

Timely Informing Clients of the Impact of Changes in Their Business Environment

*Juan Ricardo Bauer Mengelberg and
Gilberto David Velazquez Castro*

Colegio de Postgraduados, Montecillo, Edo. De Mexico, Mexico

jbauer@colpos.mx; dvelazquez@colpos.mx

Abstract

Many information services offer current data to their customers, but it often is up to these to filter it, especially regarding its relevance. Additionally, the interpretation of the new data in the context of the customer's activities is often left to the recipient. An informing service that will offer information about the impact of changes in real world data on their businesses, rather than just report these changes, is described. A customer formulates a mathematical model which not only includes his own variables, but also external data which might affect his business. As part of his model, in order to obtain actionably timely available and interpretable information, the customer may include criteria to define situations in which he wishes to be warned, with some degree of urgency, something has happened which merits an action or decision on his part. The client designates certain variables of his model as critical, and criteria are defined which detect significant variations in the values of these variables during the execution of a client's model, usually triggered by some changes in his business environment's data. These in turn result in the generation of an alarm, which is delivered to the customer by means of a text message to his cell phone or an email to inform him of this circumstance. Thus, he receives this warning almost immediately after the change happened, and he can use his model to determine the impact on his affairs. In the paper, the components of the alarms criteria, the program that allows the client to include them in his model and the generation and delivery of an alarm as a result of the execution of the model are described in detail.

Keywords: informing service, communication, actionably timely available, interpretable, mathematical models, alarms

Introduction

Services that offer up-to-date information to their customers are blossoming, and their coverage is ever more extensive. In many parts of the world, it is now possible to be informed, almost immediately, that something has happened that either affects you, or interests you in other ways. Cellular phone networks have added another

dimension to the already impressive advances the Internet contributed to these possibilities of knowing what is happening. However, not all the information received will be useful or even interesting. Thus, ample opportunities arise in the field of Informing Science to improve the informing process, if the focus is to "provide their *clientele* with *information* in a *form*, *format*, and *schedule*

Material published as part of this publication, either on-line or in print, is copyrighted by the Informing Science Institute. Permission to make digital or paper copy of part or all of these works for personal or classroom use is granted without fee provided that the copies are not made or distributed for profit or commercial advantage AND that copies 1) bear this notice in full and 2) give the full citation on the first page. It is permissible to abstract these works so long as credit is given. To copy in all other cases or to republish or to post on a server or to redistribute to lists requires specific permission and payment of a fee. Contact 0HPublisher@InformingScience.org to request redistribution permission.

that maximizes its effectiveness” as the fields that comprise IS do, according to Cohen (1999). As remarked in a previous paper by the same author (Cohen 1997), the definition points to three interrelated components: the client (who has a task to perform that requires information for its completion), the delivery system (for providing information), and the informing environment that creates information to aid the clients complete their tasks.

Gackowski (2005) suggests that information should have four primary attributes: it must be interpretable, significant, timely available and credible. In particular, timely access to information that significantly impacts the client’s interests plays a fundamental role, since it may constitute a significant competitive advantage. Specifically, in a business environment, Gackowski includes four secondary attributes of the quality of information, when the economic aspect is relevant; he formulates them in terms of its use in decision making or carrying out specific actions that take advantage of the information obtained. The information must be interpretable at a non-excessive cost, should be ready on time and credible, which implies that it must be objective, unbiased, accurate, correct and current, and must be presented in a way suitable for the receiver, so that it can be interpreted easily and without error. But the concept of “actionably timely available” (Gackowski, 2005) is probably the most important attribute of the information a businessman needs: information that arrives too late to make a difference loses its significance.

Typically, the client decides what he needs (will receive) and how he will use the information. In turn, the service should adapt to the request. Cohen (2009) states it succinctly: “The framing phenomenon can be viewed within the Informing Science framework as the way in which the informer creates the message in view of the informer’s need or task and bias, and the client interprets the message in terms of his/her own need or task and bias”. Birdsall (2009) states that “the role of the client can be considered from at least two perspectives: (1) as a participant in the use of an informing process; (2) as a participant in the process of designing and developing an informing process”. Since there seemed to be an opportunity for it, we decided to define a service that would offer information to its customers, in such a way that it makes it easy for both sides to do precisely that: define their own needs and satisfy the other’s. Especially, its objective is to empower the client to ask for, and receive, precisely what he needs, and in the form that will maximize its usefulness to him. He should also be able to indicate when and how he wishes to be informed.

This led us to the design of a service which we called FLAG - cash flow in agribusinesses, due to the initial applications for which it was conceived: to enable enterprises in the agronomic sector to update predictions of their cash flows immediately, whenever significant changes happen (Bauer-Mengelberg, in press). The main tools which furnish the informing process with all the desirable attributes are mathematical models, formulated by its clients. The model will include real world variables which reflect circumstances of the customer’s business environment. FLAG will update the values of these variables with the frequency indicated by the owner of the model, through information agents or brokers or other means of obtaining up-to-date information.

Changes in these values will trigger the execution of the model: the calculated variables will be recomputed via their corresponding formulae. This enables the customer, by way of queries available anytime and in different ways, to know the impact of the changes on his cash flow or other indicators he has included in his model. Additionally, as a result of the execution of a model, whenever the new values of its indicators change sufficiently, an urgent message to the owner is delivered by the service. In FLAG, these warnings are called *alarms*. An alarm can be defined, as in Dictionary.com (<http://dictionary.reference.com/browse/alarm>, accessed August 25, 2009) as “any sound, outcry, or information intended to warn of approaching danger.”. Perhaps the definition of warning as “something that makes you understand there is a possible danger or problem, especially one in the future”, from the same source, is better suited to its meaning in Flag.

For our purpose, the use of the term alarm was extended to include a warning or notice of special or exceptional circumstances.

