

ABSTRACT

**ESTABLISHING CRITERIA FOR EVALUATING A PROBLEM SOLVING SYSTEM**

By Rosemary Cook, Applied Research and Engineering Sciences (ARES) Corporation  
Scott Alan Jones, Executive VP Technology, Decision Systems, Inc.

Recognizing the growing need for both a unified problem solving methodology for resolving and heading off operations problems, and a unified system for case and solution data to be archived for immediate lessons learned access within the aerospace industry, The ARES Corporation studied, tested and evaluated available technologies for deployment. Critical data-quality elements of verifiable accuracy, consistency, objectivity, reliability and relevancy dominated their concerns. This paper details the criteria that were developed and established to study and evaluate root cause analysis methodology and lessons learned concepts, and explains how the new system soon to be

deployed meets those criteria to improve and unify mission problem solving capabilities. In the study, criterion for data quality was ranked highest in importance. Critical elements of data accuracy and completeness, and their relevance to effective solutions and response are examined. Relevancy of the criteria to the aerospace industry, as they impact processes, goals and strategies for operations reliability is discussed. The methodology and lessons learned design capabilities of the new Root Cause Analysis and Lessons Learned System are explained and illustrated as they apply to the satisfaction of the evaluation/study criteria, as well as the actual problem solving process.

## ESTABLISHING CRITERIA FOR EVALUATING A PROBLEM SOLVING SYSTEM

Paper #: 2001-4608

(Prepared for presentation at the AIAA Space 2001 Conference and  
Exposition)

By Rosemary Cook, Applied Research and  
Engineering Sciences (ARES) Corp.  
Houston, Texas

Scott Alan Jones, Executive VP Technology, Decision  
Systems, Inc. Longview, Texas

Recently the ARES Corporation established criteria for an industry team that was charged with the task of evaluating the various root cause analysis and lessons learned, problem solving and engineering systems that were available, as well as existing analysis methods being used within the aerospace industry. The process for establishing criteria for the selection of a system consists of 1) determining what is expected from the system, and 2) then determining what systems criteria will provide the functions that will best produce those expectations. The manner in which the system produces its results, and the quality of those results are then a function of the extent to which the system meets the criteria.

The Team determined four fundamental activities in which critical quality considerations would impact the success of the root cause analysis and lessons learned system: collection of data, analysis of data, communication of data, and archiving of data. These interrelated and interdependent functions were believed to be equally critical to success. None-the-less, a hierarchical criteria structure was adopted, recognizing that no amount of analysis and reporting rigor would be able to remedy or compensate for the lack of accurate and complete data.

Later, we will be explaining and demonstrating how the system that was selected for deployment meets the following set of criteria. For the reader, we have provided a criteria document for reference, and are providing reference coding in the text.

### **Technical Accuracy of Data (Category 1)**

The evaluating Team placed highest priority upon the technical accuracy of the data that were to be produced by the problem solving method. Of the criteria associated with data quality, the ability of the process itself to validate the accuracy and relevancy of the data

was assigned the highest priority (1.1). The utility of a built-in, procedural process that would require investigators to gather adequate data by driving them to systemic root causes was felt to be the next most significant desired capability within the methodology (1.2). The capabilities to reliably filter out irrelevancies through application of process (1.3), and to identify the need to acquire additional information when appropriate (1.4) were considered important keys to generating technical accuracy of the data, and were assigned high priority as well.

In summary, relevant criteria for technical accuracy describes a system with rigor that through repeatable process generates and validates accurate and complete data for problem solving decision support.

### **User Friendly Software (Category 2)**

Not surprising, the evaluation Team placed the next highest overall priority, after technical accuracy of data, on the need for the system to be user friendly. Within that category, they prioritized Ease of Use (2.1), which they described as “easy to understand and work through logic”, the highest single priority criterion within the entire evaluation standard. Three additional criteria were developed to describe and assess the user friendliness of the software system: minimum time required to input data (2.2), minimum time to generate reports (2.3), and understandable cause code layout (2.4).

For efficiency, the Team sought the capability for the software system itself to automatically generate analysis reports, including narratives (2.5) and fault trees/flow charts to visually display causes and links in error chains (2.6). In summary, the Team wanted a system that facilitated exact answers with a minimum of time and effort, and without system impediments.

