doi: 10.5028/jatm.v8i3.609

PRO-ELICERE: A Hazard Analysis Automation Process Applied to Space Systems

Tharcius Augusto Pivetta[1], Glauco da Silva[1,2], Carlos Henrique Netto Lahoz[1,2,3], João Batista Camargo Júnior[4]

ABSTRACT: In the last decades, critical systems have increasingly been developed using computers and software even in space area, where the project approach is usually very conservative. In the projects of rockets, satellites and its facilities, like ground support systems, simulators, among other critical operations for the space mission, it must be applied a hazard analysis. The ELICERE process was created to perform a hazard analysis mainly over computer critical systems, in order to define or evaluate its safety and dependability requirements, strongly based on Hazards and Operability Study and Failure Mode and Effect Analysis techniques. It aims to improve the project design or understand the potential hazards of existing systems improving their functions related to functional or non-functional requirements. Then, the main goal of the ELICERE process is to ensure the safety and dependability goals of a space mission. The process, at the beginning, was created to operate manually in a gradual way. Nowadays, a software tool called PRO-ELICERE was developed, in such a way to facilitate the analysis process and store the results for reuse in another system analysis. To understand how ELICERE works and its tool, a small example of space study case was applied, based on a hypothetical rocket of the Cruzeiro do Sul family, developed by the Instituto de Aeronáutica e Espaço in Brazil.

KEYWORDS: ELICERE, Hazard analysis, Safety, Dependability, Quality attributes, Space systems.

INTRODUCTION

Critical systems or high-integrity systems are those in which a failure can lead to a severe consequence, such as economic, environmental or even human losses. In this context, aerospace systems can be highlighted, such as spacecraft, test facilities and ground equipment. One of the main activities of the safety engineering is performing hazard analysis, which aims to define potential hazards, consequent failures and defects into the system, identifying unplanned behaviours, problems related to exchanges information, wrong procedures execution, among others (Stark *et al.* 2004).

In 2009, a safety and dependability (S&D) analysis process called ELICERE was developed, whose intent was to improve the quality level of critical computer systems (Lahoz 2009). The "elicere" word is derived from infinitive of the Latin verb "*elicio*", which means to elicit, to extract.

In general, the elicitation activity consists of the extraction and identification of the system and software requirements. Requirements can be classified into functional and non-functional. Basically, functional requirements describe the main features of the product under the user's perspective. Non-functional requirements describe various quality factors, or attributes, which affect the functional requirements, such as usability, dependability and safety. Dependability covers other safety-related features, as reliability, availability and maintainability and other factors related to the critical functioning of a product. These particular systems are known as safety critical or high-integrity systems.

1.Departamento de Ciência e Tecnologia Aeroespacial – Instituto Tecnológico de Aeronáutica – São José dos Campos/SP – Brazil. **2.**Departamento de Ciência e Tecnologia Aeroespacial – Instituto de Aeronáutica e Espaço – São José dos Campos/SP – Brazil. **3.**Massachusetts Institute of Technology – Department of Aeronautics and Astronautics – Cambridge/MA – USA. **4.**Universidade de São Paulo – Escola Politécnica – Departamento de Engenharia de Computação e Sistemas Digitais – São Paulo/SP – Brazil.

Author for correspondence: Tharcius Augusto Pivetta | Departamento de Ciência e Tecnologia Aeroespacial – Instituto Tecnológico de Aeronáutica | Praça Marechal Eduardo Gomes, 50 – Vila das Acácias | CEP: 12.228-900 – São José dos Campos/SP – Brazil | Email: tharcius@yahoo.com.br

Received: 02/02/2016 | **Accepted:** 05/24/2016

ELICERE brings together goal-oriented requirements engineering technique — known as ISTAR (Yu 1995) — and features of safety engineering techniques such as Hazards and Operability (HAZOP) and Failure Mode and Effect Analysis (FMEA). The idea is to perform the hazard analysis over the system requirements model, in order to identify potential mitigation actions and improvements in the system. The process uses a questionnaire based on guidewords that are applied under the modelled system elements. The outcomes of the questionnaires are mitigation actions based on a set of quality attributes (factors) related with the technique to assure its integrity.

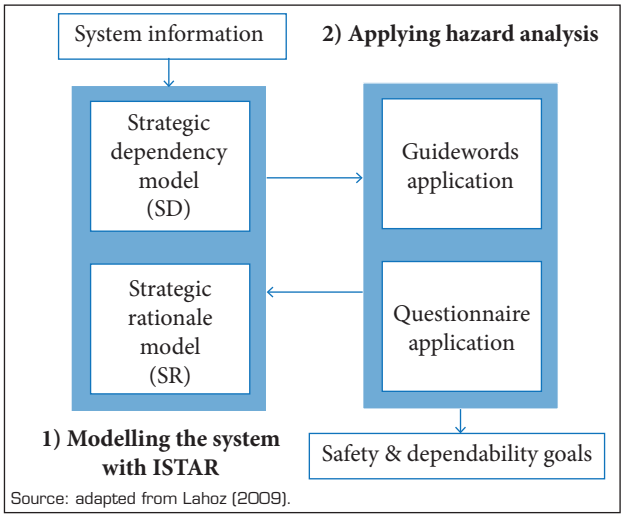
However, the ELICERE generates a large amount of information and requires a computational structure to deal with the relationship between the hazards and the quality attributes that could mitigate it and then to suggest prioritizations of hazard that should be treated. Besides, it is desirable that the results can be recorded as a database of knowledge in order to reuse it for future analysis in other projects, creating a statistical and historical database. To meet these needs, the PRO-ELICERE automated tool has been proposed to improve the process.

This paper introduces the main features of PRO-ELICERE, describing the architecture of the first prototype proposed, the automation of each step, how to run this tool and which results are expected, initially applied for a studied case related to space system.

