(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
8 November 2001 (08.11.2001)

PCT

(10) International Publication Number
WO 01/84764 A2

(51) International Patent Classification[7]: H04L

(21) International Application Number: PCT/US01/14137

(22) International Filing Date: 2 May 2001 (02.05.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
09/565,584 4 May 2000 (04.05.2000) US

(71) Applicant *(for all designated States except US)*: **MICROSOFT CORPORATION** [US/US]; One Microsoft Way, Redmond, WA 98052 (US).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: **HORVITZ, Eric, J.** [US/US]; 330 Waverly Way, Kirkland, WA 98033 (US).

(74) Agent: **AMIN, Himanshu, S.**; Amin & Turocy, LLP, 24th Floor, National City Center, 1900 East 9th Street, Cleveland, OH 44114 (US).

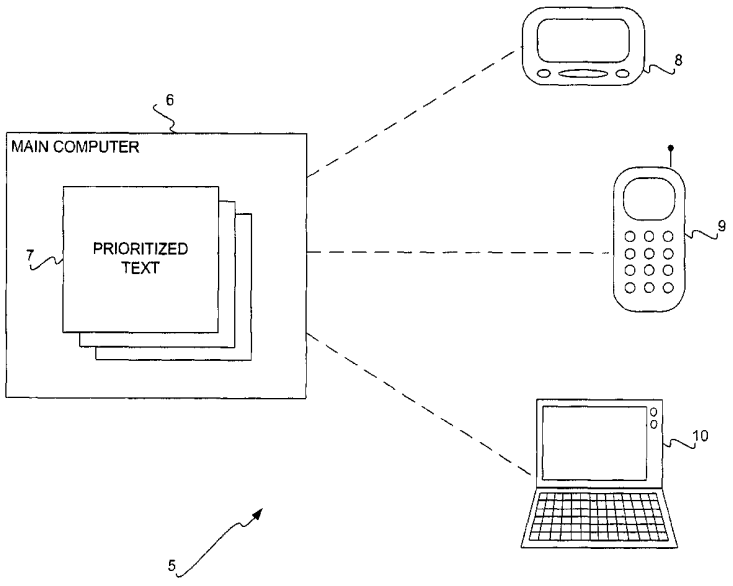
(81) Designated States *(national)*: AE, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, CA, CH, CN, CR, CU, CZ, DE, DK, DM, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL,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

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

[Continued on next page]

(54) **Title:** TRANSMITTING INFORMATION GIVEN CONSTRAINED RESOURCES



(57) **Abstract:** Prioritizing text (7) such as e-mail messages, given constrained resources (8, 9, 10), is disclosed. In one aspect, a method (5, 36) first receives a plurality of messages (37), such as e-mail messages. The method (5, 36) prioritizes the messages (37), based on a predetermined criteria (38) given a constrained resource (38). The method then outputs the messages (39) to a user, as pioritized (39) given the constrained resource (38). Specific constrained resources (38) include limited bandwidth, limited storage space, and limited time in which to receive the texts. Other aspects of the invention include policies (41, 42, 43) that include priority-sensitive truncation and summarization.

WO 01/84764 A2

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.

Title: TRANSMITTING INFORMATION GIVEN CONSTRAINED RESOURCES

Technical Field

The present invention relates generally to the transmitting of information such as text, images, and software components given constrained resources, including the case of limited or expensive bandwidth and/or situations having intermittent connectivity of a client device to a network server as may be the case in a mobile situation.

Background Art

Computer users today frequently use their computers and computerized devices, such as handheld computers and cell phones, in a mobile manner. For example, in the past a computer user may have expected to read his or her e-mail while at a main desktop computer at the office, whereby the main computer was constantly connected to the company network, and the Internet. However, users are now frequently connecting to network or communication facilities intermittently, from a variety of different locations and utilizing a variety of different devices. For example, a user who travels frequently may connect his or her laptop to the office network while at the airport over a dial-up phone connection just before boarding the plane, or call the office network after checking into a hotel. A user may also have their e-mail forwarded to a pager, or a cellular phone, or to other forms of wireless network technology. These connections may be brief in which to download some e-mail or other information, before disconnecting. Intermittency may also result even with optimally permanent connections. For example, more sophisticated mobile communications systems may allow for "always-on" wireless services, but in reality signal strength may not be sufficient in all the places users travel, such that the resulting connection is intermittent. Additionally, a system may be designed to limit its usage of bandwidth by establishing connections and/or transmitting information intermittently.

Intermittent connections and even some permanent connections present difficulties in receiving important e-mail messages and other information, because of the constrained resources that frequently are inherent with such connections. For

example, a cellular phone or a pager may only be able to receive a certain number of messages, such that each is of a certain length. As another example, a user in an airport who wishes to connect to his or her desktop or e-mail system over a public phone generally is limited to a relatively slow modem connection, and may not have time to receive e-mail messages if the user is awaiting a connecting flight.

Disclosure of the Invention

The following presents a simplified summary of the invention in order to provide a basic understanding of some aspects of the invention. This summary is not an extensive overview of the invention. It is intended to neither identify key or critical elements of the invention nor delineate the scope of the invention. Its sole purpose is to present some concepts of the invention in a simplified form as a prelude to the more detailed description that is presented later.

The present invention relates to prioritizing information such as instant messages, e-mail messages, or relevant web pages, for example, given constrained resources. The invention relates to substantially any type of information, referred to generally herein as texts, such as e-mail messages, software components, and/or graphical information, for example. However, it is noted that several aspects of the invention are described by example as relating to e-mail, although it is understood that the invention itself is not so limited. According to one aspect of the present invention, a method first receives a plurality of texts, such as e-mail messages. The method prioritizes the texts, based on a predetermined criteria given a constrained resource. The method then transmits the information to a user, as prioritized given the constrained resource.

For example, in one aspect of the invention, a user may have in their e-mail box ten or so e-mail messages, including several important e-mails, a few unsolicited, commercial or "junk" e-mail messages, and several e-mail messages with very large attachments. The user may be away from their main computer, such that the user receives e-mail over a pager or cell phone display, or dial in to a text-to-speech system that can read e-mail to the user. Therefore, predetermined criteria in one such aspect may specify that the junk e-mail is not to be forwarded to the pager, nor the attachments – that is, the important e-mail and the text of the other e-mail messages (that originally had attachments) are forwarded. The constrained resource in this instance is the fact

that the pager does not have a large amount of storage space, and also does not necessarily have sufficient bandwidth to receive large amounts of data in a timely manner. Thus, the method outputs the e-mail messages to the user by forwarding them to the pager, given these constrained resources.

The following description and the annexed drawings set forth in detail certain illustrative aspects of the invention. These aspects are indicative, however, of but a few of the various ways in which the principles of the invention may be employed and the present invention is intended to include all such aspects and their equivalents. Other advantages and novel features of the invention will become apparent from the following detailed description of the invention when considered in conjunction with the drawings.

Brief Description of Drawings

Fig. 1 is a schematic block diagram of a representative operating environment in conjunction with constrained resources in accordance with an aspect of the present invention.

Fig. 2 is a graph illustrating linear cost functions of high, medium and low priority texts, according to an aspect of the present invention.

Fig. 3 is a graph illustrating a non-linear cost function for a text, according to an aspect of the present invention.

Fig. 4 is a block diagram illustrating a classifier in accordance with an aspect of the present invention.

Fig. 5 is a flowchart of a method according to an aspect of the present invention.

Fig. 6 is a diagram of a system according to an aspect of the present invention.

Fig. 7 is a schematic block diagram illustrating message classification in accordance with an aspect of the present invention.

Fig. 8 is a schematic block diagram illustrating a scalar classifier output in accordance with an aspect of the present invention.

Fig. 9 is a schematic block diagram illustrating texts classified according to a class and scalar output in accordance with an aspect of the present invention.

Fig. 10 is a diagram illustrating a model for determining user activity in accordance with an aspect of the present invention.

Fig. 11 is a diagram illustrating an inference-based model for determining current user activity in accordance with an aspect of the present invention.

Fig. 12 is a diagram illustrating an inference-based model for determining alerting costs in accordance with an aspect of the present invention.

Fig. 13 is a diagram illustrating a more detailed inference-based model for determining alerting costs in accordance with an aspect of the present invention.

Fig. 14 is a diagram illustrating a more detailed inference-based model for determining alerting costs in view of a fidelity loss in accordance with an aspect of the present invention.

Fig. 15 is a flow chart diagram illustrating a methodology for generating and determining priorities in accordance with an aspect of the present invention.

Fig. 16 is a diagram illustrating a text generation program and classifier in accordance with an aspect of the present invention.

Fig. 17 is a schematic block diagram illustrating an alerting system in accordance with an aspect of the present invention.

Fig. 18 is a diagram illustrating a routing system and an alerting system in accordance with an aspect of the present invention.

Fig. 19 is a schematic block diagram of a system illustrating a notification platform architecture in accordance with an aspect of the present invention.

Fig. 20 is a schematic block diagram illustrating a context analyzer in accordance with an aspect of the present invention.

Fig. 21 is a schematic block diagram illustrating notification sources and sinks in accordance with an aspect of the present invention.

Fig. 22 is a diagram illustrating a utility of notification curve in accordance with an aspect of the present invention.

Fig. 23 is a diagram illustrating a user specification interface for notifications in accordance with an aspect of the present invention.

Fig. 24 is a diagram illustrating context information sources in accordance with an aspect of the present invention.

Fig. 25 is a diagram illustrating a rules-based system for determining context in accordance with an aspect of the present invention.

Fig. 26 is a schematic block diagram illustrating an inference-based system for determining context in accordance with an aspect of the present invention.

Fig. 27 is a diagram illustrating an inference model for determining context in accordance with an aspect of the present invention.

Fig. 28 is a diagram illustrating a temporal inference model for determining context in accordance with an aspect of the present invention.

Fig. 29 is a flow chart diagram illustrating a methodology for determining context in accordance with an aspect of the present invention.

Fig. 30 is a flow chart diagram illustrating a methodology for notification decision-making in accordance with an aspect of the present invention.

Fig. 31 is a schematic block diagram illustrating a suitable operating environment in accordance with an aspect of the present invention.

Fig. 32 is a diagram of an exemplary computerized device in accordance with the present invention.

Mode(s) for Carrying Out the Invention

The present invention relates to a notification system, wherein received messages can be prioritized and forwarded to a communications modality associated with a constrained resource such as a limited transmission bandwidth, a connection interval, a download time, and a storage capability. The system includes a prioritizer to assign a priority to one or more received messages and a notification platform that forwards the one or more received messages to a communications modality based upon an associated resource of the modality and the priority of the one or more received messages. The one or more received messages can be forwarded based upon a determined loss of value with time associated with a delayed review of message content and can be forwarded based upon a cost-benefit analysis in order that expected costs of the delayed review are minimized. As will be described in more detail below, other considerations such as rules-based policies are provided to transmit messages given the constrained resources of the communications modality.

Referring initially to Fig. 1, a main computer 6 associated with a user includes one or more prioritized text 7. The main computer 6 of the user may be, for example, a desktop computer that is located in the user's office or home. The prioritized text 7 may include, for example, a series of e-mail messages that have been received by the main computer 6, and prioritized according to a predetermined criteria, given a constrained resource, as is described in more detail below.

The environment 5 of Fig. 1 presumes that the user desires to access the prioritized text 7, but not other text that the computer 6 may have received. For

example, the user may have received several junk e-mail, several unimportant e-mail that were circulated to the user as well as to the user's friends and colleagues, important e-mail that include attachments that are very large, and/or e-mail from the user's supervisor. In such an instance, the user may desire to read the e-mail from his or her supervisor, and the text of the important e-mail but not the attachments thereto.

The user, however, may not be able to access the text including the prioritized text 7 for a length of time – for example, hours if the user is in a meeting, or even days or weeks if the user is on a business trip or on vacation. Instead, the user may desire access to the prioritized text 7 only – that is, not the unimportant messages – forwarded to a pager 8, or a cell phone 9, or may desire to call into the server for the network of which the main computer 6 is a part (not shown), or call into the main computer 6 directly, using a modem on a laptop computer 10 to download the text 7. Each of the devices 8, 9 and 10, however, is, has or represents a constrained resource.

For example, the pager 8 typically has meager storage space as compared to the main computer 6, and also typically has limited transmission bandwidth. Therefore, too many messages, or a very large message, can indicate that the pager 8 cannot receive all of the user's associated messages. Furthermore, attempting to upload the messages to the pager 8 can take an inconveniently long time, because of limited transmission bandwidth. To this end, the pager 8 is, has or represents a constrained resource. The user, therefore, may desire to have only the prioritized text 7 sent to the pager 8, and may also desire that only the first "*x*" characters of respective messages are sent, for example.

With respect to the cell phone 9, the constrained resources are similar. Like the pager 8, the cell phone 9 typically has meager storage space as compared to the main computer 6, and also typically has limited transmission bandwidth. Thus, with respect to the phone 9, the user may only desire to have the prioritized text 7 sent to the cell phone 9, and that the first "*x*" characters of each message sent.

Finally, with respect to the laptop 10, storage space is generally greater than the pager 8 or the cell phone 9; however, other constrained resources are present. Unlike the cell phone 9 and the pager 8, which can receive information unaided by the user, and generally wherever the user happens to be located, the laptop 10, when accompanied by a modem, typically requires the user to actively download information from the computer 6. That is, the user generally must hook up the laptop 10 to a phone

jack, dial the server or the computer 6, and stand by while the prioritized text 7 is downloaded. Furthermore, the modem of the laptop 10 generally has limited transmission bandwidth, albeit usually better than that of the pager 8 and the phone 9. However, the user may find him or herself, for example, in an airport, whereby the user has a limited amount of time to download important messages before having to catch a connecting flight. Therefore, with respect to the laptop computer 10 as well, the user may desire to have the text 7 sent to the laptop 10, and/or limited to the first "*x*" characters of the message sent.

The present invention provides for prioritization of texts, based on a predetermined criteria, given a constrained resource as has been described. This enables a user to receive the most important texts at his or her auxiliary device (*e.g.,* the pager 8, the phone 9, or the laptop 10) of the texts that have been received at the main computer 6 of the user. Those of ordinary skill within the art can appreciate, however, that the invention is not limited to the constrained resources described in this section, and that description in this section of constrained resources is provided only for illustrative purposes as to the types of environments in which aspects of the invention may be practiced.

In this section of the description, a predetermined criteria is described for prioritizing text, given a constrained resource such as the constrained resources described in the previous section of the description.

Methods for downloading content under limited bandwidth and/or intermittent connectivity are now described. The downloading methods center on the creation of classifiers with the ability to recognize the loss of value with time associated with the delayed review of content based on properties of the content. For the example of e-mail, the classifiers take as input a set of features, including the time of day a message is sent, single words, phrases, information about the sender (*e.g.,* someone in a user's address book or someone within some distance in an online organizational tree versus others) and recipient(s) (directly to the user as compared to a user being cc'ed), the length of the body of the message, the presence of one or more questions, and variables representing summaries of statistics about more abstract linguistic features (*e.g.,* presence of future versus past tense, *etc.*).

Features for discriminating the criticality of content for e-mail include those often seen in low priority messages, including words or patterns of attributes associated

with unsolicited or "junk" e-mail in the document such as the quantity and pattern of non-alphanumeric text content such as asterisks and dashes, and quantity of white space. A time-dependent loss function is assigned that captures the loss of value with delayed review of new documents or items received and utilizes this value to prioritize the downloading of e-mail content. It is noted that the present invention does not require classifiers and is not limited to such. For example, sets of rules can be employed to assign and parameterize time-dependent loss functions based on the presence or absence of one or more features associated with new content or messages.

Given a measure of urgency based on features associated with the nature or source of content, a system can download them according to a cost—benefit analysis so that the expected cost associated with delayed review is minimized (or, equivalently, so that the expected value of the transmission is maximized). In a basic approach, messages can be fully downloaded according to a cost—benefit analysis. For respective items, a cost-benefit ratio is determined, by dividing the value of the transmission by the amount of resources required for transmitting the information, for example. Furthermore, the difference in the value of seeing the message now versus later, per an estimate of the expected time that a message will be reviewed if the message is not relayed now, can be determined as well.

According to one aspect of the present invention, the entire message can be downloaded, in order of urgency. Downloading is continued until a communications deadline is reached. In another approach, a cost-benefit ratio is utilized. In this approach, the gains in value associated with having the document now versus seeing it later (*e.g.*, during review of e-mail at a time when there is greater bandwidth), divided by the size of the document, are considered. If downloading by this metric, an additional consideration of a single test to check whether the value of any single message is worth more than the total value of messages downloaded by the latter ordering can also be performed. More ideal orderings can be sought by performing local or more general searching over multiple candidate documents for transmission in pursuit of additional savings in value (or, equivalently, the minimization of cost) for any amount of available time or memory.

Furthermore, the value of truncated documents is considered. Rather than downloading entire documents, a function can be employed that truncates a document and that modifies the value of downloading the document – ultimately employing such

assignments to download smaller documents that have lesser value. In a basic approach for truncation, large documents can be truncated to a maximum transmitted document size. This value is then utilized in a cost-benefit analysis. A value function can also be considered over transmitting portions of documents and employing methods that automatically segment and truncate documents in a manner that maximizes returns given multiple truncations dictated by such utility models.

Beyond the use of truncation, intelligent summarization of text can be enabled, as well as options for the user-specified policy of removing attachments or of dynamically removing them given a cost-benefit analysis of the value of the attachments. A user can also specify such summarization policies as providing a threshold in value (or urgency), such that documents that have higher value are kept intact, and documents that are below the value are truncated or attachments are removed – or users can be allowed to dictate truncation as a function of the value or urgency of a communication. Other policies include the policy wherein documents are kept intact, unless they are greater than some size, at which time a value-specific truncation or summarization occurs. A variety of nonlinear or linear functions can be specified for such policies.

Furthermore, context can also be considered. This approach considers notions of locality and context and modify urgency scores by the p(message relevant | context), wherein p is a probability. In one approach to context-sensitive caching of information, respective documents have associated therewith an urgency that is conditioned on the presence of a context (*e.g.*, such as a location and/or time of day). At run-time, the urgencies are assigned to documents based on the context. If context is uncertain, an expected urgency can be computed as the mean urgency computed as the average of the urgencies of each document j weighted by the likelihood of each context, $\sum_{i=1..n}$ p(Value j | Context *i*)p(Context *i*).

The user's prior knowledge may also be considered. This approach considers the likelihood a user already knows the information in a communicated message (*e.g.*, the likelihood he or she has already heard the information from another source or communication). The expected value of information j changes with changes in the probability, p(User does not know j | E), wherein E refers to evidence including the time since an incident occurred, a news story broke, how widespread the information is likely to be based on the type of information being transmitted, *etc*. Typically, this

probability decreases with time because the likelihood that a user has received the information from another source increases with time. In the general case, the expected value of information j contained in a message is p(User does not know j | E) * Value(j, j not already known) + [1- p(User does not know j | E)]* Value(j, j already known), wherein Value(j, j already known) and Value(j, j not already known) refer to the value or urgencies of receiving information j when the information is already known and not known respectively. If Value(j, j already known) is zero, the value of transmitting the information reduces to p(User does not know j | E) * Value(j, j not already known).

Thus, the expected value of transmitting j is diminished by multiplying it with the probability, p(User does not know j | E). More subtle cases of partial knowledge consider the probability that some partial information is available from another source, and that partial value has some value. For example, a high-level summary or rumor may already have been known by a user. For such situations, the probability of the user knowing the high-level information, and the Value(j, high-level summary of j already known) are considered.

Various approaches to prioritizing documents such as e-mail have been described. In general, it is noted that the overall goal is to maximize the utility for the user given information availability constraints – that is, the constrained resources. Put another way, the goal is generally to minimize the expected cost of delayed review of text received at the user's main computer, when the user is away from the computer, and his or her ability to receive the text is hampered by a constrained resource.

Given a series of messages $msg_1 \ldots msg_N$, for text of equal length (for example, as each message is truncated to a length l), a variable i is selected to maximize

$$\sum_{i=1}^{N} C(msg_i)$$

wherein $C(msg_i)$ is the expected cost of delayed review of the text message msg_i received at the user's main computer. This can also be generalized for text of unequal length.

The cost function can further vary according to the type of text received. For example, the general case is illustrated in Fig. 2, which is a graph 12 of linear cost functions dependent on the priority of a text. In the graph 12, as time increases, the cost of not having reviewed a text also increases. However, the cost increases more for a high priority message, as indicated by the line 13, as compared to a medium priority

message, as indicated by the line 14, or a low priority message, as indicated by the line 15. That is, the high priority line 13 may have a slope of 100, the medium priority line 14 may have a slope of 10, and the low priority line 15 may have a slope of 1. These slope values can then be utilized by a text classifier (described below) to assist in assigning a priority to a given text, for example, by regression analysis.

Some messages, however, do not have their priorities well approximated by the use of a linear cost function. For example, a message relating to a meeting will have its cost function increase as the time of the meeting nears, and thereafter, the cost function rapidly decreases – since after the meeting is missed, there is not much generally a user can do about it. This situation is better approximated by a non-linear cost function, as illustrated in the graph 16 of Fig. 3. In the graph 16, a cost function 17 rapidly increases until it reaches the time of the meeting demarcated by the line 18, after which it rapidly decreases.

In more general representations of the change in value with delayed review of information, general functions for describing the change in cost with delayed review are considered. For example, changes in value can be represented with a sigmoid function. In considering such nonlinear cost functions, a classifier can be employed to learn classes of email, each associated with a single, directly assessed sigmoid curve. In a more detailed approach, classifiers can be employed to learn the parameters describing the height and shape of the sigmoid curve. A set of classifiers can provide a description of such a function including the initial value associated with the importance of an email message, in addition to the temporal dynamics of the loss of value or cost with delayed review. Thus, those of ordinary skill within the art can appreciate that depending on a message's type, the cost function can be approximated by one of many different representative cost functions, both linear and non-linear.

For a nonlinear function, the cost with delayed review depends on the age of the message. For this more general situation, optimization methods such as dynamic programming and general searching are considered in performing a cost—benefit analysis in ordering messages for transmission under limited bandwidth. In one approximation, the reduction in cost of delay with transmitting each message divided by the size of the transmission is considered. The value of transmitting a message now to a user versus waiting until a later time is identified, such as the time the user will likely see the message when returning from a mobile setting, or a later transmission. A single

time, $t_{\text{delay, no transmit}}$, or a probability distribution over the time until a user will see the message if it is not transmitted at the current time, given evidence about the context E, $p(t_{\text{delay, no transmit}}|E)$, where E can represents evidence about the time of day or location of the user, is determined or estimated. The value of transmitting each message is the difference between the value of transmitting the message now and the value when it will likely be seen later. An estimated time delay, $t_{\text{delay, transmit}}$, that the user will review the message if it is downloaded now, or a probability distribution over this time, $p(t_{\text{delay, transmit}}|E)$ can additionally be considered.

For the case of a single estimate of times, the gain in value, ΔValue, with transmission of the message is,

$$\Delta\text{Value} = \text{Value}(t_{\text{delay, transmit}}) - \text{Value}(t_{\text{delay, no transmit}})$$

For the case of probability distributions, the expected change in value is,

$$\Delta\text{Value} = \text{Integral}_{x=0..1}\ p(t_{\text{delay, transmit}}|E=x)\text{Value}(x)dx$$

$$-\ \text{Integral}_{x=0..1}\ p(t_{\text{delay, no transmit}}|E=x)\text{Value}(x)dx$$

As in the case of constant urgency, downloading policies is driven now by ΔValue. In one approach the downloading of entire messages is ordered, in order of ΔValue and downloading is continued until a communications deadline is reached. A more ideal downloading policy employs a cost-benefit ratio, where documents are downloaded in the order of the ratio of the ΔValue of each document divided by the size of the document. If downloading by this metric, with the additional consideration of a single test to check whether the value of any single message is worth more than the total value of messages downloaded by the latter ordering, the result will be within a factor of two of optimal for any amount of communication time.

According to one aspect of the present invention, prioritization of a text based on a predetermined criteria given a constrained resource can be achieved *via* a text classifier. In such an instance, the predetermined criteria is the manner by which the text classifier is trained, either explicitly, *via* initial training, or implicitly, *via* continual training, or both. The text classifier in one aspect can be a Bayesian classifier, as known within the art, while in another aspect it can be a support vector machine (SVM) classifier, as also known within the art.

Referring now to Fig. 4, a text/data classifier 20 can be trained explicitly, as represented by the arrow 22, and implicitly, as represented by the arrow 24 to perform

classification in terms of priority. Explicit training represented by the arrow 22 is generally conducted at the initial phases of constructing the classifier 20, while the implicit training represented by the arrow 24 is typically conducted after the classifier 20 has been constructed - to fine tune the classifier 20, for example, *via* a background monitor 34. Specific description is made herein with reference to an SVM classifier, for exemplary purposes of illustrating a classification training and implementation approach. Other text classification approaches include Bayesian networks, decision trees, and probabilistic classification models providing different patterns of independence may be employed. Text classification as used herein also is inclusive of statistical regression that is utilized to develop models of priority.

According to one aspect of the invention Support Vector Machines (SVM) which are well understood are employed as the classifier 20. It is to be appreciated that other classifier models may also be utilized such as Naive Bayes, Bayes Net, decision tree and other learning models. SVM's are configured *via* a learning or training phase within a classifier constructor and feature selection module 26. A classifier is a function that maps an input attribute vector, $\mathbf{x} = (x1, x2, x3, x4, xn)$, to a confidence that the input belongs to a class - that is, $f(\mathbf{x}) = confidence(class)$. In the case of text classification, attributes are words or phrases or other domain-specific attributes derived from the words (*e.g.*, parts of speech, presence of key terms), and the classes are categories or areas of interest (*e.g.*, levels of priorities).

An aspect of SVMs and other inductive-learning approaches is to employ a training set of labeled instances to learn a classification function automatically. The training set is depicted within a data store 30 associated with the classifier constructor 26. As illustrated, the training set may include a subset of groupings G1 through GN that indicate potential and/or actual elements or element combinations (*e.g.*, words or phrases) that are associated with a particular category. The data store 30 also includes a plurality of categories 1 through M, wherein the groupings can be associated with one or more categories. During learning, a function that maps input features to a confidence of class is learned. Thus, after learning a model, categories are represented as a weighted vector of input features.

For category classification, binary feature values (*e.g.*, a word occurs or does not occur in a category), or real-valued features (*e.g.*, a word occurs with an importance weight **r**) are often employed. Since category collections may contain a large number

of unique terms, a feature selection is generally employed when applying machine-learning techniques to categorization. To reduce the number of features, features may be removed based on overall frequency counts, and then selected according to a smaller number of features based on a fit to the categories. The fit to the category may be determined *via* mutual information, information gain, chi-square and/or substantially any other statistical selection technique. These smaller descriptions then serve as an input to the SVM. It is noted that linear SVMs provide suitable generalization accuracy and provide suitably fast learning. Other classes of nonlinear SVMs include polynomial classifiers and radial basis functions and may also be utilized in accordance with the present invention.

The classifier constructor 26 employs a learning model 32 in order to analyze the groupings and associated categories in the data store 30 to "learn" a function mapping input vectors to confidence of class. For many learning models, including the SVM, the model for the categories can be represented as a vector of feature weights, **w**, wherein there can be a learned vector of weights for each category. When the weights **w** are learned, new texts are classified by computing the dot product of **x** and **w**, wherein **w** is the vector of learned weights, and **x** is the vector representing a new text. A sigmoid function may also be provided to transform the output of the SVM to probabilities **P**. Probabilities provide comparable scores across categories or classes from which priorities can be determined.

The SVM is a parameterized function whose functional form is defined before training. Training an SVM generally requires a labeled training set, since the SVM will fit the function from a set of examples. The training set can consist of a set of N examples. Each example consists of an input vector, ***x***i, and a category label, **y**j, which describes whether the input vector is in a category. For each category there can be N free parameters in an SVM trained with N examples. To find these parameters, a quadratic programming (QP) problem is solved as is well understood. There is a plurality of well-known techniques for solving the QP problem. These techniques may include a Sequential Minimal Optimization technique as well as other techniques. Referring briefly to Fig. 7, a text input 56 that has been transformed into an input vector **x** is applied to the classifier 20 for each category. The classifier 20 utilizes the learned weight vectors **w** determined by classifier constructor 26 (*e.g.*, one weight vector for each category) and forms a dot product to provide a priority output 58, wherein

probabilities **P** may be assigned to the input text 56 indicating one or more associated priorities (*e.g.*, high, medium, low).