### **Analysis of Data (Category 3)**

Confident that the preceding criteria would quickly produce the level of data necessary to support mission critical decisions, the Team devised criteria designed to provide reliable decision support for immediate resolve of problems, as well as early detection of developing problems in processes. To foster the capability to respond quickly to identified immediate problems, a criterion was established that called for some procedural way to objectively measure and compare available solution options and action plans, in order to assess their potential for immediate prevention benefit

and their overall control impact upon the identified problem system (3.1). The Team established two additional analysis criteria intended to provide the means to detect early developing trends: analysis capability to identify common causes (3.2), and the ability to research results of similar undesired outcomes (3.3).

**Archiving and Communication of Data (Categories 4 and 5)**

Only one evaluation criterion was established for accessibility of data. However, it was assigned the same high priority that was assigned Ease of Use. The criterion called for “Data viewable to all locations via the Web” (4.1). Thus, the three highest priority issues pinpointed by the criteria describe a user friendly, easy to use system that will generate technically accurate data that can be broadly accessed by all facilities and personnel via the Web.

As an adjunct to the criterion requiring easy and broad access to the data, the Team adopted additional criteria for a system that would provide adaptability to existing systems: Integration with existing databases (5.1), Export/import capabilities (5.2), and Incorporation of existing cause codes (5.3).

Now let’s see how these criteria translate into functional utility. We are going to take you through the integrated process that is provided through application of the selected root cause analysis system. As we experience the process together, we will note for you when the function of the software directly relates to the satisfaction of the criteria requirements. Then we will have a chance to discuss and address questions regarding the long-range implications of the criteria and the data produced by the system.

**Satisfying the Criteria Requirements**

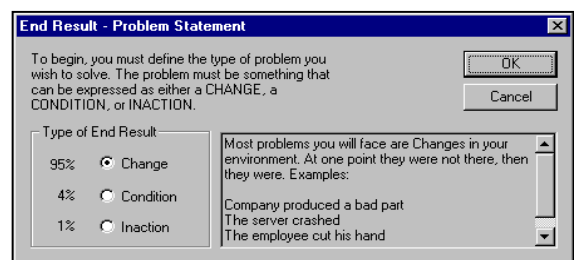
We have chosen an example problem that deals with everyday elements that do not require more than everyday knowledge or experience, although the system applies just as validly to complex space issues, exotic

processes, and hostile environments. Let me quickly tell you what happened.

**Case Facts:**

“Paul, a relatively new night shift employee, arrived late for work again. His supervisor was angry. He told Paul that they would need to discuss his repeated late arrivals after the shift; but first, Paul needed to refuel the forklift. The Supervisor explained that there were several problems with the forklift. The lights on the forklift did not work. The supervisor warned Paul that there was a leak in the fuel line between the fuel line coupling and the tank. It would leak only during the refueling process. Both problems had been turned into maintenance, but the forklift had been returned without being repaired. The supervisor told Paul that it should not be a problem because there were lights at the fueling depot that would automatically switch on when it became dark. The supervisor explained the process for refueling: couple up to the pump, push the single start/stop button to begin fueling, watch the gauge on the forklift and press the button again to stop when the tank is full. As Paul drove out of the plant through the bay doors, he could see the outline of the refueling depot in the distance. The sun was just setting and it was getting dark. Paul continued to the refueling station. When he arrived it was completely dark, however the auto-lights had not switched on. He coupled up by feel, and started the pump. Paul could not see the gauge to know when the tank was full. Paul was worried about his job, what the supervisor might say, taxes, and the new baby on the way. At that moment he had a decision to make; would he go back through the pitch-black night to his angry foreman and tell him that he did not get his job done, or would he come up with another way to get the job done? He tried jostling the tank, tapping on the tank to see if he could hear when it was full. Paul thought, “if I could just get enough light on the gauge for a moment, then I would be able to estimate how long it would take to fill the tank.” So, Paul lit a match. A fuel gas explosion resulted. Paul was seriously burned.”

To begin, our focus will be on the Criterion 1.1 that required a built-in process that guides the investigator to gather adequate data. The expert system software starts this process by guiding the investigator to accurately and precisely express the problem, as a change, or a condition or an in action.

