Overview of the EIA 632 Standard: Processes for Engineering a System

James N. Martin

Raytheon Systems Co.

6600 Chase Oaks Blvd M/S 8497

Plano, TX 75023

j-martin@ti.com

Abstract. The EIA 632 standard has been developed that describes the "processes for engineering a system." This standard evolved from EIA/IS 632—the interim standard that described a systems engineering process. This paper will describe the structure of the standard, elements of the process, and key concepts such as stakeholder requirements, enabling products, building blocks, and development layers.

INTRODUCTION

In April 1995, the G47 Systems Engineering Committee of the Electronic Industries Association (EIA) chartered a working group to convert the interim standard EIA/IS 632 into a full standard. This full standard has been developed over the last three years and is planned to be released in July 1998.

The interim standard, EIA/IS 632, was titled “Systems Engineering.” The full standard was expanded in scope to include ***all the technical processes for engineering a system***. It is intended to be a higher level abstraction of the activities and tasks found in the IS version plus those other technical activities and tasks deemed to be essential to the engineering of a system.

This paper describes the elements of the process, and key concepts. The intended purpose is to give the reader of the standard some background in its development and to help other standards activities in developing their own standard. (Since this standard was not released at the time of this paper submittal, some differences may be found between this paper and the released version.)

There is a separate paper that describes the evolution from an interim standard to the full standard [Martin 1998]. This standard is intended to be a “top tier” standard for the ***processes essential to engineering a system***. It is expected that there will be second and third tier standards that define specific practices related to certain disciplines (e.g., systems engineering, electrical engineering, software engineering) and industry domains (e.g., aircraft, automotive, pharmaceutical, building and highway construction).

It is important to understand several things that are ***not*** covered by this standard:

a) Does not define what “systems engineering” is;

b) Does not define what a “systems engineer” is supposed to do; and

c) Does not define what a “systems engineering organization” is supposed to do.

STRUCTURE OF THE STANDARD

Clause 1 ***Scope***

Clause 2 ***Normative references***

Clause 3 ***Definitions and acronyms***

Clause 4 ***Requirements***

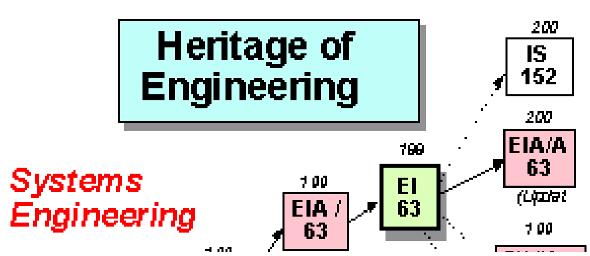
Clause 5	***Application guidance***
Clause 6	***Application context***
Annex A	***Glossary***
Annex B	***Enterprise-based life cycle***
Annex C	***Process tasks***
Annex D	***Planning documents***
Annex E	***Technical reviews***
Annex F	***Unprecedented and precedented development***