The manner in which the client defines certain criteria that will be applied when values of the variables of his model change due to variations in external data, and the systems to offer these alarms in the total context of the service, will be described and commented. The components of FLAG necessary to understand the rest of the paper are described using a simple example; details are only included if they contribute to the understanding the concepts related to the alarms component of the service.

FLAG – The Informing Service

Brief Description of FLAG

Many businesses use their own systems, or special programs such as spreadsheets, to interpret the data they receive from information services, or acquire through some other communication channel. Others use the models of the information provider, so they obtain the data processed in their own contexts. For example, a stock broker's model might reflect the change in the portfolio of one of his customers due to a variation in the price of a certain stock. In FLAG, the customer defines the model, but it is used by the information provider to process the data, and deliver the information in a way that the client will be able to interpret immediately. It is somehow the best of two worlds: the client defines what and when he wants to obtain actionably timely available information, and the service can process recently acquired data in a way that satisfies the client's specific needs.

FLAG, described in detail in Bauer-Mengelberg (in press) and outlined in FLAG (2010), is an informing service, as well as the corresponding computer application, that informs its customers, with the frequency required in each case, of the impact on their affairs resulting from changes in data values of their business environments (*real-world variables*). The real-world variables, in conjunction with the client's variables and how they interrelate, integrate a client's *model*. The information contained in the client's variables reflects data he knows or determines by himself.

Figure 1 outlines how the system, and consequently, the planned service to customers, works.

The customer formulates his model, probably with the aid of a consultant or adapting one of the sample models offered by his contact with the service. The service offers a set of variables, whose values are updated with the frequency resulting from its customers' demands. When changes in these values occur, or by other triggering mechanisms, the models' calculated variables are computed using their formulae – a process we call the *execution* of the models, and thus the impact of the changes on indicators or other variables are determined. Besides queries, which they may formulate at any time, they may formulate criteria placed on certain variables, to indicate when they need an *alarm* informing them about significant changes in the values of such variables.

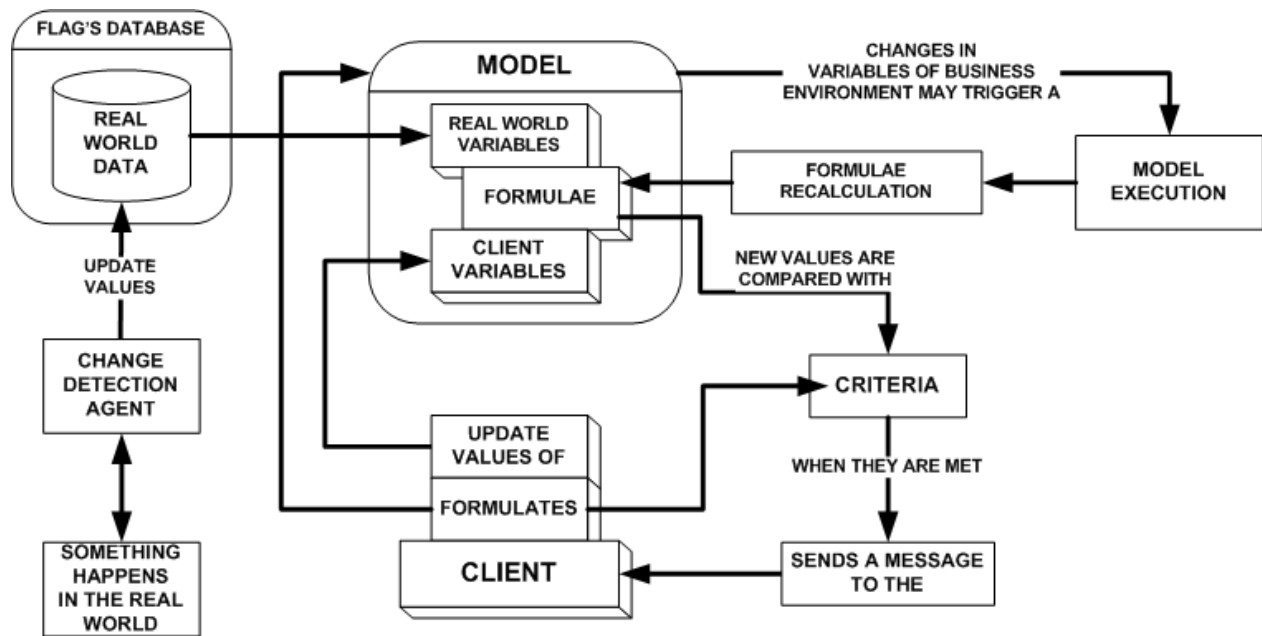


Figure 1. The service offered by FLAG

The Farmer's Cash Flow Example

A simple example is used to point out and explain the relevant functions provided by FLAG. The particular model was selected to simplify its comprehension, whilst still satisfying its objective. It shows a situation involving few variables, and those the customer decided are critical are easily identified. Of course the model neither reflects all the characteristics and generalities of the types of models currently implemented in FLAG, nor constitutes a usable cash-flow model, but this will not affect its usefulness in the context of this paper.

A farmer grows corn in a region that does not have irrigation facilities. Thus, rain is his only source of water. Because of this, it is vital for the farmer to know the weather forecast for the seeding period, so he can estimate his crop yield and the resulting harvest. A more sophisticated user of FLAG might use the rainfall prediction to decide when he should fertilize and sow, based on the predicted start of the rainy season. Additionally, the price of seeds and fertilizer will affect his profits, or even keep him from growing anything, due to his cash-flow situation. His operating profit is computed, based on the supposition that the entire harvest will be sold at market price at the end of the season. The cash flow components of the model were oversimplified, since they are not the objective of the example. He will group other income such as interests or debt collection, as well as expenditures and other data that affects his cash flow directly, in the corresponding variables included in the model. He also indicates his cash flow at the beginning of his planning horizon.

