

Requirements Changing in System Engineering and Impact on Safety

Abdelaziz Babiker
Sudan University of Science and Technology
(SUST), Khartoum Sudan

Abd-El-Kader Sahraou
LAAS-CNRS, University de Toulouse, CNRS, UT2J
Toulouse, France

ABSTRACT

The requirements evolution with a systems engineering approach can be modeled by systems engineering processes at least for the on development phases of the product. This paper addresses requirements changing issues and systems engineering deployment, based on EIA-632 standards; and a methodological approach is proposed. The paper discuss some important research directions in requirements traceability and discuss a simple traceability model at low level.

Keywords

Requirements engineering, systems engineering, requirements change, safety requirements.

1. INTRODUCTION

The actual systems are more and more complex, because they integrate a variety of technologies. The system need often long period of development. When we take a change on requirements the objectives are to improve the functionality, the cost or the delay of systems; but unfortunately, this modification can be affect a problem with others requirements those involved with the safety of the system.

The requirements change occurs in two main cases:

1. Concerns updating existing requirements that led to an actual version of the systems to adapt to new environment (Larsen and Buede, 2002).
2. When new technology is occurred then new requirements will implement, consequently for reasons of cost or feasibility

When we consider the change of requirements, we focus mainly in this paper to study the effect of the change on the security of the system; we try to see if any change cannot harm the system and the risk to affect the systems security. You have two type of change for requirements. The first such as the requirement change and the system is in the phase of development, the second when we take a modification for a realize requirements

The main idea in this part is to present the effect of change on the system if one or many requirements change, and to present what are the considerations we must make, and what is effect on safety requirements.

Our goals in this paper are to study:

1. The problematic if we take a modification (on one or many) requirements, how can be sure that we have not any problem on safety of the system?
2. The problematic of How can we take a study of the effects on safety when we have a changes?
3. Where is change in (functional requirements or non-functional requirements, physical requirements,

operational requirements...) and is it makes a problem of safety of the system?

2. REQUIREMENTS ENGINEERING

Requirements Engineering covers all activities related to the elicitation, modelling, analysis, specification and management of requirements throughout the systems development life cycle. RE is a multi-disciplinary and communication rich part of software development that utilises a variety of techniques and tools at different stages of development and for different types of application domains.

The processes are viable for the engineering or reengineering of the end products that forms the system, as well as the development of enabling products required to provide life-cycle support to system end products. Figure 1 shows the relationships between the processes.

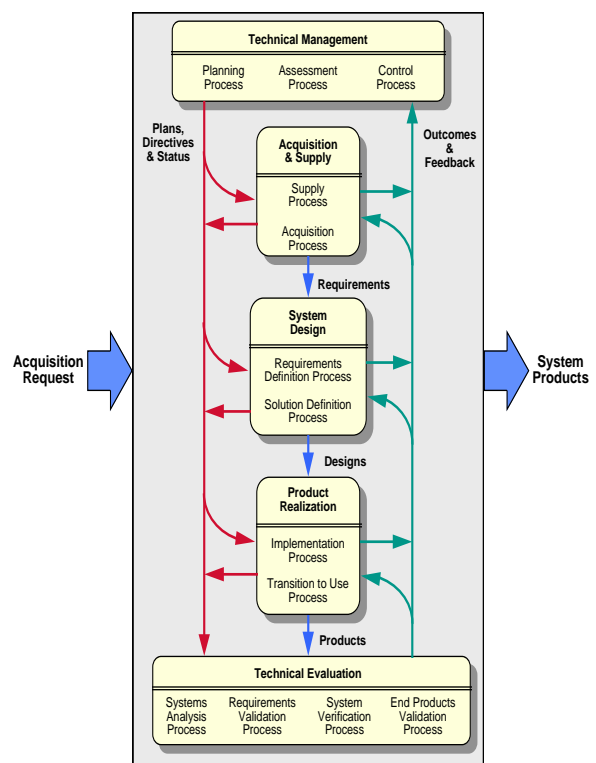


Figure 1: Requirements for Engineering System

Our context study will, through the system engineering framework, give importance to end products aggregation of end products.

Enabling products are used to perform the associated process functions of the system develop, produce, test, deploy, and support the end products; train the operators and maintenance staff of the end products; and retire or dispose of end products

that are no longer viable for use. The end products and the enabling products are either developed or reused, as appropriate. The relationship of these system elements is shown in Figure 2.

