group of lawyers walks into the queuing location with dark suits on.

Likewise the queuing location does have an incentive to enable as many surveys as possible so that its patrons are provided with surveys that match their interests. In many instances the queuing location benefits financially from completed surveys in that the survey system can provide a credit that the survey respondent/queuing location patron may use on that visit to pay for goods or services. This infusion of payments to benefit the patron, plus any additional handling fee, results in another source of revenue for the queuing location.

Thus a night club is likely to provide the queuing location locale profile information and then update the type of performance for a given night into a Time, Place, and Manner status file. The update can be done before the queuing location becomes busy. This one update might result in several dozen patrons answering jazz surveys. Failure to update may mean that the jazz audience is asked about wedding dress preferences since the night club hosted a wedding fashion show the night before.

Some pieces of information could logically be characterized as either locale demographic or Time, Place, and Manner demographic information. For example, does one fill out a separate locale demographic profile for the frequent travelers that wait in the member only airline lounges from the general locale of the airport? Or does one pick up that information as part of the Time Place and Manner profile for one part of the airport. It really does not matter whether they are

ultimately characterized as locale or Time, Place, and Manner. In fact with object-oriented programming, the information may not be stored in separate "files" in the traditional sense of the word file. Although the queue of people waiting for a taxi at the airport will have some of the same people that could be found at some time in the frequent flyer lounges, the overall population demographics of the two groups will differ—it is important the information useful for efficiently targeting demographic pools be made available to the Digital Survey System, not what the information happens to be called in the abstract.

BRIEF DESCRIPTION OF DRAWINGS

10 The following figures illustrate the function and form of the invention. The first seven figures describe the invention as an object-oriented design. The Figures 8 through 12 describe the invention as a structured design. The drawings are summarized as:

FIG. 1 is an analogy that is useful for envisioning one layer of qualification conditions;

15 FIG. 2 is a flowchart for the various activities related to a particular survey, including: the creation, distribution, completion of the survey, and the post completion activities to provide the promised incentives to the participants and the collection of surveys to the requesting Marketer.

FIG. 3 is a series of diagrams showing the interaction of elements to accomplish various steps in Fig. 2;

20 FIG. 4 shows the overall interaction of the various software components that encompass the invention;

FIG. 5 shows an example of the hardware components that may be used to implement the invention;

FIG. 6 shows an example of the administration of a survey;

FIG. 7 shows an example of palm manager synchronization;

25 FIG. 8 shows an example of credit authorization;

FIG. 9 shows an example of collector synchronization;

FIG. 10 shows an example of survey delivery;

FIG. 11 shows an example of survey distribution;

FIG. 12 shows an example of survey collection;

30 FIG. 13 shows an example of survey selection;

FIG. 14 shows an example of survey acquisition; and

FIG. 15 shows an example of survey respondent lookup.

DETAILED DESCRIPTION OF THE DRAWINGS

A) Short Circuiting the Qualification Process

FIG. 1 is an analogy that is useful for envisioning one layer of qualification conditions. As an analogy, Figure 1 does not represent any particular circuit that must be present in an embodiment of the present invention. Turning now to Figure 1, the set of qualifications for a target audience for a particular survey can be represented by a set of switches 110 in circuit 100. This particular set of qualification conditions is more complicated than most in that there are eight different ways that a person could qualify as indicated by the possible paths between side 102 and side 104 of circuit 100. Each switch represents a binary condition for each of the various qualifying questions (income over 50k, gender is female, gender is male, has a child under the age of 3, operates an in-home day care facility, respondent works outside the home, respondent shops at warehouse stores, and so forth). If a particular combination of desired answers effectively closes all the switches along one or more paths, then the respondent is qualified to answer this detailed survey designed to help the marketers convince working moms to drive to a warehouse store to stock up on certain disposable diapers.

Note that the goal for a marketer is usually going to be to obtain timely information to use in some analysis conducted by the marketer or to document a trend in opinion in a certain group over time. In either case, the marketer wants to obtain information quickly while limiting the results to qualified respondents taking the survey under suitable conditions. For example, the marketer may want a quick sample of 100 qualified people to see if a proposed marketing strategy is likely to work so the marketer can obtain funding for a larger more detailed study. If taken to extreme, a large number of qualifying questions would cause a potentially qualifying respondent to stop filling out the survey before finishing both the qualifying question and the survey questions.

