



Safety Management Method in Complex System Engineering

Romaric Guillerm, Hamid Demmou

► To cite this version:

Romaric Guillerm, Hamid Demmou. Safety Management Method in Complex System Engineering. 8th Annual IEEE International Systems Conference (SysCon 2014), Mar 2014, Ottawa, Canada. 7p. hal-01828535

HAL Id: hal-01828535

<https://laas.hal.science/hal-01828535>

Submitted on 3 Jul 2018

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Safety Management Method in Complex System Engineering

Romaric Guillermin, Hamid Demmou

CNRS, LAAS, 7 avenue du Colonel Roche, F-31400, Toulouse, France

Université de Toulouse, UPS, LAAS, F-31400, Toulouse, France

guillermin@laas.fr ; demmou@laas.fr

Abstract—The main objective of System Engineering is the successful development of complex system. It is based on the application of iterative and recursive processes on each phase or step of the system development. One critical process is the requirement management, particularly when it deals with the safety requirements. These one are non-functional requirements and are related to emergent properties, which come from the integration of the different system components. They must be identified as soon as possible, because they are guards to validate or not the system, which can require changes in system architecture. Moreover, they are formulated at system level and need to be derived at sub-system level.

The objective of this paper is to propose a safety management method based on well-known safety methods, in order to organize the different tasks to make the system safe. The method focuses mainly on the definition of the system safety requirements following risk and hazard analysis, and also on their derivation according to a top-down approach. It is based on the well-known Failure Mode, Effects, and Criticality Analysis (FMECA) and the use of Fault Trees and Event Trees.

Keywords—Safety requirement; Requirement engineering; Complex system

I. INTRODUCTION

Modern systems are increasingly complex [1]. Indeed, they integrate more and more different technologies, offer more functions, and have complex interactions between their components. The processes and the design methods must evolve to reflect this growing complexity [2], [3]. In particular, for our purposes, the management of properties such as reliability or security [4] must evolve accordingly, to ensure and enable the necessary level of confidence [5]. For an effective consideration of safety in the design process, it is necessary to consider safety in overall studies by using the engineering system processes. The safety properties must be defined globally; that is to say elicited [6]. Once these safety properties are identified, they must be derived locally to be actually realized by the system. The local properties associated with subsystems must be satisfied to ensure the global properties, reaching issues of traceability [7], [8] and requirements engineering [9].

Requirements Engineering (RE) is one of the System Engineering (SE) processes. RE is a crucial process within the

development of complex system. Safety requirements are classified as non-functional requirements and are related to emergent system properties. They cannot be attributed to a single system component. Furthermore, non-functional requirements are fundamental to determine the success of a system. Two activities are defined in RE. The first one concerns requirements development including the processes of elicitation, documentation, analysis and validation of requirements. The second one concerns requirement management which includes the processes of maintainability management, changes management and requirements traceability.

The work presented in this paper concerns a part of our approach for the integration of safety in system engineering processes [10]. It is an improvement and extension of the method presented in [11], that was inspired from [12] with an engineering process and requirements point of view. The originality of the approach is to take into account the safety requirements in system engineering process to facilitate traceability of these requirements throughout the life cycle of the system. It concerns the two activities of RE: the development and the management activities. The paper presents a method that allows to define, derive and derive system safety requirements, with the combination of several FMECA (Failure Mode, Effects, and Criticality Analysis) [13], Fault Trees analysis [14] and Event Trees analysis [15].

This paper contains five parts. The second one presents the system engineering framework of the method. The third one exposes the method for safety requirement definition and derivation, with its different steps. The fourth one presents briefly the result of one first case study. Finally, the last section concludes the paper and presents some perspectives.

II. CONTEXT

In this part, the context of our work is exposed. The first section presents the System Engineering notion. Then, the SE standard that we adopt is presented with its useful concept of building block that split the design into different system layers. To finish, a focus is done on the safety requirement management.

A. System Engineering

System Engineering (SE) is an interdisciplinary approach, whose objective is to assist the development of new systems. It

contains collaborative and interdisciplinary processes of resolution of problems, supporting knowledge, methods and techniques resulting from the sciences and experiment to define a system, which satisfies an identified need, and is acceptable for the environment, while seeking to balance the total economy of the solution, on all the aspects of the problem in all the phases of the development and the life of the system. SE concepts are adequate specifically for complex problems [16].