Figure 2 shows the example's variables and, using symbols to indicate their operands and operations, the formulae for the calculated variables. The variables are numbered, and RW variables are shown with the prefix "R".

The values of these variables can be scalars or random variables. In the latter case, their probability distributions must be discrete and limited to a maximum of 4 values. Both these decisions were part of the design of the system. Actually, every variable is a vector of 24 data elements, corresponding to successive time periods. We call the data values *periods* of the variable. Though the periods in the example represent months, they could just as well indicate quarters, fortnights or even days or years.

Formulae were kept as simple as possible, without losing their power. The minimum and maximum functions are added to the basic arithmetic operations (addition, subtraction, multiplication, division and square root). Additionally, there is a *shift* operation. For example, if periods are used as subindices, shown in parentheses, the operation $V50 = [\text{shift } V75 \text{ by } +1]$ would result in values $V50(j+1) = V75(j)$ for $j = 1$ to 23. $V50$ would have no value for period 1.

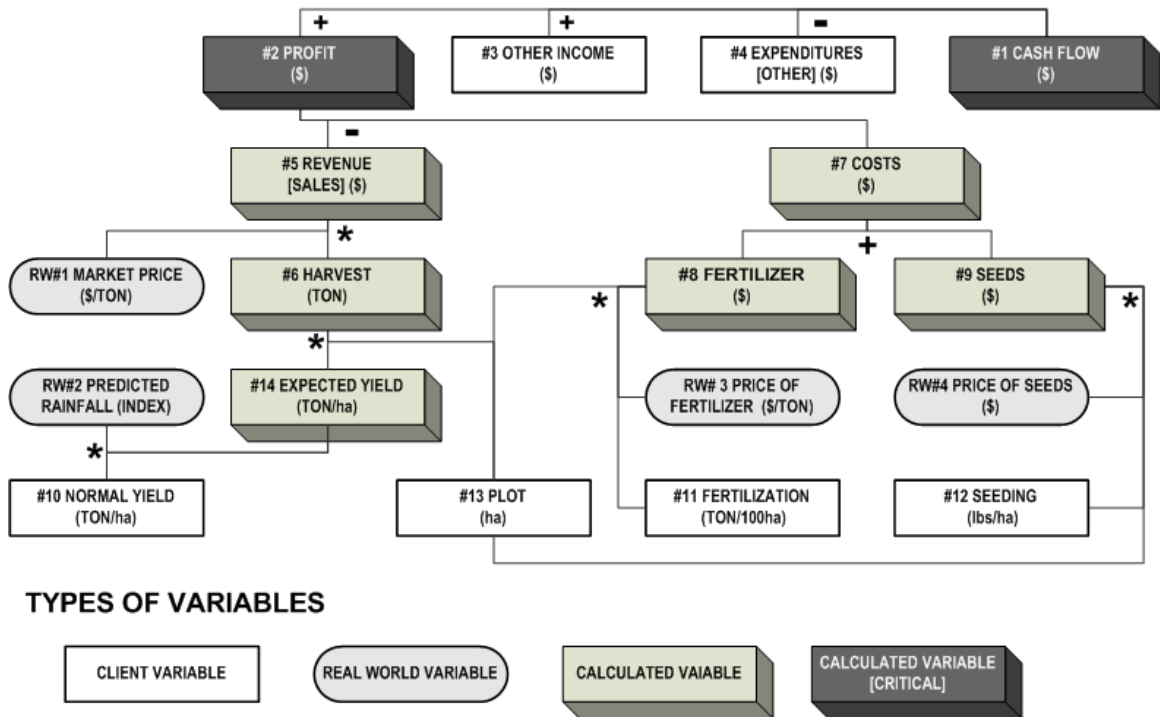


Figure 2. The farmer's model.

The formulae are defined using subtotals, which replace parentheses, since this apparently makes life easier for mathematics-adverse users. The last subtotal defined is the result of the formula, and thus constitutes the value (for each period) of the variable being calculated by the formula. Details about formulae can be found in the aforementioned references. For example, the formula for Variable # 1 (cash flow) in the example would be:

$$\text{Subtotal (1)} = \text{Shift (v1 by } +1) \quad \text{subtotal (2)} = \text{subtotal (1)} + V2 + V3 - V4.$$

The service will obtain up-to-date information about the real world variables of all its customers, from the sources that offer it. Another (as yet unpublished) component of FLAG performs these functions, typically using information agents and brokers.

Changes in real world values will trigger executions of the affected models. Yet another component of the system is in charge of allowing customers to specify rules that indicate when their model should be executed, according to the changes in their real world variables.

FLAG has a powerful querying tool. Besides being fairly easy to use, it allows storing the customers' most frequently used queries, including the parameters to adapt them to any specific use, typically a range of periods or a change in the variables to report.

Of course, FLAG has an access control component which will prevent non-authorized users from seeing a customer's data. Should a customer elect to do so, he can even avoid staff members of

the service having access to parts of or all of his data, though this might limit some of the services offered to him.

An obvious question arises from the description of the models: why not use a spread-sheet to implement the models? In favor of choosing this platform, “Excel has a complex and finely tuned algorithm for choosing the fastest sequence and the minimum number of cells required to calculate the correct answer” (Decision Models, 2008b). This would obviously have saved a lot of work, since not only the correct order of execution of formulae would have been guaranteed (no formula should be calculated before its operands have been recomputed previously), but also some performance issues could have been avoided. However, the specifications of models such as the ones used by FLAG, including 24 values for the variables, would have been very difficult, even though array processing is included as a feature. “Array formulae are one of Excel’s most powerful features, although not always the easiest to use” (Decision Models, 2008a). The inclusion of random variables, with their probability distributions, further complicates matters. Since FLAG was designed for clients who do not necessarily have the skills to do this, it was well worth the effort to design interfaces that eliminated many of the difficulties and subtleties involved in preparing a spread-sheet. And of course, several of its features would have been almost impossible to include without specific routines, which even in Excel are programmed in Visual Basic.