In "The ELICERE Process" section, the process is described in summary, explaining how the ISTAR modelling language works, the approach for hazard analysis and, finally, the questionnaire submitted to the Analyst. "The PRO-ELICERE tool" section presents the tool that builds upon the ELICERE and shows the main features, mainly in terms of its questionnaire and how to present the mitigation options. "The Case Study Example" section is about a case study based on a hypothetical rocket called V-ALFA. The "Conclusions" section discusses the results and possible improvements for the next version of PRO-ELICERE.

THE ELICERE PROCESS

The ELICERE is an S&D process applied to critical computer systems and was created to support hazard analysis of space systems (Lahoz and Camargo Júnior 2011). ELICERE adopts the ISTAR framework (Yu 1995) for modelling the systems behaviour and the guidewords based on HAZOP and FMEA to extract mitigations provisions and goals related to S&D. In general, this activity comprehends the establishment of the general business and technical goals, an outline description of the problem to be solved and the identification of the system constraints. ELICERE helps to define what the system cannot do, or what the system should do in order to minimize problems related to safety, security, reliability and so on, typically non-functional requirements. In addition, the process improves the product quality, mitigates problems such as ambiguity, risk behaviour, unclearness, besides omission of non-functional requirements. Figure 1 presents the two main activities of ELICERE.



Source: adapted from Lahoz (2009).

Figure 1. The ELICERE's two main activities.

ACTIVITY 1: MODELLING THE SYSTEM WITH ISTAR

The purpose of this activity is to create a system model, through the modelling language called ISTAR. The ISTAR (also called i* or i* framework) is an organizational requirements modelling technique suitable for use in early phase of system design in order to better understand the problem domain. This modelling language describes dependencies among actors through their four basic elements: goal, soft-goal, task and resource. Actors depend on each other for goals to be achieved, tasks to be performed and resources to be furnished. The ISTAR consists of two main modelling components: strategic dependency model (SD), which describes a network of dependency relationships among various actors in an organizational context, and the strategic rationale model (SR), which allows modelling the reasons associated with each actor and their dependencies and provides information about how actors achieve their goals and soft-goals. The ISTAR is used into ELICERE to represent a system in such a way that its hazards

and vulnerabilities become evident, as well as the mission goals, the elements needed by the system to operate, and so on. The constraints should be identified and represented by the actors and its other four modelling elements. ELICERE apply the SD model to obtain a general overview of the relationship between actors, mainly concerned about goals and their soft-goals, tasks and resources. The SR model is used to represent a boundary of an actor, as well a view "in deep" about how each actor works to attend the goals and soft-goals. The main ISTAR features are represented in Table 1 (Yu and Grau 2006).

Figure 2 shows an example of ISTAR model as adapted to ELICERE. It shows that there is a relationship between two actors, where **GS** is the ground system and **OP_GS** is the operator of the ground system. This relationship shows, in a

Table 1. ISTAR features.

Symbol	Definition
Actor	Actor: active entity which leads to goals achievement, exercising their abilities
Goal	Goal: the affirmation or goal that the actor or system must meet or achieve
Soft-Goal	Soft-goal: the affirmation or goal related to non-functional requirements that an actor or system must meet or achieve
Task	Task: it concerns the activity that an actor or system must play in achieving a goal
Resource	Resource: system entity that provides some kind of information, product or service to the actor
	Dependency link: representation of the dependency relationship ("Dependum") between two actors of the system: one is a "Depender" (consumer or depending actor on a dependency relationship) and the other is a "Dependee" (producer or the actor who is depended upon on a dependency relationship). In the graphical notation, the arrowhead points are presented from the "Depender" to the "Dependee"
	Decomposition link: task element which is linked to its component nodes by decomposition links. A task can be decomposed into four types of elements: a subgoal, a subtask, a resource, and/or a soft-goal
	Means-End link: ISTAR element that indicates a relationship between an end and a means for attaining it (the end). The "means" is expressed in the form of a task, since the notion of task embodies how to do something, while the "end" is expressed as a goal. In the graphical notation, the arrowhead points are presented from the means to the end.

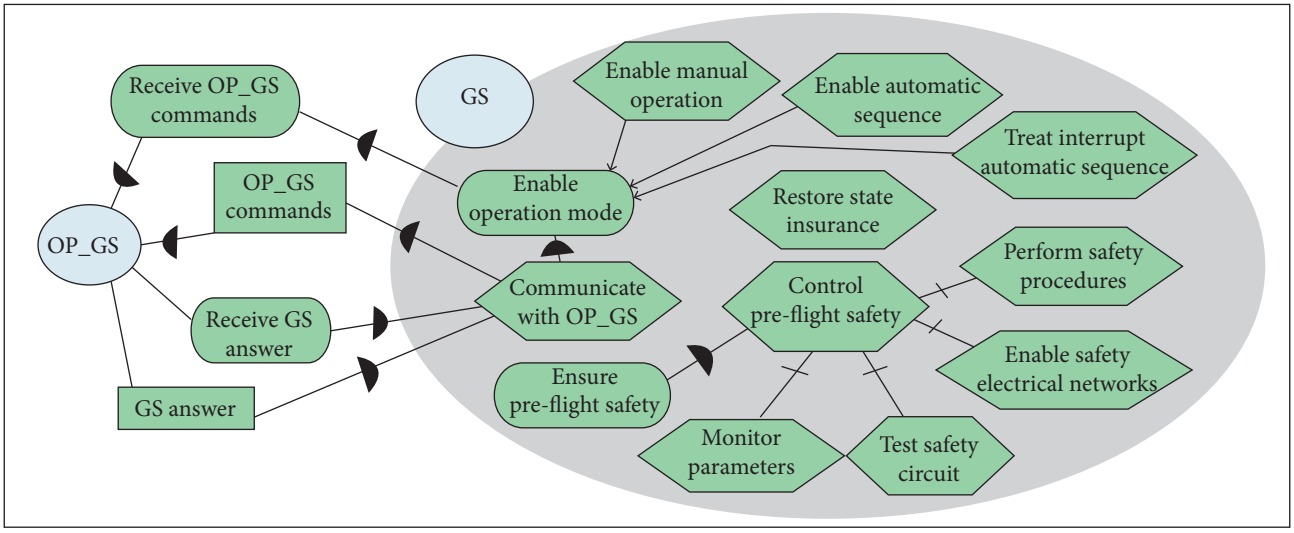


Figure 2. ISTAR SD and SR model example.

simply way, the message exchange between the operator and the system to enable the operational mode during the pre-launch, performing the tests of safety circuit and the change of the modes of operation, in a way to assure that the electrical network is ready to fly.