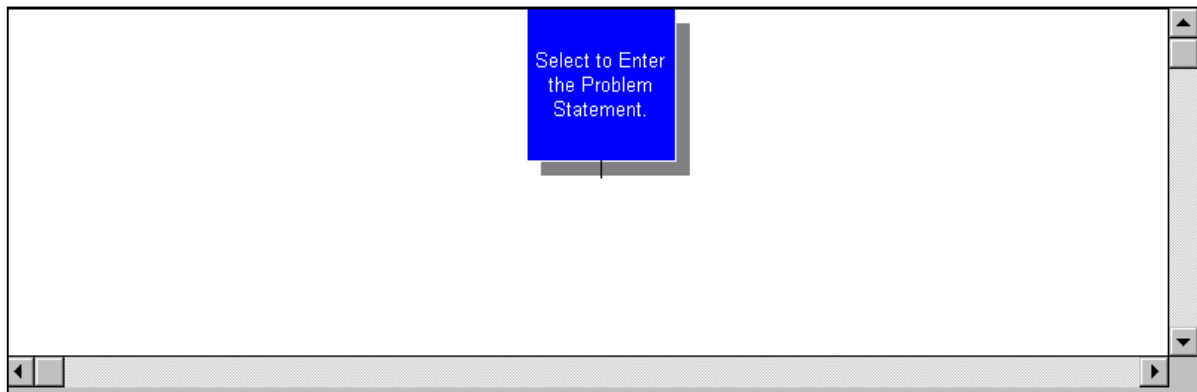


Because the explosion is a dramatic example of a change, we will click Change.

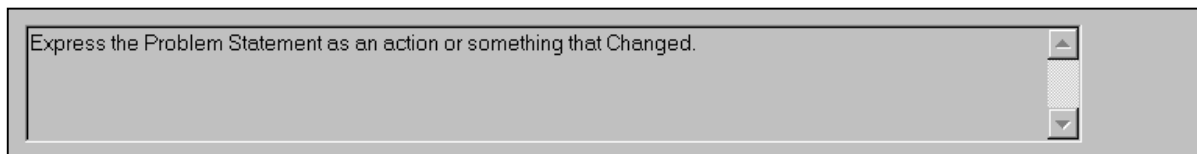
guides the user to clearly state the data. The idea is to look one-step-at-a-time at the system that produced the problem.

In response, the software provides a work area to begin gathering relevant information: a cause set area, an advise area in which questions and counsel are provided to keep the analysis on track, and a template area that

The Set Area displays the set of causes as it is being built.

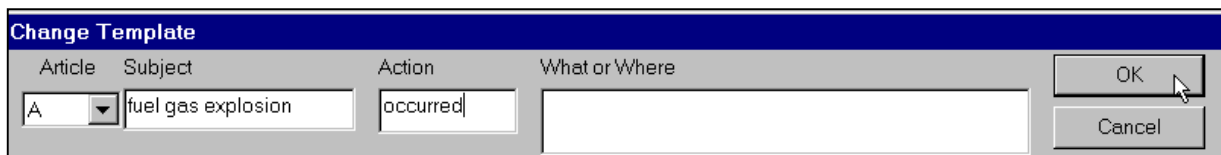


The Advise Area, where questions and counsel are provided to facilitate the investigation.



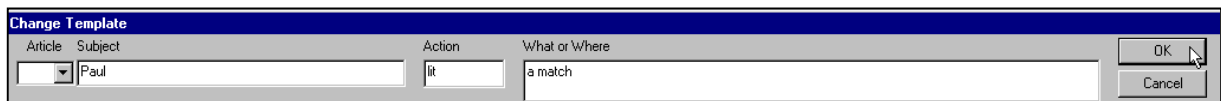
And the Template Area, where we input the change information to start building the first set in our model of

the problem. "A" [tab] "fuel gas explosion" [tab] "occurred"



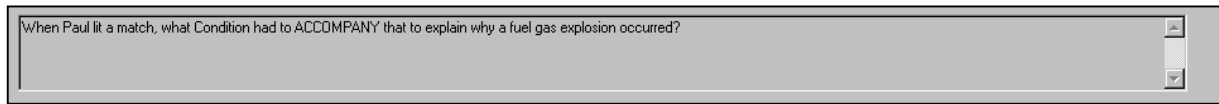
The Advise area then asks. "At one point a fuel gas explosion did not occur, then DID. What triggered this; what changed?" We know that the change that

produced the explosion was when Paul lit a match; so we will enter that fact.