Referring back to Fig. 4, training of the text classifier 20 as represented by the arrow 22 includes constructing the classifier in 26, including utilizing feature selection. In the explicit training phase, the classifier 20 can be presented with both time-critical and non-time-critical texts, so that the classifier may be able to discriminate between the two, for example. This training set may be provided by the user, or a standard or default training set may be utilized. Given a training corpus, the classifier 20 first applies feature-selection procedures that attempt to find the most discriminatory features. This process employs a mutual-information analysis. Feature selection can operate on one or more words or higher-level distinctions made available, such as phrases and parts of speech tagged with natural language processing. That is, the text classifier 20 can be seeded with specially tagged text to discriminate features of a text that are considered important.

Feature selection for text classification typically performs a search over single words. Beyond the reliance on single words, domain-specific phrases and high-level patterns of features are also made available. Special tokens can also enhance classification. The quality of the learned classifiers for e-mail criticality, for example, can be enhanced by inputting to the feature selection procedures handcrafted features that are identified as being useful for distinguishing among e-mail of different time criticality. Thus, during feature selection, one or more words as well as phrases and symbols that are useful for discriminating among messages of different levels of time criticality are considered.

As the following examples illustrate, tokens and/or patterns of value in identifying the criticality of messages include such distinctions as, and including Boolean combinations of the following:

Information in a Message Header

For example:

To: field (Recipient information)

Addressed just to user,

Addressed to a few people including user,

Addressed to an alias with a small number of people,

Addressed to several aliases with a small number of people,
Cc:'d to user,
Bcc:'d to user.

From: field (Sender information)

Names on pre-determined list of important people, potentially segmented into a variety of classes of individuals, (*e.g.,* Family members, Friends)
Senders identified as internal to the user's company/organization,
Information about the structure of organizational relationships relative to the user drawn from an online organization chart such as:

- Managers user reports to,
- Managers of the managers of users,
- People who report to the user,

External business people.

Past tense Information

These include descriptions about events that have already occurred such as:
We met,
meeting went,
happened,
got together,
took care of,
meeting yesterday.

Future tense Information

Tomorrow,
This week,
Are you going to,
When can we,
Looking forward to,
Will this,
Will be.

Meeting and coordination Information

Get together,

Can you meet,

Will get together,

Coordinate with,

Need to get together,

See you,

Arrange a meeting,

Like to invite,

Be around.

Resolved dates

Future vs. past dates and times indicated from patterns of text to state dates and times explicitly or typical abbreviations such as:

On 5/2,

At 12:00.

Questions

Words, phrases adjacent to questions marks (?)

Indications of personal requests:

Can you,

Are you,

Will you,

you please,

Can you do,

Favor to ask,

From you.

Indications of need:

I need,

He needs,

She needs,

I'd like,
It would be great,
I want,
He wants,
She wants,
Take care of.

Time criticality

happening soon,
right away,
deadline will be,
deadline is,
as soon as possible,
needs this soon,
to be done soon,
done right away,
this soon,
by [date],
by [time].

Importance

is important,
is critical,
Word, phrase + !,
Explicit priority flag status (low, none, high).

Length of message

Number of bytes in component of new message.

Signs of Commercial and Adult-Content Junk e-mail

Free!!,
Word + !!!,
Under 18,

Adult's only,
Percent of capitalized words,
Percent non-alphanumeric characters.

It is noted that the word or phrase groupings depicted above illustrate exemplary words, groupings, or phrases that may be utilized from which to conduct classifier training. It is to be appreciated that other similar words, groups, or phrases may be similarly employed and thus the present invention is not limited to the illustrated examples.

Furthermore, still referring to Fig. 4, implicit training of the classifier 20, as represented by the arrow 24, can be conducted by monitoring the user work or usage patterns *via* the background monitor 34 that can reside on the user's desktop or mobile computer, for example. For example, as users work, and lists of mail are reviewed, it can be assumed that time-critical messages are read first, and lower-priority messages are reviewed later, and/or deleted. That is, when presented with a new e-mail, the user is monitored to determine whether he or she immediately opens the e-mail, and in what order, deletes the email without opening, and/or replies to the e-mail relatively in a short amount of time. Thus, the classifier 20 is adapted such that a user is monitored while working or operating a system, the classifier is periodically refined by training in the background and updated for enhancing real-time decision-making. Background techniques for building classifiers can extend from those that update the classifier 20 with new training messages.

In this section of the detailed description, a computer-implemented method according to an aspect of the invention is described, and a computerized system according to an aspect of the invention is described. With respect to the method, the method is desirably realized at least in part as one or more programs running on a computer -- that is, as a program executed from a computer-readable medium such as a memory by a processor of a computer. The program is desirably storable on a machine-readable medium such as a floppy disk or a CD-ROM, for distribution and installation and execution on another computer.

Referring to Fig. 5, a flowchart of a method 36 according to an aspect of the invention is illustrated. At 37, one or more texts are received. Respective texts can be an e-mail message, or any other type of text; the invention is not so limited. In 38, the

texts are prioritized based on a predetermined criteria, given a constrained resource, as has been described in the previous section of the description. For example, in one aspect, prioritization is achieved using a text classifier, such that the manner of training the classifier is the predetermined criteria according to which the prioritization is performed. Thus, in one aspect, 38 includes initially training and continually training the text classifier, as has been described. Prioritization may also include first truncating each text to a predetermined length, and/or prioritizing the texts based on a maximization of utility of information for the user, given the constrained resource, as has been described. The invention is also not limited to the constrained resource; in one aspect of the invention, the resource is one or more of: limited bandwidth, limited time to download, and limited memory (storage space), as have been described.

In one aspect, as particularly indicated by the dotted-line box 40, the prioritization of the texts is accomplished for respective texts as follows. In 41, the expected loss of non-review of a text, at the current time, is determined, as described in in the previous section of the description. In 42, the expected cost of alerting the user of the text, at the current time, is determined, as has also been described in the previous section of the description. In 43, if the expected loss of non-review determined in 41 is greater than the expected cost of alerting now determined in 42, then it is concluded that the text is of sufficient priority that the text should be transmitted to the user. Thus, in this aspect, the texts can be divided into two categories: those having priority and which will be transmitted to the user, and those that do not have priority and which will not be transmitted to the user. As can be appreciated, the aspect of the invention represented by the dotted-line box 40 is not a limitation or a requirement of the invention itself, however.

The priority of the text is then output in 39, as prioritized in 38 given the constrained resource. In one aspect, this includes transmitting the texts to the user, who is presumed to be away from his main computer at which the texts were received at 37, for example. This may include transmitting the texts to a device such as a cell phone or a pager. Or, this may include a device such as a laptop computer downloading the texts as prioritized (that is, such that the main computer or a server of a network of which the computer is a part uploading the texts to the laptop computer). Again, however, the invention is not so limited.

Referring next to Fig. 6, a diagram of a system according to an aspect of the

invention is shown. The system includes a first program 45, a mechanism 46, and a second program 49. One or more of the programs 45 and 49, and the mechanism 46, include a computer program executed by a processor of a computer from a computer-readable medium thereof. However, the invention is not so limited.

The first program 45 generates a plurality of texts 47. In one aspect, the program includes an electronic mail program that receives e-mail, which then serve as the text. The mechanism 46 receives the plurality of text, and prioritizes them based on a predetermined criteria, given a constrained resource, as has been described. The mechanism 46 thus generates the prioritized texts 48. The mechanism 46 in one aspect may include a text classifier, and may initially truncate each message to a predetermined length, as has also been described. In one aspect, the prioritization is meant to maximize the utility of information, given the constrained resource (such as: limited bandwidth, limited time, and/or limited memory), as has been described.

The second program 49 receives the prioritized texts 48 from the mechanism 46. For example, the program 49 may be a part of a communications device communicatively coupled to the mechanism 46 in that the program 49 is able to communicate with the mechanism 46, but may not be physically coupled to the mechanism 46. Such devices include a portable computer (*e.g.,* a laptop computer), a handheld computer, a cellular phone, and a pager. However, the invention is not so limited. Thus, the device downloads the plurality of texts 48, as has been prioritized given the constrained resource by the mechanism 46. It is noted that the constrained resource may be inherent within such devices – that is, the nature of a cell phone or a pager, for example, is such that they have constrained resources.

In an alternative aspect of the present invention, larger quantities of messages can be gathered, wherein new filters are created in a batch process, either per a daily schedule, per the number of new quantities of messages admitted to the training set, and/or combinations. For each message inputted into the classifier, for example, a new case for the classifier can be created. The cases are stored as negative and positive examples of texts that are either high or low priority, for example. As an example, one or more low, medium, and high urgency classes can be recognized such that the probabilities of membership in each of these classes are utilized to build an expected criticality. Larger numbers of criticality classes can be utilized to seek higher resolution. For example, as illustrated in Fig. 7, a training set of messages 60 (*e.g.,* very

high, high, medium, normal, low, very low, *etc.*) can be initially employed to train a classifier 62, such that real-time classification is achieved, as indicated at 64, wherein new messages are classified according to the number of examples resolved by the training set 60. In Fig. 7, three such categories are illustrated for exemplary purposes, however, it is to be appreciated that a plurality of such categories may be trained according to varying degrees of desired importance. As illustrated, the new messages 64 may be labeled, tagged and/or sorted into one or more folders 66, for example, according to the priorities assigned by the classifier 62. As will be described in more detail below, the assigned priorities may further be utilized by subsequent systems to make message format, delivery and modality determinations to/for the user.

According to another aspect of the invention, an estimation of a number or value can be achieved by monitoring a user interact with e-mail, for example, rather than labeling the case or message as one of a set of folders. Thus, a classifier can be continued to be updated but have a moving window, wherein cases of messages or documents that are newer than some age are considered, as specified by the user.

For example, a constant rate of loss associated with the delayed review of messages is referred to as the expected criticality (EC) of the message, wherein,

$$EC = \sum_i C^d(H_i) p(H_i \mid E^d_\setminus)$$

wherein C is a cost function, d is a delay, E is an event, H is the criticality class of the e-mail, and EC is expressed as the sum over the likelihood of the class(es) weighted by the rate of loss described by the cost function C for the potential class(es).

As an example, referring to Fig. 7, the text, such as an e-mail message, 56 is input into the classifier 20, which based thereon generates the priority 58 for the text 56. That is, the classifier 20 generates the priority 58, measured as a percentage from 0 to 100%, for example. This percentage can be a measure of the likelihood that the text 56 is of high or some other priority, based on the previous training of the classifier 20.

It is noted that the present invention as has been described above, the classifier 20 and the priority 58 can be based on a scheme wherein the e-mails in the training phase are construed as either high priority or low priority, for example. This scheme is illustrated in reference to Fig. 8, wherein the text classifier 20 is trained by a group of texts 67 that are predetermined to be high priority and a group of texts 68 that are predetermined to be low priority. The text 56 to be analyzed is input into the classifier 20, which outputs a scalar number 69, for example, measuring the likelihood that the

text being analyzed is of high or low priority.

For example, referring to Fig. 9, a diagram illustrates a scheme wherein texts 56 are categorized into low, medium, and high priority. As described above, a plurality of other training sets may be employed to provide greater or higher resolution distinctions of priorities. The text classifier 20 is trained by a group of texts 67 that are high priority and a group of texts 68 that are low priority, and by a group of texts 70 that are medium priority. Thus, the text 56 to be analyzed is input into the classifier 20, which outputs a scalar number 69, that can measure the likelihood that the text being analyzed is of high priority, if so desired, or medium priority or low priority, for example. The classifier 20 is also able to output a class 72, which indicates the class of low, medium, or high priority that the text 56 most likely falls into. Further classes can also be added if desired.

The present invention is not limited to the definition of priority as this term is employed by the classifier 20 to assign such priority to a text such as an e-mail message. Priority can be defined in terms of a loss function, for example. More specifically, priority can be defined in terms of the expected cost in lost opportunities per time delayed in reviewing the text after it has be received. That is, the expected lost or cost that will result for delayed processing of the text. The loss function can further vary according to the type of text received.

Thus, as has been described, the priority of a text can be just likelihood that it is of one of a plurality of priorities based on the output of a classifier, or the most likely priority class the text applies to, also based on the output of the classifier. Alternatively, an expected time criticality of the text, such as an e-mail message, can determined. This can be written as:

$$EL = \sum_{i}^{n} p(critical_i)C(critical_i)$$

wherein EL is the expected loss, $p(critical_i)$ is the probability that a text has the criticality i, $C(critical_i)$ is the cost function for text having the criticality i, and n is the total number of criticality classes minus one. The cost functions may be linear or non-linear, as has been described. In the case where the function is linear, the cost function defines a constant rate of loss with time. For non-linear functions, the rate of loss changes with delayed review or processing of the text and can increase or decrease, depending on the amount of delay.

In the case where n=1, specifying that there are only two priority classes low and high, the expected loss can be reformulated as:

$$EC = p(critical_{high})C(critical_{high}) + [1 - p(critical_{low})]C(critical_{low})$$

wherein EC is the expected criticality of a text. Furthermore, if the cost function of low criticality messages is set to zero, this becomes:

$$EC = p(critical_{high})C(critical_{high})$$

The total loss until the time of review of a text can be expressed as the integration of the expressed criticality, or,

$$EL = \int_0^t p(critical_{high})C(critical_{high}, t)dt$$

wherein t is the time delay before reviewing the document.

Other measures that accord a value metric for ranking documents, such as e-mail messages, by importance. While the discussion above focused on priority as time criticality, other notions of "importance" can also be trained. For example, this can be accomplished by labeling a set of training folders: "High Importance" all the way down to "Low Importance" wherein a measure of "expected importance" can be determined. Another metric can be based on a semantic label, "messages that I would wish to hear about within 1 day while traveling" and to determine a measure for prioritizing messages for forwarding to a traveling user. Furthermore, one utilized metric is urgency or time-criticality, as it has clear semantics for decision-making, triage, and routing. In this case, the classes are labeled according to different levels of urgency and computed as an expected urgency for each message from the probabilities inferred that the message is in each class.

Extensions to criticality classification, as described in the previous section, can also be provided in accordance with the present invention. For instance, classification can include an automatic search for combinations of high-payoff features within or between classes of features. As an example, combinations of special distinctions, structures, and so forth, with words that have been found to be particularly useful for certain users can be searched for and utilized in the classification process. A combination of two features is referred as a doublet, whereas a combination of three features is referred to as a triplet, and so forth. The combination of features can enable improved classification. Classification can also be improved with the use of incremental indexing that employs a moving window in the classifier. This enables the

classifier to be routinely refreshed, as old data is timed out, and new data is brought in.

Classification can also be based on the determination of the date and time of an event specified in a message. This determination can assign features to the message that can be utilized by the classifier. For example, the features assigned may include: today within four hours, today within eight hours, tomorrow, this week, this month, and next month and beyond. This enables the classifier to have improved accuracy with respect to the messages that are classified. In general, classification can be based on the time of the referenced event, considering whether the event is in the future or has past. With respect to future events, classification thus considers the sender's reference to a time in the future when the event is to occur.

Other new features can also be integrated into the classification process. For example, an organization chart can be utilized to determine how important a message is by the sender's location within the chart. Linguistic features may be integrated into the classifier. To accommodate different languages, the features may be modified depending on the origin of the sender, and/or the language in which the message is written. Classification may vary depending on different folders in which messages are stored, as well as other scaling and control rules. In addition to e-mail and other sources, classification can be performed on instant messages, and other sources of information, such as stock tickers, and so forth.

In general, a sender-recipient structural relationship may be considered in the classification process. If the user is substantially the only recipient of a message, for example, then this message may be considered as more important than a message sent to a small number of people. In turn, a message sent to a small number of people may be more important than a message on which the user is blind-copied (bcc'ed) or carbon-copied (cc'ed). With respect to the sender, criticality may be assigned based on whether the sender's name is recognized. Criticality may also be assigned depending on whether the sender is internal or external to the organization of which the user is associated.

Other distinctions that may be considered in classification include the length of the message, whether questions have been detected, and whether the user's name is in the message. Language associated with time criticality may increase the message's importance. For example, phrases such as "happening soon," "right away," "as soon as possible," "ASAP," and "deadline is," may render the message more critical. Usage of

past tense as compared to future tense may be considered, as well as coordinative tasks specified by phrases such as "get together," "can we meet," and so on. Evidence of junk mail may lower the priority of a message. Predicates representing combinations, such as a short question from a sender proximate to the user in the organization chart, may also be considered in the classification process.

In the next section of the description, processes are described that provide a determination when to alert the user of a high-priority text, for example, a text that has a likelihood of being high priority greater than a user-set threshold, or greater than a threshold determined by decision-theoretic reasoning. That is, beyond knowing about time-critical messages, it is also important to decide when to alert a user to time-critical messages if the user is not directly viewing incoming e-mail, for example. In general, a cost of distracting the user from the current task being addressed to learn about the time-critical message is determined.

Alternatively, various policies for alerting and notification can be employed. These policies can be implemented within a notification platform architecture, for example, that is described in more detail below. Some of these policies include:

- Setting a user-specified upper bound on the total loss. This policy would specify that a system should generate an alert when the total loss associated with the delayed review of a message exceeds some pre-specified "tolerable" loss "x".

- Another policy can be a cost-benefit analysis based on more complete decision-theoretic analysis, such as NEVA = EVTA - ECA – TC, wherein NEVA is the net expected value of alerting, EVTA is the expected value of alerting, ECA is the expected cost of alerting, and TC is the transmission cost associated with communicating a message.

In general, a user should be alerted when a cost-benefit analysis suggests that the expected loss the user would incur in not reviewing the message at time t is greater than the expected cost of alerting the user. That is, alerting should be conducted if:

$$EL - EC > 0$$

wherein EL is the expected loss of non-review of the text at a current time t, and EC is the expected cost of alerting the user of the text at the current time t. The expected loss

is as described in the previous section of the description.

However, the above formulation may not be the most accurate, since the user will often review the message on his or her own in the future. Therefore, in actuality, the user should generally be alerted when the expected value of alerting, referred to as EVTA, is positive. The expected value of alerting should thus consider the value of alerting the user of the text now, as opposed to the value of the user reviewing the message later on their own, without alert, minus the cost of alerting. This can be stated as:

$$EVA = EL_{alert} - EL_{no-alert} - EC$$

wherein EL_{alert} is the expected loss of the user reviewing the message if he or she were to review the message now, upon being alerted, as opposed to $EL_{no\text{-}alert}$, which is the expected loss of the user reviewing the message on his or her own at some point, without being alerted, minus EC, the expected cost of alerting based on a consideration of distraction and on the direct cost of the transmitting the information.

Furthermore, information from several messages can be grouped together into a single compound alert. Reviewing information about multiple messages in an alert can be more costly than an alert relaying information about a single message. Such increases in distraction can be represented by making the cost of an alert a function of the its informational complexity. It can be assumed that the EVA of an e-mail message is independent of the EVA of other e-mail messages. EVA(M_i,t), for example, refers to the value of alerting a user about a single message M_i at time t and ECA(n) refers to the expected cost of relaying the content of n messages. Thus, multiple messages can be considered by summing together the expected value of relaying information about a set of n messages, wherein:

$$NEVA = \sum_{i=1} EVA(M_i,t) - ECA(n).$$

It is also noted that in order to determine the expect cost of alerting, it is useful to infer or directly access information about whether the user is present or is not present. Sensors can be employed that indicate when a user is in the office, such as infrared sensors and pressure sensors. However, if such devices are not available, a probability that a user is in the office can be assigned as a function of user activity on the computer, for example, such as the time since last observed mouse or keyboard activity. Furthermore, scheduling information available in a calendar can also be employed to make inferences about the distance and disposition of a user and to

consider the costs of forwarding messages to the user by different processes.

It is also important to know how busy the user is in making decisions about interrupting the user with information about messages with high time criticality. It can be reasoned (*e.g.,* inferential decision-making) about whether and the rate at which a user is working on a computer, or whether the user is on the telephone, speaking with someone, or at a meeting at another location. Several classes of evidence can be employed to assess a user's activity or his or her focus of attention, as illustrated in Fig. 10. A Bayesian network can then be utilized for performing an inference about a user's activity. An example of such a network is depicted in Fig. 11.

In general, a decision should be made as to when and how to alert users to messages and to provide services based on the inference of expected criticality and user activity. Decisions can be performed by utilizing decision-models, for example. Figs. 12-14 are influence diagrams illustrating how such decision models can be utilized to make alerting decisions. Fig. 12 displays a decision model for decisions about interrupting a user, considering current activity, expected time criticality of messages, and cost of alerting depending on the communications modality. Fig. 13 also includes variables representing the current location and the influence of that variable on activity and cost of alternate messaging techniques. Furthermore, Fig. 14 is expanded to consider the costs associated with losses in fidelity when a message with significant graphics content is forwarded to a user without the graphical content being present.

Alternatively, decisions as to when and how to alert users can be made by employment of a set of user-specified thresholds and parameters defining policies on alerting. User presence can be inferred based on mouse or keyboard activity, for example. Thus, a user can be enabled to input thresholds on alerting for inferred states of activity and non-activity, for example. Users can also input an amount of idle activity following activity wherein alerting will occur at lower criticalities. If it is determined that the user is not available based on the time that substantially no computer activity is detected, then messages can be stored, and are reported to the user in order of criticality when the user returns to interact with the computer. Furthermore, users can specify routing and paging options as a function of quantities including expected criticality, maximum expected loss, and value of alerting the user.

A notification and/or alerting system may also estimate when the user is expected to return, such that it transmits priorities that are expected to be important

before the user is expected to return. This can be achieved by learning user-present and user-away patterns over time. The user can then set suitable policies in terms of when he or she is expected to return to the system to review the priorities without being alerted to them. The expected time to return determination by the system may be automatically conveyed to senders of highly urgent messages, for example. In this manner, message senders receive feedback when the user is expected to return such that he or she can reply to the messages. The sender may also be informed that his or her message has been conveyed to the user's mobile device, and so forth.

Fig.15 illustrates a methodology for generating priorities and performing alerting decisions based on the priorities in accordance the present invention. While, for purposes of simplicity of explanation, the methodology is shown and described as a series of acts, it is to be understood and appreciated that the present invention is not limited by the order of acts, as some acts may, in accordance with the present invention, occur in different orders and/or concurrently with other acts from that shown and described herein. For example, those skilled in the art will understand and appreciate that a methodology could alternatively be represented as a series of interrelated states or events, such as in a state diagram. Moreover, not all illustrated acts may be required to implement a methodology in accordance with the present invention.

Referring to Fig. 15, a flowchart diagram 74 illustrates a methodology wherein priorities are generated and utilized in accordance with the present invention. At 80, a data, such as text to have a priority thereof assigned is received. The data can be an e-mail message, or substantially any other type of data or text. At 82, a priority for the data is generated, based on a classifier, as has been described. Additionally, 82 can include initial and subsequent training of the classifier, as has been described.

The priority of the data is then output at 84. As indicated in Fig. 13, this can include processing at 86, 88, 90, 92, and 94. At 86, an expected loss of non-review of the data at a current time *t* is determined. This determination considers the expected loss of now-review of the text at a future time, based on an assumption that the user will review the text him or herself, without being alerted, as has been described. At 88, an expected cost of alerting is determined, as has also been described. If the loss is greater than the cost at 90, then no alert is made at the time *t* 92, and the process proceeds back to 86, at a new current time *t*. Proceeding back to 86 may be performed since as time progresses, the expected loss may at some point outweigh the alert cost,

such that the calculus at 90 can change. Upon the expected loss outweighing the alert cost, then an alert to the user or other system is performed at 94.

The output of the alert to a user or other system is now described. A user can be alerted on an electronic device based on alert criteria, which indicates when the user should be alerted of a prioritized text. The electronic device on which the user is alerted can be a pager, cellular telephone, or other communications modality as described in more detail below. Alerts to a user on an electronic device, such as a pager or a cellular phone, can be based on alert criteria that can be adapted to be sensitive to information about the location, inferred task, and/or focus of attention of the user, for example. Such information can be inferred under uncertainty or can be accessed from online information sources. The information from an online calendar, for example, can be adapted to control criteria employed to make decisions about relaying information to a device, such as a notification sink which is described in more detail below.

Alerts can be performed by routing the prioritized text or other data based on routing criteria. Routing of the text can include forwarding the text, and/or replying to the sender of the text, in the case where the text is e-mail. For example, a sound can be played to alert the user to a prioritized document. Alternatively, an agent or automated assistant can be opened (*e.g.,* interactive display wizard). That is, the agent can appear on a display screen, to notify the user of the prioritized document. Furthermore, the prioritized document can be opened, such as being displayed on the screen. The document can receive focus. This can also include sizing the document based on its priority, such that the higher the priority of the document, the larger the window in which it is displayed, and/or centrally locating the document on the display based on its priority.

Referring now to Fig. 16, a diagram of a text generation and priorities system 100 in accordance with an aspect of the present invention. The system 100 includes a program 102 and a classifier 104. It is noted that the program 100 and the classifier 102 can include a computer program executed by a processor of a computer from a computer-readable medium thereof.

The program 102 generates a text for input into the classifier 104. The program includes an electronic mail program that receives e-mail, which then serve as the text. The classifier 104 generates a priority for the associated message. As described above, the classifier 104 can be a Bayesian classifier, a Support Vector Machine classifier, or

other type of classifier. The priority of the text output by the classifier 104 can then be utilized in conjunction with a cost-benefit analysis, as has been described, to effectuate further output and/or alerting based thereon.

Referring now next to Fig. 17, a diagram of an alternative alerting system 108 is illustrated. The system 108 of Fig.17 includes an alerting system 110. Not shown in Fig. 17 are the program 102 and the classifier 104. However, the alerting system 110 is operatively and/or communicatively coupled to the latter. The system 110 includes a computer program executed by a processor of a computer from a computer-readable medium thereof. The alerting system 110 is communicatively coupled to the Internet 112, for example, and can be the network by which the alerting system contacts an electronic device to alert the user to a prioritized text based on an alerting criteria, for example. The network is not limited to the Internet 112, however. Thus, the alerting system is able to alert the user of a prioritized text *via* contacting a pager 114, a cellular phone 116, or other electronic devices capable of receiving information from a network such as the Internet 112, and are described in more detail below.

Referring next to Fig 18, a diagram of other aspects of the present invention are illustrated. This can include a routing system 120, for example. The routing system 120 receives a prioritized text, and based on routing criteria, is able to reply to the sender of the text, in which case the system 120 is a replying mechanism. Also, based on the routing criteria, the system 120 can forward the text, for example, to a different e-mail address, in which case the system is a forwarding mechanism. The former may be useful when the user desires to indicate to the sender of a message that the user is not present, and thus may provide the sender with contact information as to how to reach the user. The latter may be useful when the user has e-mail access to a different e-mail address, such as a web-based email address, such that the user desires to be kept informed of high priority emails at the alternative address.

An alerting system 130, also depicted in Fig. 18, receives a prioritized document, and based on a predetermined criteria (*e.g.,* priority above an importance or urgency threshold), can display receive text, and/or provide a sound, as has been described. Of the documents that have been received by the system 130, and that have a priority greater than a predetermined threshold, for example, can be displayed as a prioritized list with associated priority labels and/or display formats adapted to the priority as described above.

The system 130 can include other functionality as well. For example, a priorities-oriented viewer (not shown) can be provided that performs as a view onto a user's e-mail store, in terms of its ability to filter by priority. The viewer can enable summaries of messages to be sorted in a list by priority score, for example. The viewer can also enable a user to sort and view only those messages that remain unread as an option. The viewer can also enable users to scope the sorting of messages by priority within some scoped time period, and to change the scope or periods being considered. For example, a user can specify that the viewer only display e-mail from today. Alternatively, the user can specify that the priorities list span two days, one week, or all the messages in the in-box. The viewer can also let the user prune from the display messages below a user-specified minimal threshold.

Furthermore, beyond the use of qualitatively different sounds for low, medium, and high priorities, one or more scalar parameters can be utilized that define the manner by which an alerting sound is rendered. The parameters can be functions of an inferred priority. Such parameters include variables that such as the volume of the alerting sound, for example, to continuous changes in the modulation or resonance of the sound.