The relationship of the goals "Receive OP_GS commands" and "Receive GS answer" as well as the resources "OP_GS Commands" and "GS Answer" are part of the SD model, showing the external dependencies between two actors. On the other hand, all elements within the **GS** are part of the SR model and show the interoperability of this actor, *i.e.* how the elements relate internally.

ACTIVITY 2: APPLYING HAZARD ANALYSIS

The HAZOP study was used initially in the 1960s and, despite being based initially on the systematic examination of a chemical engineering plant, it is adopted for other areas and complex software systems (Crawley and Tyler 2015). It is based on guidewords to perform a qualitative analysis to each flow of a system, suggesting deviations operations, such as no, more, less or reverse (Souza 1995). These guidewords aim to identify deviations that may result in potential hazards to the system or function. The FMEA technique was developed in the 1940s as a US military security procedure to determine failures and effects on the system and in its equipment. Later, in the 1960s, the aerospace industry started to use the FMEA during the Apollo program. It aims to classify the flaws in relation to its impact on the mission success and staff safety. To do so, it individually investigates components or system functions, determining how and how often the components of a system can fail, and analyses the effects of this failures. After the analysis execution, it is made a verification of possible ways of reducing the probability of failures or effect analysed (Storey 1996).

The HAZOP and FMEA originated approaches such as Software Hazard Analysis and Resolution in Design (SHARD), Low-level Interaction Safety Analysis (LISA) (Pumfrey 1999) and Software FMEA (Lutz and Woodhouse 1996, 1997) that were used as reference for creating the ELICERE guidewords. While SHARD and LISA are more appropriated for hardware/software deviations, the SHARD technique examines the information flow deviations, initiating with the output system or its functions. LISA examines events of time deviations, such as interruptions, and physical resources used in the system operation. The Software FMEA approach of Lutz and Woodhouse (1997) is used to verify software requirements, specifically to analyse software requirements in space vehicles.

The ELICERE specific guidewords, strongly based on the HAZOP study nodes and FMEA, will be applied in the goal, resource, task and soft-goal of ISTAR components, observing their relationship dependency. The next step of the ELICERE hazard analysis is to apply guidewords over the components of the system modelled with ISTAR. The guidewords are used as a tool for the hazard analysis conduction, aiding the evaluation of the system components, anticipating their possible risks or failures. These guidewords represent the deviation of design intent, taking into consideration mainly the ISTAR components to be used in Programmable Electronic System (PES). Computer systems, communication systems, hardware devices (sensors and actuators), software or even human interface are some important actors considered during this step. This activity will allow obtaining a characterization of the hazard evidences that should be explored and prioritize the more critical components (resource, tasks, actors or even goals) that should be analysed. Then, it is necessary to fill each questionnaire chosen and the result in a set of soft-goals for the system.

Their settings are made for each type of ISTAR element and have a generic structure, which can be used to systems in a standard way, but can evolve and has new features for specific systems. Table 2 describes the guidewords used in the context of the ELICERE process.

Table 2. ELICERE generic guidewords.

Element	Guideword
Soft-goal	Soft-goal is not achieved
	Incorrect soft-goal
	Additional soft-goal
	Soft-goal out of time/order
Goal	Goal is not achieved
	Incorrect goal
	Additional goal
	Goal out of time/order
Task	Abnormal task termination
	Task omission
	Task is incorrect
	Task is out of time/order
Resource	Absent resource
	Incorrect resource
	Additional resource
	Resource out of time/order

Source: Lahoz (2009).

To accomplish this activity, it is necessary to apply a questionnaire for each guideword to conduct the hazard analysis that could result in a set of soft-goals for the system. This is an easy way to discover goals related to non-functional requirements (soft-goals) that allow mitigating the system hazards. More details about ELICERE guidewords and its questionnaire can be found in Lahoz (2009).

THE PRO-ELICERE TOOL

The PRO-ELICERE is a software tool that aims gathering the hazard analysis and its requirements for critical computer system from a system modelling perspective. Based on the ELICERE process, this new approach proposes an intelligent layer that allows performing the analysis, as possible, automatically. PRO-ELICERE is designed in several steps, as shown in Fig. 3.