SE is the application of scientific and engineering efforts to:

- Transform an operational need into a description of system performance parameters and a system configuration, through an iterative process of definition, synthesis, analysis, design, test and evaluation.
- Integrate reliability, safety, maintainability, expandability, survivability, human engineering and other factors into the total engineering effort to meet cost, schedule, supportability and technical performance objectives.

SE is the global framework of the approach proposed in this paper.

B. EIA-632

A standard currently used in the industrial and military fields is the EIA-632 standard [17]. Our work is also based on it.

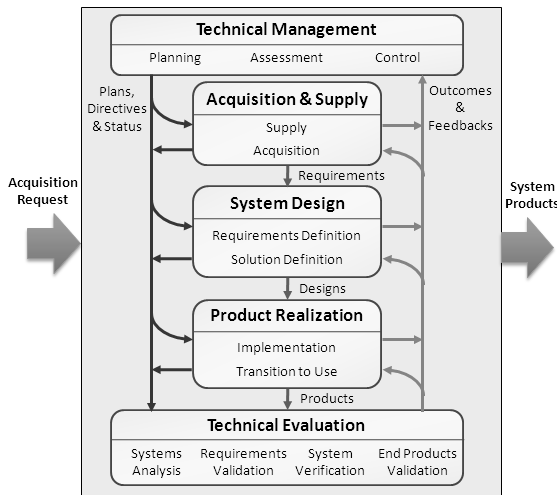


Fig. 1. EIA-632 Standard System Engineering Processes

Briefly, this standard covers the product life cycle from the needs capture to the transfer to the user. It is constituted by 13 processes grouped into 5 sets (see Figure 1):

1. Technical management processes (three processes): these processes monitor the whole process ranging from the initial idea to building a system until the delivery of the system.
2. Acquisition and supply processes (two processes): these processes ensure the supply and acquisition (and are very close to logistics).
3. System design processes (two processes): these processes deal with the elicitation and the acquisition of

requirements and their modeling, the definition of the logical design and its physical solution.

4. Product realization processes (two processes): these processes deal with the implementation issues of the system design and its use.
5. Technical evaluation processes (four processes): these processes deal with verification, validation and testing issues.

C. Building Block Concept

The EIA-632 standard adopts an original and interesting system decomposition based on the concept of "building block". A building block is the association between one (or several) final product and a set of enabling products, as shown in Figure 2.

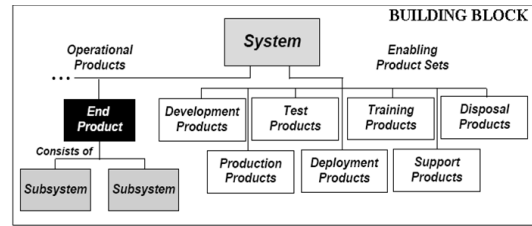


Fig. 2. One Building Block

In fact, the system is seen as a hierarchy of building blocks. The solutions defined in the upper layer (level) blocks, described by a set of specified requirements, are allocated as input requirements for the lower layer blocks (see Figure 3). Finally, the building block decomposition is stopped when blocks correspond to on-the-shelf components or when their realization can be subtracted. With this description, we identified the need of deriving the safety requirements through the hierarchical decomposition.

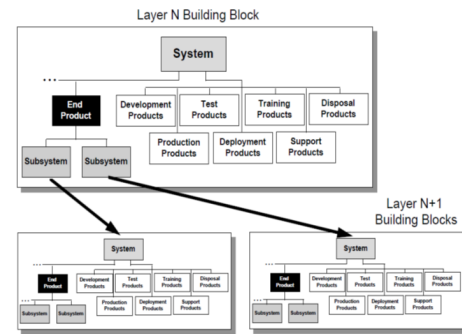


Fig. 3. Multilayer Building Block

D. Safety Requirement Management

To situate the position of our method for deriving safety requirements, the Figure 4 gives an overview of the involved EIA-632 system engineering processes.

Among the different possible sources of safety requirement we can find the requirement provided by some dependability analysis as shown in the Figure 4. In this paper we consider this source of requirements. The proposed approach is used to define and derive safety requirement with different safety analysis.