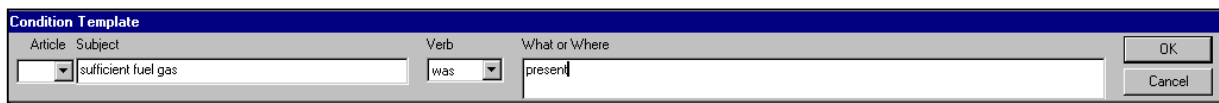
The advice area responds, **“When Paul lit a match, was there a condition already in place necessary to explain why a fuel gas explosion occurred?”**

We answer **yes**, and the Advise area now asks us, **“When Paul lit a match, what had to ACCOMPANY that to explain why a fuel gas explosion occurred?”**



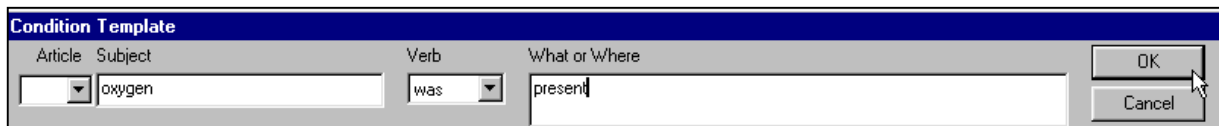
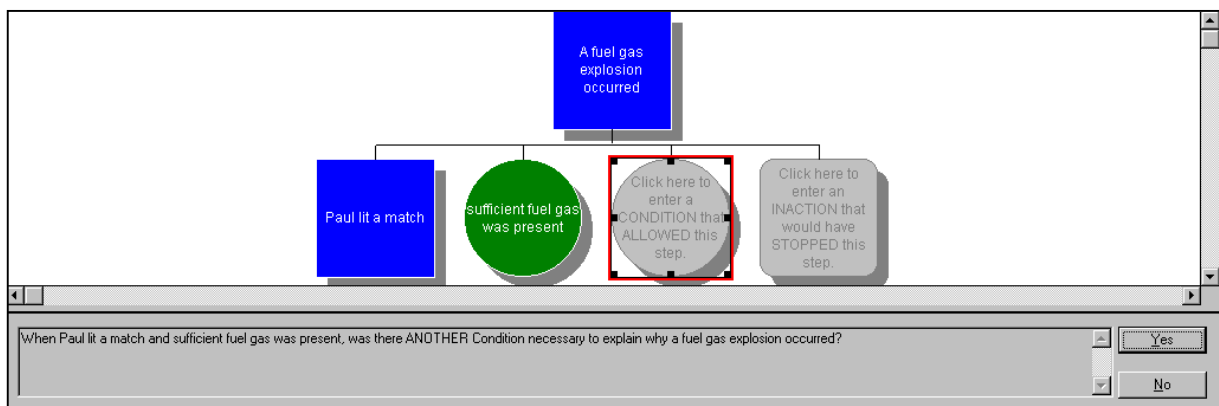
The software has presented a special template for me to answer,

[tab] **“Sufficient fuel gas”** [tab] **“was”** [tab] **“present”**



The advice area now reads: **“When Paul lit a match and sufficient fuel gas was present, was there**

**ANOTHER Condition necessary to explain why a fuel gas explosion occurred?”**



You may be saying to yourself, “Gee do we really have to input the Oxygen?” This is a point that demonstrates how the expert system software helps meet that primary Criterion (1.1) that calls for accuracy and completeness of data. The system does not let us leave out things that we personally believe might be obvious or unimportant. With this system, vital information does not fall through the cracks because someone thinks it is not important. The process itself requires us to do a

thorough analysis, to completely answer what happened and what can be changed to prevent a recurrence. To answer these questions we must include all of the facts. You will see in a minute that in this case we are not going to spend a lot of time on the fact that Oxygen was present, but we all know that in a different environment, the presence of oxygen could be critical and could spell disaster.

The advice area is asking, “When Paul lit a match and sufficient fuel gas was present and oxygen was present was there another condition that had to

accompany that to explain why a fuel gas explosion occurred?” We will answer NO.

When Paul lit a match and sufficient fuel gas was present, was there ANOTHER Condition necessary to explain why a fuel gas explosion occurred?

Yes No

Now in order to demonstrate how the system meets the criteria for accuracy and completeness of data, we are going to intentionally make an error. We will stray away from logic to demonstrate how the system deals with errors in thinking, so that you will see how the process itself helps the investigator keep on course as required by criterion 1.2.