Other functionality can be provided to users to define thresholds among different ranges of uncertainty, and wherein users can specify multiple options involving the automation of the sizing and centering of messages within each range. For example, A "While Away" briefer can be included to give the user a summary of messages that have arrived while a user was away or busy with another application. The system can be configured to bring up a summary of e-mail directed by priority values when a user returns after being away, or comes back to the viewer after working with the system in a quiet mode. The automated text summarizer can be controlled to decrease a summarization level of the text of messages as a function of the priority of the document. That is, as documents increase in priority, they are less and less summarized in the summarized view. The priorities can also be utilized to color or add other annotations, such as priority flags, icons indicating level of priority, and a special priority field itself, to e-mail headers appearing in the display.

Furthermore, a user-defined threshold can be utilized on the priority assigned to messages to set up a temporary interaction context that is active for some portion of time following an alert or summary that a message has arrived exceeding the threshold. Following an alert, and lasting for the time period that an interaction context is active,

predetermined gestures are enabled to give the user access to more details about the message that was associated with the alert. Such gestures include a simple wiggle of the mouse from side to side, for example. As an example, an audio alert may indicate that an incoming message has exceeded some threshold of criticality. The user can then wiggle the mouse quickly from side to side to see details about the message that led to the alert. The amount of time that such an interaction context is active can be made a function of the priority of the message, or can be user-defined.

Turning now to Fig. 19, a system 200 illustrates a notification architecture and priorities system according to an aspect of the present invention. The system 200 includes a context analyzer 222, a notification manager 224 (also referred to as an event broker), one or more notification sources 1 through N, 226, 227, 228, a priorities system 230 which can operate as a notification source and one or more notification sinks, 1 through M, 236, 237, 238, wherein N an M are integers, respectively. The sources are also referred to as event publishers, while the sinks are also referred to as event subscribers. There can be any number of sinks and sources. In general, the notification manager 224 conveys notifications, which are also referred to as events or alerts, from the sources 226-228 to the sinks 236-238, based in part on parametric information stored in and/or accessed by the context analyzer 222.

The context analyzer 222 stores/analyzes information regarding variables and parameters of a user that influence notification decision-making. For example, the parameters may include contextual information, such as the user's typical locations and attentional focus or activities per the time of day and the day of the week, and additional parameters conditioned on such parameters, such as the devices users tend to have access to in different locations. Such parameters may also be functions of observations made autonomously *via* one or more sensors. For example, one or more profiles (not shown) may be selected or modified based on information about a user's location as can be provided by a global positioning system (GPS) subsystem, on information about the type of device being used and/or the pattern of usage of the device, and the last time a device of a particular type was accessed by the user. Furthermore, as is described in more detail below, automated inference may also be employed, to dynamically infer parameters or states such as location and attention. The profile parameters may be stored as a user profile that can be edited by the user. Beyond relying on sets of predefined profiles or dynamic inference, the notification

architecture can enable users to specify in real-time his or her state, such as the user not being available except for important notifications for the next "x" hours, or until a given time, for example.

The parameters can also include default notification preference parameters regarding a user's preference as to being disturbed by notifications of different types in different settings, which can be used as the basis from which to make notification decisions by the notification manager 224, and upon which a user can initiate changes. The parameters may include default parameters as to how the user wishes to be notified in different situations (*e.g.,* such as by cell phone, by pager). The parameters can include such assessments as the costs of disruption associated with being alerted by different modes in different settings. This can include contextual parameters indicating the likelihoods that the user is in different locations, the likelihoods that different devices are available, and the likelihoods of his or her attentional status at a given time, as well as notification parameters indicating how the user desires to be notified at a given time.

Information stored by the context analyzer 222, according to one aspect of the present invention is inclusive of contextual information determined by the analyzer. The contextual information is determined by the analyzer 222 by discerning the user's location and attentional status based on one or more contextual information sources (not shown), as is described in more detail in a later section of the description. The context analyzer 222, for example, may be able to determine with precision the actual location of the user *via* a global positioning system (GPS) that is a part of a user's car or cell phone. The analyzer may also employ a statistical model to determine the likelihood that the user is in a given state of attention by considering background assessments and/or observations gathered through considering such information as the type of day, the time of day, the data in the user's calendar, and observations about the user's activity. The given state of attention can include whether the user is open to receiving notification, busy and not open to receiving notification, and can include other considerations such as weekdays, weekends, holidays, and/or other occasions/periods.

The sources 226-228, 230 generate notifications intended for the user and/or other entity. For example, the sources 226-230 may include communications, such as Internet and network-based communications, local desktop computer-based

communications, and telephony communications, as well as software services, such as intelligent help, background queries, and automated scheduling. Notification sources are defined generally herein as that which generates events, which can also be referred to as notifications and alerts, intended to alert a user, or a proxy for the user, about information, services, and/or a system or world event. A notification source can also be referred to as an event source.

For example, e-mail may be generated as notifications by an the priorities system 230 such that it is prioritized, wherein an application program or system generating the notification assigns the e-mail with a relative priority corresponding to the likely importance or urgency of the e-mail to the user. The e-mail may also be sent without regard to the relative importance to the user. Desktop-centric notifications can include an automated dialog with the goal of alerting a user to a potentially valuable service that he or she may desire to execute (*e.g.,* scheduling from a message), information that the user may desire to review (*e.g.,* derived from a background query), or errors and/or other alerts generated by a desktop computer. Internet-related services can include notifications including information that the user has subscribed to, such as headlines of current news every so often, and stock quotes, for example.

Other notifications can include background queries (*e.g.,* while the user is working, text that the user is currently working on may be reviewed, such that background queries regarding the text are formulated and issued to search engines), and scheduling tasks from a scheduling and/or other program. Notification sources 226-230 can themselves be push-type or pull-type sources. Push-type sources are those that automatically generate and send information without a corresponding request, such as headline news and other Internet-related services that send information automatically after being subscribed to. Pull-type sources are those that send information in response to a request, such as e-mail being received after a mail server is polled. Still other notification sources include the following:

- e-mail desktop applications such as calendar systems;
- computer systems (*e.g.,* that may alert the user with messages that information about alerts about system activity or problems);
- Internet-related services, appointment information, scheduling queries;

- changes in documents or numbers of certain kinds of documents in one or more shared folders;
- availability of new documents in response to standing or persistent queries for information; and/or,
- information sources for information about people and their presence, their change in location, their proximity (*e.g.*, let me know when I am traveling if another coworker or friend is within 10 miles of me"), or their availability (*e.g.*, let me know when Steve is available for a conversation and is near a high-speed link that can support full video teleconferencing").

The notification sinks 236-238 are able to provide notifications to the user. For example, such notification sinks 236-238 can include computers, such as desktop and/or laptop computers, handheld computers, cell phones, landline phones, pagers, automotive-based computers, as well as other systems/applications as can be appreciated. It is noted that some of the sinks 236-238 can convey notifications more richly than other of the sinks. For example, a desktop computer typically has speakers and a relatively large color display coupled thereto, as well as having a higher bandwidth for receiving information when coupled to a local network or to the Internet. Thus, notifications can be conveyed by the desktop computer to the user in a relatively rich manner. Conversely, many cell phones have a smaller display that can be black and white, and receive information at a relatively lower bandwidth, for example. Correspondingly, the information associated with notifications conveyed by cell phones may generally be shorter and geared towards the phone's interface capabilities, for example. Thus, the content of a notification may differ depending on whether it is to be sent to a cell phone or a desktop computer. According to one aspect of the present invention, a notification sink can refer to that which subscribes, *via* an event subscription service, for example, to events or notifications.

The notification manager 224 accesses the information stored and/or determined by the context analyzer, and determines which of the notifications received from the sources 226-230 to convey to which of the sinks 236-238. Furthermore, the notification manager 224 can determine how the notification is to be conveyed, depending on which of the sinks 236-238 has been selected to send the information to. For example, it may

be determined that notifications should be summarized before being provided to a selected sinks 236-238.

The invention is not limited to how the manager 224 makes its decisions as to which of the notifications to convey to which of the notification sinks, and in what manner the notifications are conveyed. In accordance with one aspect, a decision-theoretic analysis can be utilized. For example, the notification manager 224 can be adapted to infer important uncertainties about variables including a user's location, attention, device availability, and amount of time until the user will access the information if there were no alert. The notification manager 224 can then make notification decisions about whether to alert a user to a notification, and if so, the nature of the summarization and the suitable device or devices to employ for relaying the notification. In general, the notification manager 224 determines the net expected value of a notification. In doing so, it can consider the following:

- the fidelity and transmission reliability of each available notification sink;
- the attentional cost of disturbing the user;
- the novelty of the information to the user;
- the time until the user will review the information on his or her own;
- the potentially context-sensitive value of the information; and/or,
- the increasing and/or decreasing value over time of the information contained within the notification.

Inferences made about uncertainties thus may be generated as expected likelihoods of values such as the cost of disruption to the user with the use of a particular mode of a particular device given some attentional state of the user, for example. The notification manager 224 can make decisions as to one or more of the following:

- what the user is currently attending to and doing (based on, for example, contextual information);
- where the user currently is;
- how important the information is;
- what is the cost of deferring the notification;

- how distracting would a notification be;
- what is the likelihood of getting through to the user; and,
- what is the fidelity loss associated with the use of a specific mode of a given notification sink.

Therefore, the notification manager 224 can perform an analysis, such as a decision-theoretic analysis, of pending and active notifications, evaluates context-dependent variables provided by information sinks and sources, and infers selected uncertainties, such as the time until a user is likely to review information and the user's location and current attentional state.

As used herein, inference refers generally to the process of reasoning about or inferring states of the system 200, environment, and/or user from a set of observations as captured *via* events and/or data. Inference can be employed to identify a specific context or action, or can generate a probability distribution over states, for example. The inference can be probabilistic – that is, the computation of a probability distribution over states of interest based on a consideration of data and events. Inference can also refer to techniques employed for composing higher-level events from a set of events and/or data. Such inference results in the construction of new events or actions from a set of observed events and/or stored event data, whether or not the events are correlated in close temporal proximity, and whether the events and data come from one or several event and data sources.

Furthermore, the notification manager 224 can access information stored in a user profile by the context analyzer 222 in lieu of or to support a personalized decision-theoretic analysis. For example, the user profile may indicate that at a given time, the user prefers to be notified *via* a pager, and only if the notification has a predetermined importance level. Such information can be utilized as a baseline from which to start a decision-theoretic analysis, or can be the manner by which the notification manager 224 determines how and whether to notify the user.

According to one aspect of the present invention, the notification platform architecture 200 can be configured as a layer that resides over an eventing or messaging infrastructure. However, the invention is not limited to any particular eventing infrastructure. Such eventing and messaging systems and protocols can include:

- HyperText Transport Protocol (HTTP), or HTTP extensions as known within the art;
- Simple Object Access Protocol (SOAP), as known within the art;
- Windows Management Instrumentation (WMI), as known within the art;
- Jini, as known within the art; and,
- substantially any type of communications protocols, such as those based on packet-switching protocols, for example.

Furthermore, the architecture can be configured as a layer that resides over a flexible distributed computational infrastructure, as can be appreciated by those of ordinary skill within the art. Thus, the notification platform architecture can utilize an underlying infrastructure as a manner by which sources send notifications, alerts and events, and as a manner by which sinks receive notifications, alerts and events, for example. The present invention is not so limited, however.

Referring now to Fig. 20, the context analyzer 222 of the notification architecture described in the previous section of the description is depicted in more detail. The context analyzer 222 as illustrated in Fig. 20 includes a user notification preferences store 252, a user context module 254 that includes a user context profile store 255, and a whiteboard 257. The context analyzer 222 according to one aspect of the invention can be implemented as one or more computer programs executable by a processor of a computer from a machine-readable medium thereof, such as a memory.

The preferences store 252 stores notification parameters for a user, such as default notification preferences for the user, such as a user profile, which can be edited and modified by the user. The preferences store 252 can be considered as that which stores information on parameters that influence how a user is to be notified. The user context module 254 determines a user's current context, based on one or more context information sources 260 as published to the whiteboard 257, for example. The user context profile store 255 stores context parameters for a user, such as the default context settings for the user, which can be edited and modified by the user. That is, the user context module 254 provides a best guess or estimate about a user's current context information by accessing information from the profile store 255 and/or updating a prior set of beliefs in the store 255 with live sensing, *via* the one or more context

sources 260. The profile store 255 can be considered as that which stores *a priori* where a user is, and what the user is doing, for example.

The user context profile store 255 can be a pre-assessed and/or predefined user profile that captures such information as a deterministic or probabilistic profile. The profile can be of typical locations, activities, device availabilities, and costs and values of different classes of notification as a function of such observations as time of day, type of day, and user interactions with one or more devices. The type of day can include weekdays, weekends and holidays, for example. The user context module 254 can then actively determine or infer aspects of the user's context or state, such as the user's current or future location and attentional state. Furthermore, actual states of context can be accessed directly from the context information sources 260 *via* the whiteboard 257, and/or, can be inferred from a variety of such observations through inferential methods such as Bayesian reasoning as is described in more detail below.

The context information sources 260 provide information to the context module 254 *via* the whiteboard 257 regarding the user's attentional state and location, from which the module 254 can make a determination as to the user's current context (*e.g.,* the user's current attentional state and location). Furthermore, the invention is not limited to a particular number or type of context sources 260, nor the type of information inferred or accessed by the user context module 254. However, the context sources 260 can include multiple desktop information and events, such as mouse information, keyboard information, application information (*e.g.,* which application is currently receiving the focus of the user), ambient sound and utterance information, text information in the windows on the desktop, for example. The whiteboard 257 can include a common storage area, to which the context information sources 260 can publish information, and from which multiple components, including sources and the context module 254 can access this information. An event, also referred to as a notification or alert, generally can include information about an observation about one or more states of the world. Such states can include the status of system components, the activity of a user, and/or a measurement about the environment. Furthermore, events can be generated by an active polling of a measuring device and/or source of events, by the receipt of information that is sent on a change, and/or per a constant or varying event heartbeat.

Other types of context sources 260 includes personal-information manager (PIM) information of the user, which generally can provide scheduling information regarding the schedule of the user, for example. The current time of day, as well as the user's location – for example, determined by a global positioning system (GPS), and/or a user's access of a cell phone, PDA, or a laptop that can be locationally determined – are also types of context sources 260. Furthermore, real-time mobile device usage is a type of context source 260. For example, a mobile device such as a cell phone may be able to determine if it is currently being accessed by the user, as well as device orientation and tilt (*e.g.,* indicating information regarding device usage as well), and acceleration and speed (*e.g.,* indicating information as to whether the user is moving or not).

Referring now to Fig. 21, the notification sources described above are illustrated in more detail. The notification sources 226-228, and/or 230 generally generate notifications that are conveyed to the notification manager 224, which determines when notifications should occur, and, if so, which of the notifications should be conveyed to which of the notification sinks 236-238 and in what order.

According to one aspect of the present invention, notification sources 226-228 can have one or more of the following parameters within a standard description of attributes and relationships, referred to herein as a notification source schema or source schema. It is noted that schema can be provided for sources, for sinks, and for context-information sources, described above. Such schemas provide declarative information about different components and can enable the sources 226-228, 230, the notification manager 224, the sinks 236-238, and the context analyzer 222 to share semantic information with one another. Thus, different schemas provide information about the nature, urgency, and device signaling modalities associated with notification. That is, schema can be defined generally as a collection of classes and relationships among classes that defines the structure of notifications and events, containing information including event or notification class, source, target, event or notification semantics, ontological content information, observational reliability, and substantially any quality-of-service attributes, for example.

Parameters (not shown) for notification source schema can include one or more of: message class; relevance; importance; time criticality; novelty; content attributes; fidelity tradeoffs, and/or source information summary information. The message class

for a notification generated by a notification source indicates the type of communication of the notification, such as e-mail, instant message, numerical financial update, and desktop service, for example. The relevance for a notification generated by notification sources indicates a likelihood that the information contained within the notification is relevant, for one or more specified contexts. For example, the relevance can be provided by a logical flag, indicating whether the source is relevant for a given context or not. The novelty of the notification indicates the likelihood that the user already knows the information contained within the notification. That is, the novelty is whether the information is new to the user, over time (indicating if the user knows the information now, and when, if ever, the user will learn the information in the future without being alerted to it).

Fidelity tradeoffs associated with the notification indicate the loss of value of the information within the notification that can result from different forms of specified allowed truncation and/or summarization, for example. Such truncation and/or summarization may be required for the notification to be conveyed to certain types of notification sinks 236-238 that may have bandwidth and/or other limitations preventing the sinks from receiving the full notification as originally generated. Fidelity in general refers to the nature and/or degree of completeness of the original content associated with a notification. For example, a long e-mail message may be truncated, or otherwise summarized to a maximum of 100 characters allowed by a cell phone, incurring a loss of fidelity. Likewise, an original message containing text and graphics content suffers a loss in fidelity when transmitted *via* a device that only has text capabilities. In addition, a device may only be able to depict a portion of the full resolution available from the source. Fidelity tradeoffs refer to a set of fidelity preferences of a source stated either in terms of orderings (*e.g.*, rendering importance in order of graphics first, then sound) and/or costs functions that indicate how the total value of the content of the notification diminishes with changes in fidelity. For example, a fidelity tradeoff can describe how the full value associated with the transmission of a complete e-mail message changes with increasingly greater amounts of truncation. Content attributes, for example, can include a summary of the nature of the content, representing such information as whether the core message includes text, graphics, and audio components. The content itself is the actual graphics, text, and/or audio that make up the message content of the notification.

The importance of a notification refers to the value of the information contained in the notification to the user, assuming the information is relevant in a current context. For example, the importance can be expressed as a dollar value of the information's worth to the user. Time criticality indicates time-dependent change in the value of information contained in a notification – that is, how the value of the information changes over time. In most but not all cases, the value of the information of a notification decays with time. This is illustrated in the diagram of Fig. 22. A graph 280 depicts the utility of a notification mapped over time. At the point 284 within the graph, representing the initial time, the importance of the notification is indicated, while the curve 286 indicates the decay of the utility over time.

Referring back to Fig. 21, default attributes and schema templates for different notification sources or source types may be made available in notification source profiles stored in the user notification preferences store, such as the store 252 of Fig. 20. Such default templates can be directed to override values provided by notification sources or to provide attributes when they are missing from schema provided by the sources. Source summary information enables a source to post general summaries of the status of information and potential notifications available from a source. For example, source summary information from a messaging source may include information about the total number of unread messages that are at least some priority, the status of attempts by people to communicate with a user, and/or other summary information.

The notification sinks 236-238 can be substantially any device or application by which the user or other entity can be notified of information contained in notifications. The choice as to which sink or sinks are to be employed to convey a particular notification is determined by the notification manager 224.

Notification sinks 236-238 may have one or more of the following parameters provided within a schema. These parameters may include a device class; modes of signaling (alerting); and, for the associated mode, fidelity/rendering capabilities, transmission reliability, actual cost of communication, and/or attentional cost of disruption, for example. For devices that are adapted for parameterized control of alerting attributes, the schema for the devices can additionally include a description of the alerting attributes and parameters for controlling the attributes, and functions by which other attributes (*e.g.*, transmission reliability, cost of distribution) change with

the different settings of the alerting attributes. The schema for notification sinks provides for the manner by which the notification devices communicate semantic information about their nature and capabilities with the notification manager 224 and/or other components of the system. Default attributes and schema templates for different device types can be made available in device profiles stored in the user notification preferences store, such as the store 252 of Fig. 20 as described in the previous section. Such default templates can be directed to override values provided by devices or to provide attributes when they are missing from schema provided by such devices.

Each of the schema parameters is now described in term. The class of the device refers to the type of the device such as a cell phone, a desktop computer, and a laptop computer, for example. The class can also be more general, such as a mobile or a stationery device. The modes of signaling refer to the manner in which a given device can alert the user about a notification. Devices may have one or more notification modes. For example, a cell phone may only vibrate, may only ring with some volume, and/or it can both vibrate and ring. Furthermore, a desktop display for an alerting system can be decomposed into several discrete modes (*e.g.,* a small notification window in the upper right hand of the display vs. a small thumbnail at the top of the screen – with or without an audio herald). Beyond being limited to a set of predefined behaviors, a device can enable modes with alerting attributes that are functions of parameters, as part of a device definition. Such continuous alerting parameters for a mode represent such controls as the volume at which an alert is played at the desktop, rings on a cell phone, and the size of an alerting window, for example.

The transmission reliability for a mode of a notification sink 236-238 indicates the likelihood that the user will receive the communicated alert about a notification, which is conveyed to the user *via* the sink with that mode. As transmission reliability may be dependent on the device availability and context of the user, the transmission reliability of different modes of a device can be conditioned on such contextual attributes as the location and attention of a user. Transmission reliability for one or more unique contextual states, defined by the cross product of such attributes as unique locations and unique attentional states, defined as disjunctions created as abstractions of such attributes (*e.g.,* for any location away from the home, and any time period after 8 am and before noon), can also be specified. For example, depending on where the user currently is, information transmitted to a cell phone may not always reach the user,

particularly if the user is in a region with intermittent coverage, or where the user would not tend to have a cell phone in this location (*e.g.*, family holiday). Contexts can also influence transmission reliability because of ambient noise and/or other masking or distracting properties of the context.

The actual cost of communication indicates the actual cost of communicating the information to the user when contained within a notification that is conveyed to the sink. For example, this cost can include the fees associated with a cell phone transmission. The cost of disruption includes the attentional costs associated with the disruption associated with the alert employed by the particular mode of a device, in a particular context. Attentional costs are typically sensitive to the specific focus of attention of the user. The fidelity/rendering capability is a description of the text, graphics, and audio/tactile capabilities of a device, also given a mode. For example, a cell phone's text limit may be 100 characters for any single message, and the phone may have no graphics capabilities.

Turning now to Fig. 23, an interface 290 illustrates context specifications selectable by a user that can be utilized by the context analyzer 222 in determining a user's current context. The determination of user context by direct specification by the user, and/or a user-modifiable profile, is described. The context of the user can include the attentional focus of the user – that is, whether the user is currently amenable to receiving notification alerts – as well as the user's current location. The present invention is not so limited, however.

Direct specification of context by the user enables the user to indicate whether or not he or she is available to receive alerts, and where the user desires to receive them. A default profile (not shown) can be employed to indicate a default attentional state, and a default location wherein the user can receive the alerts. The default profile can be modified by the user as desired.

Referring to Fig. 23, the interface 290 illustrates how direct specification of context can be implemented, according to an aspect of the present invention. A window 291, for example, has an attentional focus section 292 and a location section 294. In the focus section 292, the user can check one or more check boxes 296, for example, indicating whether the user is always available to receive alerts; whether the user is never available to receive alerts; and, whether the user is only available to

receive alerts that has an importance level greater than a predetermined threshold. It is to be appreciated that other availability selections can be provided. As depicted in Fig. 23, a threshold can be measured in dollars, but this is for exemplary purposes only, and the invention is not so limited. The user can increase the threshold in the box 298 by directly entering a new value, or by increasing or decreasing the threshold *via* arrows 300.

In the location section 294, the user can check one or more of the check boxes 302, to indicate where the user desires to have alerts conveyed. For example, the user can have alerts conveyed at the desktop, by e-mail, at a laptop, on a cell phone, in his or her car, on a pager, or on a personal digital assistant (PDA) device, and so forth. It is to be appreciated that these are examples only, however, and the invention itself is not so limited.

The window 291, wherein there can be preset defaults for the checkboxes 296 and the box 298 of the section 292 and the checkboxes 302 of the section 294, can be considered a default user profile. The profile is user modifiable in that the user can override the default selections with his or her own desired selections. Other types of profiles can also be utilized in accordance with the invention.

Referring now to Fig. 24, a determination of user context by direct measurement, for example, using one or more sensors, is illustrated in accordance with the present invention. The context of the user can include the user's attentional focus, as well as his or her current location. The invention itself is not so limited, however. Direct measurement of context indicates that sensor(s) can be employed to detect whether the user is currently amenable to receiving alerts, and to detect where the user currently is. According to one aspect of the present invention, an inferential analysis in conjunction with direct measurement can be utilized to determine user context, as is described in a later section of the description.

Referring to Fig. 24, a system 310 in which direct measurement of user context can be achieved is illustrated. The system 310 includes a context analyzer 312, and communicatively coupled thereto a number of sensors 314-320, namely, a cell phone 314, a video camera 315, a microphone 316, a keyboard 317, a PDA 318, a vehicle 319, and a GPS 320, for example. The sensors 314-320 depicted in Fig. 24 are for exemplary purposes only, and do not represent a limitation or a restriction on the

invention itself. The term sensor as used herein is a general and overly encompassing term, meaning any device or manner by which the context analyzer 312 can determine what the user's current attentional focus is, and/or what the user's current location is.

For example, if the user has the cell phone 314 on, this can indicate that the user can receive alerts on the cell phone 314. However, if the user is currently talking on the cell phone 314, this can indicate that the user has his or her attentional focus on something else (namely, the current phone call), such that the user should not presently be disturbed with a notification alert. The video camera 315 can, for example, be in the user's office, to detect whether the user is in his or her office (*viz.*, the user's location), and whether others are also in his or her office, suggesting a meeting with them, such that the user should not be disturbed (*viz.*, the user's focus). Similarly, the microphone 316 can also be in the user's office, to detect whether the user is talking to someone else, such that the user should not be disturbed, is typing on the keyboard (*e.g.*, *via* the sounds emanating therefrom), such that the user should also not be presently disturbed. The keyboard 317 can also be employed to determine if the user is currently typing thereon, such that, for example, if the user is typing very quickly, this may indicate that the user is focused on a computer-related activity, and should not be unduly disturbed (and, also can indicate that the user is in fact in his or her office).

If the PDA device 318 is being accessed by the user, this can indicate that the user is able to receive alerts at the device 318 – that is, the location at which notifications should be conveyed is wherever the device 318 is located. The device 318 can also be utilized to determine the user's current attentional focus. The vehicle 319 can be utilized to determine whether the user is currently in the vehicle – that is, if the vehicle is currently being operated by the user. Furthermore, the speed of the vehicle can be considered, for example, to determine what the user's focus is. If the speed is greater than a predetermined speed, for instance, then it may be determined that the user is focused on driving, and should not be bothered with notification alerts. The GPS device 320 can also be employed to ascertain the user's current location, as known within the art.

In the following section of the detailed description, a determination of user context according to user-modifiable rules is described. The context of the user can include the user's attentional focus, as well as his or her current location. The invention is not so limited, however. Determining context *via* rules indicates that a hierarchical

set of if-then rules can be followed to determine the user's location and/or attentional focus.

Referring to Fig. 25, a diagram illustrates an exemplary hierarchical ordered set of rules 330. The set of rules 330 depicts rules 332, 333, 334, 335, 336, 337 and 338, for example. It is noted that other rules may be similarly configured. As illustrated in Fig. 25, rules 333 and 334 are subordinate to 332, while rule 334 is subordinate to rule 333, and rule 338 is subordinate to rule 338. The rules are ordered in that rule 332 is first tested; if found true, then rule 333 is tested, and if rule 333 is found true, then rule 334 is tested, and so forth. If rule 333 is found false, then rule 335 is tested. If rule 332 is found false, then rule 336 is tested, which if found false, causes testing of rule 337, which if found true causes testing of rule 138. The rules are desirably user creatable and/or modifiable. Otherwise-type rules can also be included in the set of rules 330 (*e.g.*, where if an if-then rule is found false, then the otherwise rule is controlling).

Thus, a set of rules can be constructed by the user such that the user's context is determined. For example, with respect to location, the set of rules can be such that a first rule tests whether the current day is a weekday. If it is, then a second rule subordinate to the first rule tests whether the current time is between 9 a.m. and 5 p.m. If it is, then the second rule indicates that the user is located in his or her office, otherwise the user is at home. If the first rule is found to be false – that is, the current day is a weekend and not a weekday – then an otherwise rule may state that the user is at home. It is noted that this example is not meant to be a restrictive or limiting example on the invention itself, wherein one or more other rules may also be similarly configured.