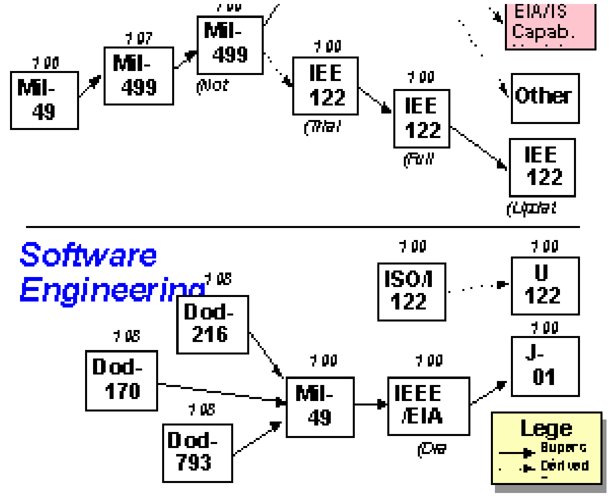
ROLE OF EIA 632

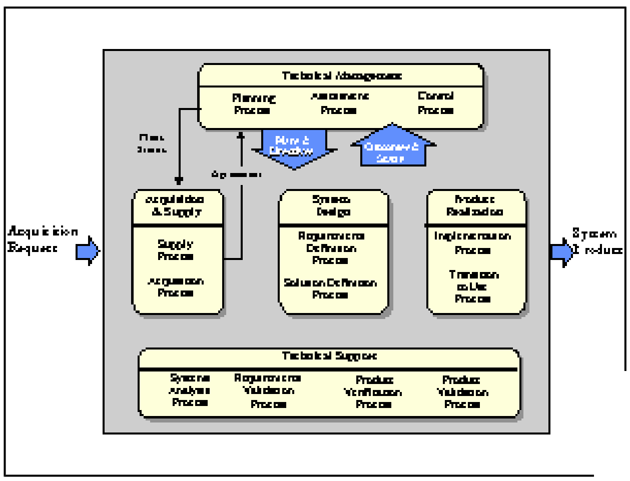
Implementation of the requirements of EIA 632 are intended to be through establishment of enterprise policies and procedures that define the requirements for application and improvement of the adopted processes from the standard.

THE PROCESSES

The figure below shows the processes described in EIA 632 and their relationship to one another. Each enterprise will determine which of these processes are implemented by systems engineering personnel.

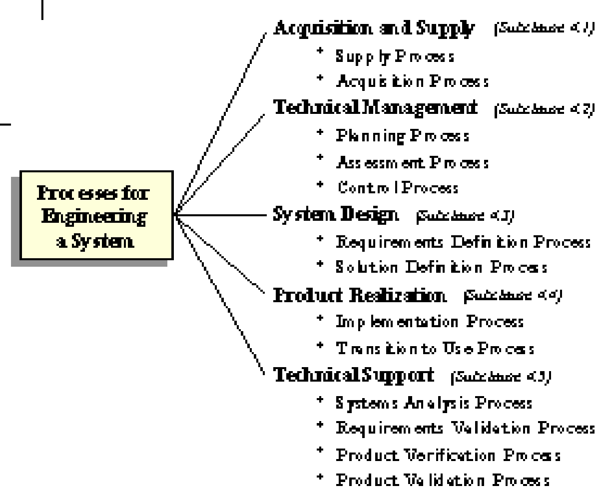






PROCESS HIERARCHY

The processes for engineering a system are grouped into the five categories as shown below. This grouping was made for ease of organizing the standard and is not a required structure for process implementation.

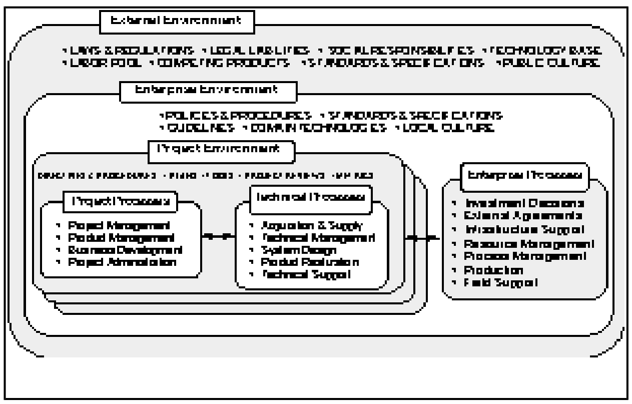


HERITAGE OF EIA 632

The figure below shows the relationship between EIA 632 and other standards on systems engineering. Some of the key software engineering standards are shown for comparison since there has been an intimate relationship between the development of both types of standards. There has been much activity recently in unifying the processes contained in each.

PROJECT CONTEXT

These "technical" processes fit into a larger context of a project (see figure below), and the project resides in some sort of enterprise, which in turn resides in an environment external to the enterprise. There are processes in the project and within the enterprise (but outside the project) that significantly affect the successful implementation of the technical processes.