The software is asking, “Realistically was there enough time after Paul lit a match and before a fuel gas explosion occurred for something to have kept a fuel gas explosion from occurring?” The expert system is trying to find out if there was an opportunity for installing a barrier for prevention, or if a barrier already in place did not function. In this case there was not the opportunity to intervene once the match was lit, but we are going to make an intentional error. We will answer yes.

Realistically, was there enough time AFTER Paul lit a match, but BEFORE a fuel gas explosion occurred for something to have kept a fuel gas explosion from occurring?

Yes No

The software asks. “AFTER Paul lit a match, what DID NOT happen that should or could have that

would have kept a fuel gas explosion from occurring?”

AFTER Paul lit a match, what DID NOT happen that should have kept a fuel gas explosion from occurring?

I

Here is where we are going to make the mistake. We are going to take a fact that is actually causal, but enter it in the **wrong place**. Remember those auto-lights that did not go on? We have the feeling already that this was

somehow causal to the explosion, and we are going to input the fact that “**The Auto-lights did not switch on.**”

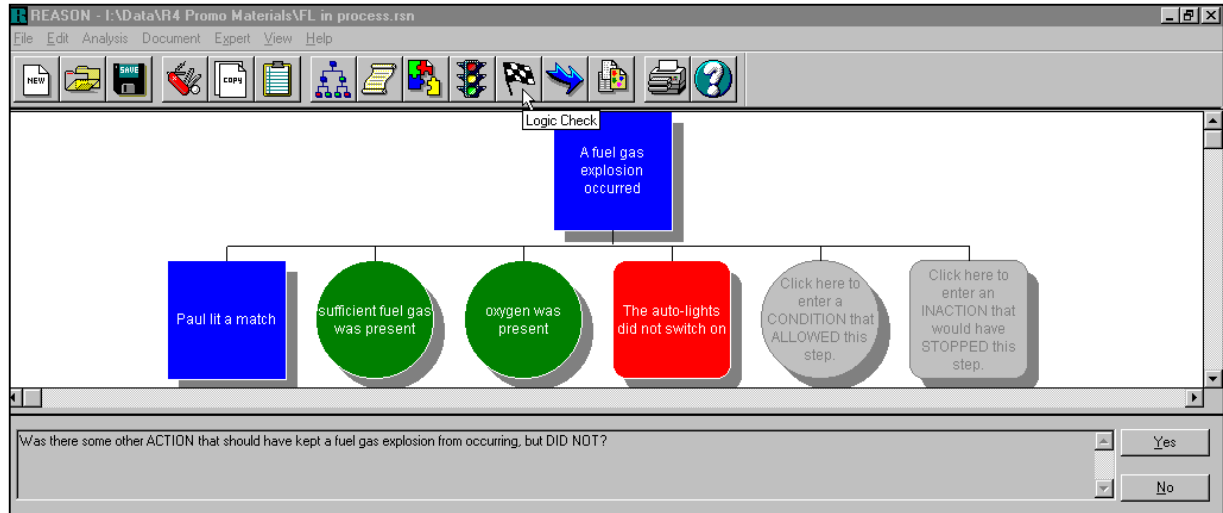
**InactionTemplate**

Article	Subject	Nots	Action	What or Where
The	auto-lights	did not	switch	or

OK Cancel

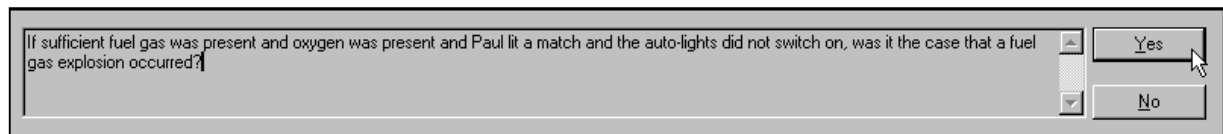
Now we are going to tell the software that we are through building the set of causes by selecting the finish flag icon on the menu bar. Note that during the inquiry process, the software automatically displays

your data in real-time model form, so that you are always reviewing the causal relationships and links. This promotes efficiency and accuracy at each step of the inquiry process and supports the Criterion 2.6.