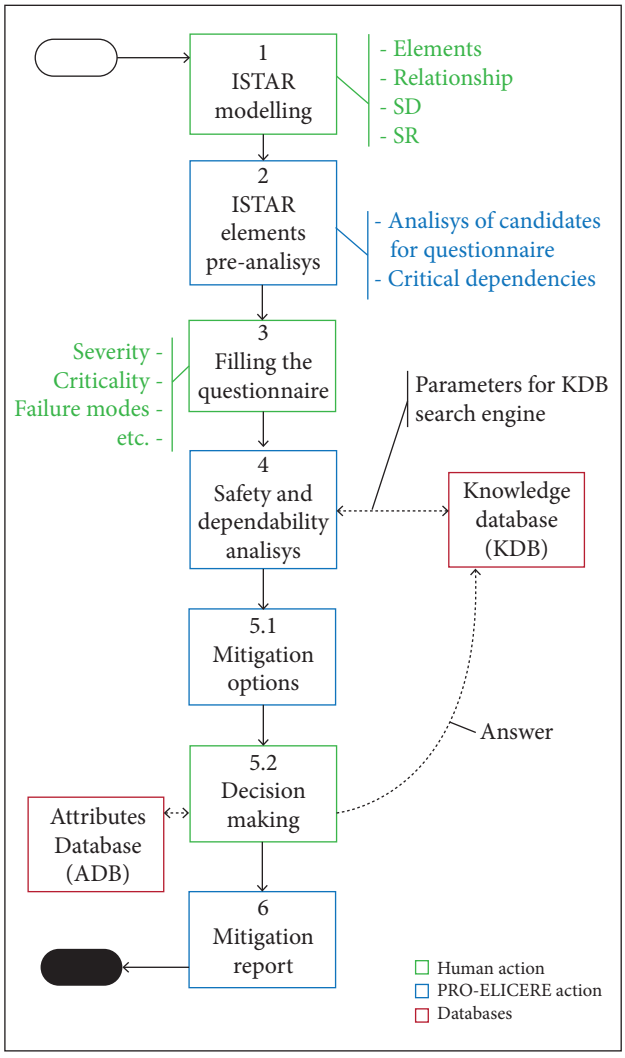


Figure 3. The PRO-ELICERE steps.

The PRO-ELICERE starts working with the ISTAR system models (inputs); after that, it creates the questionnaires based on guidewords related to ISTAR elements. Finally, it offers to the Analyst a menu of potential techniques and metrics (related to quality attributes) that could mitigate or even eliminate the hazards identified in the questionnaires. This set of techniques and metrics is an important contribution to the hazard analysis. Additionally, the knowledge database — capable of keeping the information of previous projects — become an important contribution to the Analyst's decision about which mitigation actions are more appropriated for the hazard.

Initially, it is made the system modelling (1), following the concepts of ISTAR. Thus, they shall be entered into the system (through manual entry or import from OpenOme, an open source tool) and separated by projects, identifying all the elements that make up the system and each of their relationships. After inserting the model in the system (2), a pre-analysis will be performed to identify candidates for the questionnaires. This filter is used to prioritize the elements, using criteria such as number of related elements or criticality in operation. After reviewing which questionnaires should be filled, they will be generated under the command of the Analyst.

With the questionnaires properly generated (3), the Analyst fills with the information that defines the task, resource, goal or soft-goal, to identify possible failure modes, severity, criticality, among other important factors. The automated questionnaire assists the analysis and helps to optimize the time.

The S&D analysis (4), an activity performed after the questionnaire, works with the data and recommendations (automatizing as much as possible) of the mitigation techniques to mitigate the hazards presented. It uses queries based on the attributes database (ADB) and the previous answered questionnaires. These answers are registered in the knowledge database (KDB) to find the best recommendations in futures hazard analysis queries. Originally, the ADB was created through a survey on the quality attributes of literature and techniques and methods to ensure these attributes. For another side, the KDB contains a repository of the mitigation suggestions coming from other analysis, of the same project or not.

At the end of the S&D analysis (5.1), the recommendations will be displayed and sorted by the degree of confidence or through the same parameters used in other projects. Despite the automation, the PRO-ELICERE does not choose the technique for hazard analysis. The Analyst (human inference) is still necessary (5.2), because he knows exactly the specificities and the scope of the

problem and its variables; also he is free to consult others involved in the system. The PRO-ELICERE gives some possible answers to the Analyst determine the best way to mitigate the problem posed in the questionnaire. After that, the system will store that answer to the knowledge database that can be based on one PRO-ELICERE option chosen or an own free text of the Analyst. This information recorded in the knowledge database can be used in a future analysis of other systems. Finally, at the end of all questionnaires and answers (6), the PRO-ELICERE issues mitigation reports by several parameters, such as elements, models, guidewords, relationships or criticality, for example. This report can be used as a document to record the analysis and to suggest what actions can be taken to certain elements of the project.

Working with ISTAR Modelling

The PRO-ELICERE has two options to introduce the ISTAR models: manually, inserting each element of the system and their relationships, or importing the data from the OOD format (a XML format file), chosen from the OpenOme (Yu and Horkoff 2013).

The system model elements should include the Dependency Strengths, such as Open, Committed and Critical. This type of dependency strengths helps the PRO-ELICERE to prioritize the questionnaires to be generated.

When drawing or manually including the elements of the system, the Analyst should define the relationships between them, informing dependencies, task decomposition and other kinds of features available in the ISTAR modelling. The example in Fig. 4 shows a model definition used in PRO-ELICERE.

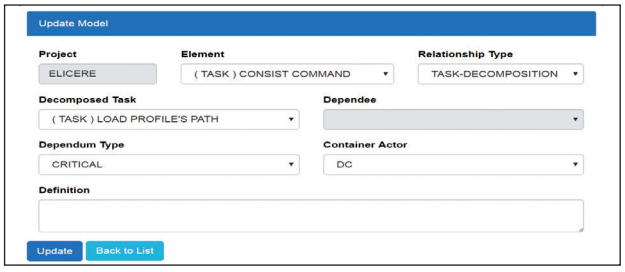


Figure 4. A PRO-ELICERE model definition.

In Fig. 4, **Relationship Type** is the field that describes the relationship between the element analysed and the system (another actor, resource, tasks etc.). This relationship could be **Task Decomposition**, **Means-End** and **Contribution Link.** Element is the element type and name analysed, such as task, resource, and goal. The **Decomposed Task** represents the task that will be fulfilled with the task analysed (in this case, the "Consist Command"). **Dependency type** shows if the task is critical. The **Container Actor** is the actor who contains this task in the S&D model. The **Actor** is the field that contains the actor ("Dependee" or "Depender") in charge of the task or resource under analysis. Lastly, the **Definition** can be filled with additional information about the ISTAR model.