In the following section of the description, a determination of user context by inferential analysis, such as by employing a statistical and/or Bayesian model, is described. It is noted that context determination *via* inferential analysis can rely in some aspects on other determinations, such as direct measurement *via* sensor(s), as has been described. Inferential analysis as used herein refers to using an inference process(es) on a number of input variables, to yield an output variable(s), namely, the current context of the user. The analysis can include in one aspect utilization of a statistical model and/or a Bayesian model.

Referring to Fig. 26, a diagram of a system 340 is illustrated in which inferential analysis is performed by an inferential engine 342 to determine a user's context 344, according to an aspect of the present invention. The engine 342 is in one aspect a computer program executed by a processor of a computer from a computer-readable medium thereof, such as a memory. The user context 344 can be considered the output variable of the engine 342.

The engine 342 can process one or more input variables to make a context decision. Such input variables can include one or more sensor(s) 348, such as the sensor(s) that have been described in conjunction with a direct measurement approach for context determination in a previous section of the description, as well as the current time and day, as represented by a clock 350, and a calendar 352, as may be accessed in a user's scheduling or personal-information manager (PIM) computer program, and/or on the user's PDA device, for example. Other input variables can also be considered besides those illustrated in Fig. 26. The variables of Fig. 26 are not meant to be a limitation or a restriction on the invention itself.

Referring now to Figs. 27 and 28, an exemplary inferential model, such as provided by a statistical and/or Bayesian model that can be executed by the inferential engine described above is illustrated in accordance with the present invention. In general, a computer system can be somewhat uncertain about details of a user's state. Thus, probabilistic models can be constructed that can make inferences about a user's attention or other state under uncertainty. Bayesian models can infer a probability distribution over a user's focus of attention. Such states of attention can be formulated as a set of prototypical situations or more abstract representations of a set of distinct classes of cognitive challenges being addressed by a user. Alternatively, models can be formulated that make inferences about a continuous measure of attentional focus, and/or models that directly infer a probability distribution over the cost of interruption for different types of notifications.

Bayesian networks may be employed that can infer the probability of alternate activity contexts or states based on a set of observations about a user's activity and location. As an example, Fig. 27 displays a Bayesian network 354 for inferring a user's focus of attention for a single time period. States of a variable, Focus of Attention 356, refer to desktop and non-desktop contexts. Exemplary attentional contexts considered in the model include situation awareness, catching up, nonspecific background tasks,

focused content generation or review, light content generation or review, browsing documents, meeting in office, meeting out of office, listening to presentation, private time, family time, personal focus, casual conversation and travel, for example. The Bayesian network 354 indicates that a user's current attention and location are influenced by the user's scheduled appointments 358, the time of day 360, and the proximity of deadlines 362. The probability distribution over a user's attention is also in influenced by summaries of the status of ambient acoustical signals 364 monitored in a user's office, for example. Segments of the ambient acoustical signal 164 over time provide clues/inputs about the presence of activity and conversation. Status and configuration of software applications and the ongoing stream of user activity generated by a user interacting with a computer also provide sources of evidence about a user's attention.

As portrayed in the network 354, a software application currently at top-level focus 366 in an operating system or other environment influences the nature of the user's focus and task, and the status of a user's attention and the application at focus together influence computer-centric activities. Such activity includes the stream of user activity built from sequences of mouse and keyboard actions and higher-level patterns of application usage over broader time horizons. Such patterns include e-mail-centric and Word-processor centric, and referring to prototypical classes of activity involving the way multiple applications are interleaved.

Fig. 28 illustrates a Bayesian model 368 of a user's attentional focus among context variables at different periods of time. A set of Markov temporal dependencies is illustrated by the model 368, wherein past states of context variables are considered in present determinations of the user's state. In real-time, such Bayesian models 368 consider information provided by an online calendar, for example, and a stream of observations about room acoustics and user activity as reported by an event sensing system (not shown), and continues to provide inferential results about the probability distribution of a user's attention.

Figs. 29 and 30 illustrate methodologies for providing portions of a notification architecture such as a context analyzer and a notification manager in accordance the present invention. While, for purposes of simplicity of explanation, the methodologies are shown and described as a series of acts, it is to be understood and appreciated that the present invention is not limited by the order of acts, as some acts may, in

accordance with the present invention, occur in different orders and/or concurrently with other acts from that shown and described herein. For example, those skilled in the art will understand and appreciate that a methodology could alternatively be represented as a series of interrelated states or events, such as in a state diagram. Moreover, not all illustrated acts may be required to implement a methodology in accordance with the present invention.

Referring to Fig. 29, a flowchart diagram 370 illustrates determining a user's context in accordance with the present invention. The process includes determining the user's location in 371, and the user's focus in 372. These acts can be accomplished by one or more of the approaches described previously. For example, a profile can be employed; a user can specify his or her context; direct measurement of context can be utilized; a set of rules can be followed; an inferential analysis, such as *via* a Bayesian or a statistical model, can also be performed. It is to be appreciated that other analysis can be employed to determine a user's context. For example, there can be an integrated video camera source that notes if someone is front of the computer and whether or not he or she is looking at the computer. It is noted, however, that the system can operate with or without a camera. For all of the sources, the system can operate with substantially any input source available, not requiring any particular source to inference about context. Furthermore, in other aspects, there can be integrated accelerometers, microphones, and proximity detectors on small PDA's that give a sense of a user's location and attention.

Referring now to Fig. 30, a flowchart diagram 373 illustrates a decision process for a notification manager in accordance with an aspect of the present invention. At 374, one or more notification sources generate notifications, which are received by a notification manager. At 375, a context analyzer generates/determines context information regarding the user, which in 376 is received by the notification manager. That is, according to one aspect of the present invention, at 375, the context analyzer accesses a user contextual information profile that indicates the user's current attentional status and location, and/or assesses real-time information regarding the user's current attentional status and location from one or more contextual information sources, as has been described in the previous sections of the description.

At 377, the notification manager determines which of the notifications to convey to which of the notification sinks, based in part on the context information

received from the context analyzer. The notification manager also makes determinations based on information regarding notification parameters of the user as stored by the context analyzer. That is, according to one aspect, in 377, the manager performs a decision-theoretic analysis as to whether a user should be alerted for a given notification, and how the user should be notified. As will be described in more detail below, decision-theoretic and/or heuristic analysis, determinations and policies may be employed at 377. Notification parameters regarding the user can be utilized to personalize the analysis by filling in missing values or by overwriting parameters provided in the schema of sources or sinks. Notification preferences can also provide policies (*e.g.,* heuristic) that are employed in lieu of the decision-theoretic analysis. Based on this determination, the notification manager conveys the notifications to the sinks at 378.

Various aspects of the invention have been described herein thus far as applicable to users. However, the invention itself is not so limited. That is, the invention is applicable to substantially any type of entity, including users. Other types of entities include agents, processes, computer programs, threads, services, servers, computers, machines, companies, organizations, and/or businesses, for example. The agent, for example, may be a software agent, which can be generally defined as a computer program that performs a background task for a user and reports to the user when the task is done or some expected event has taken place. Still other types of entities are encompassed under the invention, as can be appreciated by those of ordinary skill within the art. For example, the context analyzer according to another aspect of the invention can be generalized as a component applicable to substantially any type of entity. As another example, notification sinks can generate notifications, alerts and events regarding entities other than users. Similarly, notification sinks can receive notifications, alerts and events regarding entities other than users.

In order to provide a context for the various aspects of the invention, Fig. 31 and the following discussion are intended to provide a brief, general description of a suitable computing environment in which the various aspects of the present invention may be implemented. While the invention has been described above in the general context of computer-executable instructions of a computer program that runs on a computer and/or computers, those skilled in the art will recognize that the invention also may be implemented in combination with other program modules. Generally,

program modules include routines, programs, components, data structures, *etc.* that perform particular tasks and/or implement particular abstract data types. Moreover, those skilled in the art will appreciate that the inventive methods may be practiced with other computer system configurations, including single-processor or multiprocessor computer systems, minicomputers, mainframe computers, as well as personal computers, hand-held computing devices, microprocessor-based or programmable consumer electronics, and the like. The illustrated aspects of the invention may also be practiced in distributed computing environments where tasks are performed by remote processing devices that are linked through a communications network. However, some, if not all aspects of the invention can be practiced on stand-alone computers. In a distributed computing environment, program modules may be located in both local and remote memory storage devices.

With reference to Fig. 31, an exemplary system for implementing the various aspects of the invention includes a computer 420, including a processing unit 421, a system memory 422, and a system bus 423 that couples various system components including the system memory to the processing unit 421. The processing unit 421 may be any of various commercially available processors. It is to be appreciated that dual microprocessors and other multi-processor architectures also may be employed as the processing unit 421.

The system bus may be any of several types of bus structure including a memory bus or memory controller, a peripheral bus, and a local bus using any of a variety of commercially available bus architectures. The system memory may include read only memory (ROM) 424 and random access memory (RAM) 425. A basic input/output system (BIOS), containing the basic routines that help to transfer information between elements within the computer 420, such as during start-up, is stored in ROM 424.

The computer 420 further includes a hard disk drive 427, a magnetic disk drive 428, *e.g.,* to read from or write to a removable disk 429, and an optical disk drive 430, *e.g.,* for reading from or writing to a CD-ROM disk 431 or to read from or write to other optical media. The hard disk drive 427, magnetic disk drive 428, and optical disk drive 430 are connected to the system bus 423 by a hard disk drive interface 432, a magnetic disk drive interface 433, and an optical drive interface 434, respectively. The drives and their associated computer-readable media provide nonvolatile storage of data, data structures, computer-executable instructions, etc. for the computer 420.

Although the description of computer-readable media above refers to a hard disk, a removable magnetic disk and a CD, it should be appreciated by those skilled in the art that other types of media which are readable by a computer, such as magnetic cassettes, flash memory cards, digital video disks, Bernoulli cartridges, and the like, may also be used in the exemplary operating environment, and further that any such media may contain computer-executable instructions for performing the methods of the present invention.

A number of program modules may be stored in the drives and RAM 425, including an operating system 435, one or more application programs 436, other program modules 437, and program data 438. It is noted that the operating system 435 in the illustrated computer may be substantially any suitable operating system.

A user may enter commands and information into the computer 420 through a keyboard 440 and a pointing device, such as a mouse 442. Other input devices (not shown) may include a microphone, a joystick, a game pad, a satellite dish, a scanner, or the like. These and other input devices are often connected to the processing unit 421 through a serial port interface 446 that is coupled to the system bus, but may be connected by other interfaces, such as a parallel port, a game port or a universal serial bus (USB). A monitor 447 or other type of display device is also connected to the system bus 423 *via* an interface, such as a video adapter 448. In addition to the monitor, computers typically include other peripheral output devices (not shown), such as speakers and printers.

The computer 420 may operate in a networked environment using logical connections to one or more remote computers, such as a remote computer 449. The remote computer 449 may be a workstation, a server computer, a router, a peer device or other common network node, and typically includes many or all of the elements described relative to the computer 420, although only a memory storage device 450 is illustrated in Fig. 31. The logical connections depicted in Fig. 31 may include a local area network (LAN) 451 and a wide area network (WAN) 452. Such networking environments are commonplace in offices, enterprise-wide computer networks, Intranets and the Internet.

When employed in a LAN networking environment, the computer 420 may be connected to the local network 451 through a network interface or adapter 453. When utilized in a WAN networking environment, the computer 420 generally may include a

modem 454, and/or is connected to a communications server on the LAN, and/or has other means for establishing communications over the wide area network 452, such as the Internet. The modem 454, which may be internal or external, may be connected to the system bus 423 *via* the serial port interface 446. In a networked environment, program modules depicted relative to the computer 420, or portions thereof, may be stored in the remote memory storage device. It will be appreciated that the network connections shown are exemplary and other means of establishing a communications link between the computers may be employed.

In accordance with the practices of persons skilled in the art of computer programming, the present invention has been described with reference to acts and symbolic representations of operations that are performed by a computer, such as the computer 420, unless otherwise indicated. Such acts and operations are sometimes referred to as being computer-executed. It will be appreciated that the acts and symbolically represented operations include the manipulation by the processing unit 421 of electrical signals representing data bits which causes a resulting transformation or reduction of the electrical signal representation, and the maintenance of data bits at memory locations in the memory system (including the system memory 422, hard drive 427, floppy disks 429, and CD-ROM 431) to thereby reconfigure or otherwise alter the computer system's operation, as well as other processing of signals. The memory locations wherein such data bits are maintained are physical locations that have particular electrical, magnetic, or optical properties corresponding to the data bits.

Referring to Fig. 32, a diagram of an exemplary computerized device 500 that can be employed in conjunction with various aspects of the present invention is illustrated. The computerized device 500 can be, for example, a desktop computer, a laptop computer, a personal digital assistant (PDA), a cell phone, *etc.*; the invention is not so limited. Those skilled in the art will appreciate that the invention may be practiced with other computer system configurations, including hand-held devices, multiprocessor systems, microprocessor-based or programmable consumer electronics, network PC's, minicomputers, mainframe computers, and the like. The invention may also be practiced in distributed computing environments wherein tasks are performed by remote processing devices that are linked through a communications network.

The device 500 includes one or more of the following components: processor(s) 502, memory 504, storage 506, a communications component 508, input device(s) 510,

a display 512, and output device(s) 514. It is noted, that for a particular instantiation of the device 100, one or more of these components may not be present. For example, a PDA may not have any output device(s) 514, while a cell phone may not have storage 506, *etc.* Thus, the description of the device 500 is to be utilized as an overview as to the types of components that typically reside within such a device 500, and is not meant as a limiting or exhaustive description of such computerized devices.

The processor(s) 502 may include a single central-processing unit (CPU), or a plurality of processing units, commonly referred to as a parallel processing environment. The memory 504 may include read only memory (ROM) and/or random access memory (RAM). The storage 506 may be any type of storage, such as fixed-media storage devices such as hard disk drives, flash or other non-volatile memory, as well as removable-media storage devices, such as tape drives, optical drives like CD-ROM's, floppy disk drives, *etc.* The storage and their associated computer-readable media provide non-volatile storage of computer-readable instructions, data structures, program modules and other data. It should be appreciated by those skilled in the art that any type of computer-readable media which can store data that is accessible by a computer, such as magnetic cassettes, flash memory cards, digital video disks, Bernoulli cartridges, random access memories (RAMs), read only memories (ROMs), and the like, may be employed.

Because the device 500 may operate in a network environment, such as the Internet, intranets, extranets, local-area networks (LAN's), wide-area networks (WAN's), *etc.*, a communications component 508 can be present in or attached to the device 500. Such a component 508 may be one or more of a network card, such as an Ethernet card, an analog modem, a cable modem, a digital subscriber loop (DSL) modem, an Integrated Services Digital Network (ISDN) adapter, *etc.*; the invention is not so limited. Furthermore, the input device(s) 510 are the mechanisms by which a user indicates input to the device 500. Such device(s) 510 include keyboards, pointing devices, microphones, joysticks, game pads, satellite dishes, scanners, *etc.* The display 512 is how the device 500 typically directs output to the user, and can include, for example, cathode-ray tube (CRT) display devices, flat-panel display (FPD) display devices, *etc.* In addition, the device 500 may indicate output to the user *via* other output device(s) 514, such as speakers, printers, *etc.*

What has been described above are preferred aspects of the present invention. It

is, of course, not possible to describe every conceivable combination of components or methodologies for purposes of describing the present invention, but one of ordinary skill in the art will recognize that many further combinations and permutations of the present invention are possible. Accordingly, the present invention is intended to embrace all such alterations, modifications and variations that fall within the spirit and scope of the appended claims.

Industrial Applicability

This invention has industrial applicability in the area of computers, computer software and information technologies.

Claims

What is claimed is:

1. A notification system, comprising:

a prioritizer to assign a priority to one or more received messages;

a notification platform that forwards the one or more received messages to a communications modality based upon an associated resource of the modality and the priority of the one or more received messages.

2. The system of claim 1, wherein the associated resource is at least one of a transmission bandwidth, a connection interval, a download time, and a storage capability.

3. The system of claim 1, wherein the one or more received messages are forwarded based upon a determined loss of value with time associated with a delayed review of message content.

4. The system of claim 3, wherein the one or more received messages are forwarded based upon a cost-benefit analysis in order that expected costs of the delayed review are minimized.

5. The system of claim 4, wherein the one or more received messages are forwarded in order of urgency based upon the priority.

6. The system of claim 4, wherein the one or more received messages are forwarded based upon a cost-benefit ratio, the cost-benefit ratio determined as the gain in value of forwarding the messages in the present versus forwarding the messages in the future divided by the size of the document.

7. The system of claim 4, wherein the one or more received messages are forwarded based upon truncation of the one or more received messages.

8. The system of claim 7, wherein the one or more received messages are truncated to a predetermined size.

9. The system of claim 8, wherein the truncated one or more received messages are utilized in the cost-benefit analysis.

10. The system of claim 4, wherein the one or more received messages are forwarded based upon a summarization of the one or more received messages.

11. The system of claim 10, wherein the summarization of the one or more received messages is based upon at least one of a user-specified policy for removing message attachments, dynamically removing attachments based upon a cost-benefit analysis of the value of the attachments, a threshold value of urgency such that messages assigned an urgency value above the threshold remain intact, and a policy enabling documents to remain substantially intact unless the document is above a predetermined size.

12. The system of claim 1, wherein the one or more messages are forwarded based upon a determined context of the user by the notification platform, the determined context including at least one of the time of day and the location of the user.

13. The system of claim 12, wherein the determined context is a probability that a user has already received at least some of the information associated with the one or more received messages.

14. The system of claim 13, wherein the determined context is computed as an expected value.

15. The system of claim 14, wherein the expected value is provided as information "j" contained in a message is expressed as:

p(User does not know j | E) * Value(j, j not already known) + [1- p(User does not know j | E)]* Value(j, j already known),

wherein Value(j, j already known) and Value(j, j not already known) refer to the value or urgencies of receiving information j when the information is already known and not known respectively, p is a probability, and E is evidence since an incident occurred.

16. The system of claim 15, wherein the incident is at least one of when a news story broke and how widespread the information is.

17. The system of claim 3, wherein the loss associated with delayed review of the one or more messages is described by at least one of a linear and a non-linear cost function.

18. The system of claim 17, wherein the cost function is represented as a value of transmitting the one or more messages now versus the value when the message will be observed in the future.

19. The system of claim 18, wherein the cost function is represented as a difference in value as: $\Delta\text{Value} = \text{Value}(t_{\text{delay, transmit}}) - \text{Value}(t_{\text{delay, no transmit}})$, wherein $t_{\text{delay, transmit}}$ is an estimated time delay the user will review the message now, and $t_{\text{delay, no transmit}}$ is an estimated time the user will observe the message in the future.

20. The system of claim 1, wherein the prioritizer is a classifier.

21. The system of claim 20, wherein the classifier is at least one of a Bayesian, Support Vector Machine (SVM), and decision tree classifier.

22. A method for communicating messages, comprising:

determining available resources associated with a modality;

prioritizing at least one message based on a predetermined criteria; and

transmitting the at least one message as prioritized to a user based upon the determined resources of the modality.

23. The method of claim 22, wherein prioritizing the at least one message further comprises, utilizing a user-determined set of rules to prioritize the at least one text.

24. The method of claim 22, wherein prioritizing the at least one message further comprises, determining a change in value with delayed review of information contained in respective messages.

25. The method of claim 22, wherein prioritizing the at least one message further comprises, performing a cost-benefit analysis.

26. The method of claim 22, wherein prioritizing the at least one message further comprises:

determining an expected loss of non-review of the message at a current time;

determining an expected cost of alerting the user of the message at the current time; and

upon determining that the expected loss of non-review exceeds the expected cost of alerting the user, concluding that the message should be transmitted to the user.

27. The method of claim 26, wherein determining the expected loss of non-review of the message at the current time considers the expected loss of non-review of the message at a future time when the user would otherwise review the message.

28. The method of claim 26, wherein determining the expected loss further comprises, determining a current expected rate of lost opportunity as a function of present time.

29. The method of claim 26, wherein determining the expected cost further comprises, determining a probability inference as to a likelihood of the user being present.

30. The method of claim 26, wherein determining the expected cost further comprises, utilizing at least one of a linear function and a non-linear function.

31. A system for communicating messages, comprising:

means for determining available resources associated with a modality;

means for prioritizing at least one message based on a predetermined criteria; and

means for transmitting the at least one message as prioritized to a user based upon the determined resources of the modality.

32. A machine-readable medium having instructions stored thereon for execution by a processor to perform a method for text transmission to a user given a constrained resource, comprising:

for each of at least one text,

determining an expected loss of non-review of the text at a current time;

determining an expected cost of alerting the user of the text at the current time;

upon determining that the expected loss of non-review exceeds the expected cost of alerting the user, determining that the text should be transmitted to the user; and

transmitting each of the at least one text for which it was determined that the text should be transmitted to the user.

33. A computerized system, comprising:

a prioritization mechanism designed to prioritize at least one text, given a constrained resource, based on a predetermined criteria; and

a device designed to transmit the at least one text as prioritized by the prioritization mechanism.