The Alarms Component of Flag

A customer may request a special urgent notification – an alarm – of some change, if he thinks he might be able to take advantage of the knowledge gained through such a warning. To have access to these alarms, he will define certain criteria to determine when he wants to be warned, and how the alarm should be delivered.

An alarm has two components: 1) a message to be transmitted, containing the information of the impact caused by changes in the model; 2) the means by which it is conveyed to the interested party. We refer to the latter as the *communication channels or communication media*, through which the alarm is actually delivered to the customer. FLAG includes the four communication channels listed below, as ways to warn the customer. However, other means may be included in the future, especially in view of the rapid emergence of new communication channels, for example Tweeter, social networks, Facebook and similar ways people communicate with each other. The numbering of these 4 media will become significant later.

Medium # 1. The client is asked, through an e-mail message, to check his model; he will use the queries he has defined and catalogued, or *ad hoc* inquiries he can formulate anytime.

Medium # 2. An e-mail is sent to the client, but here the text contains the message generated as part of the alarm.

Medium # 3. The message is sent to the client via an SMS to his mobile device.

Medium # 4. A FLAG operator tries to reach the client by phone to deliver the message in person. The customer may ask some specific questions, since the operator will be able to query the customer’s model during the phone call.

Outlines of the processing necessary to actually send an email and an SMS message were included as an Appendix of this paper, since they do not contribute anything to the understanding of the concepts and functions involved, but might be interesting or even useful to some readers.

How Alarms Are Generated During the Execution of a Model

The role of the alarms component of FLAG, shown in Figure 3, and its terminology, can be briefly described as follows. The crucial event occurs whenever updated values for variables used by FLAG's clients' models are detected and trigger the execution of some of the models. During these executions, the previous and new values of certain variables are compared, using the alarm criteria placed on the *critical* variables. If the values satisfy these criteria, an alarm will be generated.

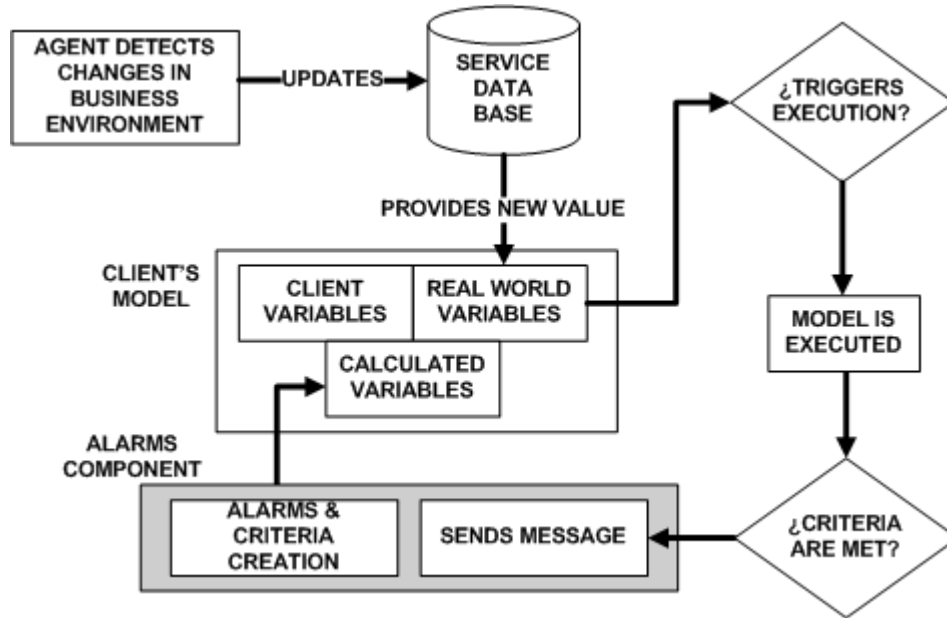


Figure 3. The ALARMS component of FLAG

Alarm Criteria

The generation and the subsequent delivery of an alarm are based on the criteria defined by the client. The way these will eventually determine the need for an alarm is presented first, so that the rest of the explanations may be understood.

The mechanism that will trigger an alarm is the *severity index* of the particular execution of the model: if the value of the index is high enough, an urgent message will be sent to the customer. The term severity (grievousness; hard or trying character or effect) was chosen over similar words, especially gravity (serious or critical nature) and weight (importance, moment, consequence, or effective influence), where all three definitions were taken from the above cited Dictionary.com. The severity index is the sum of the *contributions to the severity* of the critical variables of the model.

Roughly, an alarm criterion is a *critical interval* which excludes the current value of the variable, with an associated contribution to the severity, or just *severity*. However, the terminology we adopted was to say that the interval includes the current value, and the critical event occurs whenever a new value falls *outside* this interval. FLAG allows for two such intervals, where the second – called the *emergency interval* – is wider than the first. Figure 4 depicts these intervals around a value x , with their respective severities.

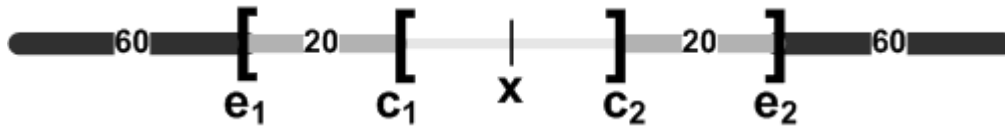


Figure 4. An example of the critical and emergency intervals for a variable.

The criterion reflects the client’s view of the effect a change of the value of this variable will have on his business, so that an alarm will be raised if the severity index is high enough. Only calculated variables can have an associated critical interval. Should the client want to place a critical criterion on a real world value, he will have to define a dummy variable – with identical values than the original one.