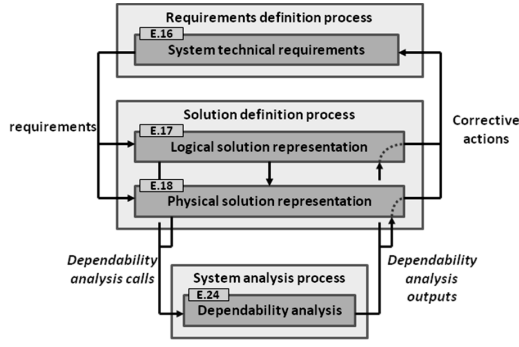


Fig. 4. Dependability analysis as a source of requirements

III. THE SAFETY MANAGEMENT METHOD

In this second part, the safety management method is presented. First, an overview of the method is given, followed by an explanation about the different kinds of safety requirements that are taken into account in the current version of the method. Afterwards, the 9 steps of the method are explained in details.

A. Overview

The method assumes that a complex system is composed of some subsystems (the principle of Building Block of the EIA-632 standard). It combines FMECA, fault trees and event trees, and has the objective to define all safety requirements at system level and to derive them locally at subsystems level with a goal of traceability. The Figure 5 summarizes the process associated to the method and illustrates how the different steps are integrated together.

B. Classification of the considered Safety Requirements

The method enables to identify and deals with several kinds of *safety requirements*. We classify these requirements into subcategories, which are:

- *Reliability requirement*, that claims a quantitative objective in term of reliability properties.
- *Architectural requirement*, that defines an architectural design to deal with safety (like redundancies).

- *Active functional security requirement*, that is related to additional security equipment (protective barrier) that can participate to reduce the probability of an accident.
- *Passive functional security requirement*, that is related to an additional protective or mitigation equipment that can reduce the severity of an accident.

C. Step 1: Risks Identification

The first step is to identify and classify all the system risks. These can be human actions, external failures, internal failures (those of the system) or environmental conditions. The classification must be done in two groups: the risks representing internal failure modes and the other risks.

D. Step 2: System Failure Modes Analysis

The second step is to begin the analysis of risks that correspond to system failure modes. The recommended method is the FMECA [13], that is a technique used to identify, prioritize, and eliminate potential failures from a system, a design or a process. Concretely, this step is to complete few columns of the FMECA table (others than severity, probability, criticality and corrective action) (see Table I). For each system function, we identify failure modes, causes of these modes and effects on the system (possibly depending on the phase, state or mode). For the identification of failure modes, lists of generic modes have been defined in some standards like CEI 60812 [18]. The effects are here the potential accidents.

TABLE I. FMECA TABLE

Function	Failure Mode (FM)	Probability (of the FM)	Detectability (of the FM)	Causes	Effect (Accident)	Condition	Probability (of the Effect)	Severity (of the Effect)	Detectability (of the Effect)	Criticality (of the Effect)	Corrective Actions	Acceptable
...	...	...	...	...	...	...	...	...	...	...	...	YES/NO

