Abstract. Research is being conducted at the RMIT University on the ‘Improvement of Aircraft Accident Investigation through Expert Systems’. The outcome of this research presented in this paper is a novel investigation tool in the form of a data mining method designed towards giving aircraft accident investigators improved utilization of forensic data. The tool, named GP1020, is primarily a data sorting tool that sifts through a huge amount of data, namely those regarding aircraft accidents/incidents and investigations conducted including a number of causal factors for accidents/incidents associated with their consistent evidence. The GP1020 program interface asks the user a tree-based set of questions related to conditions of wreckage, accident site and other circumstances relevant to accidents/incidents. Given enough information, the program is capable of narrowing down all known possibilities to indicate the most probable cause(s) of the accident/incident.

1. INTRODUCTION
The research project ‘Improvement of Aircraft Accident Investigation through Expert Systems’ aims to analyse the current accident investigation process and to review some of the off-the-shelf tools that support these investigations. An analysis of the process and tools will provide a possible avenue for updating the investigation process and implementing mitigation measures to enhance air traffic safety. The research framework is presented below in Figure 1.

Figure 1: The design and development process for expert systems applied to aircraft accident investigation

The work starts with discussing the procedure of aircraft accident investigation, which is defined as a process conducted for the purpose of accident prevention and focused on the circumstances of the accident including gathering, recording and analysis of all available information, the drawing of conclusions, including the determination of accident causes [3].

Despite the attitude and commitments to achieving the above purpose, accident investigations may become a cumbersome work associated with significant costs and uncertainty. This can potentially contribute to some accidents being assigned an unknown cause as described in the global aircraft accident statistics for the past 50 years. Therefore, investigation process has been subjected to constant review in order to improve its outcomes and to help enhance air traffic safety.

The work concludes that an intuitive and interdisciplinary approach must be vital elements of any contemporary methods used for establishing a set of priorities for further improvement of the aircraft accident investigation process. Hence interpolations methods including an analysis of accident statistics and Delphi enquire are appropriate tools in analysing and drawing conclusions for further improving the investigation process. The statistical data examined contained the number of accidents which occurred between 1950 and 2004 including accident distribution over the past, causal factors, and casualty count. In addition a Delphi enquiry, which is formalized regarding the procedure for carrying out research activities, has provided a comprehensive analysis of the whole procedure of investigation. A team of experts created for this purpose has conducted a qualitative and quantitative analysis of the influential factors having an impact on investigation outcomes. The Delphi study has indicated that there is a great potential for further improvement of the aircraft accident investigation process. It has pointed out the sections where significant improvements of the aircraft accident investigation process could be achieved.

Overall, the current results of this research have addressed the importance of several points with respect to air accident outcomes as follow:

- Every accident occurs as a result of a chain of errors, omissions, or malfunctions.
- Although all aircraft accidents are different, there are certain common elements in accident
causes, and there are a number of causes which frequently result in accidents.

- Accident investigation could be facilitated if its distinguishing features could be quickly identified from large amounts of data in order to help predict possible causes of the accident.
- Improving aircraft accident investigation could be achieved by creating and using advanced databases for storing and analysing the data of aircraft accidents.

These points address the need of creating a tool, in the form of a computer program, which can use stored expert knowledge coupled with an inference engine to process this knowledge and provide safety event analysis to users of the program (This entirely refers to the definition of Expert Systems).

To summarise, this research has shown that investigation could be significantly improved with the application of a global expert system as a tool for storing and analysing the forensic data of aircraft accidents worldwide.

As a consequence the computer program ‘GP1020’ has been created in order to demonstrate how expert systems could contribute to facilitating and enhancing the investigation results. This paper is focused on GP1020 computer tool, its design and features.

2. EXPERT SYSTEMS TOOL ‘GP1020’

2.1 Expert Systems – Outline

The most common form of an expert system is an interactive computer program that examines data stored and provides a problem solution following a set of predefined rules. An expert system involves two principal components: a problem dependent set of data stored known as knowledge data and problem independent program known as the inference engine. Interaction between the user and the inference engine is performed via the user interface, which asks questions and supplies the user’s replies to the engine. However, the possession of expert knowledge is vital for the successful application of expert systems.