Thus to the extent possible, the survey should quickly assess if the respondent is qualified. Thus the question of whether the respondent shops at warehouse stores is unnecessary when the survey is taken at a warehouse store. This particular qualifying switch can be set to "closed" just based on the location where the question is being asked. More specifically, the survey reads information stored about the retail location and answers what questions can be answered from that information.

If information previously accumulated from the respondent is available, other qualifying information can be filled in without taking time to ask the respondent. For example, a user profile filled out by the respondent could be used to get the gender, age, and income information. Information stored from respondent responses to other surveys (either qualification questions or survey questions) can be accessed to fill in additional question switches.

In addition to a profile created for use in the answering of surveys, the warehouse store may have

its own information about the respondent in its Point of Sale ("POS") computer system. The information may include questions such as the gender, and the occupation of the respondent (for example--patent attorney). For example the answers to a membership card application for the warehouse store may already reveal that the respondent is a patent attorney. This information
5 about the respondent's occupation may be used to infer that the respondent is not a provider of in-home day care. Acting upon this information, the logic accompanying the survey would set the answer to "no" without asking the respondent whether the respondent is a provider of in-home day care. Since it is only unlikely and not impossible for someone to be an in-home day care provider during the day and a patent attorney at night, making this inference raises the risk that a
10 qualified respondent that owns a day care center is being prematurely disqualified because the respondent works as a patent attorney at night. The marketer may feel the tradeoff of making the qualification process go faster merits some small risk of missing a qualified respondent.

The respondent is left to answer several qualification questions that are closely connected to this particular survey such as whether she lives within a half hour drive to a warehouse store. Since
15 these questions asked do not require repetition of information provided before, the respondent is not annoyed by having to answer the several remaining qualification questions.

When the marketer desires to only target a certain survey respondent demographic, the present invention will only offer a survey respondent the opportunity to participate in the survey after verifying that the consumer matches the survey respondent demographic. Whether the survey
20 respondent matches the target audience may be ascertained from information the survey respondent volunteers.

The marketer may also target a certain class of locales. This can be accomplished by maintaining a database of locale profiles and only distributing the surveys to locales that match the target. The type and class of business, the geographic location, or the typical clientele can specify the
25 type of locale.

Under the Time, Place, and Manner demographics, the marketer may also target a specific time period. Most of us are aware of a restaurant, movie theater, or other location that has a different distribution of people at one time of day from another time of day. The movie matinee crowd is a different crowd from the midnight movie crowd. The dining crowd during the time period for the
30 "early bird dinner special" is often a different crowd from those who come to dine at prime time and pay full price. The look and feel of an airport is different on a Saturday morning than it is on a Friday afternoon. Whether this is a specific time of day, week, month or year, the invention can ensure that only the target audience interacts with the survey.

B) Cross sectional surveys

35 The Digital Survey System has the additional capability to create an explicit distribution survey across a cross-section of survey respondent demographics, locale demographics and Time, Place,

and Manner demographics. In the prior art, surveys could be sent to a number of homes and then the responses may be categorized later. However, it is difficult to accurately predict how many mailed surveys will arrive at each the homes of each demographic group without spending a good deal of time picking the survey audience beforehand. Even if a balanced survey distribution is sent out, it is highly unlikely that the survey response will be either 100% or if less than 100% that the survey response rate will be the same for all demographic groups. Some groups tend to be less trusting of anyone asking a great number of questions. Some groups do not believe that any response to a survey sent to their home address could ever be anonymous. Some groups tend to have less time to answer surveys.

The Digital Survey System will allow the marketer to specify a survey that is to be distributed to a variety of profiles with preference given to certain demographics. For instance, a pollster may wish to poll differences in political views between races and economic classes. However, certain demographics may be more desirable for the survey because they often are under represented in traditional polling methods or they are simply difficult to locate. For example, people that have been homeless in the last 10 years, parents of pregnant teenagers, great grandchildren of European immigrants, spouses of avid golf players, people interested in traveling to Greece, bachelors who cook more than four times a week. Therefore, a larger credit may be used as an incentive for people who fit that desirable demographic, or the number of surveys for other demographics may be limited. (A more detailed description of the process of awarding incentive credits is provided below). This information can be encoded into the survey on the survey device and controlled from the information servers. When the Digital Survey System has received the requested number of surveys for the less desired (less rare) demographics it can continue to seek survey response but only from the types of respondents needed to complete the desired distribution.