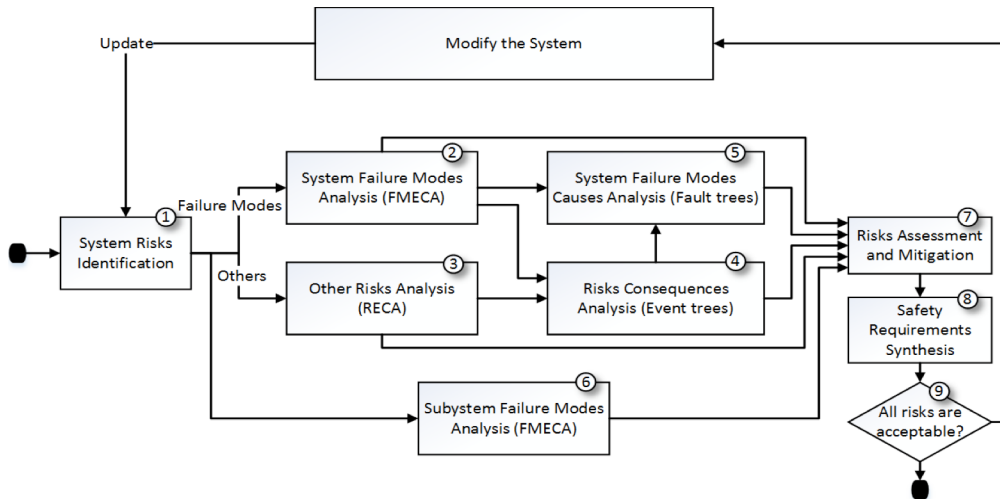


Fig. 5. Overview of the safety management method

In fact, we also propose some changes in the classical FMECA to clarify the method, (see Table I). A distinction is made between the probability and the detectability of the failure modes and those of the effects. Indeed, between a failure mode and an effect (accident), there is a set of involved "cofactors" (protection barrier, environmental condition ...), recorded in the "condition" column of the table. These conditions will be identified during the step of consequences analysis.

The assessment of the probability of the risk and the assessment of the severity, probability and criticality of the effect will be done during the step of risk assessment. The corrective actions will be proposed at the risk mitigation step.

E. Step 3: Other Risks Analysis

This step is similar to the previous step of system failure modes analysis, but focuses on the other risks (external). The recommended method is to use the principle of an FMECA, that we can call here RECA (Risks, Effects, and Criticality Analysis) (see Table II). This step is to complete few columns of the RECA table (others than severity, probability, criticality and corrective action). The effects are also the potential accidents.

TABLE II. RECA TABLE

Actors	Risk	Probability (of the Risk)	Detectability (of the Risk)	Reason	Effect (Accident)	Condition	Probability (of the Effect)	Severity (of the Effect)	Detectability (of the Effect)	Criticality (of the Effect)	Corrective Actions	Acceptable
...	...	...	...	...	...	...	...	...	...	...	...	YES/NO

The same remarks as for the FMECA remain true concerning the probability, the detectability and the severity of the failure modes and the effects, and the conditions.

F. Step 4: Consequences Analysis

In this step, the consequences of all the identified risks (system failure modes and others) must be analyzed. It is to identify how the risks contribute to an accident. It can be done using event trees [15] to visualize the possible chains of events that led from the risk to the accident, through branching points representing protective measures or interventions (cofactors) (see Figure 6). The minimal cuts associated with the various accidents are also identified.

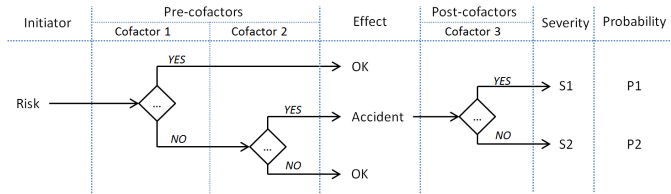


Fig. 6. Event tree

A generic example is given in Figure 6. The effects correspond to accidents, whereas the term consequence refers to events or factors involved in the causes to consequences relationship, starting from the analyzed risk.

G. Step 5: Causes Analysis

The fifth step is to conduct an internal analysis of the system by identifying the causes of system failure modes. These causes analysis must lead to subsystems failure modes. For this step, the use of fault trees [14] is recommended. Indeed, a fault tree provides a simple modeling way to represent the interactions between components from the point of view of reliability. Static fault trees use traditional Boolean logic functions to represent the combination of component failures (events) that cause system failure.

So, the top event of each tree corresponds to a system cause. The objective is to determine the causes of the top event (using logical operators such as AND and OR) in the sub-systems. The leaves of the fault tree correspond to sub-systems failure modes (see Figure 7).

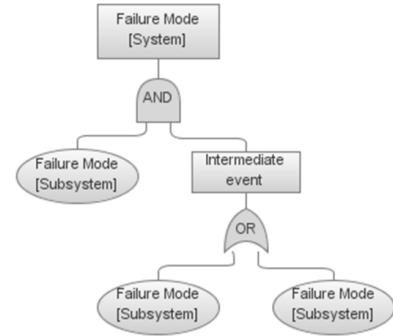


Fig. 7. Fault tree

In fact, the analysed system failure modes correspond either directly to a system risk (defined in the first step), or to a cofactor of an event tree which is a system failure mode.