ISTAR Elements Pre-Analysis

The project elements description (and their relationships with other elements) is important to understand how they interact and to open the possibility of analysing the most critical items, using the dependency criteria, for example.

The PRO-ELICERE presents a set of guideword for each type of element helps to create the hazard questionnaires. With the models stored in the system, the Analyst will be able to identify which are the elements, and the related guidewords, that will have a critical importance. Figure 5 shows an example of the list of the potential questionnaires of an element.

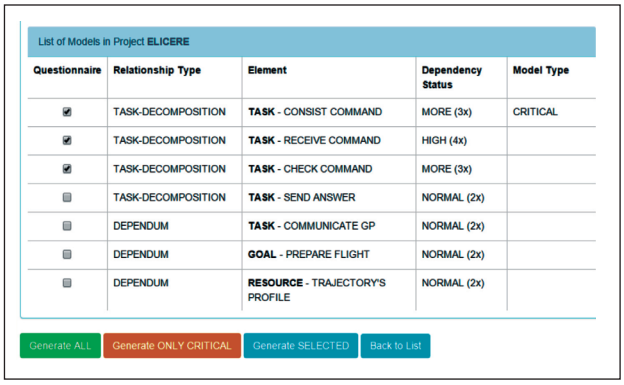


Figure 5. Example of the list of potential questionnaire generation.

One of the sorting options for choosing which questionnaires should be created is observing how many relationships (Dependency Status field) this element has in the whole system model.

For this, all design elements and relationships are seen, and the Analyst will check the items with more complexity, criticality or dependence. The PRO-ELICERE has the option to generate all questionnaires, to select only a few, or to choose just those suggested by the tool. For each element, the questionnaires will be generated according to 4 generic guidewords classified by ISTAR.

The Questionnaire Automation

The questionnaire proposed in ELICERE methodology had specific fields, but they were open; therefore, they did not have

a fill pattern and would hardly be capable to be reused in other projects, like PRO-ELICERE aims. To avoid this, the PRO-ELICERE uses predetermined options from menus, aiming to enhance the Analyst's task, becoming the answer more direct and standardized. The example of the PRO-ELICERE questionnaire is presented in Fig. 6.

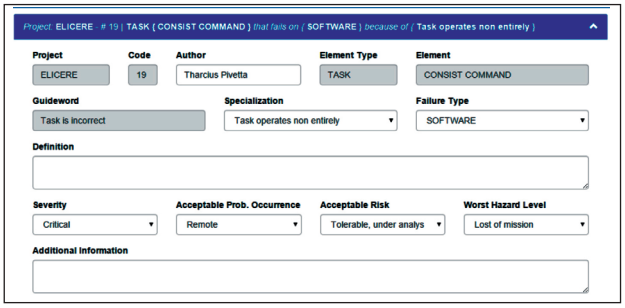


Figure 6. A PRO-ELICERE questionnaire example.

There are several fields to identify the hazard, but the most important are described as follows:

- Element Type: (1) resource, (2) task, (3) goal, and (4) soft-goal.
- Specialization: specific level of guideword to identify more objectively the deviation, such as "Saturated data", "Task occurs very early", "It was not provided the resource", "Sensor failure to send data", among others.
- Failure type: defines if the failure is (1) human, (2) software, (3) hardware or (4) environmental.
- Acceptable risk: defines the acceptable risk for failure. It is given by combining the Acceptable Probability of Occurrence and the Severity, being represented by the options: (1) intolerable, (2) undesirable, (3) tolerable under analysis, and (4) acceptable.

Safety and Dependability Analysis

To identify technical recommendations for the completed questionnaire, 2 possible approaches were applied: identify attributes and techniques directly from the ADB or through parameters from the questionnaire related to the KDB. Upon finished the questionnaire, the tool will suggest the recommendations, presented as a menu of options, from the ADB attributes. The option(s) will be chosen manually by the Analyst and will be recorded in the KDB. The PRO-ELICERE will create a database relationship table that matches some fields of the questionnaire (a hazard combination) such as "Element + Guideword + Specialization + Failure Type" related to the mitigation techniques chosen by the Analyst. This relationship table will help the tool to present future recommendations, in case of some hazard combination occurs. It is considered that the hazard analysis is interactive (many interactions could happen) until finish the questionnaire with the final consideration about how to mitigate the hazard. The feedback from other projects about recommendations of the same kind of problem should be considered. These recommendations will be stored in a KDB, as well as the parameters filled of the hazard analysis questionnaire. With this, the tool can combine options based on knowledge, with techniques that can mitigate the hazard analysed, ensuring recommendations that are more reliable.

To illustrate how the questionnaire works, the PRO-ELICERE will analyse a failure related to the data transmission of an OP_GS and the GS (as illustrated in the model of Fig. 2). The resource "OP_GS Commands" and the task "Receive OP_GS commands" are related with 2 actors, OP_GS and GS, respectively. In the specific case of the **Element Type** "Resource", the questionnaire field **Guideword** is filled with "Resource Missing"; the **Specialization** of the hazard is "Loss or lack of message" for a **Failure Type** "Hardware". The tool will check in the KDB, in previous projects, if this similar scenario (communication problems) exists. Then, the tool could suggest a mitigation technique related to other questionnaires like "execute a Ping/Echo command", which recommends sending a standard signal to the sensor and waiting for the correct answer to check if the middle of transmission works well (Bass *et al.* 2003), based, for example, on the **Specialization** and **Element Type** fields. If the Analyst did not agree with this recommendation, the PRO-ELICERE can make other combinations to find other possible techniques disregarding parameters like **Specialization** (loss or lack of message). It can find 2 options, the first being already informed ("execute a Ping/Echo command") and a second, that is more generic, such as "create a passive redundancy", which is installing more than one sensor to read the same message and assure the availability of the data (Bass *et al.* 2003). As mentioned, the analysis is in charge of the final decision, although in many cases these options may be the best and the more reasonable solutions for the problem.