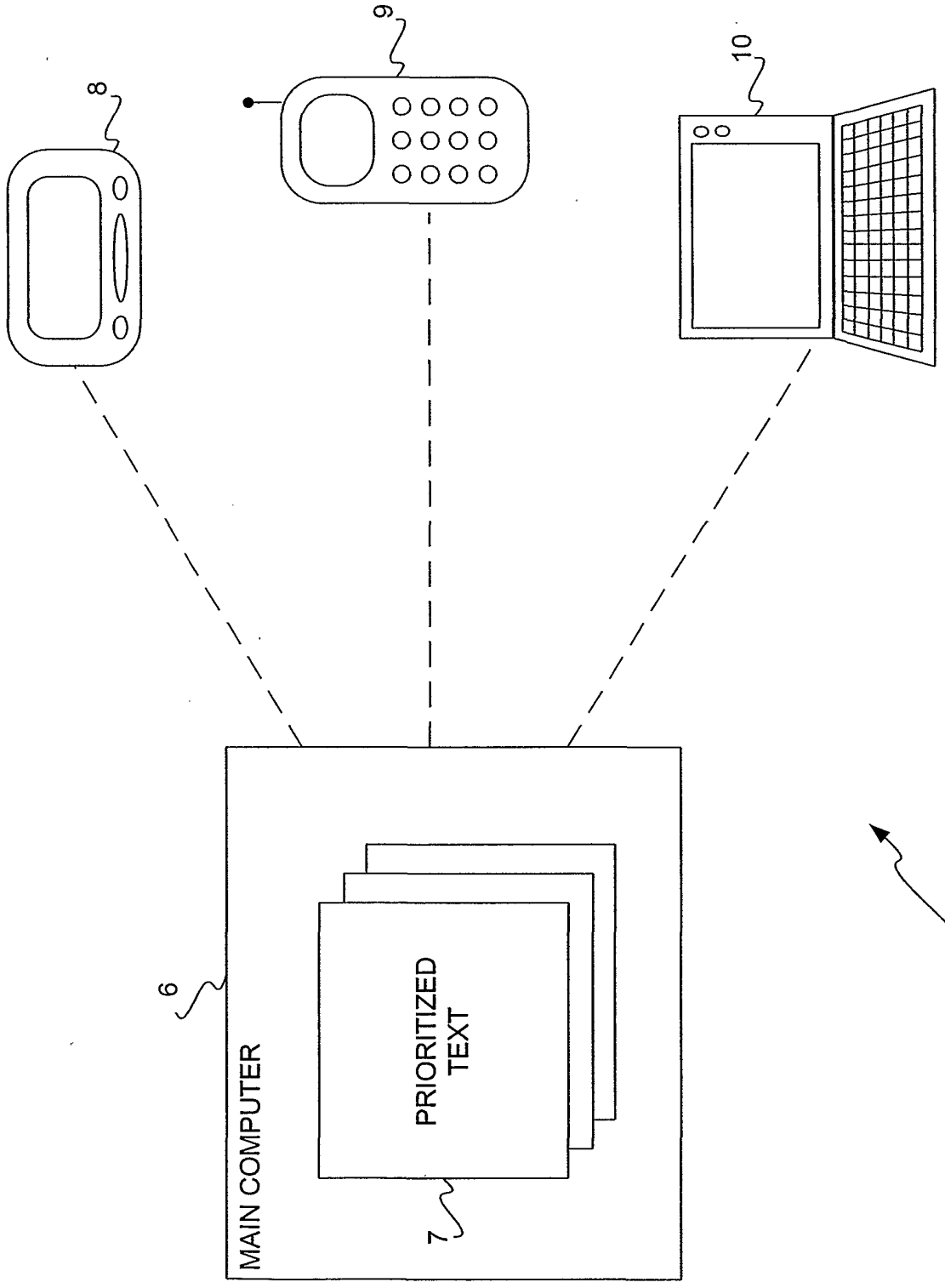


Fig. 1

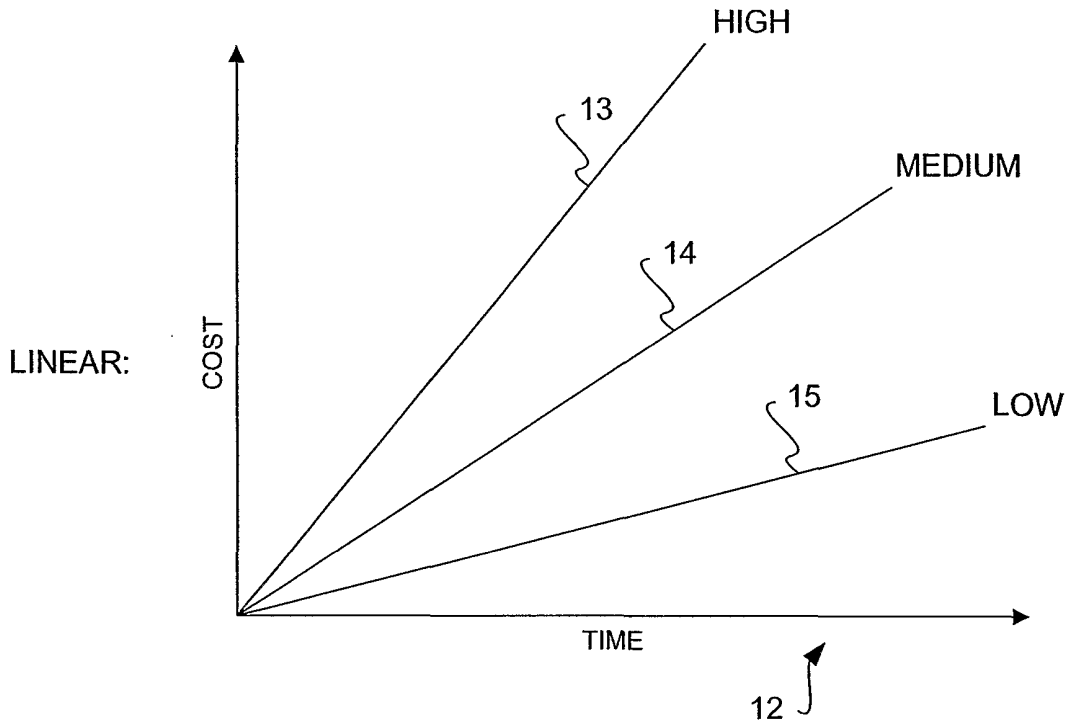

LINEAR:
HIGH
13
MEDIUM
14
15
LOW
COST
TIME
12

Fig. 2

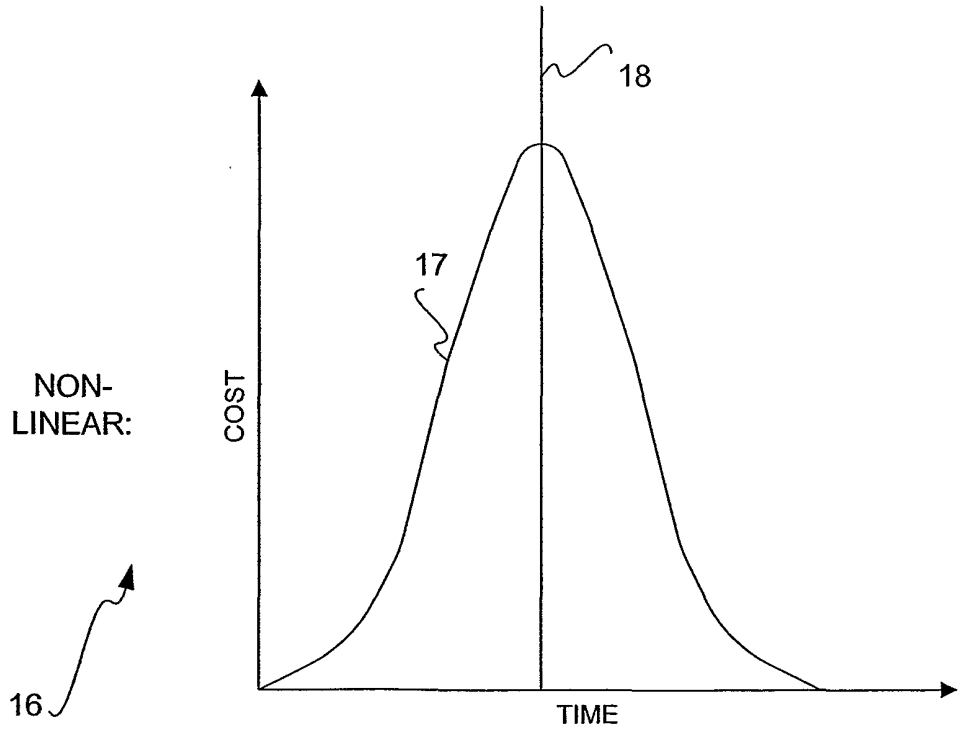

NON-
LINEAR:
18
17
COST
TIME
16

Fig. 3

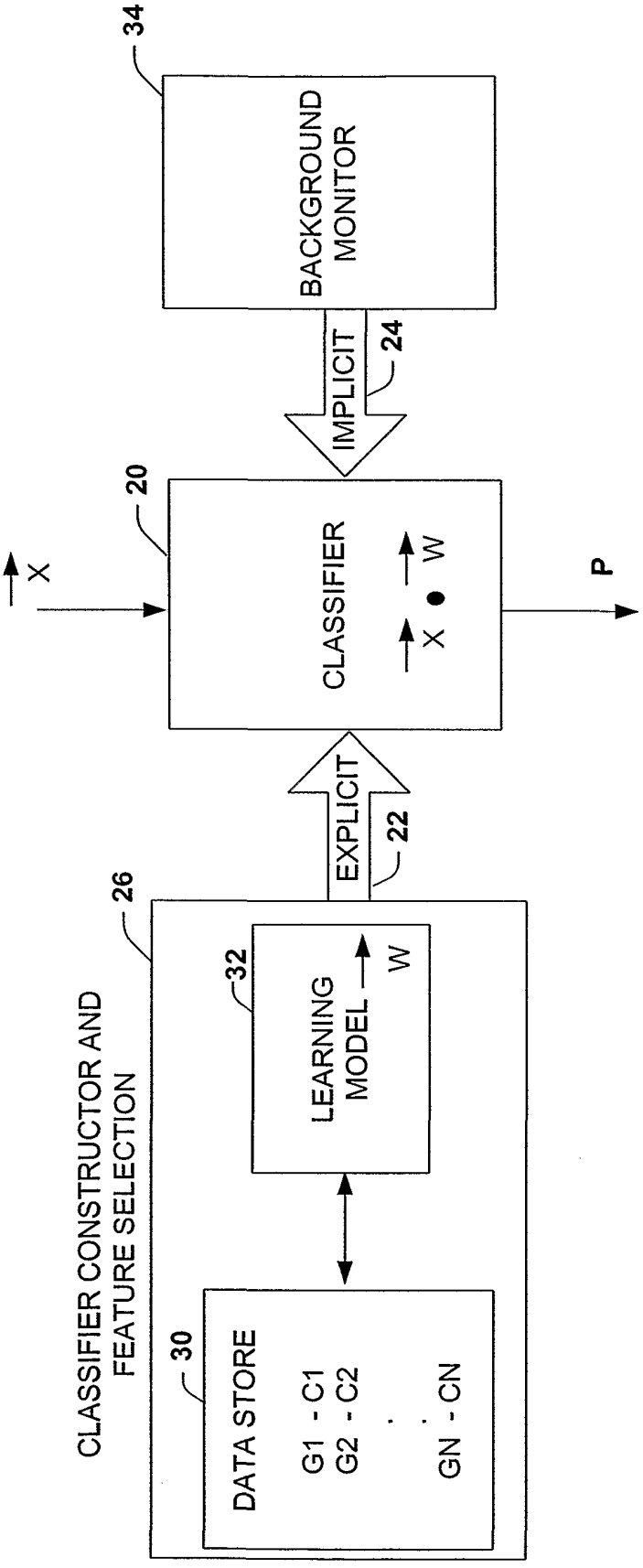

34
BACKGROUND MONITOR
IMPLICIT
24
20
X
CLASSIFIER
X • W
P
EXPLICIT
22
26
CLASSIFIER CONSTRUCTOR AND FEATURE SELECTION
32
LEARNING MODEL
W
30
DATA STORE
G1 - C1
G2 - C2
GN - CN

Fig. 4

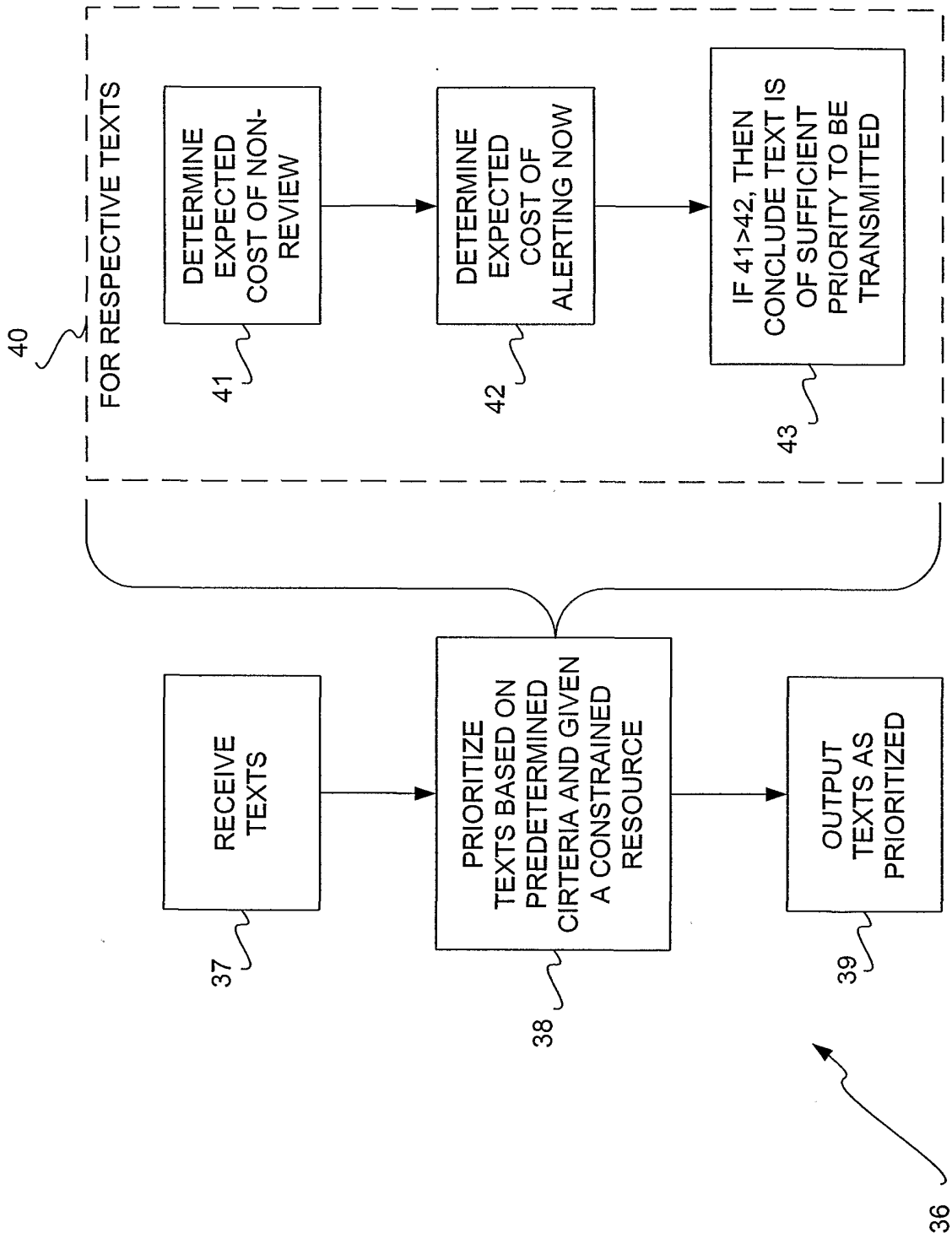

RECEIVE TEXTS
37
PRIORITIZE TEXTS BASED ON PREDETERMINED CIRTERIA AND GIVEN A CONSTRAINED RESOURCE
38
OUTPUT TEXTS AS PRIORITIZED
39
36
40
FOR RESPECTIVE TEXTS
DETERMINE EXPECTED COST OF NON-REVIEW
41
DETERMINE EXPECTED COST OF ALERTING NOW
42
IF 41>42, THEN CONCLUDE TEXT IS OF SUFFICIENT PRIORITY TO BE TRANSMITTED
43

Fig. 5

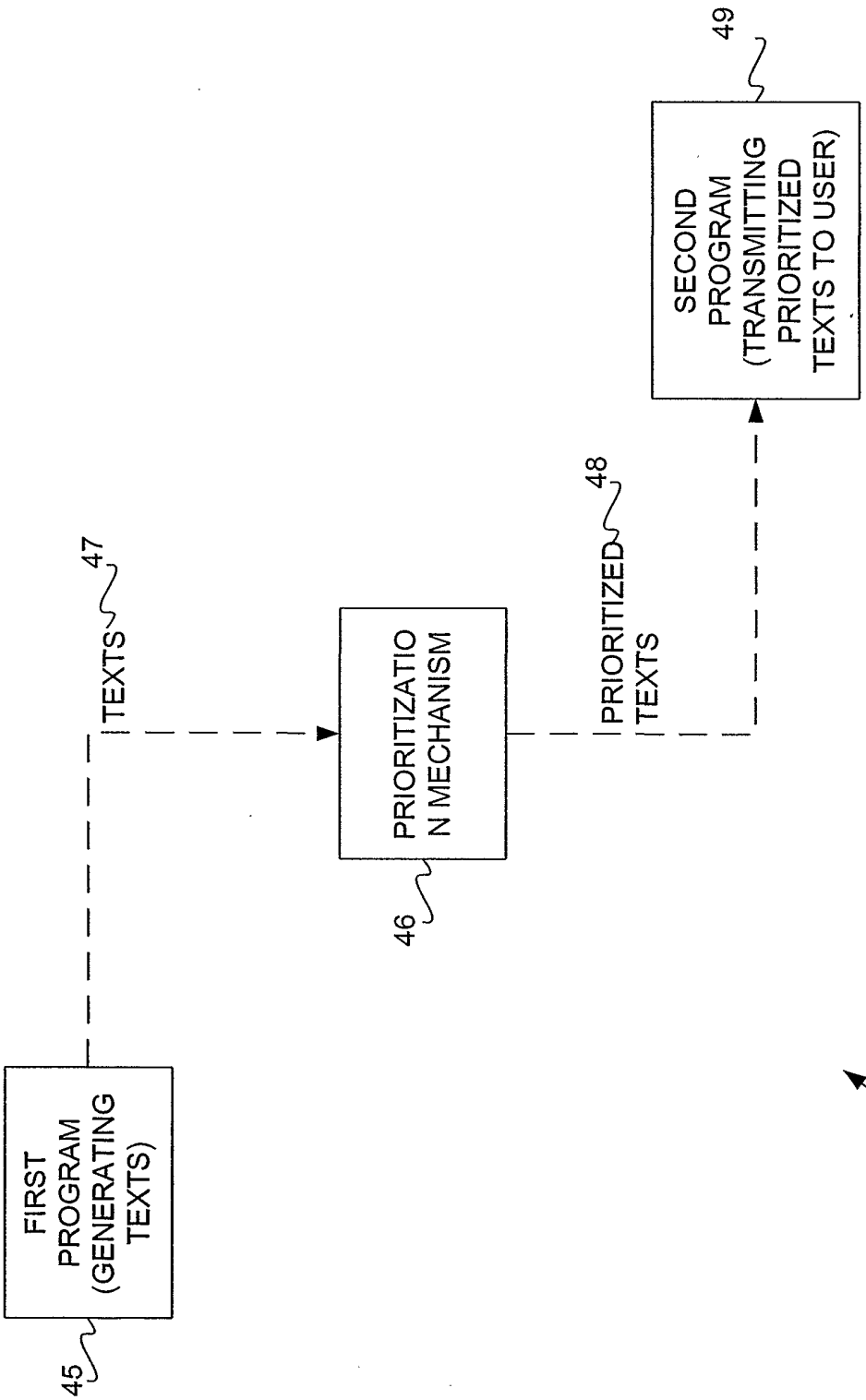

FIRST PROGRAM (GENERATING TEXTS)
45
TEXTS
47
PRIORITIZATIO N MECHANISM
46
PRIORITIZED TEXTS
48
SECOND PROGRAM (TRANSMITTING PRIORITIZED TEXTS TO USER)
49
44

Fig. 6

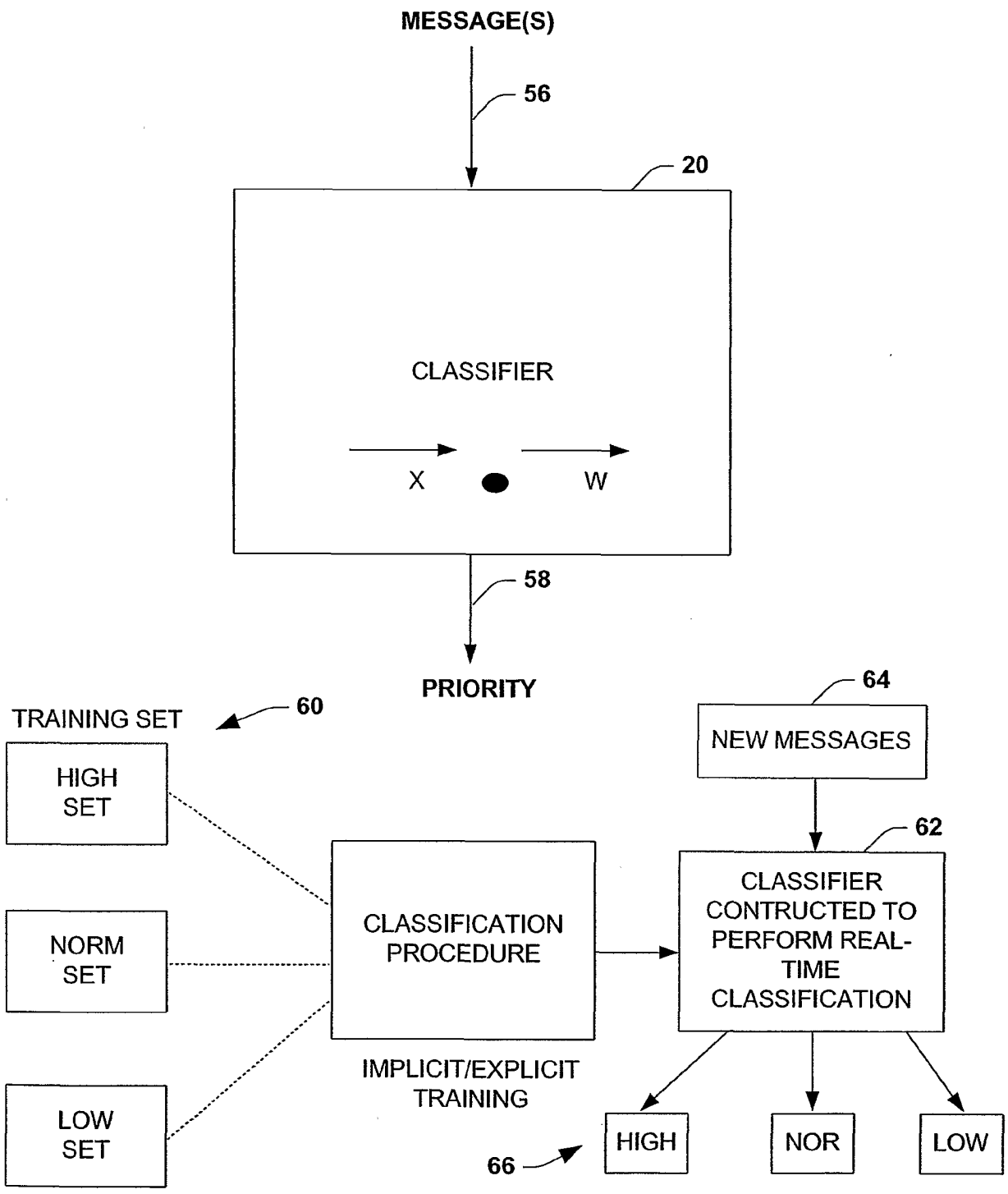

MESSAGE(S)
56
20
CLASSIFIER
X
W
58
PRIORITY
TRAINING SET
60
HIGH SET
NORM SET
LOW SET
CLASSIFICATION PROCEDURE
IMPLICIT/EXPLICIT TRAINING
66
64
NEW MESSAGES
62
CLASSIFIER CONTRUCTED TO PERFORM REAL-TIME CLASSIFICATION
HIGH
NOR
LOW

Fig. 7

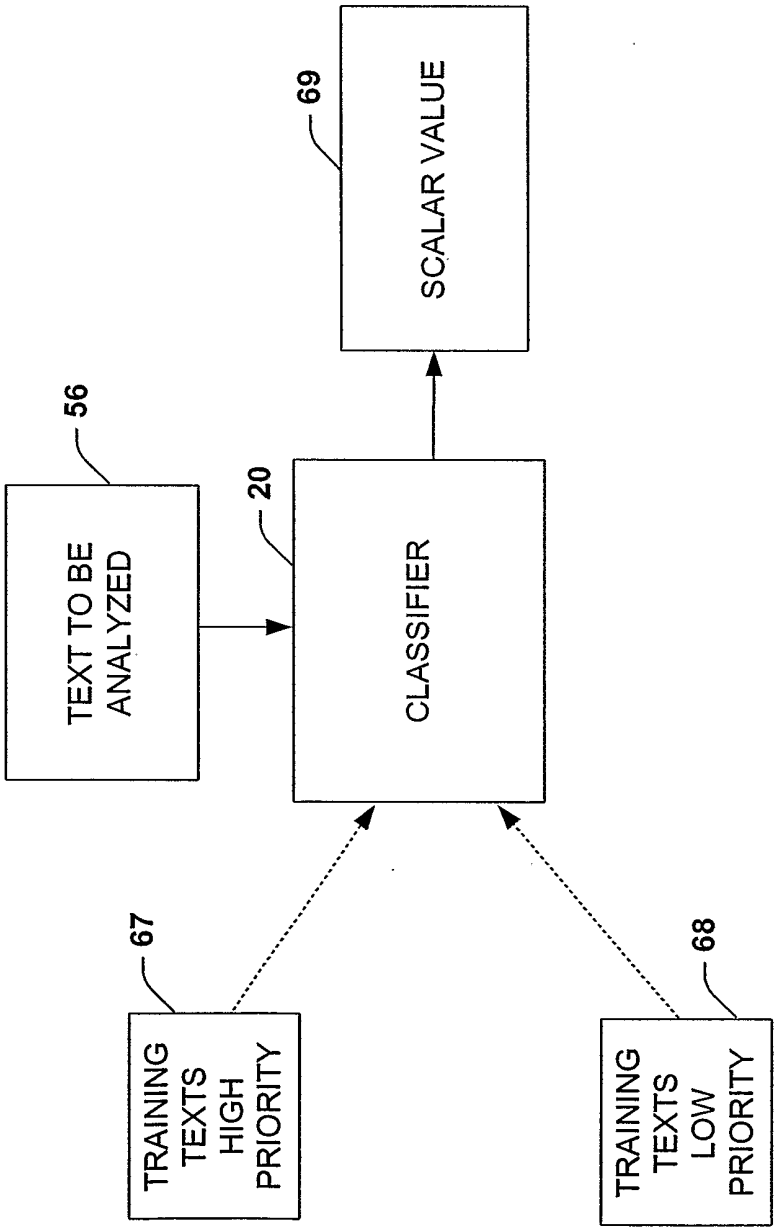

69
SCALAR VALUE
56
TEXT TO BE ANALYZED
20
CLASSIFIER
67
TRAINING TEXTS HIGH PRIORITY
68
TRAINING TEXTS LOW PRIORITY

Fig. 8

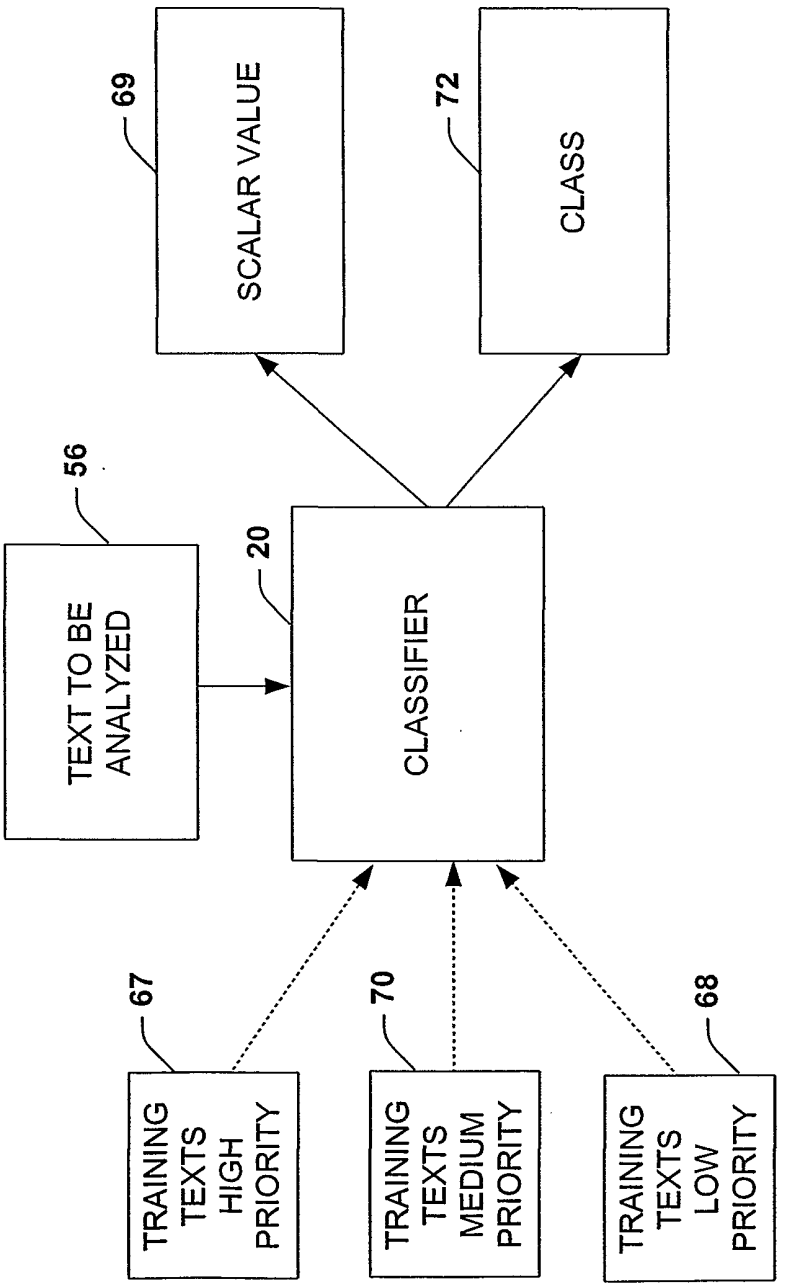

69
SCALAR VALUE
72
CLASS
56
TEXT TO BE ANALYZED
20
CLASSIFIER
67
TRAINING TEXTS HIGH PRIORITY
70
TRAINING TEXTS MEDIUM PRIORITY
68
TRAINING TEXTS LOW PRIORITY

Fig. 9

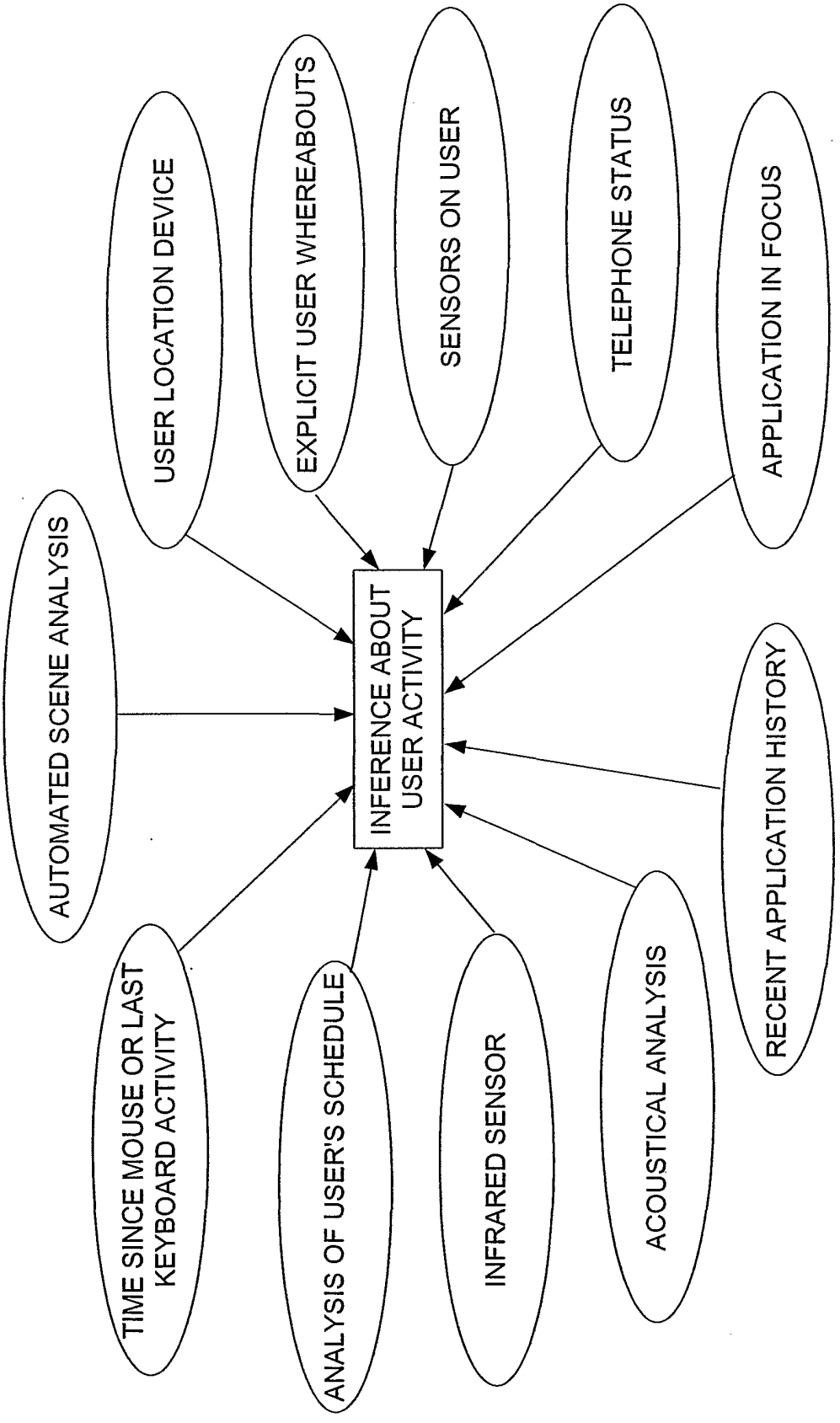

INFERENCE ABOUT USER ACTIVITY
AUTOMATED SCENE ANALYSIS
USER LOCATION DEVICE
EXPLICIT USER WHEREABOUTS
SENSORS ON USER
TELEPHONE STATUS
APPLICATION IN FOCUS
TIME SINCE MOUSE OR LAST KEYBOARD ACTIVITY
ANALYSIS OF USER'S SCHEDULE
INFRARED SENSOR
ACOUSTICAL ANALYSIS
RECENT APPLICATION HISTORY