It should be pointed out that the client can use his model as a simulator. He can input changes in values of the real world variables, and study their impact on his indicators. This can help him determine the parameters of his critical intervals. For this purpose, FLAG includes a program that allows manual changes of values of these external values, of course using a copy of the unique file, used by all clients, that contains the values of the real word variables.

How Alarm Criteria are Defined and Stored

The alarm criteria defined by a client are stored in *his* database (each client has his own database in FLAG), in the tables described in Figure 5.

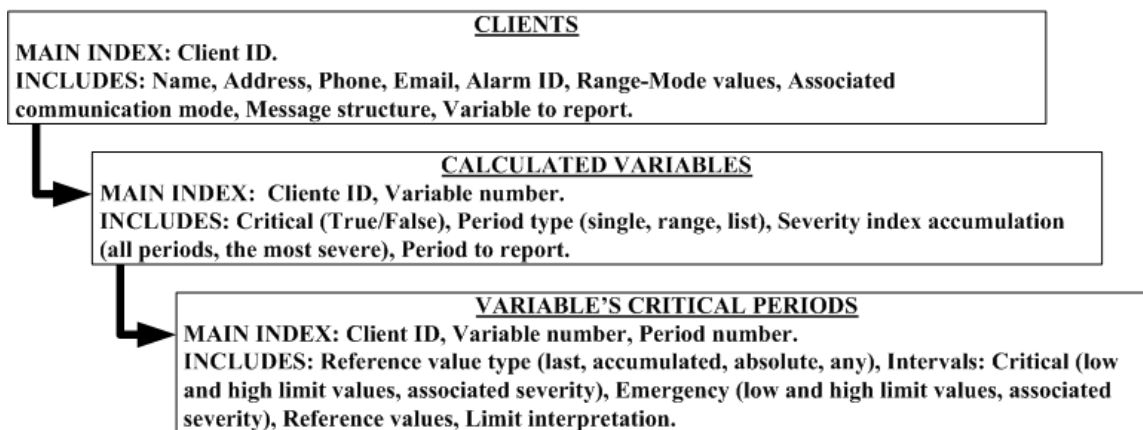


Figure 5. The tables of the database used to store alarms criteria.

The actual tables in FLAG contain other – purely technical - fields, for example, to protect other fields from being modified or determine when an update took place. Screen-shots of the program’s main routines that update each of these tables are used to explain and illustrate the components of the criteria.

The elements of the criteria stored in the CLIENTS table are updated using the form depicted in Figure 6. The customer will specify, for the ranges of the severity index he indicates, the corresponding communication channel, in ascending order of the numbers assigned to them when they were enumerated and described before, since this order somehow reflects the urgency of the alarm.

Since the message formulated as part of an alarm will refer to only one variable, a constraint we felt would not affect its usefulness, he must also furnish a rule that determines which critical variable will be included in the text of the alarm, whenever more than one of the critical variables contributed to the severity index of the execution of the model.

Using the same form, the critical variables are selected, that is, those for which a criterion will be provided. A click on the “ADD A CRITICAL VARIABLE” button causes the system to display a list of all calculated variables of the model – not shown, since it appears superimposed on the same form, and one or more of them can be selected to be included in list of the critical variables. After selecting one of them (the number and name of the variable appear to avoid confusion) the EDIT button will take him to the next step for that variable, namely the specification of its alarm criterion.

ALARM MODE AND CRITICAL VARIABLES

CUSTOMER: 1

NAME: John Dennis Smith E-MAIL: john@hisdomain.com
 ADDRESS: 314 Central Street PHONE: 5553216

ALARM MODE	VALUE OF INDEX UP TO
NO ALARM	<input type="text" value="20"/>
EMAIL (QUERY YOUR DATA)	<input type="text" value="40"/>
EMAIL (THE MESSAGE)	<input type="text" value="60"/>
SMS CONTAINING MESSAGE	<input type="text" value="80"/>
PERSONAL PHONE CALL	<input type="text" value="100"/>

Key in a new value and press ENTER.
Overlapping values are ignored.

NUMBER	SHORTIE	DESCRIPTION
1	CF	CASH FLOW
2	PROF	PROFIT

VARIABLE TO REPORT

THE MOST SEVERE
 THE FIRST THAT CHANGED
 THE LAST CALCULATED

SELECTED VARIABLE **1 CASH FLOW**

Figure 6. Selection of critical variables and the modes corresponding to the severity index.

Figure 7 depicts the form in which the elements of the criterion for the variable are updated. Its data elements are explained with the aid of a brief guide to the use of the functions used to update them.

- The variable id is shown to avoid confusions.
- The periods considered critical for the variable must be indicated. This can be done specifying a unique period, a range of consecutive periods or a list of non-consecutive periods. The labels of the 24 periods for that particular variable are shown (they might differ from one variable to the next, though they all refer to the same unit of time, in this case, months). The “change periods” button displays the options (once again, not shown) and the use of one of them will result in the list of selected critical periods that appears, as shown. In the example, the selection of periods MAY through AUGUST reflects the fact that the client is alert to predictions regarding probable rain fall, through their effect on the expected yield of his corn crop (in Mexico, the rainy season usually starts in May).

Timely Informing Clients

- Since several periods might produce a contribution to the severity index, the client must indicate whether all these contributions are to be added to the severity index of the execution, or only the one corresponding to the period that resulted in the largest severity should be used to determine the contribution of the severity of that variable.
- One of the options offered as the “period to report” must be chosen, to indicate which period will be included in the text of the alarm.