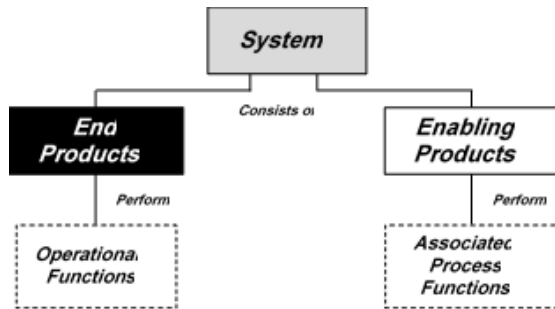


Figure 2: System concept

The system make up the basis for a larger structure, called the building block, shown in Figure 3. The building block. Effectively, it can be seen the end product is separated from other enabling product issues; this framework will help companies to enhance their RE process development when considering critical issues that often get mixed up with final product; the infrastructure for agile development relies highly on what is often called logistics but in fact they are the enabling product in systems engineering view: training, deployment products, test, development, product etc. as shown in figure 3.

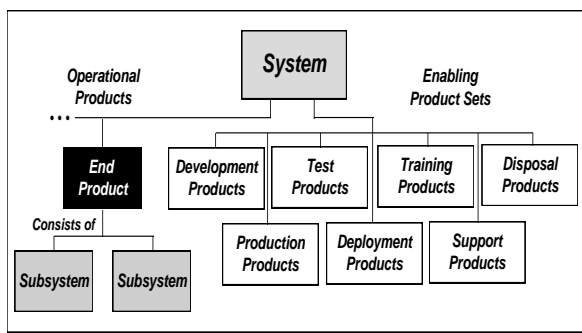


Figure 3: Building block

For such objective, standard EIA-632 will be considered as the mostly popular and effective standard being used; such standard has been deployed in major space, manufacturing, military and aeronautic industries.

Errors in requirements specifications can have a major impact on software costs and project duration. The detection and correction of potential problems during requirement analysis may mitigate many much larger problems later on during testing and maintenance. Requirements engineering is a process from system engineering, only active in an upstream phase, the objectives of this process is to identify the stakeholders, to collect and explain, formalize, analyze, validate, and allocate requirements. One specification for any system contains a highly number of requirements. The specification is necessary to make requirements very coherent, and to control, to change, to manage requirements.

Many studies in Requirements Engineering show that 90% of costs are engage in the first phase of development of System Engineering, and that is better to change one or many requirements in the first phase of development of the system. However when apply a modification on specification there have an impact not only in cost but also in safety. In

Concurrent Engineering (CE) there should be a virtual data bus communicating between every process so to enable to activate parallel processes when a change during systems development.

3. MAPPING SE TO CE

The mapping we prone is how to customise the SE processes to CE. Such mapping is not systematic as it depends how CE is deployed in every enterprise.

As stated in European research for the integration in manufacturing Information technologies already have a pervasive influence on modern manufacturing and engineering. This ranges from supporting the design and engineering process, through production planning and control, and on the toe control of manufacturing equipment and distribution systems. However, there remains enormous scope for IT growth in the manufacturing sector.

The main topics covered within the Integration in Manufacturing research area include:

1. Product data exchange and modelling which deals largely with the internal representation of all product-related design and manufacturing data.
2. Factory automation, which includes the development of software tools to support human-centered production concepts; advanced work in robotics; and software developments in the field of simulating products, the manufacturing process, and the design and layout of manufacturing cells, assembly work-stations and entire plants.
3. Communications and logistics, to improve the integration of distributed manufacturing applications, both within a plant and between enterprises collaborating in a supply chain.

Unify processes for the needs until the disposal cannot be attained sequentially in any product or service development.

The systems development is based on unified process. These processes make abstraction of the systems nature. In [Sahraoui et al, 2004) we prone numerous research issues related to the subject.

3.1 A Decision focused framework for Life-Cycle based Architecting and design in SE

3.1.1 Definition of the Problem

Systems engineering is a multi-disciplinary problem definition and problem solving process that is implemented by people. There are as many definitions of this process as there are systems engineers with no real agreement on an underlying theory that unifies the process. Most systems engineers will agree to the following characterization of systems engineering:

1. Focus: a process and systems management focus that will result in the engineering of a trustworthy product or operational system that will satisfy user and customer requirements and for which these stakeholders will pay
2. Scope: entire life cycle of the system, including the definition of user and customer requirements, development of the system products and enabling products. These enabling product systems include test system, deployment system, training system,

operational support (logistics, maintenance, etc.), refinement system, and retirement system

3. Products: Systems Engineering Management Plan, Operational Concept for the product, hierarchy of requirements documents for each key system (starting with the system-level requirements document and following the physical decomposition of the system), architectures and hierarchy of interface control documents that define the interfaces at each level of the physical decomposition
4. Characteristics of SE Process: Combination of executable models to examine the behavioral (functional, non-functional) characteristics of alternate designs and architectures and qualitative and quantitative.