Fig. 10

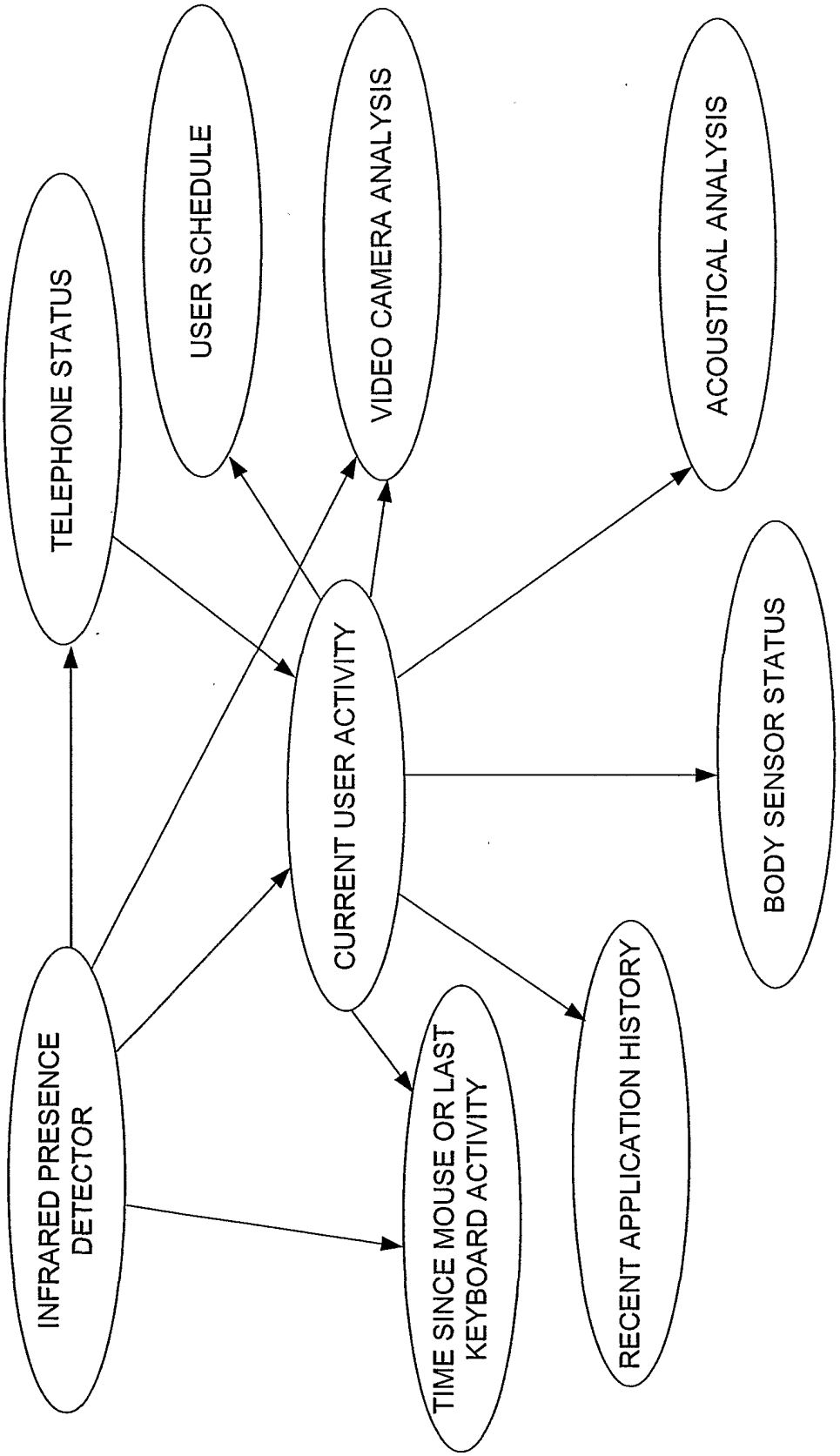

INFRARED PRESENCE DETECTOR
TELEPHONE STATUS
USER SCHEDULE
VIDEO CAMERA ANALYSIS
ACOUSTICAL ANALYSIS
CURRENT USER ACTIVITY
BODY SENSOR STATUS
TIME SINCE MOUSE OR LAST KEYBOARD ACTIVITY
RECENT APPLICATION HISTORY

Fig. 11

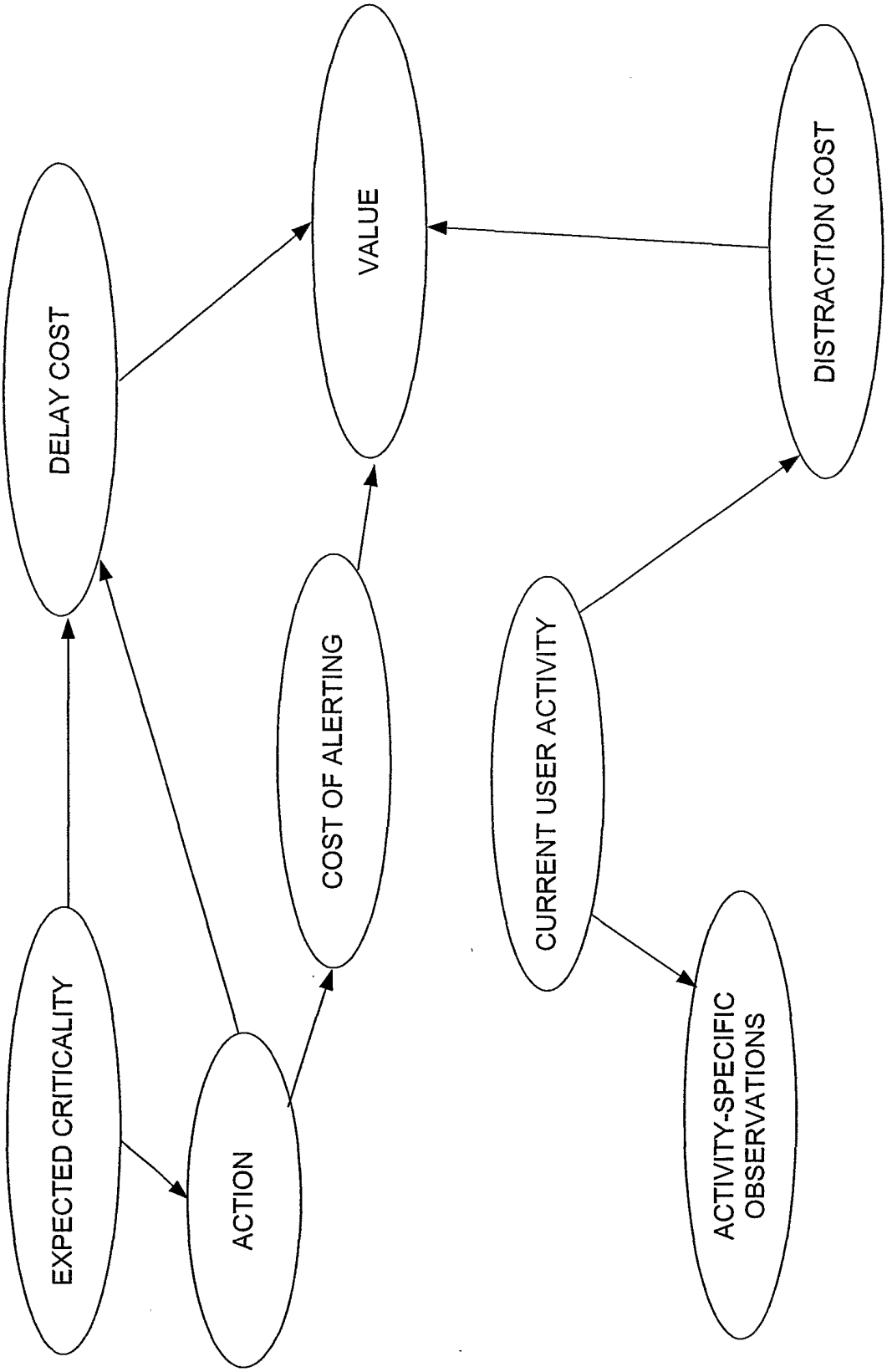

EXPECTED CRITICALITY
ACTION
DELAY COST
COST OF ALERTING
VALUE
CURRENT USER ACTIVITY
ACTIVITY-SPECIFIC OBSERVATIONS
DISTRACTION COST

Fig. 12

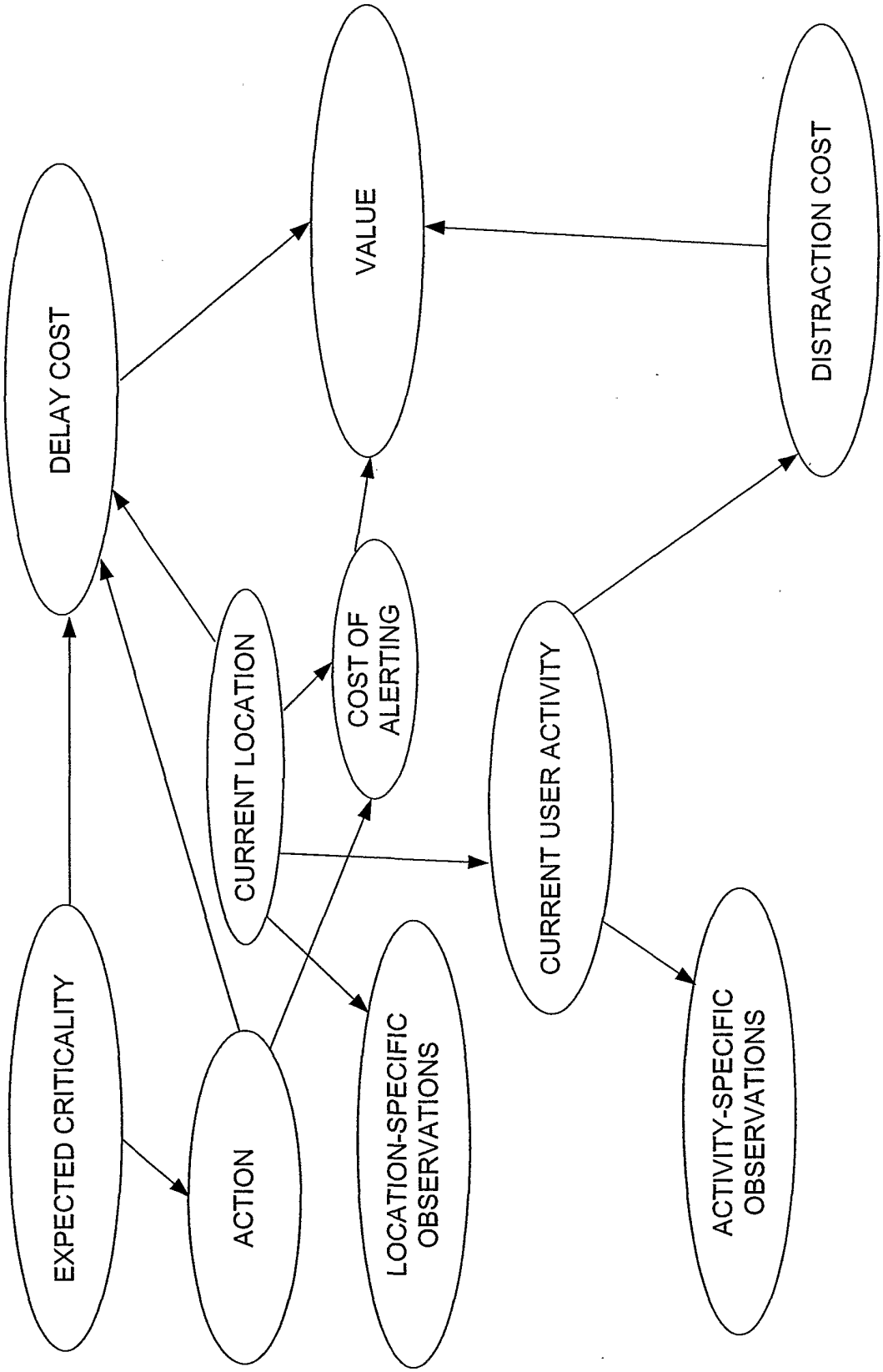

DELAY COST
VALUE
DISTRACTION COST
CURRENT LOCATION
COST OF ALERTING
CURRENT USER ACTIVITY
EXPECTED CRITICALITY
ACTION
LOCATION-SPECIFIC OBSERVATIONS
ACTIVITY-SPECIFIC OBSERVATIONS

Fig. 13

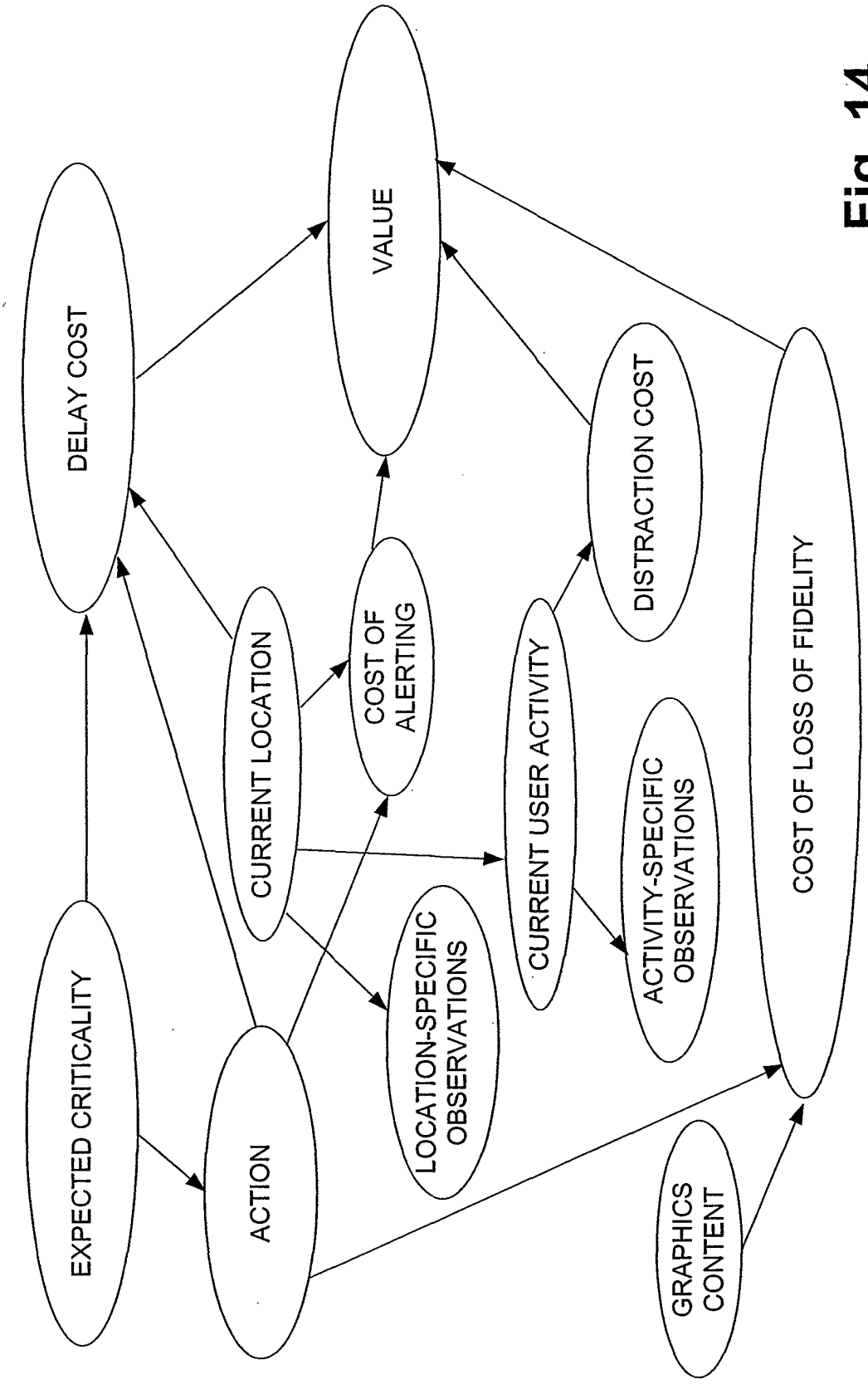

VALUE
DELAY COST
EXPECTED CRITICALITY
ACTION
CURRENT LOCATION
COST OF ALERTING
LOCATION-SPECIFIC OBSERVATIONS
CURRENT USER ACTIVITY
DISTRACTION COST
ACTIVITY-SPECIFIC OBSERVATIONS
COST OF LOSS OF FIDELITY
GRAPHICS CONTENT

Fig. 14

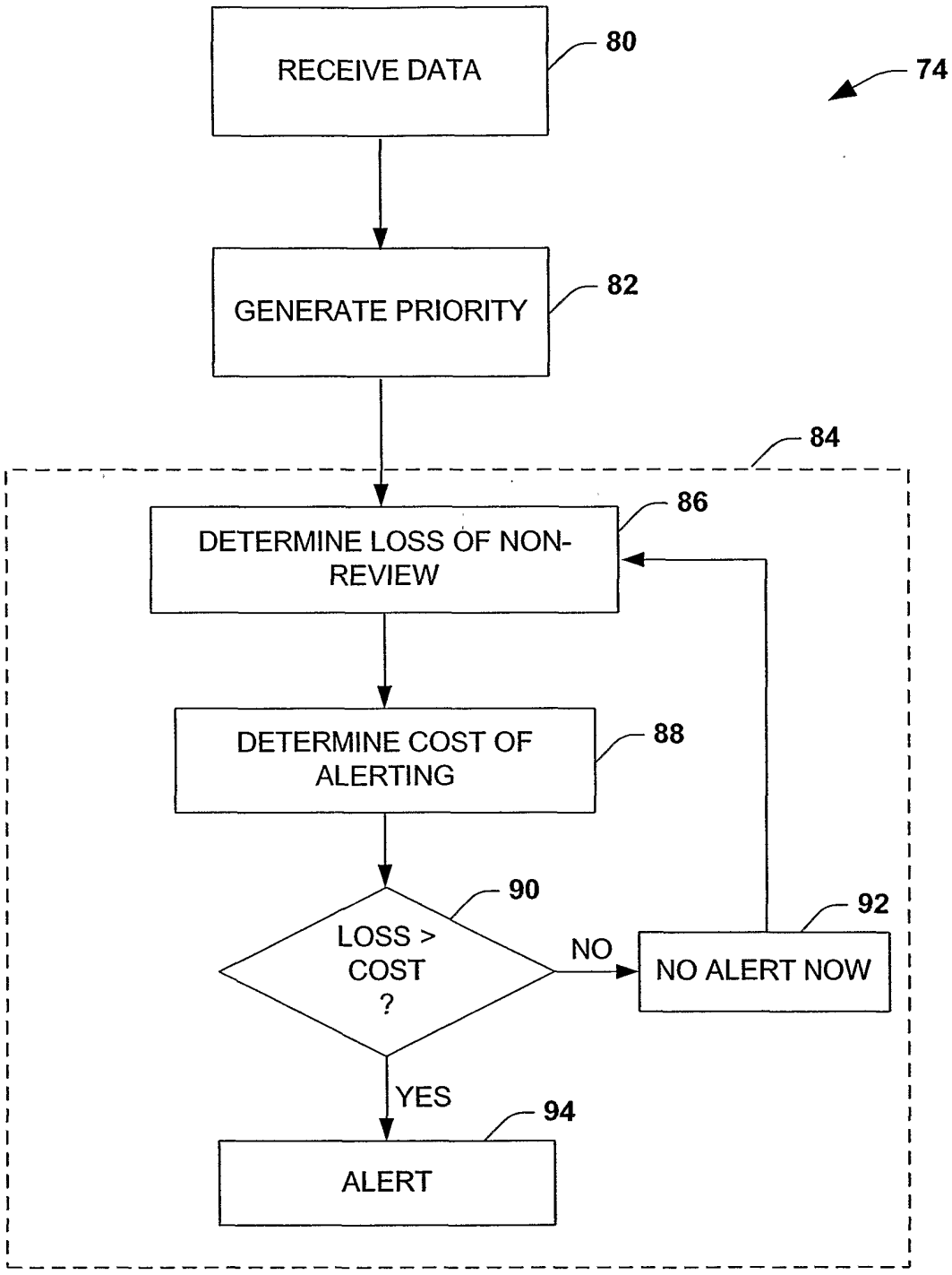

RECEIVE DATA
80
74
GENERATE PRIORITY
82
84
DETERMINE LOSS OF NON-REVIEW
86
DETERMINE COST OF ALERTING
88
LOSS > COST ?
90
NO
NO ALERT NOW
92
YES
ALERT
94

Fig. 15

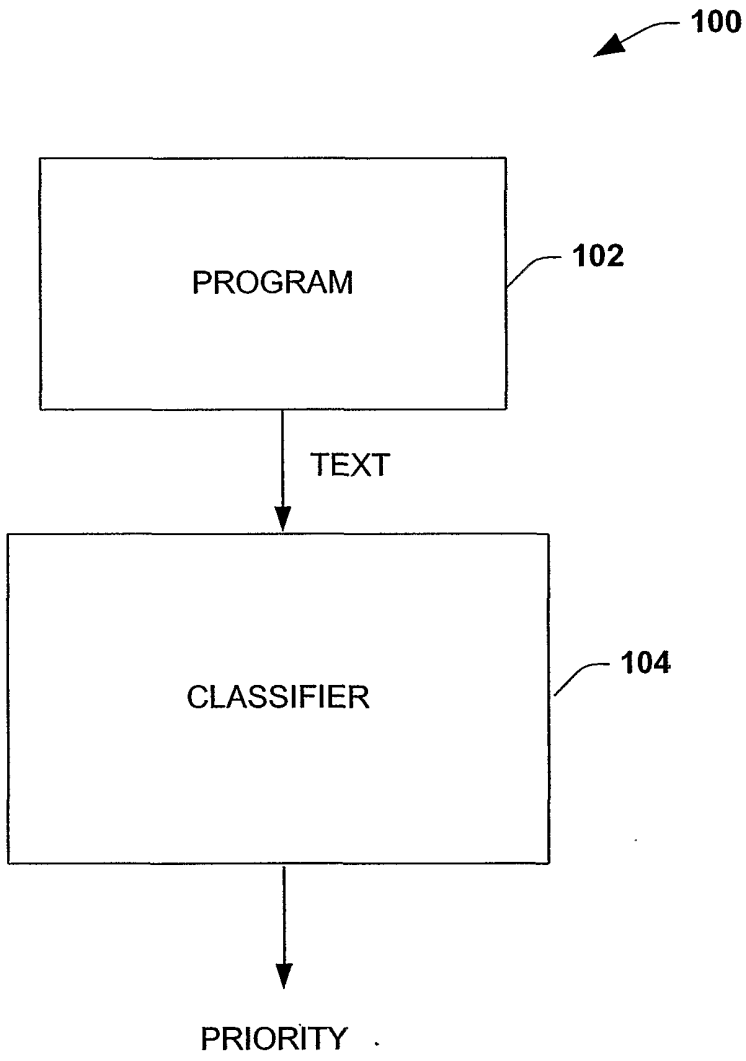


Fig. 16

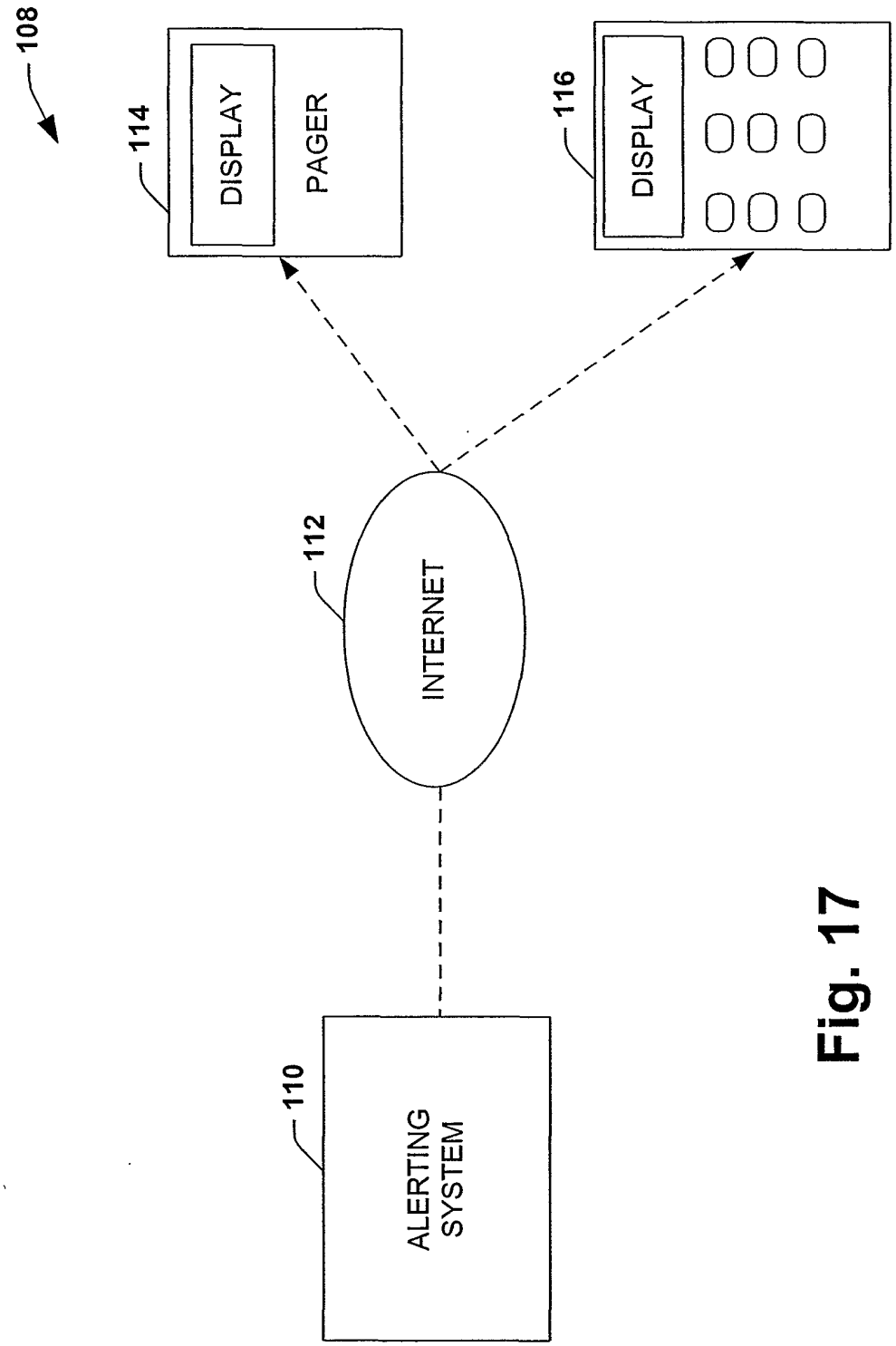

108
114
DISPLAY
PAGER
116
DISPLAY
112
INTERNET
110
ALERTING SYSTEM

Fig. 17

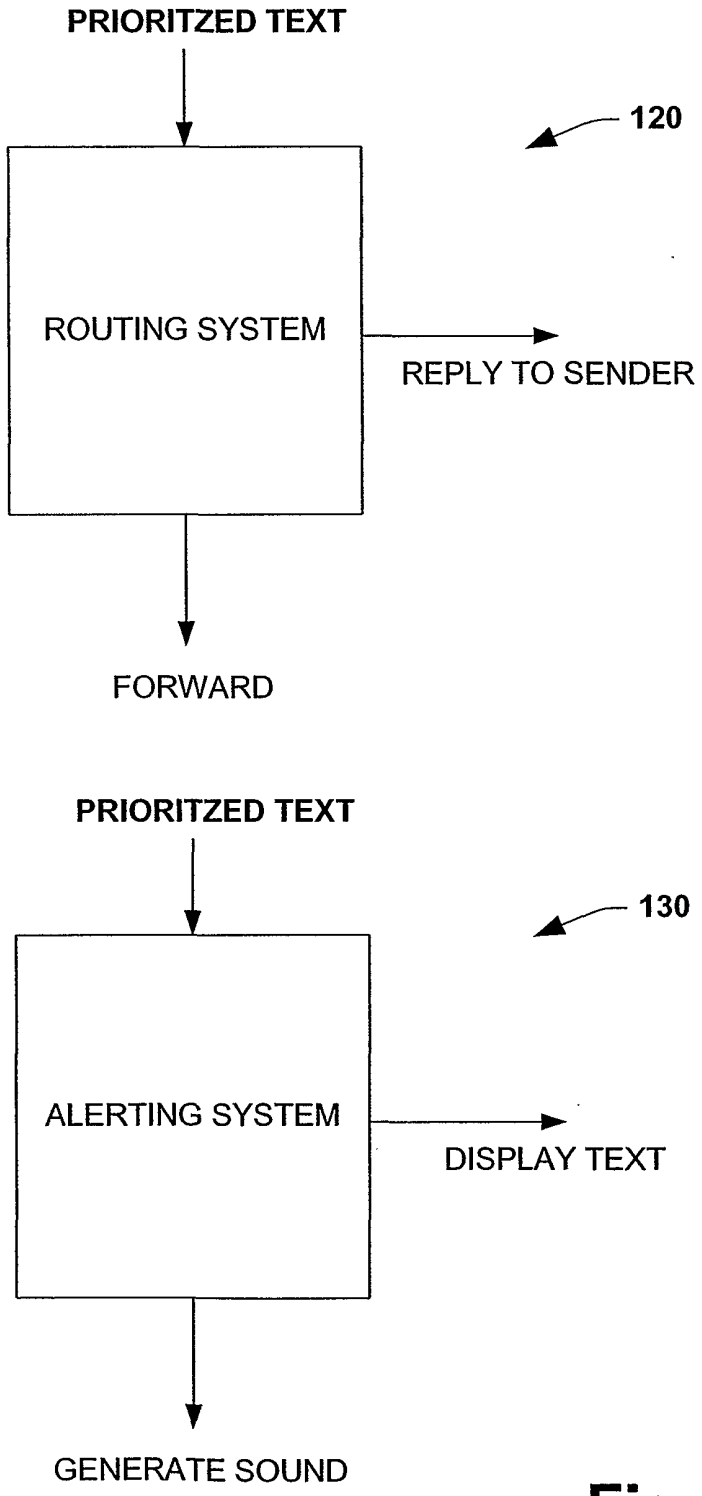

PRIORITZED TEXT
120
ROUTING SYSTEM
REPLY TO SENDER
FORWARD
PRIORITZED TEXT
130
ALERTING SYSTEM
DISPLAY TEXT
GENERATE SOUND