PERIODS SELECTION

VARIABLE: - 1 - CASH FLOW

PERIODS:

1-6	JAN/2009	FEB/2009	MAR/2009	APR/2009	MAY/2009	JUN/2009
7-12	JUL/2009	AUG/2009	SEP/2009	OCT/2009	NOV/2009	DEC/2009
13-18	JAN/2010	FEB/2010	MAR/2010	APR/2010	MAY/2010	JUN/2010
19-24	JUL/2010	AUG/2010	SEP/2010	OCT/2010	NOV/2010	DEC/2010

SELECTIONS:

CHANGE PERIODS (ERASES PREVIOUS SELECTION)

PERIOD TO REPORT (ONLY LIST OR RANGE)

THE FIRST THAT CHANGED

THE LAST THAT CHANGED

THE MOST SEVERE

INDEX ACCUMULATION

ALL PERIODS

THE MOST SEVERE

SELECTED PERIODS

RANGE OF PERIODS

MAY/2009
JUN/2009
JUL/2009
AUG/2009

BACK TO CLIENT CRITERIA

Figure 7. The interface used to update the criteria for a critical variable.

The list of selected periods is used to invoke the last step: specifying the criterion for each period of the variable. A click on one of the periods will result in the display of the form shown in Figure 8, which illustrates the selection of MAY/09.

The criterion for a period of a critical variable is defined by the attributes described below.

- The *reference value* against which the new value of the variable for the period will be compared. The alternatives are: the previous value of the variable, resulting from the prior execution of the model; the value of the variable after the last execution that produced an alarm (which may well be the last value of that variable known to the customer); finally, if an absolute interval is specified, this option is ignored.
- The lower and upper limits of the intervals, as well as the severity associated to each of them, are introduced into the corresponding fields of the form. Of course, if no emergency interval is needed, the client will not use it (he will assign a severity of 0 to the interval).
- The limit-interpretation (how limits are interpreted) indicates how they are used to build the intervals. The examples below show the resulting critical intervals for each of the three options, given that the specified limits were -10 and +5. RF indicates the value of reference, and the symbol “*” denotes multiplication.

- Percentage variation: $[RF * .90, RF * 1.05]$. The limits are computed based on the reference value, applying the corresponding percentages indicated as left and right limits.
- Range interval (numeric variation): $[RF - 15, RF + 15]$. The limits are added or subtracted to the reference value to obtain the critical intervals.
- Absolute interval: $[-10, 5]$. The critical interval is the one specified by its left and right limits. No comparison with previous values is performed: if the new value falls outside the interval, the severity applies.

Figure 8. Form used to update the criterion's intervals for a period of the variable.

Alarms Resulting From the Execution of a Model

The need to send an alarm to a customer arises during the execution of his model, when the values of its calculated variables are computed using the updated values of the operands of their respective formulae. Figure 9 shows how, when a critical variable is computed, its alarm criteria are applied, and the resulting severities are added to the execution's index of severity.

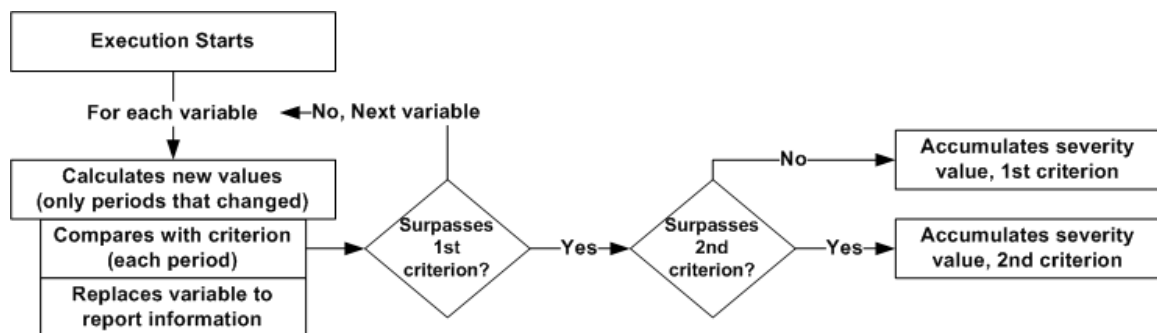


Figure 9. Alarm criteria are applied when new values of a critical variable are computed

Simultaneously, if the value of reference is the “previous value” – the one resulting from the last execution of the model, its value is updated with the new value. This is not done when the reference value is the last reported value, since at this point it is not always known if an alarm will be generated. (Sometimes the severity of the variable being calculated will be sufficient to generate an alarm by itself, but this will not always be the case). The program applies the rule specified as “variable to report”, and if appropriate, will update the number and new value of the variable in the text-building routine, which will in turn be used to build the message of the alarm, should one result from the execution.

When the last calculated variable has been recomputed, the associated index of severity will determine both the need for an alarm to be generated, as well as the corresponding communication channel. Both of these are determined by the range of the index-associated communication channel table defined as part of the client’s alarm criteria. The message is built and stored, with other data needed to deliver the alarm, in the Generated Alarms Table, whose fields are shown in Figure 10.

The last step of the execution of the model is performed only when an alarm was generated. The reference values used by the criteria of all critical values are updated, since now these are the last reported values.

Since FLAG can have a great number of customers, whose models will be executed whenever the need arises, there is a process, diagrammed in Figure 10, that periodically, for example every 20 minutes, scans the alarms in the Generated Alarms Table, and processes those that have not been sent (SENT = FALSE).