Mitigation Options versus Human Inference

The main goal of the PRO-ELICERE is to find mitigation recommendations in the databases and display the results to the Analyst. The Analyst will have the power to choose the options presented from the data from KDB. If the mitigation

that fits the hazard analysis is not found, the Analyst can also manually choose among all techniques registered in the ABD or write his own mitigation action, creating a new input in the ABD and KDB.

Figure 7 shows a result that comes from the KDB for the questionnaire of the guideword "Task Incorrect".

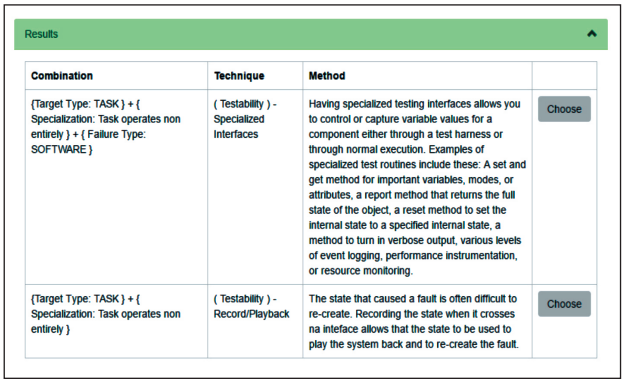


Figure 7. A PRO-ELICERE mitigation options example.

It is important to emphasize that the Analyst always takes the last decision about the best mitigation action. The PRO-ELICERE may suggest the recommendations through the criteria presented, but the final decision will always be of the Analyst, because he has the expertise about the project and the responsibility to determine the best solution to the problem presented.

Mitigation Report

Upon completion of the questionnaires, as well as the choice of recommendations for the presented analysis, the tool will allow the management of some reports, which can be used as formal documents of the hazard analysis. Among them, the "Questionnaires Report", which presents each performed analysis with details of hazards, criticality, failure mode and severity, as well as the recommendations of mitigation actions, may be cited. These reports will be described and exemplified in the next section.

THE CASE STUDY EXAMPLE

According to the Programa Nacional de Atividades Espaciais (PNAE, 2012) of the Agência Espacial Brasileira (AEB), a new launch vehicle program, called Cruzeiro do Sul, was established. The main goal is a continuation of the development carried out for the VLS-1 (Brazilian Satellite Launch Vehicle) — a medium-size solid propellant rocket motor, which is comprised of five new vehicles to be developed and qualified by the Instituto de Aeronáutica e Espaço (IAE). The first vehicle of the family is a 3-stage launch with an expected capability of transporting up to 400 kg payload into low inclination orbits of 400 km altitude (Moraes *et al.* 2006; Villas Boas 2006).

To understand how to perform the ELICERE hazard analysis and its PRO-ELICERE tool, some features of the VLS-1 were selected, related to the on-board to ground communication functionalities, potentially reused in the first prototype of the Cruzeiro do Sul family, denominated for this study V-ALFA.

THE V-ALFA MODEL EXAMPLE

For a better understanding of the model in ELICERE, it was created in the OpenOme tool a macro view of the goals models used by the actors/agents. The goal in ISTAR is an objective or function that the system should reach. To do so, it must perform tasks and use resources among actors and agents and even accomplish intermediate goals. These views are very important to be able to view which actors, goals, tasks and resources are critical for any project, noting their interdependencies. With the two ISTAR models created for the V-ALFA, it can be seen that one of the actors with the highest number of relationships is the Digital Controller (**DC**), so their tasks are critical for the V-ALFA meet the space mission. The role of the PRO-ELICERE is helping to identify soft-goals in order to assure this mission.

To illustrate the development analysis, a goal was chosen to show the model questionnaire, mitigation suggestion and finally the analysis report. Many goals can perform many tasks using many resources. It is advisable to check each goal separately, but after the analysis it must be validated the inter-relationship with the other goals and actors.

When the vehicle is in the launch pad, many activities are performed, such as testing the **DC** communication with the actuators and sensors, testing the pyrotechnic valves, checking of the destruction system, and loading the "profile of the trajectory", which contains the parameters relating to the V-ALFA flight profile. The file with the trajectory profile should be loaded correctly through the communication link between the Ground Support equipment (**GP**) and the on-board Digital Controller (**CD**), in the pre-flight phase. If this profile was loaded incorrectly, the goal "Prepare to Flight" will be not accomplished, and the mission probably will fail. This ISTAR model can be represented in Fig. 8.

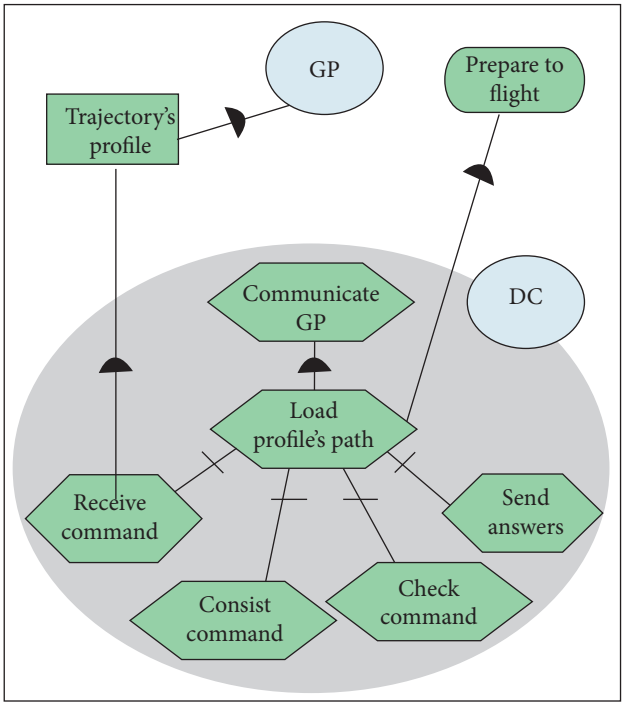


Figure 8. ISTAR model of the "Prepare to flight" goal.