H. Step 6: Sub-systems Failure Modes Analysis

An analysis of the subsystems failure modes should be done in parallel, using FMECA. The subsystems failure modes used in step 5 re-appears (the principle of the FMECA analysis). This FMECA will define the corrective actions at the sub-systems level that are representative of subsystem reliability requirements.

I. Step 7: Risks Assessment and Mitigation

The seventh step is the central one. It deals with risks assessment and risks mitigation with definitions of corrective actions.

1) Assessment

The risk assessment consists in defining the severity and the probability of the identified accidents, in order to evaluate the criticality. This information must be recorded in the various FMECA and the RECA tables. Concerning the probability, this one must be evaluated based on the fault trees and the event trees. Finally, it must be decided whether the risks are acceptable or not.

2) Mitigation

The risk mitigation consists in advocating corrective actions (to be filled in the FMECA and the RECA) for the risks qualified as "non-acceptable" during the risk assessment step, in order to make them become "acceptable".

The corrective actions can:

- Reduce the *probability* of the accident, by:
 - Fixing an *objective of reliability* with a *reliability requirement* (at system or subsystem level).
 - Modifying the system architecture for a better reliability (with redundancies for example) with an *architectural requirement*, that derives from the *reliability requirement* of the *objective*.
 - Adding additional security equipment (protective barrier) with an *active functional security requirement*, that derives from the *reliability requirement* of the *objective*. During the next iteration of the method, reliability requirements will be defined for this security equipment based on the analysis of the failure modes in which it participates.
- Try to satisfy a *criterion*, for example:
 - A *single failure criterion*, adding a security equipment (barrier) to increase the **necessary** number of failures **that lead to** the occurrence of an accident, with an *active functional security requirement*.
 - A *spatial dispersion criterion*, with an *architectural requirement*.
 - A *redundancy with separate development criterion*, with an *architectural requirement*.
- Reduce the *severity* of the accident, by:
 - Adding a protection or mitigation equipment, with a *passive functional security requirement*.

Note: This is not the only possible corrective actions (preventive maintenance for example). Other types of corrective actions will be incorporated in future work to improve the process.

J. Step 8: Safety Requirements Synthesis

Before eventually transferring the change requests to modify the system, this step will summarize the results in terms of requirements, derivation of requirements, and traceability links between requirements and between requirements and accidents. As in the first version of the method [11], the derivation part is based on the following 3 types of relations:

- System causes and system corrective actions,
- System causes and sub-systems failure modes,
- Sub-systems failures modes and sub-systems corrective actions.

A generic example of this synthesis is given in Figure 8.

K. Step 9: Stop Criterion

The process is finished once all the risks are considered as "acceptable". If this is not the case, a change request must order to modify the system and the method must be reapplied from the beginning by updating the different analysis.

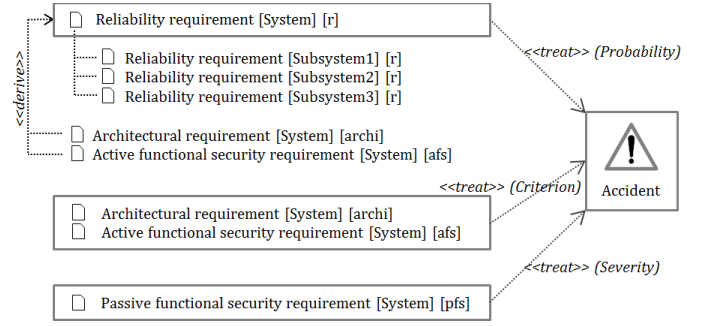


Fig. 8. Requirements traceability synthesis

IV. APPLICATION ON AN EXAMPLE

In order to validate the potential benefit of the proposed method, we have applied it on a short running example. It is the "lift" system with the following initial considered subsystems: "cable" and "motion controller". The main considered function is to "bring up or down the passenger of the cabin".

A. Step 1: Risks identification

Two risks are identified and considered:

- "Cabin fall", as a system failure mode,
- "Fire", as an environmental condition.

B. Step 2: System Failure Modes Analysis

The "cabin fall" risk appears in the FMECA of the system because it corresponds to an internal failure mode. The final version of the FMECA table is given in Figure 9, where all columns are already fielded (which is normally not the case at the beginning of the study).