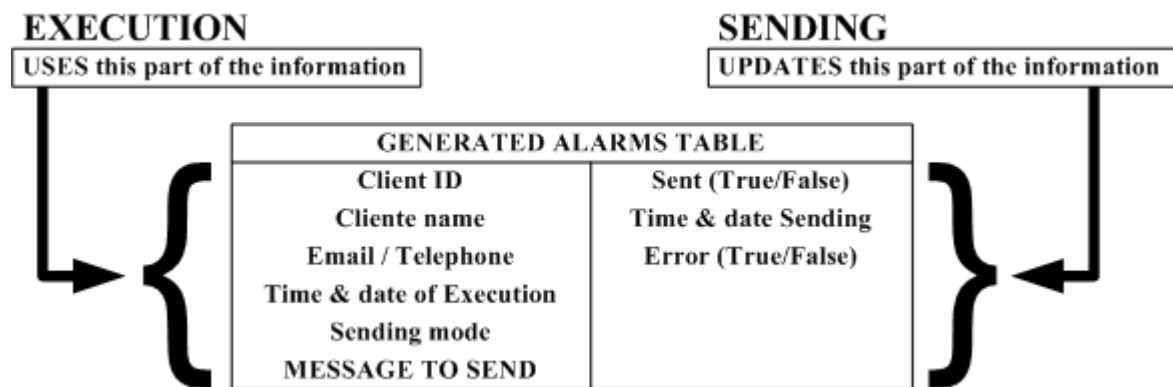


Figure 10. Alarm generation during the execution of the model.

This will result in the generation of an email message or an SMS, or in the most severe situations, an employee will make a personal phone call. The program will set the SENT field to TRUE whenever the process is successful (the message was delivered) so that it will not be included in the next process. Should an error occur, the ERROR field will be set to true.

To implement the personal phone calls, if there are pending alarms to be delivered in this manner, another program displays all such alarms. For each one, the program shows the message to be delivered, but also has an option to connect a querying facility directly to the client’s database, so the operator can answer specific questions and provide values of any variable the client might wish to know.

Remarks on the Design of the Alarms Component

The description of the alarms criteria might elicit a remark in the sense that they are unnecessarily complex and sophisticated. The main object of their design was to balance two sometimes conflicting attributes of such criteria. They should be powerful, or as we like to call it, flexible, but they must also take into account the resulting complexity of the tasks involved. After citing Cohen (1999): “The driving force behind the creation of informing environments and delivery systems is that a task needs to be accomplished”, Gill and Hicks (2006) add: “As the driving force in the development of informing systems, the role played by task is of particular significance to the individuals responsible for developing such systems, since better insights into the task to be accomplished should improve their understanding of the resources that will be required. In this context, task complexity appears to be a particularly important characteristic”. Gill (1996) exposes a factor which adds another dimension to the design: “When IT is introduced, the situation often becomes even more convoluted – as such technologies can enable task performance by new individuals”. Thus, when alternatives regarding the ways to present FLAG’s clients with ways to define their alarms criteria were evaluated, we took into consideration several factors which contribute to the complexity of the associated tasks. Since we used them in this evaluation, we quote some of the 13 “existing task complexity constructs” presented in table format on page 8 of the above quoted paper by Gill and Hicks, where the numbers correspond to the ones provided in the authors’ table: 1) Degree of difficulty; 3) Degree of stimulation; 4) Amount of work required to complete the task or information load associated with the task; 5) Amount of knowledge; 6) Size; 7) Number of paths; 8) Degree of task structure; 9) Nonroutineness or novelty; 12) Function of alternatives and attributes.

On the other hand, precisely taking into account that persons with different needs and access to different communication modes would have to use them, the criteria had to be flexible enough to allow an individual client to obtain what he wanted or needed. The inclusion of more than one critical variable, added to the fact that changes in several of their periods could be significant, complicates matters considerably. Finally, though most customers would probably choose one of the communication modes, it seemed appropriate, and not too difficult to understand, to offer the feature that would transform the urgency – given by the severity index – into a quicker alarm delivery.

To deal with this complexity and its probable implications, FLAG offers several versions of the program described in this paper to define alarm criteria. A user will use the version that corresponds to his knowledge and motivation to understand the different ways to define his alarm criteria. At the beginner level, only one interval for a single critical variable will be offered, and the choice of periods will be simplified. A second level will offer the possibility to choose more than one critical variable, and finally, the complete version can be offered. As far as indicating ranges for the different communication modes, the customer will generally select only one of them, so that difficulty is easily sidestepped. In fact, we do this in most of our information systems and software products: the user will either choose or be assigned an expertise level, and the programs will take this level into account, offering options or alternative paths only when they will not confuse or affect the user in some other way.

Conclusions

One of the salient characteristics of FLAG is to inform its clients immediately when something has happened that merits their attention. The way to implement this in the system was designed, and the resulting programs and procedures tested. As far as the usefulness of this type of service, owners of small agribusinesses, as well as farmers, were informally asked if they would use a service such as FLAG. The response was unanimous: they would welcome having up-to-date information, especially if they could quickly interpret it in the context of their businesses. The in-

clusion of the possibility to get a warning only added to this enthusiasm. However, severe doubts arose regarding the need to formulate their own models and criteria; the remark that specialized personnel would aid them in these tasks was not sufficient, so it must be taken into account by anybody who wishes to offer this service. As to the formulation of the alarms criteria, we only showed them an example with a single period of one variable. Even so, only a few understood what we were talking about, which leads us to believe that our selection of the structures and procedures to define them did not achieve one of its goals: to reduce the associated task complexity. As pointed out, the software product contains several versions of the program which allows a client to formulate his criteria: they offer default paths and options. In conjunction with some well prepared tutorials and explanations, they should enable the definition of the alarms criteria. Future research regarding the alarms component of the service, besides considering alternative ways to implement alarms, includes finding and implementing other communication channels, as new alternatives appear on the market, as well as changing the structure and contents of the messages themselves. FLAG, and its alarm component described in this paper, may well be a valuable tool for many persons who can obtain information at a reasonable cost, but in a way which may provide them with a competitive advantage.