3.1.2 Research approach

A design process is characterized by a collection of decisions. In this, we use the fundamentals of decision analysis in which a decision is characterized by alternatives (what you can do – designs), values (objectives hierarchy with a quantitative value model to describe the trade-offs of the stakeholders across the key measures of effectiveness), and facts about what is known and not known. Within this context view systems engineering as a risk mitigation strategy that includes architecture, design, and testing. We must recognize that the entire process must adhere to the following principles:

1. Coherent value structure across all decisions.
2. Managed by an adaptive, feedback-control process for decision making.
3. Focused, cost-effective, risk management of both the (life cycle) design and design process.

3.1.3 Expected Results

1. Integration of values across all decisions for the system's life cycle
2. Architecture and design framework for an integrated and coordinated decision-making framework with a schedule that identifies serial and concurrent decision-making activities
3. Structure for reviews of key products that is based on the principles of feedback-control systems as well as the coordinated decision framework and is sensitive to the uncertainties
4. Framework for risk management that is sensitive to the integrated values across the system's life cycle and the decision framework.
5. Process that can be generalized to other problem solving situations.

4. AN APPROACH: EIA STANDARD APPLICATION TO CE

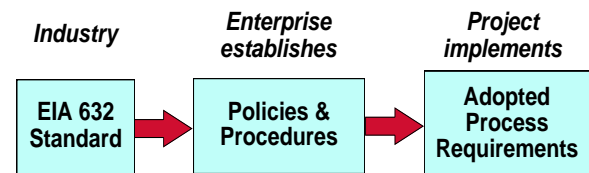
The EIA 632 as briefly presented in part 1 is common standard adopted in aeronautics and space industry and still to be deployed in other sector with other standard as IEEE – P1220 and ISO-15288.

Our work in such limited only to the design and verification processes (8 processes from the total 13 processes); the left out processes concern the management, supply and acquisition processes.

We are concerned also by the products and enabling products, that is really implicit in concurrent engineering and recommended in EIA 632.

MIL-STD-499B was intended to be the first military standard to address systems engineering as a whole. MIL-STD-499A addressed the management of systems engineering and, thus, had a different focus, although MIL-STD-499B was intended to supersede it. MIL-STD-499B uses a lengthy definition for systems engineering which concentrates on integration of disciplines, full life-cycle coverage, assurance of interface integrity, management of technical risks, and validation that the system meets the needs and requirements. The actual work of systems engineering is clarified only by reading the full standard. This work tends to lean heavily toward the Technical Management definition of systems engineering.

The standard begins with five pages of definitions and follows that with one chapter on General Requirements and one on Detailed Requirements. The General Requirements chapter is ordered by life-cycle phase, while the Detailed Requirements call out individual systems engineering work products. These work products clearly imply a large military contract environment, calling out numerous customer reviews.



Application of the standard

While this standard was never released, the May 1992 copy in particular has enjoyed a long applicable life in that it served as the basis for both the IEEE 1220 standard and the EIA IS 632 standard, both of which were fairly minor modifications.

4.1 Main Product

The product to be designed: from requirement to retrieval; such product can be decomposed in sub products and so on.

4.2 Enabling Product

5. The Requirement Evolution in CE Context

Our approach and contribution will focus mainly on impact of requirement change and development a methodology for requirements change. This will be carried on the basis of:

- a) Traceability model
- b) The concurrent processes
- c) A formal framework for the requirement change

5.1 General Approach

We are investigating many approaches to such issue. However, the global approach is thought as an operational view as illustrated by the following figure. The formal basis for such approach is not tackled yet but some items are thought to be useful and to be discussed in latter section.

This preliminary approach is systems engineering context characterised by the interaction by four models; these involve respective processes. Our concern is the development of the change model and its interaction with all other models. We will present the traceability model dynamics that have been

used in earlier work [Hellouin1 et al.] [Hellouin2], and make abstraction of the development and system configuration management models as their basic characteristics are known for long time. We know that any requirement change will concern and trigger all four models. In our first approach we will be concerned the change, traceability and development models. However, some principles will guide towards the deepening of the approach as future work will focus mainly on refining the approach:

1. Any change request either at any step of development model suppose the availability of a traceability model.
2. A change request for an operation module will necessarily require tracing back the original requirement
3. Make distinction between functional and non-function requirements
4. Identify security/safety requirements.
5. Create link between associated function and safety requirement.

5.2 Traceability Model

As discussed earlier, providing traceability of requirements to their sources and the outputs of the system development process can be along several dimensions. Different stakeholders contribute to the capture and use of traceability information, often with different perspectives. A user has a different vision from an audit specialist, a system designer or a validation engineer. Some typical questions are often asked:

What are the systems components that are affected by a specific requirement?

Why are the components affected by such requirements?

How are the components affected by such requirement?

What are the sources of a low level requirement?

Why and how two requirements are related?

And so on ...

An object can belong to one of the following classes: requirement, design, components, system/subsystem, etc. Attributes and operations (activities) are associated with each class, subclass.