Fig. 18

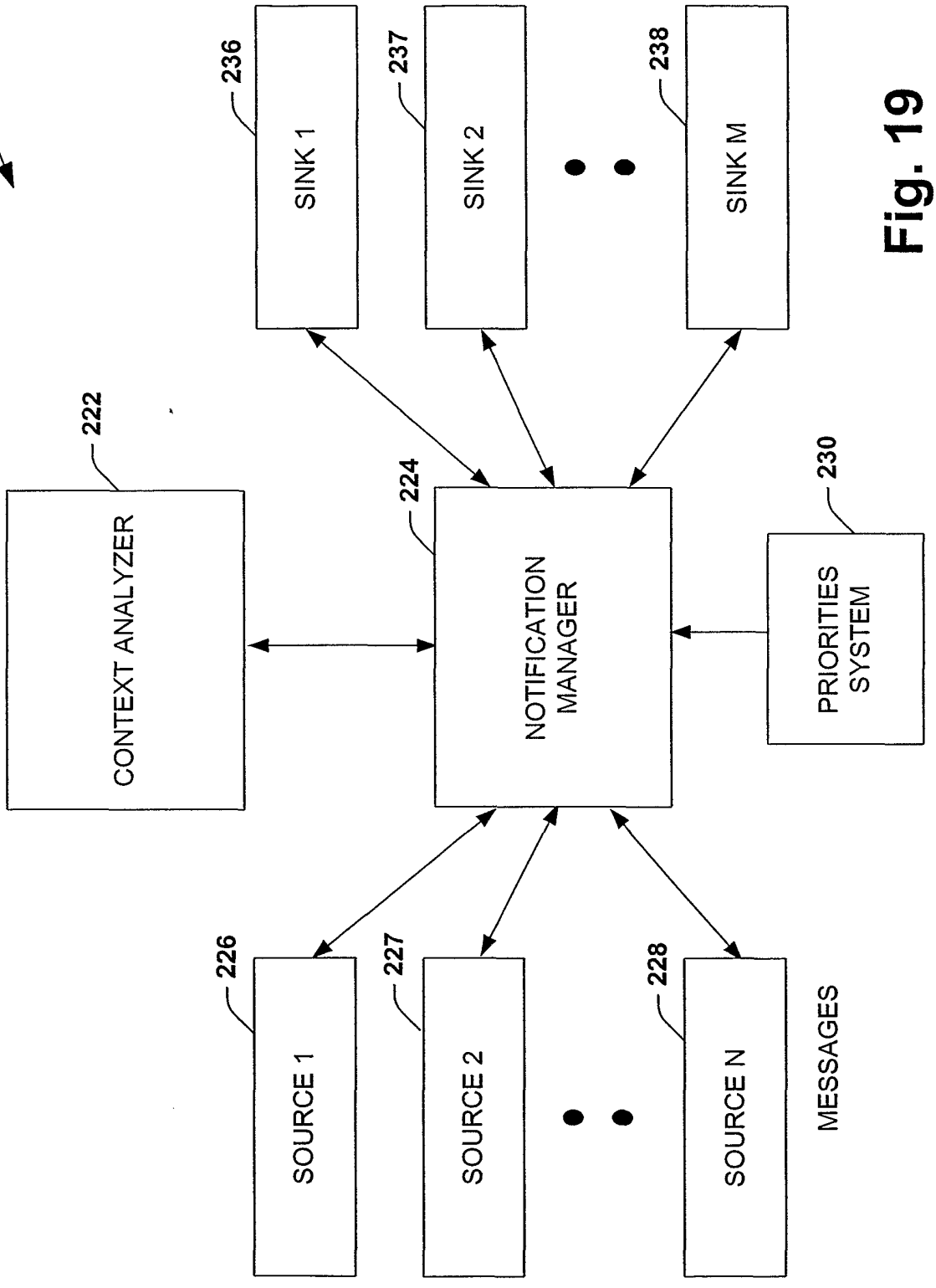

200
236
SINK 1
237
SINK 2
238
SINK M
222
CONTEXT ANALYZER
224
NOTIFICATION MANAGER
230
PRIORITIES SYSTEM
226
SOURCE 1
227
SOURCE 2
228
SOURCE N
MESSAGES

Fig. 19

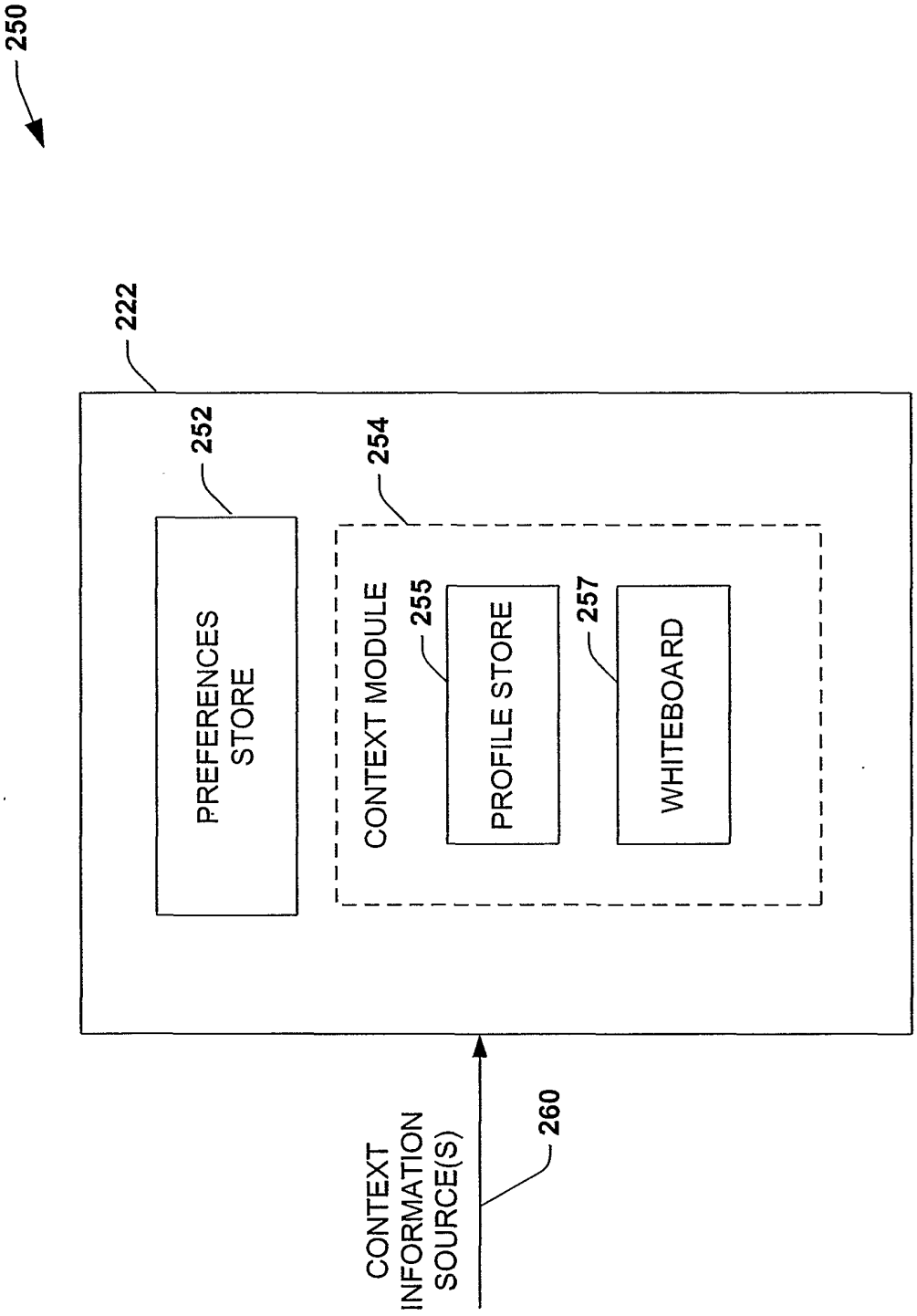

250
222
252
PREFERENCES STORE
254
CONTEXT MODULE
255
PROFILE STORE
257
WHITEBOARD
CONTEXT INFORMATION SOURCE(S)
260

Fig. 20

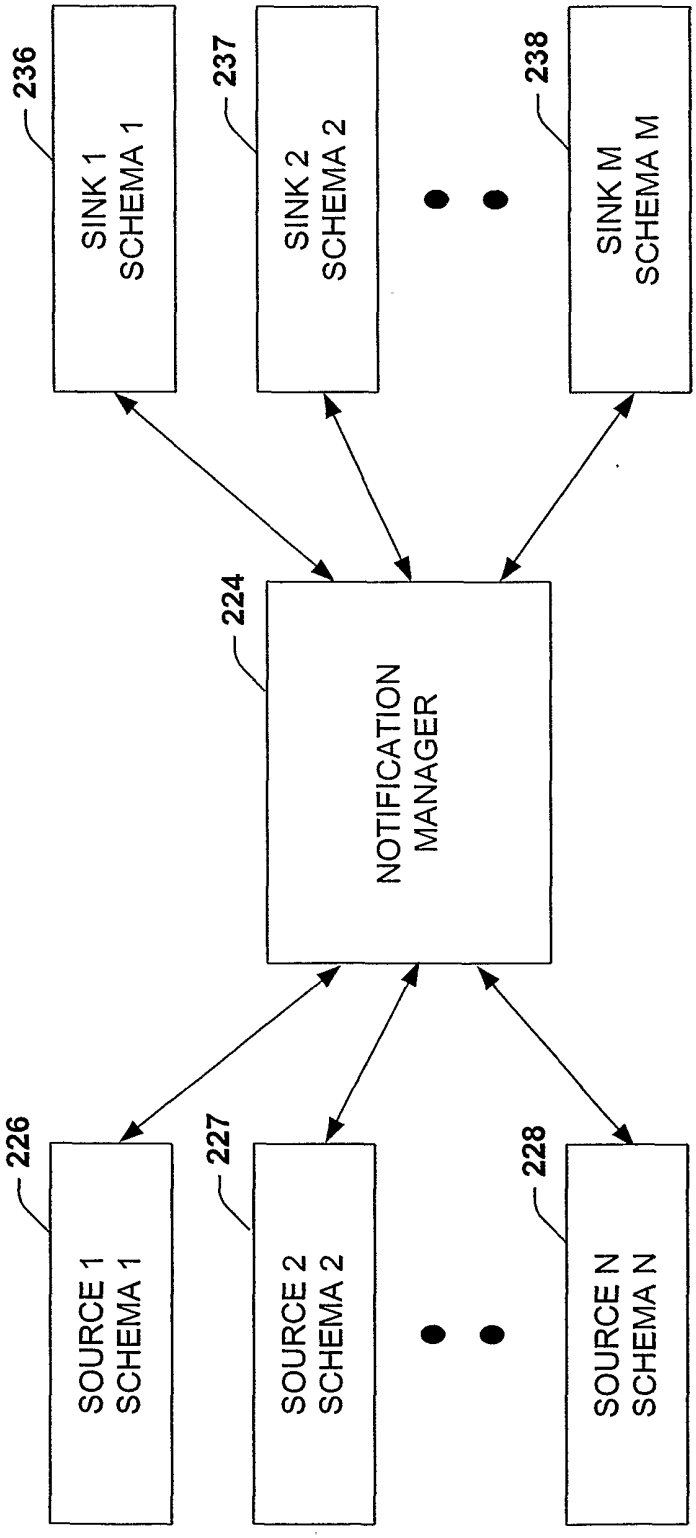

236
SINK 1
SCHEMA 1
237
SINK 2
SCHEMA 2
238
SINK M
SCHEMA M
224
NOTIFICATION
MANAGER
226
SOURCE 1
SCHEMA 1
227
SOURCE 2
SCHEMA 2
228
SOURCE N
SCHEMA N

Fig. 21

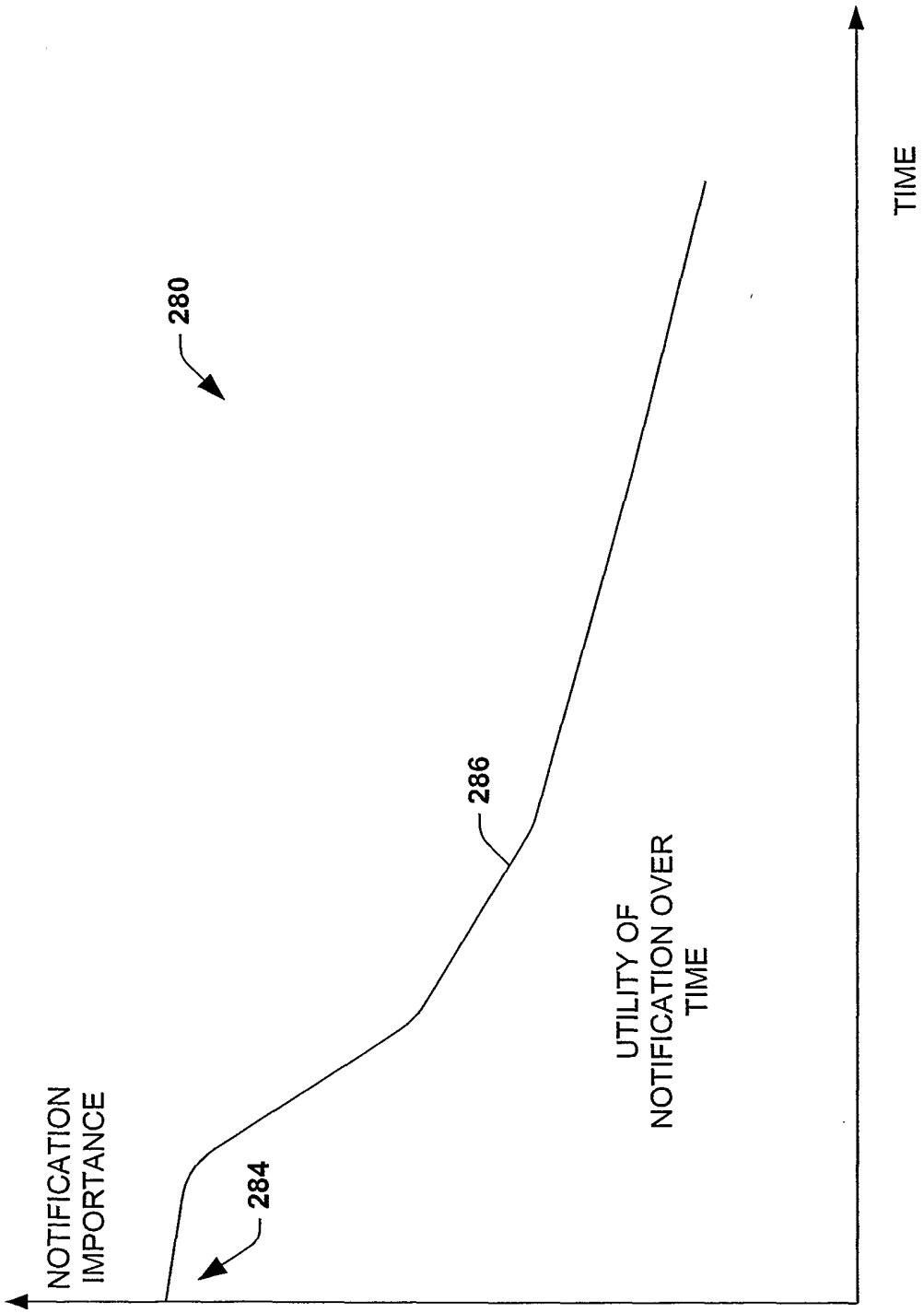

NOTIFICATION IMPORTANCE
TIME
UTILITY OF NOTIFICATION OVER TIME
280
284
286

Fig. 22

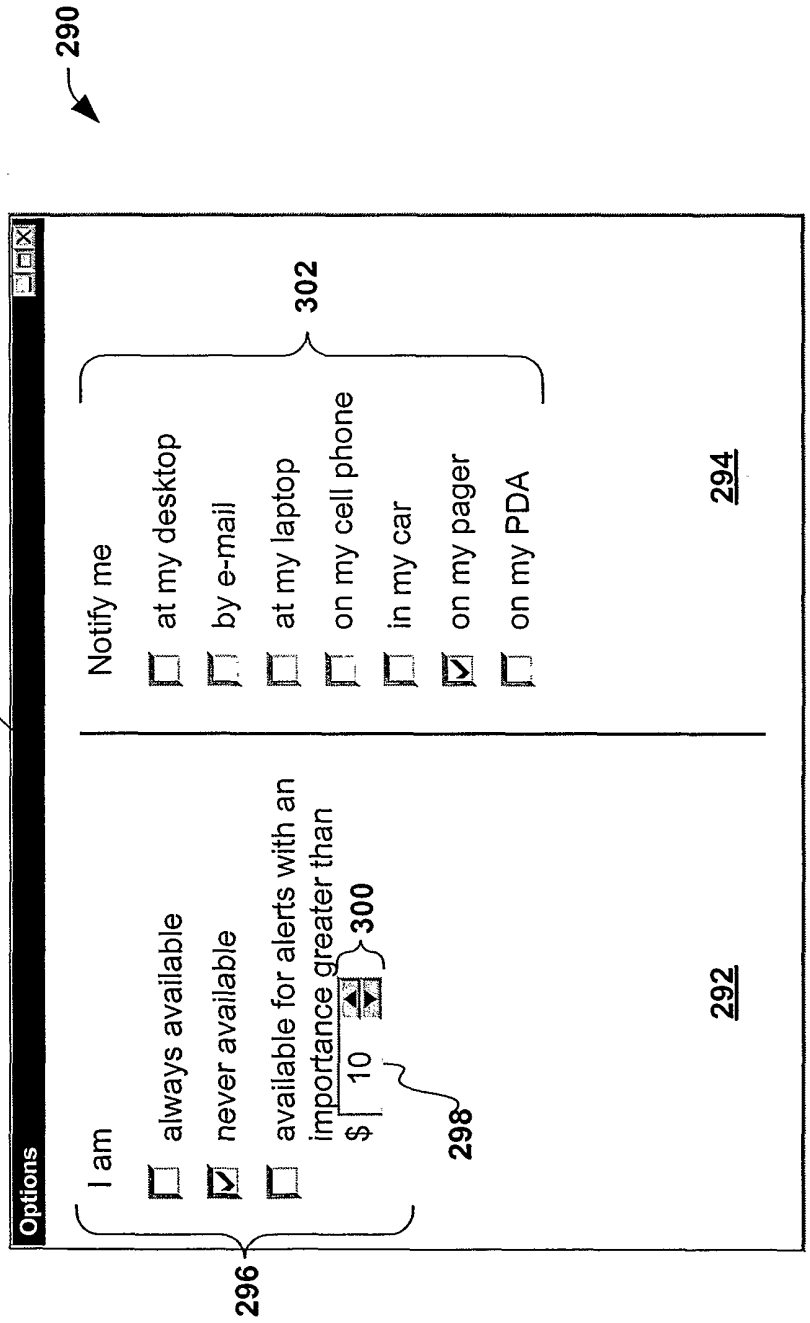

290
291
Options
I am
always available
never available
available for alerts with an importance greater than
$ 10
300
298
296
292
Notify me
at my desktop
by e-mail
at my laptop
on my cell phone
in my car
on my pager
on my PDA
302
294

Fig. 23

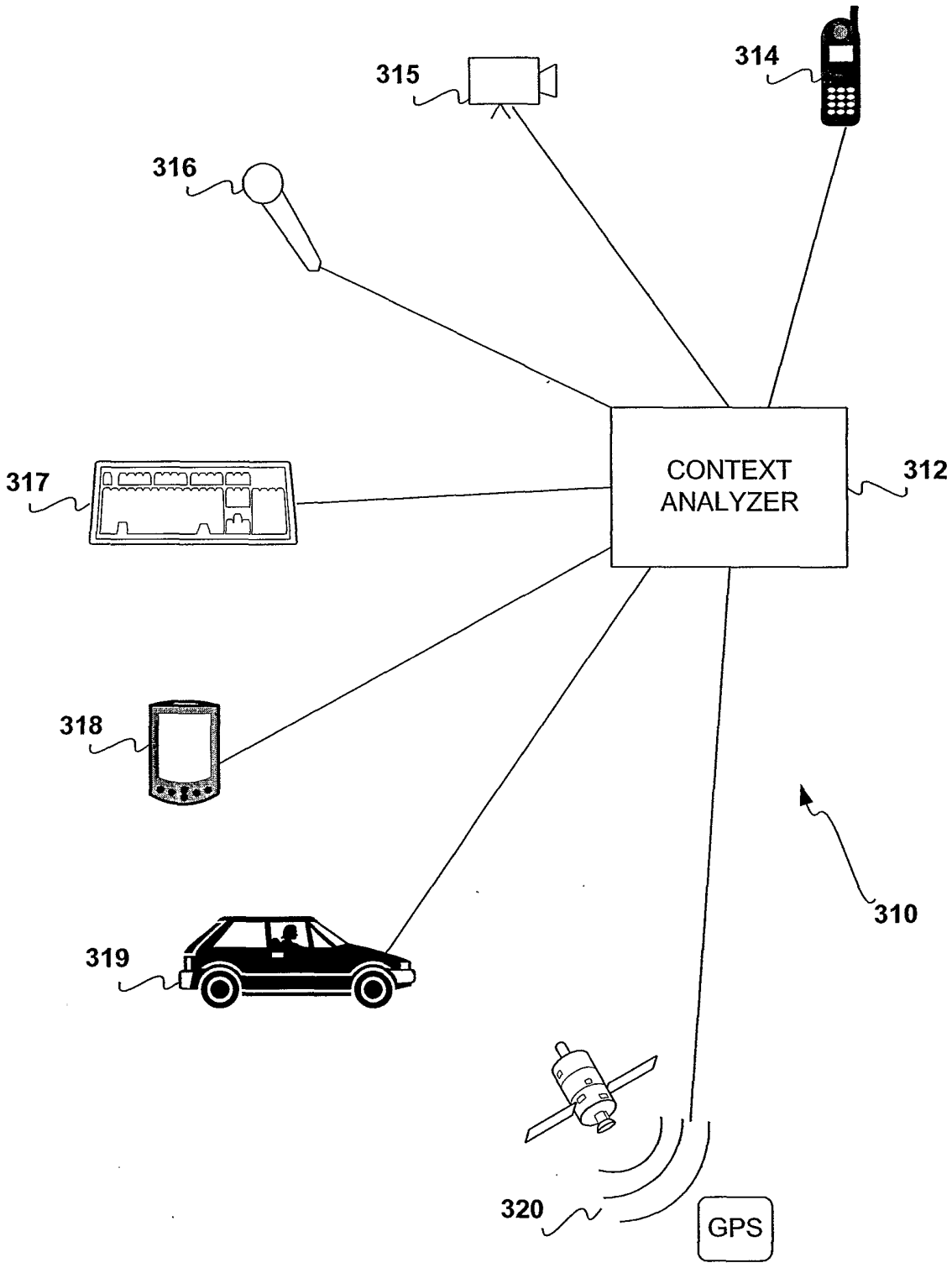


Fig. 24

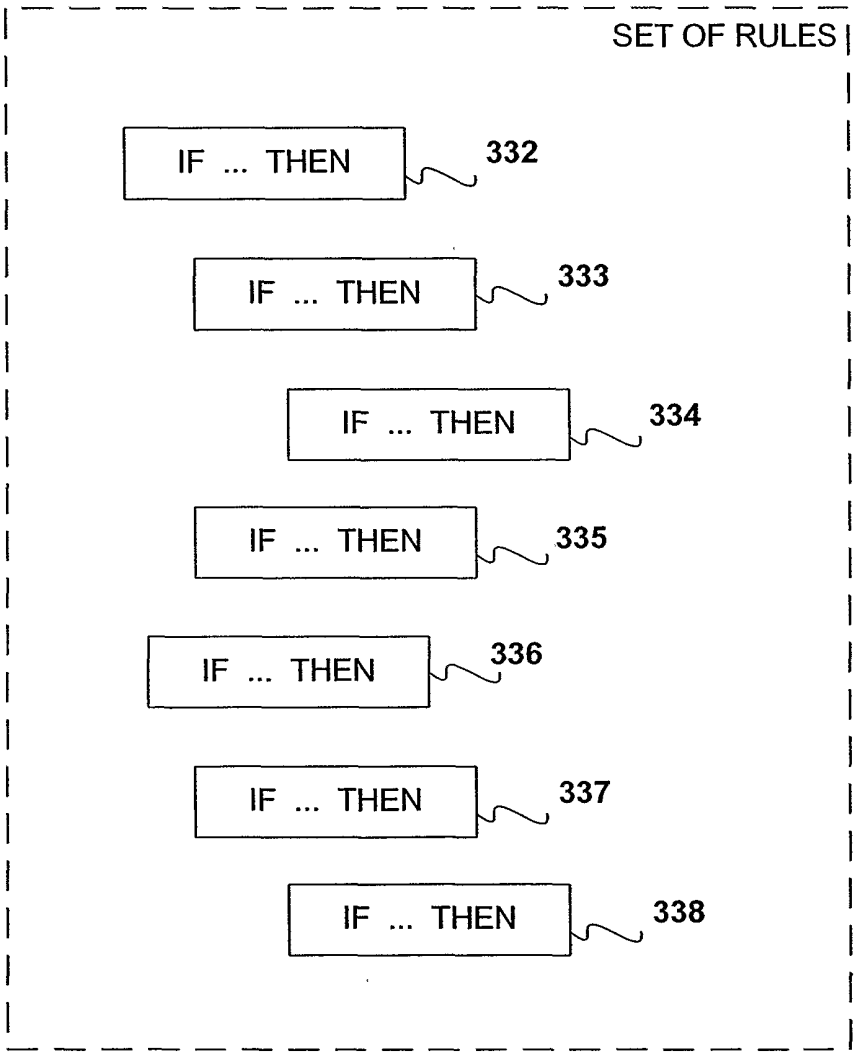

SET OF RULES
IF ... THEN
332
IF ... THEN
333
IF ... THEN
334
IF ... THEN
335
IF ... THEN
336
IF ... THEN
337
IF ... THEN
338