Function	Failure Mode (FM)	Probability (of the FM)	Causes	Effect (Accident)	Condition	Probability (of the Effect)	Severity (of the Effect)	Corrective Actions	Acceptable
Bring up/down the users	Cabin fall	1E-7/h	Uncontrolled descent of the cabin	Crash of the cabin causing possible death of the users	Failure of the emergency brake	1E-9/h	Catastrophic	- Crash of the cabin must have a frequency < 1E-9/h [r] - Must have 1 emergency brake [afs] - Must have 2 cables in redundancy [archi]	YES

Fig. 9. FMECA of the "lift" system

We can notice the difference that we do in our new version of FMECA between the probability of the failure mode and the probability of the accident (which is the effect). This difference can be justify with the event tree of the step 4.

C. Step 3: Other Risks Analysis

The second risk, "fire", appears in the RECA of the system. One more time, it is the final version of the RECA after several iterations of the methodology that is given in Figure 10, where all columns are already fielded.

Actor	Risk	Probability (of the Risk)	Reason	Effect (Accident)	Condition	Probability (of the Effect)	Severity (of the Effect)	Corrective Actions	Acceptable
Environment	Fire	1E-7/h	Fire in the building	System damaged by fire, that could cause its loss and death of the users	There is a fire extinguishing system	1E-7/h	Major	- Must have 1 fire extinguishing system [pfs]	YES

Fig. 10. RECA of the "lift" system

D. Step 4: Consequences Analysis

Then, the event tree of the Figure 11 exposes the result of the consequences analysis. The “cabin fall” failure mode can have the “crash of the cabin” accident as consequences, if the emergency brake is out of service. This emergency brake was added after an iteration of the methodology in the step 7 - risks assessment and mitigation - in order to reduce the probability of the accident caused by the “cabin fall” risk.

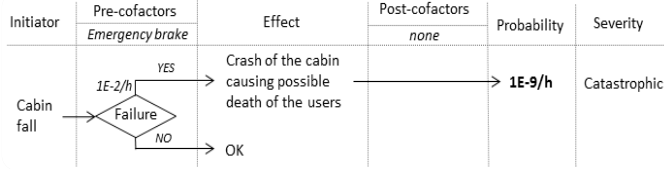


Fig. 11. Event tree of the “cabin fall” failure mode

About the consequences of the “fire” risk, a second event tree, Figure 12, shows that it brings to an accident, but with different severity depending if the fire extinguisher is operational or not. This fire extinguisher was added after an iteration of the methodology in the step 7 - risks assessment and mitigation - in order to reduce the severity of the accident caused by the “fire” risk.

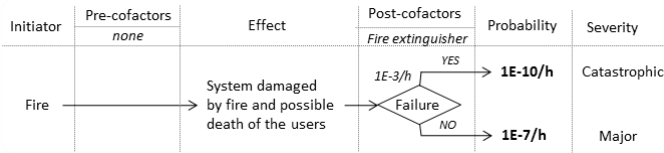


Fig. 12. Event tree of the “fire” risk

E. Step 5: Causes Analysis

In the Figure 13, the fault tree of the “cabin fall” failure mode shows the result of the causes analysis. The “cabin fall” happens if there is an undesirable descent order from the motion controller or if the two cables are broken together.

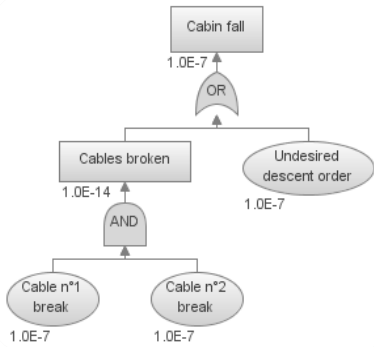


Fig. 13. Fault tree of the “cabin fall” failure mode

F. Step 6: Sub-systems Failure Modes Analysis

The Figure 14, 15 and 16 show respectively the FMECA of the “emergency brake”, “motion controller” and “cable” sub-systems. These sub-systems are the one that are involved in the causes of the “cabin fall” failure mode or in the consequences of the two risks, “cabin fall” and “fire”. The

failure modes that appears in these sub-system FMECA corresponds to leaves in the fault tree or to cofactors in the event trees.