C) The Process

The preferred implementation of the present invention is to use a top level computer at one or more locations to handle the distribution and processing of surveys.

The top level computer will need the capacity to send, receive, and in some cases handle a variety of data formats. The data formats that are most likely to arise include the formats for audio, images, video, and various computer processing languages including those that operate in an internet browser environment such as JAVA, C++, Visual Basic and others.

D) It Starts With a Survey

The top level computer needs to be able to receive surveys that are in a form suitable for transmission in a digital means. Transmission in a digital means would cover the preferred mode of transmission across the Internet. The transmission could be along other channels such as a digital telephone connection if there is sufficient bandwidth to transmit the necessary information.

The computer could re-use surveys that were used at one time and then stored for subsequent use. It is likely that a mechanism would be used to modify previously used surveys to update the survey questions. These modified surveys would then be made available to the queuing locations. Although less convenient, the surveys could be provided on a machine-readable media such as a computer disc, or a Compact Disc. As an additional service, the surveys could be converted from another format, including a paper survey. The survey after conversion would be made available for use in the top level computer.

E) Qualifications

As discussed throughout this specification, the surveys will have the parameters that the marketer wishes to use as a set of qualifications such as the set represented in Figure 1. The qualifications cover, respondent qualifications, locale qualifications, and Time, Place, and Manner qualifications. Other layers of qualifications may be added on such as Time, Place, and Manner qualifications required by the queuing location. These qualifications provide levels prerequisites for selecting a given end-user as an appropriate participant in a survey.

As discussed above, marketers may be most interested in finding people within certain hard-to-find categories but also interested in getting responses from one or more easier to find categories to use as a control group. These marketers might offer the same survey to both the hard-to-find and the easier find categories of respondent. However, the marketer may offer a more valuable incentive or "premium" to the hard to find category of respondent. Since there are two different premium levels with different qualifications, these are now effectively two different surveys.

Some of the benefits of the Digital Survey System arise from the use of surveys that use logic associated with the survey or survey system to tailor the survey to the known information about the respondent to reduce the number of redundant or silly questions (asking males if they are pregnant) and to otherwise expedite the survey or to resolve apparent conflicts between two pieces of information provided by the respondent.

The survey anticipated for use in the present invention is one that has onboard intelligence although other static surveys can be used by the present invention. Thus for the intelligent surveys (sometimes called dynamic surveys), the set of questions presented to the end-user is not the same for all end-users. Some surveys may use some questions that sample the qualification data and then discontinue this line of questioning if the sampled questions match the qualification criteria. The survey may terminate with a polite explanation that the person was provided the survey in error and ask some questions to see what was the source of the error. The survey may take information stored in a user profile or a session profile and use that information to answer one or more questions without asking the end-user. The survey may take session information or user information and then ask the user for confirmation or permission to use that information as the response to this survey.

The preferred embodiment of the present invention would use information provided at the qualification stage to alter or tailor the questions presented to the survey respondent. The alteration could include adding or deleting questions from the core set of survey questions. For example, if a survey respondent qualifies to take the minivan survey based on having several
5 children, the survey may ask additional questions such as the number of children in car seats, the approximate height of the children (if this is relevant to legroom questions). The same survey would not ask these same questions about children to another respondent that is without children but has a hobby of buying antique furniture at auctions. This antique collector might be asked questions on specific topics such as the use of tie-downs to secure loads or the use of a trailer
10 hitch.

The survey may simply instantiate the names of the respondent and the people or things that are significant to the respondent if the names have been given. So the questions about pet food might insert the name of a respondent's pet dog be it "Bella", "Batdog", or "Fleabag".