Users of the program usually see an expert system through an interactive dialog. The dialog is composed of a set of questions whereby through complex feedbacks, conclusions are drawn. Dialogs are created from the current information and the content of knowledge base.

In general, the knowledge basis of an expert system contains a large number of ‘if then’ type of clauses that gives the expert systems the ability to use them together to draw conclusions. The knowledge that is stored in the expert systems appears in the rulebase, which is composed of four different types of objects such as classes, parameters, procedures and rule nodes. [7]

Among the many different approaches available to classifying a huge amount of data is the widely used method of classification by decision tree induction. It constructs a tree in which internal nodes are split as a result of yes/no decisions (Figure 2).

![Figure 2: Classification by decision tree induction [7]](image)

2.2 Expert Systems To Aircraft Accident Investigation – GP1020

Using the positive practices of expert systems applications in many fields of science, the computer program GP1020, designed for assisting aircraft accident investigation, was created.

![Figure 3: Introduction page of the application GP1020](image)

The user of this program is asked a number of different questions that initially look at the wreckage and accident site followed by examining the human, aircraft, and weather causal factors. According to the answers given, the program will choose the set of most appropriate questions in order to determine the cause(s) of accident/incident. Finally when GP1020 assesses that there is a significant amount of evidence derived it will release the probable cause(s) of this particular safety event.

GP1020 includes two major features:

- A forensic approach to the procedure of an aircraft accident investigation, such that the flow of information, procedures, and the rise of knowledge about the occurrence during a real investigation is also followed by GP1020.
- Being a simple and efficient IT solution in determining the probable cause(s) of an aircraft accident occurrence.

2.2.1 Forensic approach to an aircraft accident/incident occurrence within GP1020

The procedure of GP1020 accident enquiry follows the steps of a real investigation of an accident/incident occurrence. GP1020 asks a broad range of questions relating to the factual information and analysis undertaken of a safety event.

At the initial stages of data collection the GP1020 program intends to learn general information about an accident/incident event such as: event type, aircraft type and category, type of flying, and so forth. In addition there are questions which intend to ascertain whether the aircraft was flown in visual or instrument meteorological conditions or/and experienced any problems during the flight.

After collecting the general information of a safety event, GP1020 will strive to learn about aircraft damage, wreckage distribution, witness location, probable flight path, and occurrence of fire or explosion occurred. GP1020 might ask for information about the person injury category, and cause(s) of death.

GP1020 using a number of questions examines the human, aircraft, and weather induced causal factors. Thus, GP1020 learns if the crew undertook voluntary acts that are poorly performed, failed to act when particular actions were appropriate, or failed to take immediate action, follow air traffic control instructions, use checklists, maintain direction control, monitor weather, and/or monitor instrumentation. There are also questions that assess potentially inadequate preparation or supervision, poor judgment, improper use of equipment, alcohol or other drug use, improper maintenance, improper aircraft modifications, and inadequate procedures.

Other set of questions examines the aircraft systems condition and assess the involvement of those factors as possible cause(s) for accident/incident. Among them are the questions relating to: wreckage and its systems condition, possible breaking of aircraft limits, readings of instruments, data recorders, and emergency procedures carried out.

Next, GP1020 will ask several more questions designed to consider the weather conditions during the accident. It learns event temperature, dew point, sky condition and whether the aircraft experienced events like air turbulence, birds strike, volcanic ash, and dust.

2.2.2 Classification of the causes of aircraft accidents within GP1020

The causal factors and their consistent evidence within GP1020 are stored and available to the program in several different ways. First of all, the contributory factors are sorted within the three well-known causal factors: human, design and manufacturing, and environmental factors [2]. Secondly, the causes of accidents are additionally sorted in accordance with the nature of accidents: taxing accidents, takeoff accidents, collisions, tail spins, fires while midair, forced landings, landing, and other accidents [1]. Finally, the list of all possible causes of accidents is classified according to problems that may be experienced during the accident as follows: lift, thrust, flight control, and weather and environmental induced problems, as well as smoke, fire and fumes, explosion, and other problems.