The main elements of the specific "Prepare to flight" ISTAR model are:

- **GP** actor – ground support equipment: system responsible for the preparation to launch flight. It commands the tests and verifies the on-board networks, sending the trajectory profile with the parameters and inputs to the control algorithms.
- **DC** actor – digital controller: equipment responsible to communicate with the ground system. It performs all pre-flight tests and loads the trajectory profile to perform the control of the vehicle during the first stages of flight.
- **Trajectory's Profile** resource –contains the parameters for the V-ALFA flight profile. This resource must be loaded properly through the means of communication between the GP and DC, in the pre-flight phase. If any of the parameter is wrong, the goal will not be met, because the DC will not load the data with the specified values. There will be the need to carry this resource again with the corrected values.
- **Load Profile's Path** — this task is in charge of receiving data relating to the proposed flight trajectory. If the task sends wrong or absent values from the GP, the target will be missed and will create an incorrect table profile path leading to loss of mission. This task is decomposed into 4 sub-tasks: "Communicate with GP", "Receive Command", "Consist Command", "Check Command" and "Send Answer".

THE V-ALFA QUESTIONNAIRE EXAMPLE

After the PRO-ELICERE generates all the possible questionnaires, the Analyst can select all or only some of them, following some criteria, such as the high criticality items, number of dependencies, level of risk etc. Figure 9 presents a questionnaire completed, analysing the "Receive Command" sub-task, decomposed from the "Load Profile's Path" task.

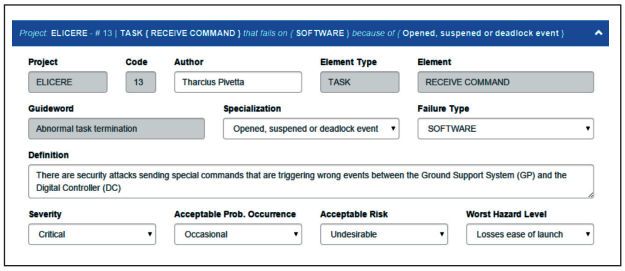


Figure 9. Example of questionnaire filled.

By performing all the hazard analysis, the system will allow the Analyst to extract certain system reports, such as the list of all filled questionnaires or the list of interaction (each other) of the elements in the system. Figure 10 presents an example of one report that informs the number of completed questionnaires, their risk analysis, and details about the mitigation action, such as the technique related, the quality attribute associated and the explanation about the method to achieve it.

In addition, it is shown a summary of the status of each project (in this context, V-ALFA), analysing quantitatively the questionnaires answered, such as:

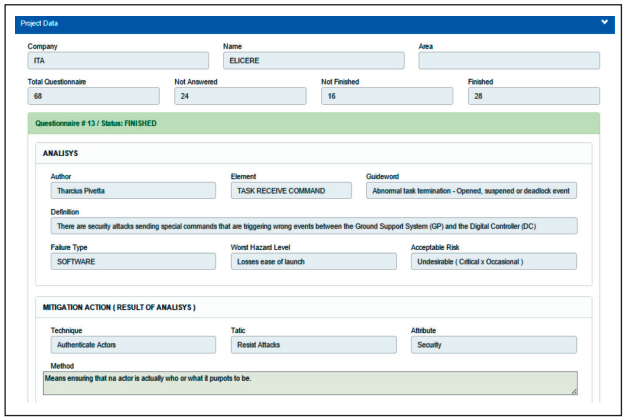


Figure 10. Questionnaire report example.

- **Total Questionnaire**: total of questionnaires generated by the elements in combination with de guidewords, selected in the step of Questionnaire Generation (Fig. 5).
- **Not Answered**: quantity of questionnaires created, but with information not filled like risk, probability occurrence, and other vital information for the understanding of the hazard.
- **Not Finished**: questionnaires fully answered, but without the mitigation technique chosen by the Analyst. It means the hazard needs one more step to finish.
- **Finished**: questionnaires fully answered and with proper mitigation technique chosen.

Figure 9 presents a report extracted from the Questionnaire #13 – "Receive command" task, which picked the mitigation action "Authenticate Actors". This technique suggests creating means to ensure that the actor access is authorized (Bass *et al.* 2003).

CONCLUSIONS

This paper described the main features of the automation of the ELICERE process, a methodology of hazard analysis through the system modelling, and its guidewords analysis. Due to the large amount of information to be manipulated and aiming to reuse the mitigation actions, the software tool called PRO-ELICERE was created. First, the system is described through the ISTAR modelling language. Then, the analysis of its elements is performed using guidewords, such as HAZOP technique and then a questionnaire is filled with the hazard information as proposed by the FMEA.

A database repository (ADB) has been created with a well-known set of quality attributes and related techniques identified in the literature, as well as another database (KDB) with the results of the previous hazard analysis and its mitigation actions. The first one was created using several literature references (Bass *et al.* 2003; Romani *et al.* 2010; Lahoz *et al.* 2012) and the other was created to record the results of the hazard analysis performed with the PRO-ELICERE. These quality attributes cover availability, interoperability, modifiability, performance, security, testability, usability, modularity, traceability, simplicity and robustness. Eighty-seven techniques are suggested to assure that elements under analysis meet this quality goal. For example, if the hazard analysis identified that an element of the system needs to improve the Availability, the PRO-ELICERE could suggest applying techniques like "make reconfiguration", "include passive redundancy", "perform self-test" or "send Ping/echo" etc. If the element analysed needs to improve its Security, for instance, "detect Intrusion", "verify message integrity" and "check authorization login" are suggested. Another possible analysis performed by PRO-ELICERE is about problems related to development, when a guideword questionnaire detects that one of the system's element needs to improve its Modularity. The tool could suggest "Split the functionalities into small components" and so on.