Function	Failure Mode (FM)	Probability (of the FM)	Causes	Effect	Corrective Actions
Brake the cabin in case of emergency	Inoperative emergency brake	5E-3/h	Emergency brake is inefficient	Impossible to stop the cabin with the brake if needed	- The inoperative emergency brake must have a frequency < 5E-3/h [r]

Fig. 14. FMECA of the “emergency brake” sub-system

Function	Failure Mode (FM)	Probability (of the FM)	Causes	Effect	Corrective Actions
Move the cabin up/down	Undesired descent order	1E-7/h	Control order stuck in the down state	The cabin move down in an uncontrolled way, resulting in a possible crash	- The undesired descent order linked to a controller order stuck in the down state must have a frequency < 1E-7/h [r]

Fig. 15. FMECA of the “motion controller” sub-system

Function	Failure Mode (FM)	Probability (of the FM)	Causes	Effect	Corrective Actions
Mechanically link the cabin to the motion controller	Cable break	1E-7/h	Mechanical link broken	The cabin is no longer retained by the cable	- Cable break must have a frequency < 1E-7/h [r]

Fig. 16. FMECA of the “cable” sub-system

G. Step 7: Risks Assessment and Mitigation

Finally, we succeed to make these two risks acceptable after two iterations of the method, by:

- Modifying the system architecture with a redundancy of two cables, in order to reach the reliability for the “cabin fall” system failure mode.
- Adding an “emergency brakes” sub-system as one active security equipment, in order to satisfy the single failure criterion.
- Adding a “fire extinguisher” sub-system as one passive security equipment, in order to reduce the severity of the “fire” environmental condition.

H. Step 8: Safety Requirements Synthesis

The final safety requirements synthesis for this case study is presented in figure 17, where appears all the necessary requirements and there traceability links with the risks.

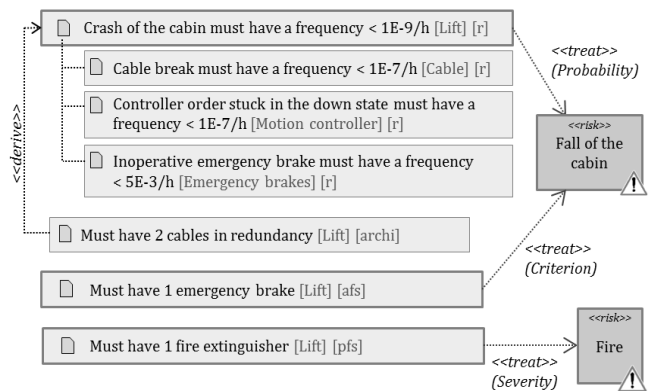


Fig. 17. Requirements traceability synthesis for the “lift” example

The “cabin fall” risk is treated by two main system safety requirements which are “*crash of the cabin must have a frequency $< 1E-9/h$* ” and “*must have 1 emergency brake*”.

The first one reduces the probability of this risk and is derived into three safety requirements at subsystem level:

- “*Cable break must have a frequency $< 1E-7/h$* ”, associated to the “cable” subsystem,
- “*Controller order stuck in the down state must have a frequency $< 1E-7/h$* ”, associated to the “motion controller” sub-system,
- “*Inoperative emergency brake must have a frequency $< 5E-3/h$* ”, associated to the “cable” sub-system.

Another safety requirement derive from this first system safety requirement, which is “*must have 2 cable in redundancy*”, in order to reach the reliability level of the failure mode. It corresponds to a modification of the architecture of the system with a redundancy.

The second one - “*must have 1 emergency brake*” - participates also to the reduction of probability of the accident caused by the failure mode. But mainly, it was added to satisfy the single failure criterion, that’s way it is linked directly with the risk, and not as a derivation of the first system safety requirement.

Concerning the “fire” risk, in this example, it is treated by only one system safety requirement: “*must have 1 fire extinguisher*”. The objective of this requirement is to reduce the severity of the effect of this risk.