The expert system now conducts a quality assurance logic check to make sure that the data entered are sufficient to explain the explosion, and that all data entered are necessary to explain what produced the

explosion. This is where in the process the expert system meets the criteria 1.2 and 1.3 for a built-in process for validating data accuracy and completeness, and filtering out error.

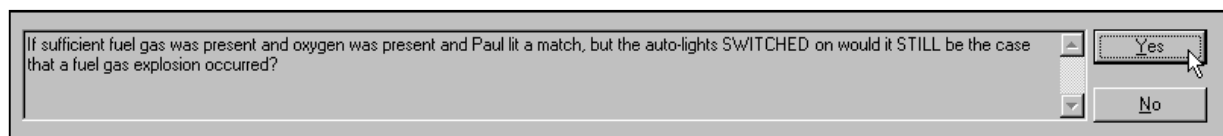


The software is asking if the lit match, gas and oxygen will produce an explosion in the dark. Obviously, it will.

your set of answers has enough data at this point to explain why the explosion occurred.

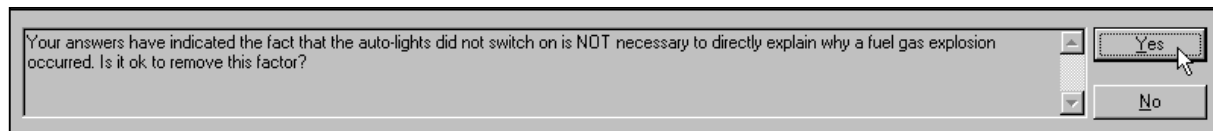
Note that there could have been many other facts included here as well, and still the statement would be sufficient. For example it might have been Thursday, it might have been the supervisor's birthday, and New Years Eve. In these cases the explosion would still have occurred. This part of the check makes sure that

Now The software's logic check wants to validate each causal fact in the set separately to make sure that each is necessary at this point to explain why the explosion occurred. It asks, **"If sufficient fuel gas was present and oxygen was present, and Paul lit a match and the auto-lights SWITCHED on, would it still be the case that a fuel gas explosion occurred?"**



If you have an ignition source in a fuel-gas-rich environment with Oxygen, do you get an explosion, even if the lights are on? Of course, I'll answer yes. Watch how the system handles this.

The software responds, **“Your answers have indicated the fact that the auto-lights did not switch on is NOT necessary to directly explain why a fuel gas explosion occurred. Is it okay to remove this factor?”**



We answer yes. The expert system removes the error from the model, and takes me to the next logical point to continue the investigation. Several of the criteria for a root cause system deal with the utility and efficiency of the time and effort required to conduct the analysis. This accuracy check is another safeguard that avoids needless time and effort being spent pursuing red hearing facts that later prove to be irrelevant, keeping the input time to a minimum, as required by Criterion 2.2.

To continue with this investigation, we would follow the same procedure under each of these factors in order to identify all of the facts that had to be there to explain why each step in the problem occurred.

What do you think the answer to the **‘Oxygen was present’** factor would be? If the presence of oxygen

was unexpected or accidental or unwanted, we would want to know exactly how and why the oxygen was present, but in this case the oxygen was expected and wanted in this environment. The system enables the user to determine that this factor is **non-correctable (NC)**. A non-correctable fact is something that we either cannot change, or do not want to expend resources to change. This illustrates a valuable function of the system software. Causal information about a factor that may be considered by an individual to be beyond the capability of the organization to control is not subjectively filtered out of the process before the analysis is conducted and presented to management. The fact is instead carefully recorded and identified as non-correctable. This is another process safe guard that reinforces the compliance to the Criterion 1.1.