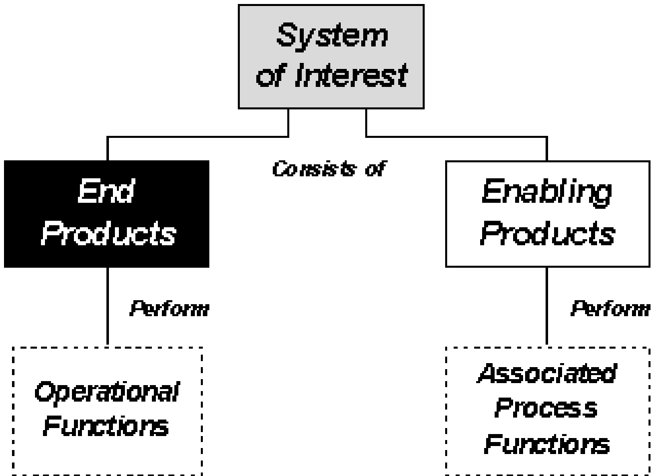
KEY CONCEPTS

To understand the processes as described in this standard, it is essential to understand the distinct use of certain terms and the conceptual models that underlie each process. Some of the key terms are: system, product, verification, and validation. Some of the key concepts are: building block, end products, associated processes, and development layers.

What is a system? The term "system" is commonly used to mean the set of hardware and software components that are developed and delivered to a customer. This standard uses this term in a broader sense in two aspects.

First, the system that needs to be developed consists of not only the "operations product" (that which is delivered to the customer and used by a user), but also the enabling products associated with that operations product. The operations product consists of one or more end products (so called since these are the elements of the system that "end up" in the hands of the ultimate user). The associated processes are performed using enabling products that "enable" the end products to be put into service, kept in service, and retired from service.

Second, the end products that need to be developed often go beyond merely the hardware and software involved. There are also people, facilities, data, materials, services, and techniques. See figure below.



This is not intended to be an exhaustive list of the “basic” product types since these will vary depending on the particular business or technology domain. For example, in the television industry, “media” is certainly one of the system elements that constitute the overall system that is developed. “CBS News” might be considered the system, for example, with end products like:

a) cameras, monitors (hardware)

b) schedule management tools, video compression algorithms (software)

c) camera operators, news anchor (personnel)

d) studio, broadcasting tower (facilities)

e) script, program guide (data)

f) pictures, stories (materials)

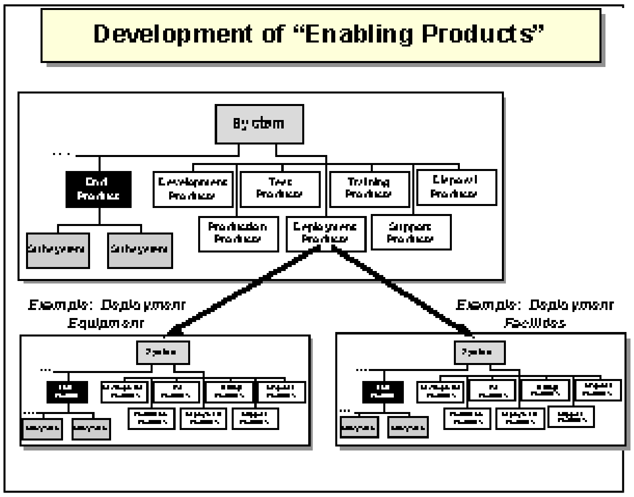
g) airplane transportation, telephone (services)

h) presentation method, editing procedures (techniques)

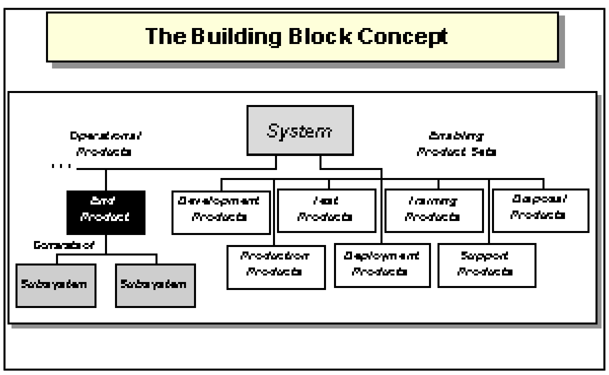
i) airwaves, worldwide web (media)

Note that any or all of these end products could be “off the shelf” (OTS). But some of them may need to be conceived, designed, and implemented. Even if one of these items is truly OTS, it may still need some enabling product to allow effective use of that item. For example, even if you can use existing editing procedures, you may need to develop a training program to train the editors.