This approach of multiple classifications of causes of accidents within GP1020 allows the creation of a huge and flexible database appropriate for a quick across search. Thus, within GP1020 a large ‘m,n’ matrix has been created, where ‘m’ represents the number of all possible causes of accident (rows) and ‘n’ represents the number of different portions of evidence that may be recovered during an investigation (columns).

Table 1: A list of possible causes of accident versus their distinguishing features

<table>
<thead>
<tr>
<th>All</th>
<th>Evidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2 3 4 5 6</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>List of Possible Causes</th>
<th>Distinguishing Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>Human Factors</td>
<td></td>
</tr>
<tr>
<td>Cause 1</td>
<td>x x x</td>
</tr>
<tr>
<td>Cause 2</td>
<td>x x x</td>
</tr>
<tr>
<td>Design and Manufacturing Factors</td>
<td>x x x</td>
</tr>
<tr>
<td>Cause 1</td>
<td>x x x</td>
</tr>
<tr>
<td>Cause 2</td>
<td>x x x</td>
</tr>
<tr>
<td>Environmental Factors</td>
<td></td>
</tr>
<tr>
<td>Cause 1</td>
<td>x x x</td>
</tr>
<tr>
<td>Cause 2</td>
<td>x x x</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The matrix above has been converted into another format, shown below in Figure 4, which is appropriate to the needs of the GP1020 program [4, p238].

Each column of the matrix, which represents a different portion of evidence, must be associated with a question. This means that GP1020 is composed of a large number of questions similar to the number of distinguishing features of all possible causes of accidents.

The second table of Figure 4 shows how GP1020 may ask the user questions in order to determine the cause(s) of accident/incident. For instance (in the case presented in Figure 4) if GP1020 asks the question associated with E5 (or QE5) then a positive answer provided by user will automatically finish the procedure as only the ‘Cause 4’ includes evidence E5. On the other hand if GP1020 asks the QE1 and the user provides a positive answer to it then the program will generate further questions related to C4, C8, C2, C1, and C7 that include this evidence, and so on.
The current version of the GP1020 software was to act as a prototype purely to test some of the fundamental mechanisms required by components of the expert system methodology. Thus, a basic version of a knowledge base, inference engine and user interface had to be developed in order to further understand the required interaction between each component of the program.

**Programming Software.** The use of Excel to develop this software turned out to be the most appropriate choice for GP1020. It had built in commands for data mining, sorting, statistical analysis and linguistics analysis. It also has numerous accuracy checks to ensure that any cell formulae and scripts are as correct as possible. Essentially it allowed focus on the development of the fundamental logic over any of the program code required for allowing application execution in any specific operating system. It also avoided the entire issue regarding how to insert the historical data into the program as Excel is already capable of handling numerous table data formats. About the only thing it may, and apparently did, lack is efficiency in terms of computation speed. Throughout the development of the GP1020 prototype this relatively high level of inefficiency was the most prevalent limiting factor to the prototype’s capabilities.

**Knowledge Base.** Originally it was hoped to develop a program that could interface with raw historical data and be ready for questions moments after the integration. However it was apparent early on that a separate program would be needed to alter the historical data records into a form that the inference engine could use.

The simplest, and therefore currently ideal, form for the knowledge base to suit the comparative programming that GP1020 uses would be as a table where each investigation or case is represented by a single row, with consistently occurring types of information represented by various columns.

As this prototype is fairly basic, the only historical data that was really needed to create a fundamental program was some generic information to easily identify the case as well as the causes and factors pertaining to the case (sequence of events, occurrences, flight phases, subjects and modifiers).

In the future, when the program’s operating environment is known, the inefficiencies given by Excel should be largely reduced when the fundamental logic in the knowledge base’s preprocessing is transferred or altered to suit the environment’s chosen operating system or program. At the very least, the data limitations of the knowledge base and its preprocessor should grow parallel to the increase in computing power that will probably become available to it. Given that the program was computed on a personal computer,
increasing the knowledge base to include all forms of relevant recorded data should be fairly feasible.