References

- Bauer-Mengelberg, J. (2010). An informing service based on models defined by its clients. *Informing Science: the International Journal of an Emerging Transdiscipline*, 13, 87-119. Retrieved from <http://www.inform.nu/Articles/Vol13/ISJv13p087-119Bauer552.pdf>
- Birdsall, W. F. (2009). The role of the client in informing science: To be Informed and to Inform. *Informing Science: the International Journal of an Emerging Transdiscipline*, 12, 147-157. Retrieved from <http://www.inform.nu/Articles/Vol12/ISJv12p147-157Birdsall535.pdf>
- Cohen, E. B. (1997). IS as an Evolving Field. *Proceedings of the World Multiconference on Systemics, Cybernetics, and Informatics, Caracas, Venezuela*, July 7-11,1997, ed. by N. Callaos, C. M. Khoong, and E. Cohen.
- Cohen, E. B. (1999). Reconceptualizing information systems as a field of the transdiscipline informing science: From ugly duckling to swan. *Journal of Computing and Information Technology*, 7(3), 213-219.
- Cohen, E. B. (2009). A philosophy of informing science. *Informing Science: the International Journal of an Emerging Transdiscipline*, 12, 1-15. Retrieved from <http://www.inform.nu/Articles/Vol12/ISJv12p001-015Cohen399.pdf>
- Decision Models. (2008a). *Array formulae*. Retrieved April 4, 2008, from <http://www.decisionmodels.com/optspeedj.htm>
- Decision Models. (2008b). *Excel's smart recalculation*. Retrieved April 6, 2008 from <http://www.decisionmodels.com/calcsecrets.htm>
- FLAG. (2010). <http://jbauerm.com/flag/english/index.html> Last updated March 8, 2010.
- Gackowski, Z. (2005). Informing Systems in business environments: A purpose-focused view. *Informing Science: the International Journal of an Emerging Transdiscipline*, 8, 101-122. Retrieved from <http://www.inform.nu/Articles/Vol8/v8p101-122Gack.pdf>
- Gill, T. G. (1996). Expert systems usage: Task change and intrinsic motivation. *MIS Quarterly*, 20(3), 301-329. Retrieved February 15, 2010 from <http://www.misq.org/archivist/vol/no20/issue3/vol20n3art3.html>
- Gill, T. G. & Hicks, R. C. (2006). Task complexity and informing science: A Synthesis. *Informing Science: the International Journal of an Emerging Transdiscipline*, 9, 1-30. Retrieved February 20, 2010 from <http://www.inform.nu/Articles/Vol9/v9p001-030Gill46.pdf>

Appendix

A Program to Generate an Email to Convey a Message

Figure A1 shows a program, written in Visual Basic Net®, that invokes a component called System.Net.SMTPCLIENT to generate an email, after receiving the values of the necessary parameters.

```

Private mailserv As String = "MAIL SERVER"
Private sender As String = "SENDER"
Private reciver As String = "RECIVER"
Private subject As String = "SUBJECT"
Private body As String = "MESSAGE BODY"

Sub send__message()
  Try
    Using message As New MailMessage(sender, reciver, subject, body)
      Dim relay As New SmtplibClient()
      Dim user As New System.Net.NetworkCredential("USER", "PASSWORD")
      relay.Host = mailserv
      relay.UseDefaultCredentials = False
      relay.Credentials = user
      System.Threading.Thread.Sleep(1000)
      mailClient.Send(message)
    End Using
    MsgBox("MESSAGE SENT")
  Catch ex As FormatException
    MsgBox(ex.Message)
  Catch ex As SmtplibException
    MsgBox(ex.Message)
    MsgBox(ex.InnerException)
  End Try
End Sub

```

Figure A1. The use of MailMessage, SmtplibClient and Catch.

A Program to Send a Text Message (SMS) to a Cell Phone

Sending an SMS message can be done with Microsoft SMSSender® (Microsoft SMS Sender (2003) *Readme File*, México: Microsoft Corporation). Though not included in Windows XP, it can easily be downloaded from Microsoft's website. Figure A2 shows the code that invokes this command, after furnishing the necessary data via its parameters: the intended receiver's cell phone number and the message itself. The final action is produced by the "Shell" command. Of course, the routine will detect any failure, and the programs will deal with such situations according to their special needs.

```

Private Sub prepare_data()
    path = "C:\Microsoft SMS Sender\smssender.exe"
    telephone = "RECIVER PHONE"
    message = "[MESSAGE BODY...]"
    phone = CDbI(telephone)
    parameters = "" & "/p:" & phone & "/m:" & """" & message & """"
    MsgBox(parameters)
    package = path & " " & parameters
    MsgBox(package)
End Sub

Private Sub send_sms_message()
    ID = Shell(package, AppWinStyle.Hide)
    If ID = 0 Then MsgBox("MESSAGE NOT SENT")
End Sub

```

Figure A2. Routine that sends an SMS message.

Biography



John Bauer Mengelberg, after obtaining a degree in Mathematics from the Universidad de Buenos Aires, Argentina, got a PhD in Statistics and Operations Research from the University of Wisconsin, at Madison, where he also taught courses in the area of Stochastic Programming. He has since worked in Mexico, where besides teaching at the Colegio de Postgraduados, a school primarily involved in the field of Agronomy but which has both Statistics and Applied Computing departments, has held several positions, always connected with the field of Information Systems, in which he has also been a consultant all his professional life. He is primarily concerned with the subject of “systems that work”, a concept he has extended to signify that they work even under abnormal circumstances. He has created and implemented many computer packages, and is currently working on several software products regarding publishing papers or books in the electronic media, and what he calls subsetting in very large data collections, specifically by means of a different type of database. He has often complained he has to work by himself, and his main interest in attending conferences seems to consist in finding ways to change this.



Gilberto David Velázquez Castro, after obtaining a degree in Energy Engineering at the Universidad Autónoma Metropolitana, Unidad Iz-tapalapa (Mexico City), completed work for his Masters degree in Applied Computing at the Colegio de Postgraduados, Campus Montecillo (Mexico) precisely with a thesis based on the alarms component of FLAG. He has taught at the high school and undergraduate levels, besides participating in the design, implementation and management of the data and communications network at Carson & Brasch Corporation, a financial services company.