Fig. 25

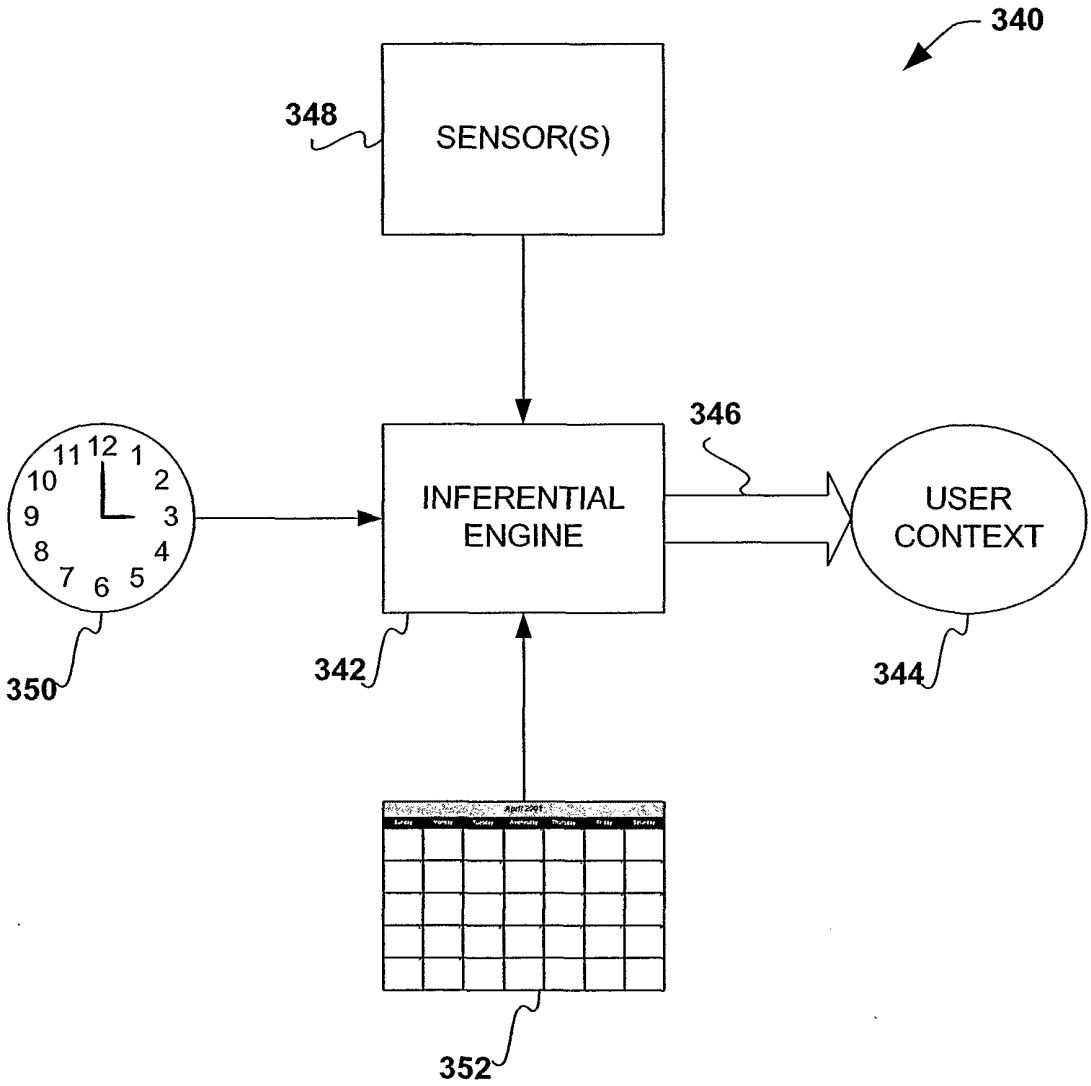

340
348
SENSOR(S)
346
INFERENTIAL ENGINE
USER CONTEXT
350
342
344
352

Fig. 26

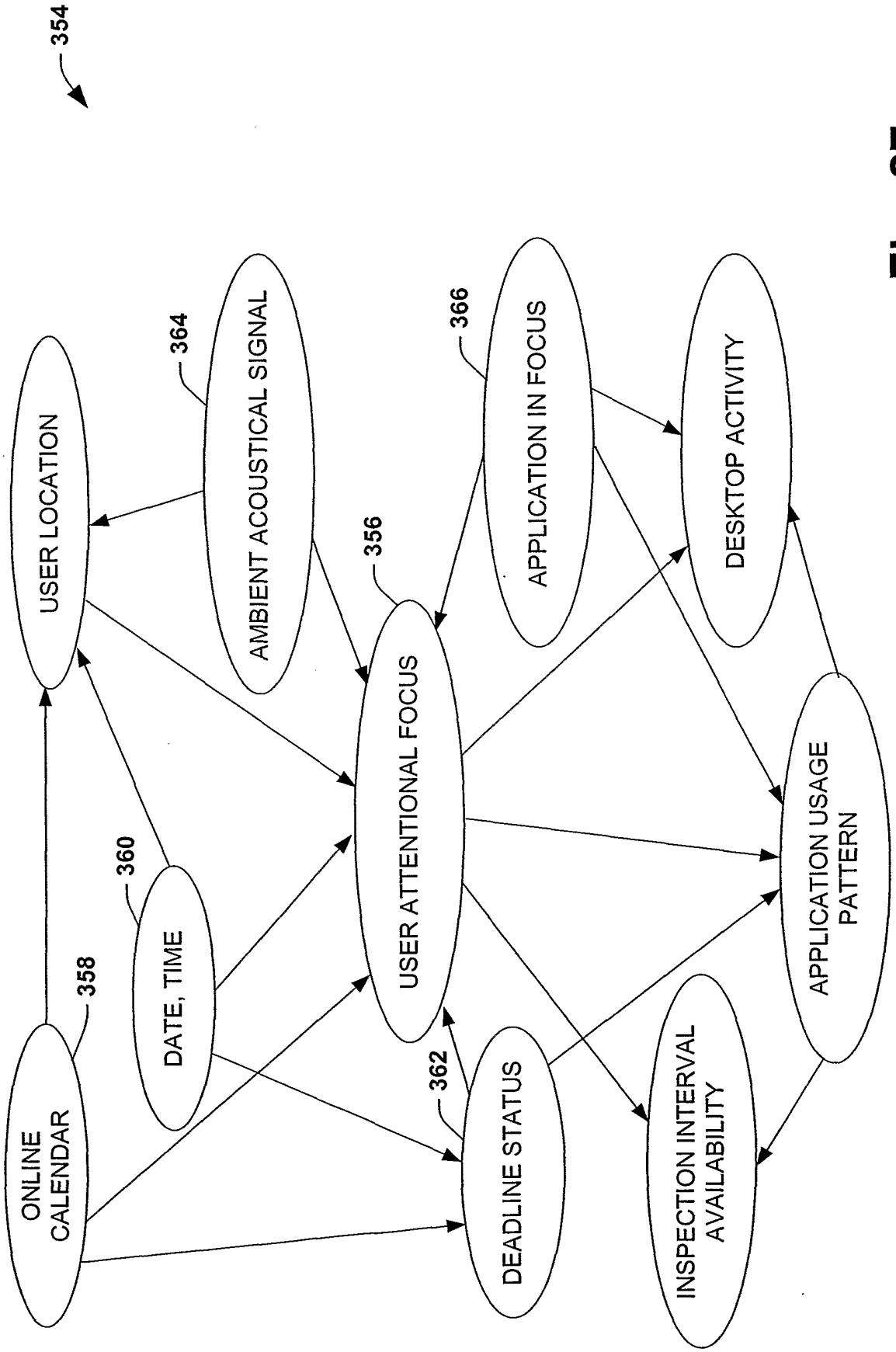

354
USER LOCATION
358
ONLINE CALENDAR
360
DATE, TIME
364
AMBIENT ACOUSTICAL SIGNAL
356
USER ATTENTIONAL FOCUS
362
DEADLINE STATUS
366
APPLICATION IN FOCUS
DESKTOP ACTIVITY
APPLICATION USAGE PATTERN
INSPECTION INTERVAL AVAILABILITY

Fig. 27

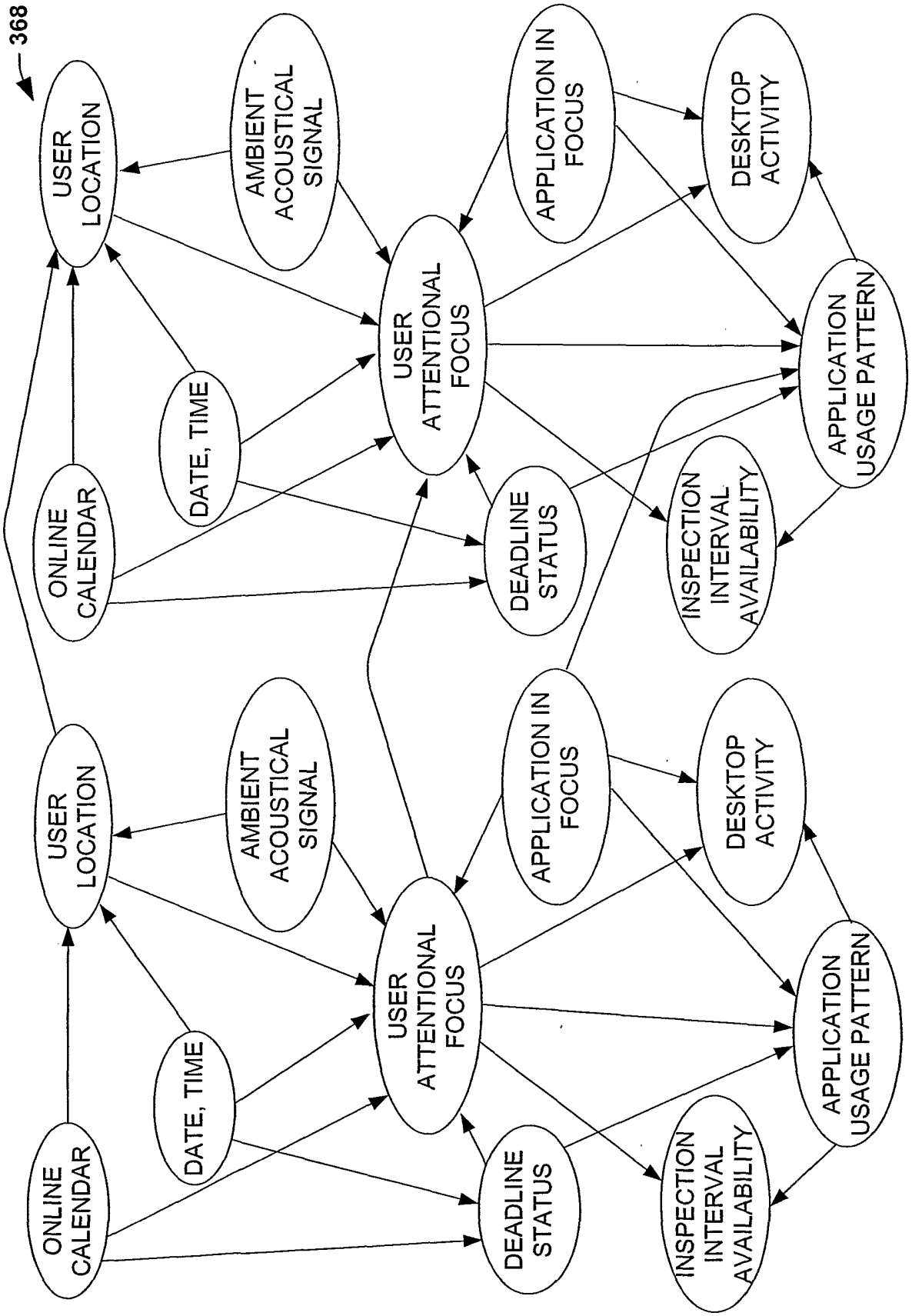

368
USER LOCATION
AMBIENT ACOUSTICAL SIGNAL
APPLICATION IN FOCUS
DESKTOP ACTIVITY
USER ATTENTIONAL FOCUS
APPLICATION USAGE PATTERN
DATE, TIME
ONLINE CALENDAR
DEADLINE STATUS
INSPECTION INTERVAL AVAILABILITY
USER LOCATION
AMBIENT ACOUSTICAL SIGNAL
APPLICATION IN FOCUS
DESKTOP ACTIVITY
USER ATTENTIONAL FOCUS
APPLICATION USAGE PATTERN
DATE, TIME
ONLINE CALENDAR
DEADLINE STATUS
INSPECTION INTERVAL AVAILABILITY

Fig. 28

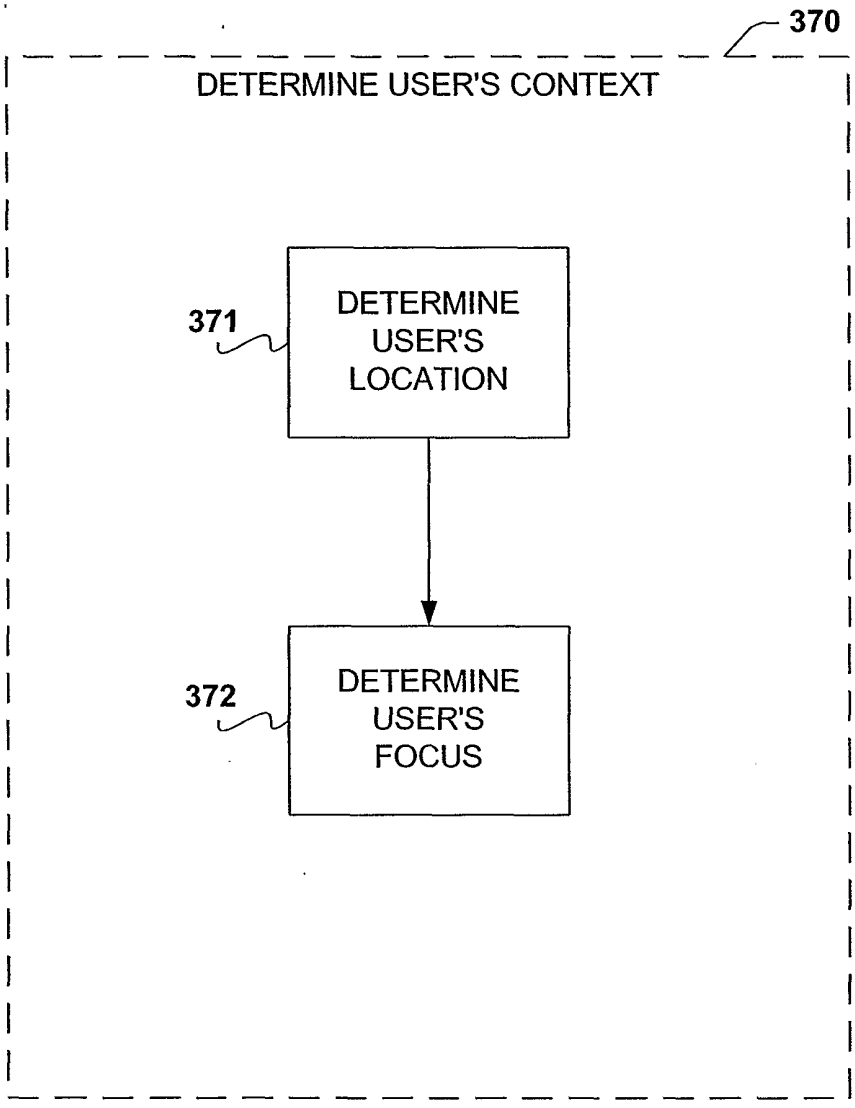

370
DETERMINE USER'S CONTEXT
371
DETERMINE USER'S LOCATION
372
DETERMINE USER'S FOCUS

Fig. 29

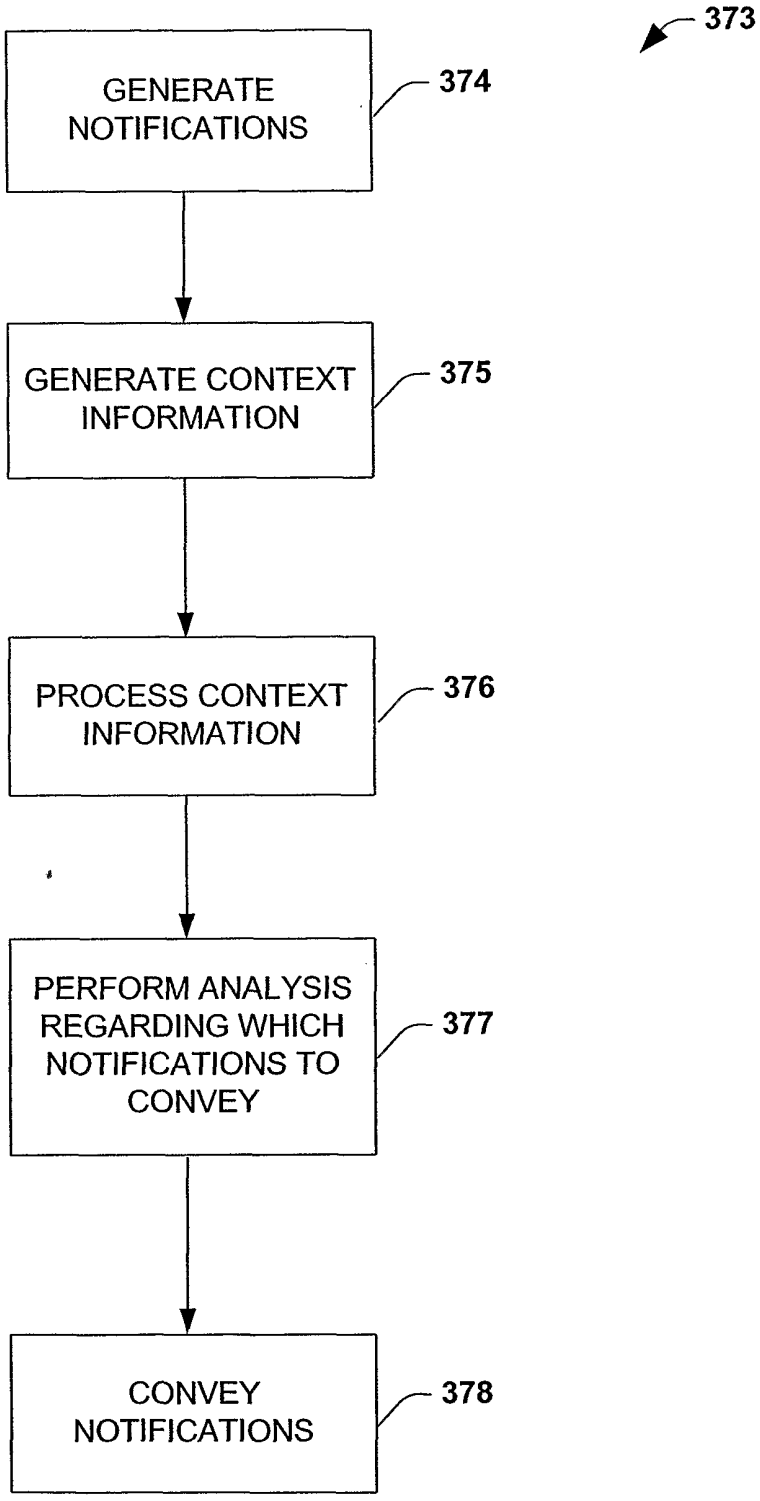

373
GENERATE NOTIFICATIONS
374
GENERATE CONTEXT INFORMATION
375
PROCESS CONTEXT INFORMATION
376
PERFORM ANALYSIS REGARDING WHICH NOTIFICATIONS TO CONVEY
377
CONVEY NOTIFICATIONS
378

Fig. 30

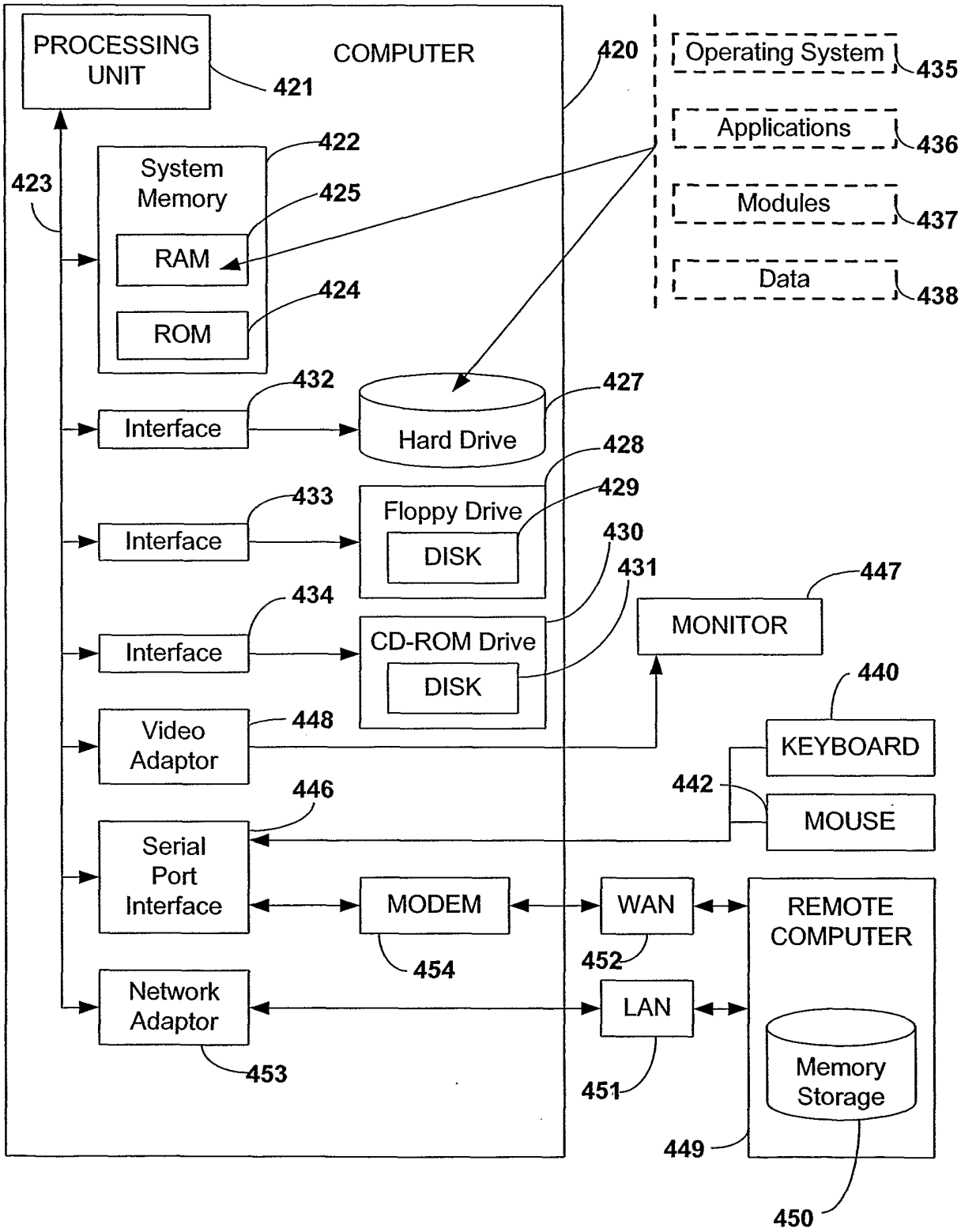

PROCESSING UNIT
421
COMPUTER
420
Operating System
435
Applications
436
Modules
437
Data
438
423
System Memory
422
425
RAM
424
ROM
Interface
432
Hard Drive
427
428
Interface
433
Floppy Drive
429
DISK
430
431
Interface
434
CD-ROM Drive
DISK
447
MONITOR
440
Video Adaptor
448
KEYBOARD
442
MOUSE
446
Serial Port Interface
MODEM
454
WAN
452
REMOTE COMPUTER
Network Adaptor
453
LAN
451
Memory Storage
449
450

Fig. 31

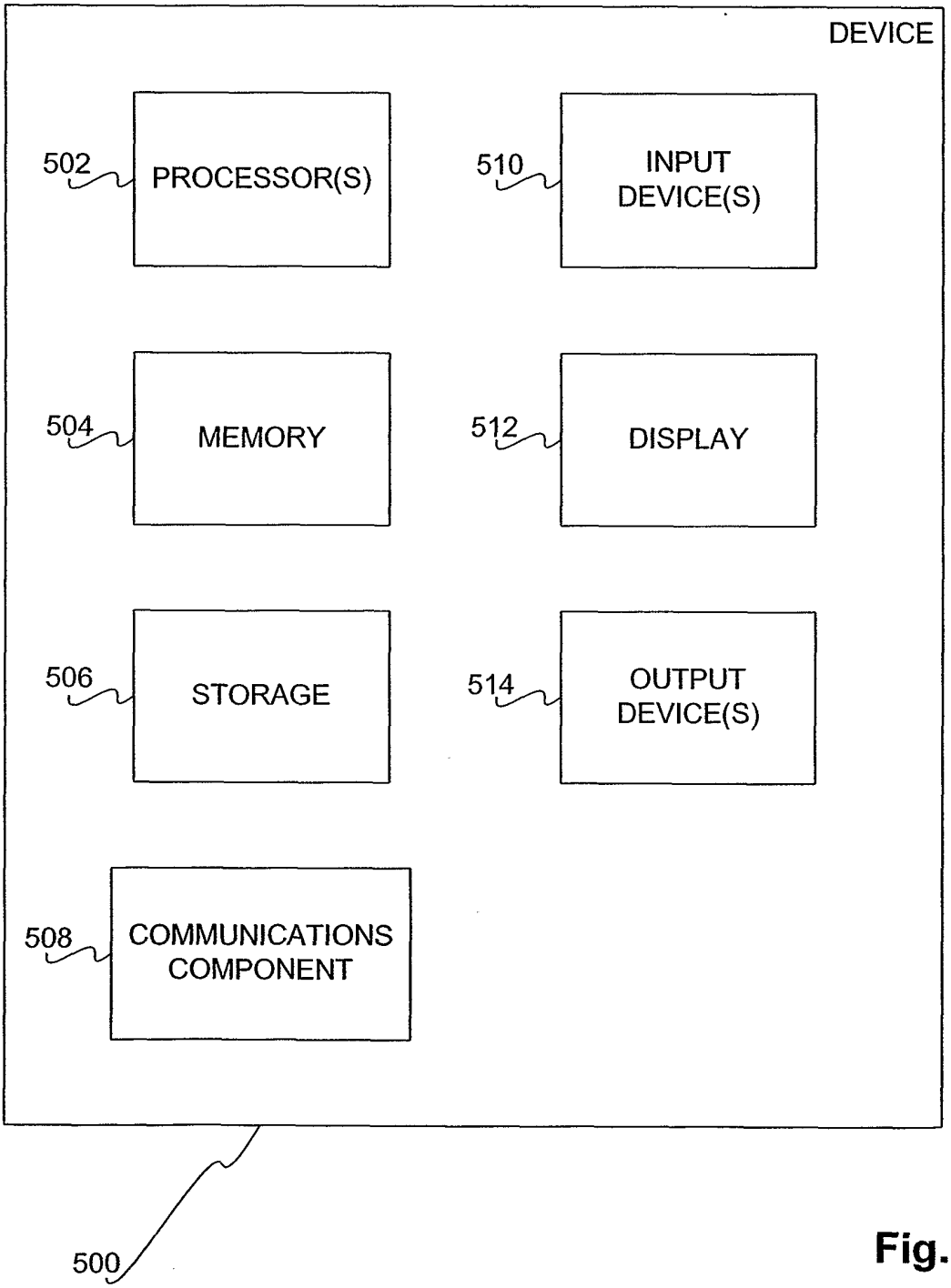

DEVICE
502
PROCESSOR(S)
510
INPUT DEVICE(S)
504
MEMORY
512
DISPLAY
506
STORAGE
514
OUTPUT DEVICE(S)
508
COMMUNICATIONS COMPONENT
500

Fig. 32