To target survey respondent demographics, the present invention must gather information about
15 the survey respondent. One source of information may come through the survey respondent filling out a survey directly. Any information that the survey respondent volunteers may be used to attempt to match the respondent to a target group for a survey's survey respondent demographics. Information may also be gathered indirectly by observing the survey respondent's use of the survey device. If the survey device has both informational and recreational content,
20 interaction with the content may indicate a survey respondent's interest in a specific topic. This indirect information can then be used to target the survey respondent with relevant surveys. As mentioned above, relevant information can also be gathered by interfacing with the Point-of-Sale computer system of the locale. Point-of-Sale computer systems typically have a large amount of information about the survey respondent and this information can be used to specify or augment
25 the demographic information stored for a particular survey respondent.

F) Tracking the Survey Respondent

To make the use of information collected about a respondent particularly effective, it is desirable to be able to track the survey respondent across interactions with the Digital Survey System as much as possible. Ideally the tracking would occur across both time and different location. The
30 ability to capture information from respondent activity at a variety of locations that do not otherwise share information with one another (such as restaurants in competition with one another, or a national park and a tire repair waiting room). One way of collecting information and associating it with a particular respondent is to use a respondent id. The respondent id may take the form of a number, a membership card or a token that the survey respondent would maintain
35 and provide. Tracking the survey respondent can also be accomplished if the survey respondent is willing to supply a piece of identifying information like a name, phone number, or address. An

additional method of tracking survey respondents is to interface with existing systems like a Point-of-Sale computer system which tracks a customer's purchases. The Digital Survey System will be subject to imperfect information at times, and therefore to track survey respondents the Digital Survey System may also need to correlate survey information to build a more complete survey respondent profile. Access to imperfect information means that sometimes the information provided may contain mistakes or may be incomplete. Therefore the attempt to recognize different interactions with the Digital Survey System as originated by the same survey respondent requires heuristic methods.

The survey respondent benefits from having to enter information time and time again from one survey to the next if the survey respondent has a user profile and the electronic survey uses the information in the user profile. The time saving aspects of a user profile are particularly large for long text strings such as name, address, telephone numbers, and e-mail address.

Another way of tracking a user across interactions with the Digital Survey System is to use a biometric instrument to measure a repeatable parameter. The range of parameters that can be measured in a repeatable way include finger or hand prints, retinal scans, voice prints, the acceleration and pressure of a stylus while signing the respondent's signature, and other biometrics. The biometric instrument may be cumbersome or expensive to place on each survey device. A solution is to provide a docking bay to allow the survey device to enter into electronic communication with a biometric instrument during the initiation of a survey session to allow identification of the respondent and downloading of the respondent profile.

G) Protecting the Survey Respondent's Privacy

The privacy of this information is protected by the Digital Survey System. The information gathered is only made available to the marketer that the survey respondent expressly responds to and then only as an anonymous response. A survey respondent's personal information is only provided to a marketer with the express consent of the survey respondent. This is done to protect the anonymity and privacy of the survey respondents. However, the survey respondent will be encouraged to volunteer an identifying datum such as a phone number, a respondent identification code, or a biometric measurement on every interaction with the Digital Survey System. This would allow the Digital Survey System to personalize itself to the survey respondent's preferences.

H) Providing the Credits

A recurring challenge in surveying is getting consumers to provide information about themselves. For this reason, marketers commonly resort to credits to convince consumers to provide information about themselves. Examples of this are business card raffles, kids clubs and rebate cards. Many local businesses offer prizes like free meals on a regular basis. To enter the raffle the customer drops a business card in an entry box. At the end of the week or month a business

card is selected from the entry box to award the prize. The objective of collecting business cards is to limit the contest to the section of the population that carries business cards (employed individuals in ownership or white collar positions.) The business gains a lot of information about their clientele, such as where they work and what kind of position they hold in their respective enterprises.

Many national fast-food restaurant chains sponsor kids clubs where they offer the participants magazines with games and puzzles and complementary birthday coupons. The chain is then able to identify and target a significant segment of their clientele, the children whom the parents seek to entertain.

10 Many manufacturers offer significant rebate coupons with their products in the hope of gaining insight into their customer base. By providing the incentive via rebate, the manufacturer learns the name and address of the purchaser and possibly other information by requiring the sales receipt (which would provide the name of the store and the sale price.) For some small commodity items such as "Zip" disks, the rebate has been up to the value of the product itself.