Sources are all available information as documents, phone call, E-mail about the object lifecycle. Traceability concerning specific decision made can be found through the relation documents.

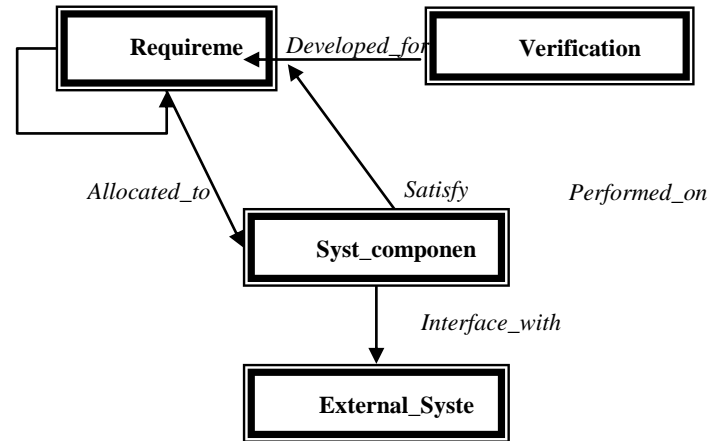


Figure 4: Traceability at Low Level

We can use this traceability model to identify any link that may be subject and constrained by a requirements change.

6. CONCLUSION

An extension of the approach for key non-functional requirement is the objecting of future research. The research discuss some related traceability issues that its necessary applied to the requirements to ensure a high quality software. The questions that appears in section 5.2 explain the problems that related to the traceability between system components that are necessary to be solved.

7. REFERENCES

- [1] Abdelaziz Babiker; Abd-El-Kader Sahraoui; On Non Functional Requirements, Their Evolution and Impact on Safety Proceedings of the International Conference on Software Engineering Research and Practice (SERP), 01/2016
- [2] Barry, E.J.; Mukhopadhyay, T.; and Slaughter, S.; Software Project Duration and Effort: An Empirical Study, 2002, Information Technology and Management, vol. 3, pp. 113-136.
- [3] Buren, J.V.; cook, D.; 1998, Experiences in the Adoption of Requirements Engineering Technologies, Journal of Defence Software Engineering, December, pp.3-10.
- [4] Hellouin, L.; Beaugrand, J.L.; and Sahraoui, A.E.K.; 2001, Requirements process and traceability issues, 11th Annual INCOSE Symposium, Melbourne.
- [5] Hellouin, L.; 2002, Contribution à l'ingénierie des exigences et à la traçabilité, PhD Thesis, LAAS-CNRS and INPT, LAAS-Report 02074.
- [6] El-Jamal, H.M: Requirements evolution and impacts on safety. IFIP 18th World Computer Congress, August 2004, Toulouse.
- [7] El-Jamal, H.M, Sahraoui, A.E.K: requirements evolution methodology and impacts on safety. Quality conference, Bordeaux, Mars 2005.
- [8] Hughes, T.; Cindy, M.; Design traceability of complex systems. In Human Interaction with Complex Systems, pages 37–41, March 1998.

- [9] Krasner, H.; 1989, Requirements Dynamics in Large Software Projects, 11th World Computer Congress (IFIP89), Amsterdam, Netherlands.
- [10] LuizMarcio, C.; Julio Cesar, S. d. P.L., Nonfunctional Requirements: From Elicitation to Conceptual ModelsIEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 30, NO. 5, MAY 2004 Engineering, Fort Collins.
- [11] Nurmaliani, N.; ZowghiD.; and Fowell S., 2004, Analysis of Requirements Volatility during Software Development Life Cycle, proceedings of the Australian Software Engineering Conference (ASWEC), April 13-16, Melbourne, Australia.
- [12] Sahraoui, A.E.K, Buede, D., Sage, A: Issues in systems engineering research. Incose Int'l Symposium, June 2004, Toulouse.
- [13] Stevens R;Brook, P.; Systems Engineering - Coping with complexity. Prentice Hall, 1998
- [14] Traoré, I.; El Jamal, M.H.; Yanguo Liu, M.; Sahraoui, A.E.K.; 2004, UML-PVS for requirements specification. Incose symposium, Toulouse, France.
- [15] Traoré, I.; Sahraoui, A.E.K.; 1997, A Multiformalism Specification Framework with Statecharts and VDM, 22nd IFAC/IFIP Workshop on Real Time Systems,Lyon, France.
- [16] Zowghi, D.;Nurmaliani, N.; 2002, A Study of the Impact of Requirements Volatility on Software Project Performance, 9th Asia-Pacific Software Engineering Conference, Gold Coast, Australia.
- [17] [Sheard, 2003] SheardS.A : Three types of systems engineering application. Incose symposium, 2003.