The main function of the KDB is to create a history of mitigation actions proposed by the previous projects in order to help the new one under analysis with the best solution. The idea is to apply knowledge discovery techniques, using ontologies, generally presented in the language Web Ontology Language/Description Logic (OWL-DL; Horrocks *et al.* 2003) to infer knowledge about safety and dependability issues based the on-going PRO-ELICERE analysis, with the set of quality attributes and its techniques previously chosen by the last projects.

Finally, in the study case presented as a way to understand how PRO-ELICERE works, 83 ISTAR elements are created, such as 13 actors, 3 agents, 13 goals, 14 resources and 40 tasks. A total of 60 **Dependency** relationships (producer *versus* consumer), 27 **Decomposed-Task** and 4 **Means-End** relationships were produced, generating a total of 68 questionnaires. The next research step involves the extraction of more data from the V-ALFA, improving its ISTAR model, performing more hazard analysis and creating a proper ontology for the PRO-ELICERE.

The main benefits of the PRO-ELICERE are the organization in database of the huge amount of data obtained with the system models, with the guidewords questionnaires and with the mitigation options that come from ADB and KDB. Also, the PRO-ELICERE's database are capable of presenting, in terms of screen views and printable reports, all the information handled, such as the list of ISTAR actors related to their goals, tasks and resources necessary to perform a system goal, number of dependencies of an actor (extracted from SD or SR diagrams), and so on. The KDB is a strategic contribution for the hazard analysis due to its capacity of storing the previous hazards analysis, then presenting a potential solution for mitigation.

REFERENCES

Bass L, Clements P, Kazman R (2003) Software architecture in practice. Upper Saddle River: Addison-Wesley.

Crawley F, Tyler B (2015) HAZOP: guide to best practice. 3rd ed. Waltham: Elsevier. Chapter 1, Introduction; p. 1-3.

Horrocks I, Patel-Schneider PF, van Harmelen F (2003) From shiq and rdf to owl: the making of a Web Ontology Language. Web Semant Sci Serv Agents World Wide Web 1(1):7-26. doi: 10.1016/j.websem.2003.07.001

Lahoz CHN (2009) ELICERE — o processo de elicitação de metas de dependabilidade para sistemas computacionais críticos: estudo de caso aplicado à area espacial (PhD Thesis). São Paulo: Universidade de São Paulo. In portuguese.

Lahoz CHN, Camargo Júnior JB (2011) Introducing ELICERE guidewords for critical computer systems. Proceedings of the IEEE Fourth International Conference on Software Testing, Verification and Validation (ICST); Berlin, Germany.

Lahoz CHN, Romani MAS, Yano ET (2012) Dependability attributes for space computer systems: quality factors approach. Proceedings of the Space Operations Conference (SpaceOps); Stockholm, Sweden.

Lutz RR, Woodhouse RM (1996) Experience report: contributions of SFMEA to requirements analysis. Proceedings of the 2nd IEEE International Conference on Requirements Engineering (ICRE); Colorado Springs, USA.

Lutz RR, Woodhouse RM (1997) Requirements analysis using forward and backward search. Ann Software Eng 3:459-475.

Moraes Jr P, Carrijo DS, Garcia A, Costa LEL, Oliveira UC, Santana Jr A, Villas Boas DJF, Yamamoto MK (2006) An overview of the Brazilian launch vehicle program Cruzeiro do Sul. Proceedings of the 57th International Astronautical Congress, International Astronautical Congress (IAF); Valencia, Spain.

PNAE (2012). Programa Nacional de Atividades Espaciais: PNAE: 2012-2021 / Agência Espacial Brasileira. Brasília: Ministério da Ciência, Tecnologia e Inovação, Agência Espacial Brasileira. p. 36.

Pumfrey DJ (1999) The principled design of computer system safety analyses (PhD thesis). York: University of York.

Romani MAS, Lahoz CHN, Yano ET (2010) Identifying dependability requirements for space software systems. J Aerosp Technol Manag 2(3):287-300. doi: 10.5028/jatm.2010.02037810

Souza EA (1995) O treinamento industrial e a gerência de riscos – uma proposta de instrução programada (Master's thesis). Florianópolis: Universidade Federal de Santa Catarina. In portuguese.

Stark J, Swinerd G, Tatnall A (2004) Introduction. In: Fortescue P, Swinerd G, Stark J, editors. Spacecraft systems engineering. 3rd ed. Chichester: Wiley.

Storey N (1996) Safety-critical computer systems. Upper Saddle River: Addison-Wesley.

Villas Boas DJF (2006) O contexto histórico das atividades espaciais e a tecnologia dos foguetes. In: Ministério da Educação, Secretaria de Educação a Distância. Da Terra ao espaço: tecnologia e meio ambiente na sala de aula. Boletim 06/2006. p. 26-37; [accessed 2016 Jan 28]. http://cdnbi.tvescola.org.br/resources/VMSResources/contents/document/publicationsSeries/1426100949736.pdf

Yu ES (1995) Modelling strategic relationship for process reengineering (PhD thesis). Toronto: University of Toronto.

Yu ES, Grau G (2006) ISTAR quick guide; [accessed 2016 Jan 02]. http://istar.rwth-aachen.de

Yu ES, Horkoff J (2013) OpenOme beta; [accessed 2015 Dec 10]. http://sourceforge.net/projects/openome