15 Therefore, a consumer gets a personal "Zip" disk for free if the consumer takes the time to reveal information to the manufacturer by filling out the rebate coupon.

Unfortunately, many consumers are not adequately motivated by the promise of a rebate check several weeks in the future. Many consumers know that many rebate coupons made it home but did not get the processing, stamp and envelope needed to complete the process. In this age of direct deposit and ATM machines, many consumers are likely to find a check for a few dollars to be a nuisance since the check will need a special trip to the bank for deposit.

20 The present invention provides a new and innovative way for marketers to convince consumers to provide information about themselves. The present invention provides a means of immediately fulfilling a credit in a retail establishment in return for the survey respondent providing one or more responses. The terms "immediate", "immediately" or other variants are used in this document and the claims that follow to mean soon enough to allow use of the credit during that retail visit. Thus, "immediately" could be within seconds or minutes for certain retail transactions such as a survey given during a gasoline fill up or a period of an hour or more for surveys taken on planes, or trains, at intermission of a show or between the wait for a table and dessert at a fine restaurant. This allows the marketer to provide the survey respondent with gratification tailored to the survey respondent's interests.

30 For instance, if a survey respondent is visiting a restaurant on Friday night the survey respondent is interested in enjoying a good dinner. For example, the present invention allows the survey respondent to fill out a survey while waiting to be seated. Use of the present invention allows the survey respondent to be gratified by receiving a free appetizer or a rebate against the survey respondent's meal expense. This provides the marketer with a new means to provide credits to

survey respondents. Prior art includes purchase incentives such as coupons or game tokens affixed to soft drink cups, bottle caps, and groceries. The present invention does not rely on purchasing an object and taking a physical piece of the purchased object as a token of any sort. The present invention does not require spending money as a means to get a coupon or game
5 piece. Unlike these prior art devices, the present invention provides an incentive for filling out a survey and submitting the set of responses.

Providing the credit to the survey respondent can be done a number of ways such as: applying a rebate to a consumer's purchases; supplying them with a promotional item in the store; or giving them access to special products and promotional items that can be ordered via the Digital Survey
10 System. The diversity of credits that a survey respondent can receive varies from locale to locale. To address this, the Digital Survey System encodes the types of credits to be applied at each locale in the surveys before they are sent to the survey devices at the various locales. This can be done using the locale database which contains information regarding the locale demographic and the types of credits the locale supports.

15 In order to provide the credit, the Digital Survey System must be able to correlate the credit with the survey respondent without compromising the survey respondent's privacy and anonymity. This can be resolved by any conventional means including: giving the survey respondent an authorization code to present to store personnel; interfacing with the Point-of-Sale computer system in a locale; or giving them a physical token including a paper token to be redeemed. In
20 the case where the credit is used in conjunction with the purchase of products, the correlation can be carried out using the shipping address, credit card or other personal information provided by the survey respondent.

D. Acquisition of Survey Information

The third fundamental issue is the actual means of acquisition of survey information. The
25 overhead involved in performing a survey can be quite significant. The prior art method of requiring a respondent to fill out a paper survey there were tremendous drawbacks. The drawbacks included the illegibility of some information, the mistakes caused by transcription of the information, and the raw cost of entering the information into computerized databases for analysis. Collectively these drawbacks present a staggering amount of overhead to the marketer.
30 Over the years many innovations have eased the burden associated with surveying. The appearance of computer scannable survey sheets allowed the marketer to automate some of the process by limiting the survey to multiple choice questions, but certain information still had to be manually entered. The advent of computerized survey stations allowed automated entry of the non-multiple choice information by the survey respondent himself. The introduction of survey
35 forms on the World Wide Web allow marketers to completely automate the collection of information for many business processes such as product information and support queries.

The present invention provides a new and innovative way to acquire and organize survey information. Digital Survey System creates a network of survey devices that are responsible for the presentation and collection of surveys. The actual topology of the equipment is not particularly important as long as the features exist within the system and communication links do not impede the process of matching survey respondents to surveys and then immediately making a credit available to the survey respondent. The survey devices may be physically networked to the central computer, wirelessly networked to the central computer or periodically synchronized with the central computer over a network. At present this may require a separate computer in the retail location to perform the collection of surveys and authorization of credits. If the survey devices do not have the capability to interface directly with the information servers, an intermediary computer may be required to make the communication possible. In such a case, it is reasonable to delegate certain responsibilities to this intermediary computer.