V. CONCLUSION

The method provides a support framework to define system safety requirements with an objective of traceability and requirements management. The interest is multiple for the safety field: the method deals with the safety elements (failure modes, safety requirements...) and it is done with a comprehensive system engineering (with traceability and requirements derivation) which is a factor contributing to safer systems. This method is compatible with the standard EIA-632 [17], and it extends the principle and strengthens the links between failure modes researches and analysis (FMECA), causes analysis and effects analysis.

In this work, several safety attributes are taken into account, like reliability, passive security and active security. They correspond to the given classification of safety requirements, which are themselves defined from the corrective actions. Other requirements concerning maintainability or availability should also be considered in further study. The probability, the severity and the criticality was treated through the FMECA.

However, the work still doesn't consider the detectability aspect. We also should update the tool “SafetyLab” (<http://homepages.laas.fr/rguiller/>) that implements the first version of the method presented in [11].

REFERENCES

- [1] Chavalarias, D., Bourguine, P., Perrier, E., Amblard, F., Arlabosse, F., Auger, P., Baillon, J.B., Barreateau, O., Baudot, P., Bouchaud, E.: French roadmap for complex systems 2008-2009. French National Network for Complex Systems (RNSC), Paris Ile-de-France Complex Systems Institute (ISC-PIF) and IXXI, Entretiens de Cargse (2008)
- [2] Juristo, N., Moreno, A.M., Silva, A.: Is the european industry moving toward solving requirements engineering problems? IEEE Software (2002) 70-77 vol.19
- [3] Komi-Sirvio, S., Tihinen, M.: Great challenges and opportunities of distributed software development - an industrial survey. Fifteenth International Conference on Software Engineering and Knowledge Engineering (2003) 489-496
- [4] Avizienis, A., Laprie, J.C., Randell, B., Landwehr, C.: Basic concepts and taxonomy of dependable and secure computing. IEEE Transactions on Dependable and Secure Computing (2004) 11-33 vol.1
- [5] Rasmussen, J.: Risk management in a dynamic society: A modelling problem. Safety Science, Elsevier Science Ltd. (1997) 183-213 vol.27
- [6] Goguen, J., Linde, C.: Techniques for requirements elicitation. 1st IEEE International Symposium on Requirements Engineering, San Diego (1993) 152-164
- [7] Gotel, O.C.Z., Finkelstein, C.W.: An analysis of the requirements traceability problem. International Conference on Requirements Engineering (1994) 94-101
- [8] Sahraoui, A.E.K.: Requirements traceability issues: Generic model, methodology and formal basis. International Journal of Information Technology and Decision Making (2005) 59-80 vol.4-1
- [9] Sommerville, I.: Software engineering (update) (8th edition). International Computer Science, Boston, MA, USA (2006) 11-33 vol.1
- [10] Guillermin, R., Demmou, H., Sadou, N.: System engineering approach for safety management of complex systems. Proceedings of European Modeling and simulation (ESM), Leicester, United Kingdom (2009)
- [11] Guillermin, R., Demmou, H., Sadou, N.: Combining fmeca and fault trees for declining safety requirements of complex systems. European Safety and Reliability Conference (ESREL), Troyes (France) (2011)
- [12] Lindsay, P.A., McDermid, J.A.: Derivation of safety requirements for an embedded control system. Engineering, Test and Evaluation Conference, Sydney (2002) 29-30
- [13] Buzzatto, J.: Failure mode, effects and criticality analysis (fmeca) use in the federal aviation administration (faa) reusable launch vehicle (rlv) licensing process. (1999) 7.A.2-1 - 7.A.2-7 vol.2
- [14] Lee, W.S., Grosh, D.L., Tillman, F.A., Lie, C.H.: Fault tree analysis, methods, and applications - a review. IEEE Transactions on Reliability (1985) 194-203 vol.1
- [15] Villemeur, A.: Sûreté de fonctionnement des systèmes industriels. Paris : Edition Eyrolles (1988) 785.
- [16] Sahraoui, A.E.K., Buede, D., Sage, A.: Issues in systems engineering research. INCOSE congress, Toulouse (2004)
- [17] EIA-632: Processes for engineering systems. Electronic Industries Alliance standard, January 7 (1999)
- [18] CEI-60812: Techniques d'analyse de la fiabilité des systèmes. (1995)