**User Interface.** The creation of the user interface was the last to be initiated. As a fundamental program, only the basics were required; the ability to ask questions, the ability to receive questions and the ability to display cases that bear the most resemblance to the case currently being investigated. It would have been a very large task had not the NTSB database have already made numerical designations for occurrences, flight phases, subjects and their modifiers. Also, all the important decisions regarding the relative positions of all the cases and all possible questions were determined by the inference engine, so the user interface only had to interact with the inference engine in terms of inputting answers and displaying results.

The decision to show multiple similar cases as opposed to showing only the most similar case was a product of practicality from two perspectives; the user and the program developer. As a program developer, there is a need to see how the displayed result evolves, or is calculated, to ensure that the appropriate functions are occurring. Constant checking of the top 20 results has heavily assisted in making sure formulae and calculations were performing accurately. From the user’s perspective there was an obvious requirement, in that one lone case record could hardly be expected to meet the user’s information needs. While the most similar case to the user’s investigation could well be entirely useful to the user’s investigation, unless it is exactly the same, the most similar case could not fully assist the user in understanding his or her investigation. It would be more likely that the sequence of events that the user is trying to establish in his or her investigation could be derived from the combination of events from two or more different historical records. Essentially, the more relevant historical records are available to the user, the better off the user should be.

The details shown in the 20 most similar cases are its event ID (a unique numeric code designation for the incident/accident), date, aircraft make, operator’s name, departure location and destination location. These details effectively tell who was flying what where when and where to. The only other detail shown is the “Level of Relevancy” which indicates the proportionate number of questions (and therefore answers) that positively match the case shown against the total number of questions asked.

**Inference Engine.** When discussing the inference engine, the entire capability of the program is dependant on its ability to use causes and factors determined in previous investigations to predict future investigations. So essentially a likely sequence of events for the user’s investigation is created from a) the user’s answers, and b) the probabilities those answers imply.

Basically the main concept to be used in the program is that causes and factors have relationships that can be defined by verifiable statistics.

The current method used in the program utilizes exact match data mining. Given partial sequence of event data (question answers), the engine calculates a score for each historical case indicating the number of exact matches between its events and the events contained within the partial data. Positive matches increase the case’s score, negative matches decrease it.

Cases are then ranked according to this score and the events from cases above a certain rank are collected and their frequency in that collection determined. The next question that is asked of the user asks whether or not the event with the highest frequency had occurred. From here, further questions are asked regarding other highly frequent possibilities.

Questioning ceases when a significant amount of variation occurs between the levels of relevancy between the cases displayed.

The next level of computation to be implemented, given more processing power, is more representative of the work done in previous section of this paper (refer to Table 1 and Figure 4). It improves case score calculation by also considering matches between events statistically related to the partial data, and not just the events within the partial data alone. It would effectively increase the rate of relevancy variation significantly.

### 2.2.5 GP1020 prototype – user instructions

User instructions are given on the introduction page of the program. By clicking on the spreadsheet titled “Query Page” the process begins.

On the sheet titled 'Query Page', two distinct sections will appear; the one at the top shows the top 20 cases most likely to resemble, and thus be used as a reference in understanding, the case that the user is currently investigating. The one at the bottom shows the questions that the software would ask depending on the data given.

To begin using the software, answers of the two questions already shown in the bottom must be provided. It is done by clicking on the yellow box next to each question and choosing an appropriate answer from the drop down menu (Figure 5).

Soon after, a third question should appear. Similarly, it is answered by clicking on the yellow cell next to the question and choosing an answer from the drop down menu. Consequently the top 20 most similar cases will be displayed on the top box and a fourth question should appear in the bottom box (Figure 7).
2.2.6 Conclusion

The research results presented in this paper has shown that expert systems methodology is an appropriate approach in analysing the aircraft accident investigation process. Thus the novel GP1020 investigation tool has been a successful demonstration of applying an expert system concept to aircraft accident investigation. The GP1020 has been designed towards giving aircraft accident investigators improved use of forensic data by sifting through a considerable amount of data related to accidents and indicating the most probable accident cause(s).

Results obtained during the testing of the GP1020 program encourage the application of a global expert system, or in other words, increasing the program’s knowledge pool to include historical data from many other sources other than those currently being used. In terms of computer power limitation, either changing code to suit a particular operating system or utilizing hardware with greater computing power would overwhelm this problem.

REFERENCES