J. Verification of Information

When collecting information it is important to prevent the inaccurate or false information from being collected. This can be of particular concern when the survey respondent providing the information is being compensated for the information. For instance, some individuals frequently respond with false information when attempting to download trial software from websites. At other times people will respond with false information in order to get candy, a mug or some other gift while filling out some application or form.

The preferred embodiment of the Digital Survey System verifies the information provided at two points. At the point-of-survey acquisition, the survey device uses does several preliminary tests to verify that information is legitimate. This includes ruling out invalid phone numbers, zip codes, addresses. When possible the survey device will take advantage of the substantial databases maintained by the Digital Survey System to cross check the accuracy of the information. This will also allow the Digital Survey System to do sophisticated checks such as preventing a consumer from filling out a survey multiple times at different locations. When done at the point of survey acquisition, the verification can prevent fraudulent use of the Digital Survey System. The second point of verification is after the responses have been collected by the information servers. At this point meticulous tests can be run against the data. This may include the verification of all phone numbers, addresses and electronic mail addresses against national databases. In this way the Digital Survey System can provide the highest quality data to the marketers as an integrated service.

K). Controlling the Presentation Environment

A marketer may have a specific need for the environment that the survey is presented. The needs may include parameters that the marketer believes are favorable to reliable results (ambient noise below a certain limit, and ambient light within a certain range, time between responses indicating

reflection but not extended periods of distraction). Another marketer may want comments on previews for a movie or a video game and want a survey device to be used that has suitable output devices to provide the samples of the video to look good and stir interest in the video through use of the survey. While the prior art use of a human survey agent would provide some control over these parameters the present invention allows the marketer to select and achieve compliance of a number of criteria without having a human agent to monitor the situation.

If the marketer only wants to show demo clips of a new computer game on survey devices with the capacity to output audio, images, or video of a minimum quality level, then the control of distribution may use multiple levels of control. As the present invention does not depend on using the same survey device at all queuing locations, it follows that some survey devices are going to be better suited for certain types of surveys. Examples include the use of waterproof devices with antiglare screens for use near pools; back-lit screens for use in certain lighting; and survey devices with headsets for music video surveys. A more sophisticated example is the use of survey devices with the capacity to gage position inside the queuing location from local position indicators or from GPS could be appropriate in grocery stores or museums. An interaction between a positioning system and a survey would include noting that the survey device carried by a respondent has stopped for more than a minute in proximity to the disposable diapers. The stop in front of the disposable diapers may indicate that the respondent is trying to decide which brand to buy. This respondent may be a prime candidate for a survey on the decision process of disposable diaper buyers. The respondent would also be a prime candidate for a point-of-sale rebate on a package of diapers (from the marketer's employer) upon completion of the survey. To control the distribution of surveys, the Digital Survey System can be programmed to distribute the survey to only those queuing locations that meet the survey requisites including the availability of survey devices with the necessary input or output capacities (touch screen, camera, suitable display screen, necessary memory to handle downloaded images, headphones, etc.).

Not every device at a particular queuing location would necessarily have the particular special use features required for a particular survey. Thus the survey may use a Time, Place, and Manner qualification to limit the availability of the survey to those devices that match the hardware requirements for that survey. As an additional level of protection, the survey designer may have added a check that the required hardware is not only present on a particular survey device but is functioning properly before qualifying the survey device to run the survey.