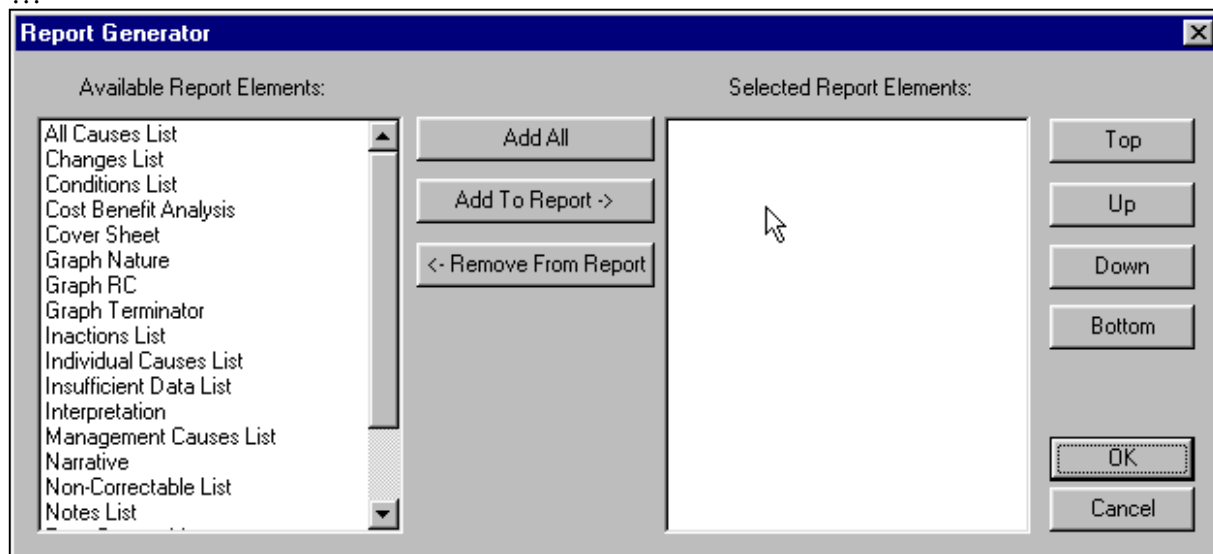
The system also allows the user to note when there is currently not enough information to answer a factor. Building the sets and applying the logic check make sure that the data are recorded. During the inquiry process, when we don't have immediate access to information, or if we just don't know the answer to a factor, the process requires us to mark the factor as Insufficient Data, in order to complete the model and the analysis process. This action marks the areas in the analysis where more investigation is needed, as required by Criterion 1.4. If additional information is obtained later, we can quickly input the new information, and resume the questioning process.

Now let's look at several capabilities of the system that are on the output side of the process. We are going to use the completed case to demonstrate how the system directly addresses many of the criteria that impact efficiency of the root cause analysis and reporting.

Once the data are gathered, the system instantly generates an analytical model of the causal system, showing how all of the causes combined and interacted to produce the problem, the root causes identified, and the organizational solutions for control. The model and the data contained by the model directly address the Criteria 2.2, 2.4, and 2.6.

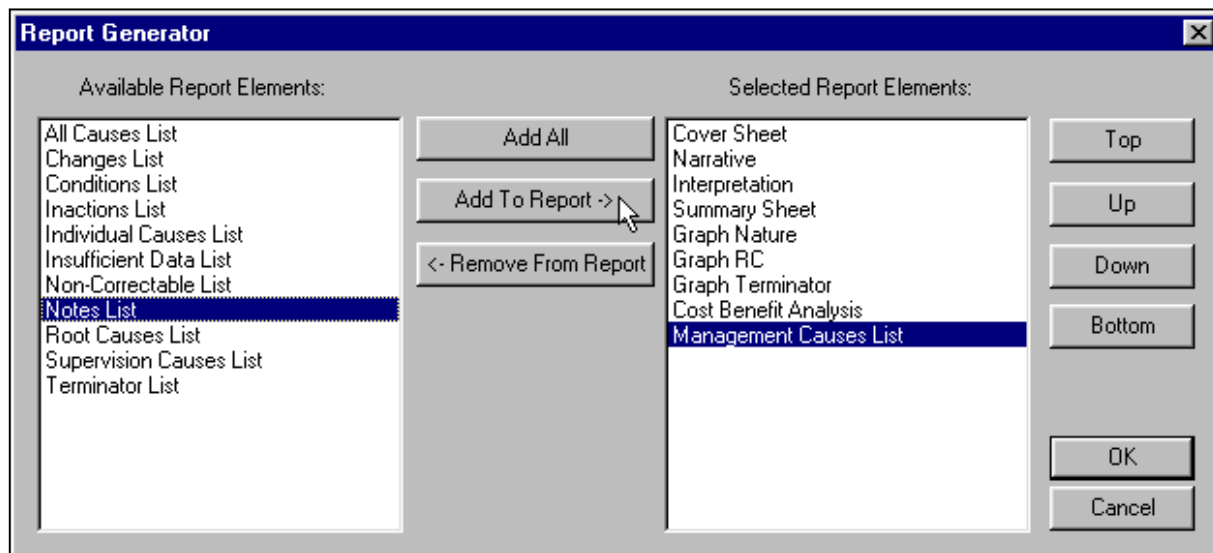
The Report Generator feature of the system provides an array of options for fast, customized decision support

and compliance reports meeting the requirements established by criteria 2.3.