Additional control on the Time, Place, and Manner of presenting a survey may include the parameters of ambient noise or light levels. The survey designer may have required that a particular survey not be presented if the ambient light is above a certain value so as to prevent the video images from looking washed out. The survey designer may limit the survey to devices that have been essentially still for several minutes to increase the odds that the survey respondent is

sitting at a table or at the bar rather than moving about in a standing-room-only waiting room for a bar. In the preferred mode of the present invention, the survey devices are portable. Thus, it may be cost effective to use some Time, Place, and Manner instruments that are not part of the mobile survey device but provide real time information that may be used in certain survey qualifications. The ambient light level at a pool cabana, and the ambient noise level in a bar, are two examples of situations where a properly placed instrument could provide the ambient information without the need for individual instruments on the survey devices. Alternatively, the results from the instruments on all or some of the survey devices could be used to get the average light level or the average noise level. The readings taken from the survey device placed near a noisy drink blender may skew the results slightly, but averaging across time and various devices will provide a reasonable estimate of ambient noise.

L) Distributing the Respondent Profile

In order to make the respondent profile information collected during the first survey session with a respondent useful at subsequent sessions, the information must be made accessible to other queuing locations. As the number of respondents and queuing locations grows it will become more difficult to move information about a given respondent to all queuing locations. The movement of information to queuing locations can be done in a number of ways within the ability of those of skill in the art. The respondent profile can be maintained at each location where the respondent has completed a survey. The local copies of the user profile could be updated in real time, but the preferred mode would be to update the files at off-peak communication periods during the night or at some weekly interval. The local copy of the user profile could be discarded after a period without use at a given queuing location. Several schemes are possible for distributing the respondent profile to queuing locations before the respondent answers a survey at that location. Local copies of the user profile could be distributed to all participating locations in the city or county where the respondent has completed a survey. Local copies of the user profile could be sent after the respondent selects a list of queuing locations to receive the local copy of the user profile. The user profile could be downloaded to the queuing location upon request from the respondent upon the respondent's arrival at the queuing location. The request from the respondent could come through the survey device; through a separate device that is used for such requests, though the survey device when docked to provide for higher communication speed when loading profiles into the survey device, or through the survey specialist at the queuing location.

FLOWCHART

The preceding sections have described the various elements in the process of distributing surveys and collecting the results. Before describing a preferred implementation of these steps in an objected-oriented-programming language, it will be useful to quickly run through the

fundamental steps to highlight the sources of information and the general sequence of events.

Figure 2 contains a general flowchart for the process for the distribution and processing of a survey. For the convenience of the reader, the elements of Figure 2 are generally verbs or process steps. These elements will have numbers in the 2xxx series.

5 Figure 3 contains a diagram of the various sources and holding places for information. For the convenience of the reader these elements will have numbers in the 3xxx series.

The following example of the process will refer to Figures 2 and 3. In order to promote focus on the elements in a certain order, element numbers will be introduced when the element is introduced rather than when the element is merely referenced in passing. In order to provide
10 context, the example will contain details for a specific marketer, survey, respondent, queuing location, and so forth. These are but details and not limitations on the scope of the invention. An example of a detail is the attributes of the survey device. In this example, the survey device has sufficient memory to store the various elements referenced below. Other options would be to
15 have a survey device with more limited memory but the ability to access the Digital Survey System through any suitable wireless connection, or through any one of a number of connection ports distributed through the queuing location. The use of a docking station in this example provides guidance on implementing the Digital Survey System in a situation where wireless communication is not the selected option.

In order to have a central story line for this example, the Survey 3000 (defined below) is followed
20 from creation to completion. Technically, the survey does not go from the central Digital Survey System to the local queuing stations and then into the survey device. Copies of the survey are made and sent based on the survey created and stored in the central Digital Survey System. However, for ease of reading, the various copies are not given unique element numbers.

A) The Creation of the Survey

25 Step 2000 Create the Survey .. Note -- to show the movement of the Survey 3000 through the various devices, the same reference number is used throughout. It is recognized by those of skill in the art, that there will actually be copies of the Survey 3000 at the various levels of the system such as a template copy on Queuing Location Digital Survey System computer and a template copy on Survey Device. In the preferred mode, the Survey 3000 is a combination of interrelated
30 elements including: the presentation of the material to the respondent; the qualifications required for becoming a survey respondent; any demographic distribution plans, incentives for respondents and queuing locations; and control measures to be used to start and stop the survey process.

Note that as a practical matter, the creation of the Survey 3000 and the various components to the survey will likely be a team product from the Marketer 3500 and one or more Digital Survey
35 System specialists on the Digital Survey System Survey Support Team 3510 that are well-versed