Because the expert system has been qualifying and ordering the data in real time while the information was being gathered, the Report Generator can produce an array of analyses to prepare instant, customized reports, as required by criterion 2.3.

Just hi-light each item that you wish to include in your report, and then click Add To Report. This will move the selected item from Available Report Elements on the left to Selected Report Elements on the right.



A typical report generated instantly by the system might include a cover sheet, a descriptive narrative that explains step-by-step how the problem occurred, an interpretive analysis that compares options for prevention benefit, and a cost benefit analysis. In

summary, the system provides the capability to produce immediate analyses and reports that are so critical to proper response in emergency situations, meeting the both criteria 2.3 and 3.1.

Now let's look at the archiving and communication of data. Paramount to the selection of a root cause system was the capability to access the information broadly and immediately. The Team's criterion was unqualified. It required access to the data by all locations via the Web. This criterion speaks to broad communication as required by Criterion 4.0. The system selected deals with the time, effort and friendly issues in reference to both the root cause analysis capabilities and the lessons learned capabilities.

Common in traditional lessons learned systems is the need to spend time preparing special, documented reports for submission to a system that relies upon an indexed word search for access. You know what happens when a large volume of indexed data is accessed by a word search. The results of the search can produce many "hits" that then have to be studied for relevancy. It seems with traditional lessons learned systems that the more valuable the database is, the less user-friendly and practical the system appears to the users, especially if they are relying upon the data in time-critical situations. The selected lessons learned system gets around both of these obstacles by providing

instant submission of the analysis to the Web –access database, and a unique search that seeks points in time in the process at which the elements combined to generate a step toward counter quality.

In short, all hits are direct hits. Summarizing, the lessons learned system meets criteria 4.1, 3.2 and 3.3, supports criteria 2.3, and removes the two most common impediments to the routine use of lessons learned data.

In conclusion, the ARES Corporation, and the other members of the aerospace Team sought a system that would meet their existing needs for accurate and reliable root cause analysis and lessons learned. A set of criteria was established that would guide the evaluation, research and study of available systems, and the eventual selection of the system. By focusing upon the four essential task functions within the problem-solving and communications activity, the Team was able to establish and apply criteria that successfully defined the quality of data and the results required for critical environment application within the aerospace industry.

— — —

**Aerospace Team Root Cause Analysis Evaluation Criteria**

Created by Rosemary Cook, ARES Corporation

**REASON<sup>®</sup> 4 Root Cause Analysis – Lessons Learned System – June 5, 2000**

**Collection of Data**

*Technical Accuracy*

1.1	Process forces users to gather adequate data (can't quit too soon)	Yes
1.2	Process validates accuracy	Yes
1.3	Process filters irrelevancies	Yes
1.4	Identifies during data collection the need for additional data	Yes

*User Friendly*

2.1	Ease of use (easy to understand and work through logic)	Yes
2.2	Minimum time required to input data	Yes
2.3	Minimum time to generate reports for management review	Yes
2.4	Understandable cause code layout	Yes
2.5	Produces narrative reports	Yes
2.6	Produces fault trees/flow charts (to visually display causes)	Yes

**Analysis of Data**

*Analysis/Trending*

3.1	Measures and compares solution options (i.e. if you eliminate one root cause you've eliminated 60% of the problem)	Yes
3.2	Ability to identify common causes	Yes
3.3	Ability to research results of similar undesired outcomes	Yes

**Archiving and Communication of Data**

*Accessibility*

4.1	Data viewable to all locations via the Web	Yes
-----	--	-----

*Adaptability*

5.1	Can integrate with existing databases	Yes
5.2	Has export/import capabilities	Yes
5.3	Can incorporate existing cause codes	Yes

If you would like more information about REASON® Root Cause Analysis Software, Training and Consulting in Australasia please contact:

REASON4 Consulting  
Level 14, 309 Kent Street  
SYDNEY 2000

t +61 2 9994 8009  
e [info@reason4.com](mailto:info@reason4.com)  
w [www.reason4.com](http://www.reason4.com)