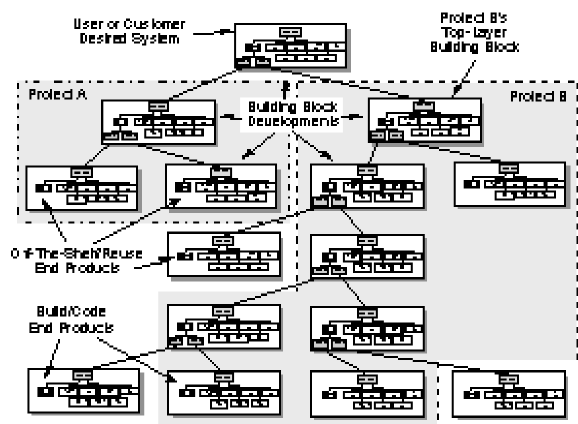
BUILDING BLOCK

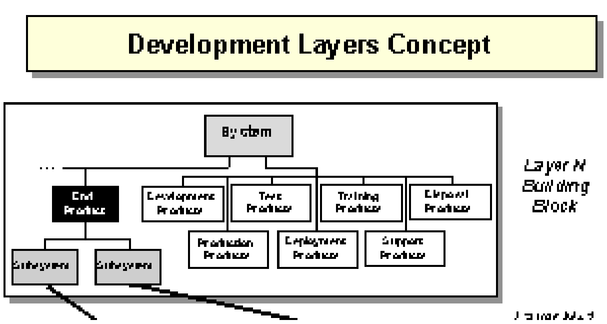


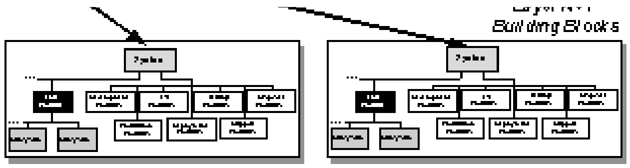
As we can see from the description above of the "CBS News" system, the non-hardware/software items may be crucial to successful realization of the whole system. You may also need to develop the associated processes along with the relevant enabling products. If we tie all these elements together, we can illustrate this using the so-called "building block" shown below. There are seven associated processes related to these sets of enabling products. These processes are used at various points in the life of a product. Hence, the use of the building block concept is intended to help the developer in ensuring that the full life cycle of the end products is properly considered.



Note that each end product can consist of subsystems. Each of these subsystems may need its own development. The building block can be used at the various "development layers" of the total system. These development layers are illustrated in the figure below.







DEVELOPMENT OF ENABLING PRODUCTS

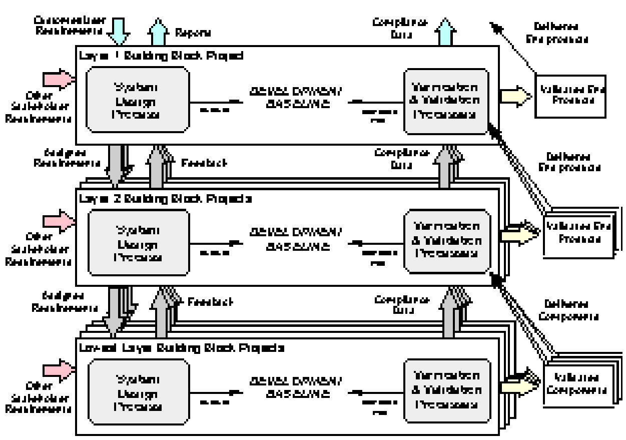
As I mentioned above, the enabling products may need to be developed also. For each associated process, there could be enabling products that either exist already or that need some degree of development. The figure below shows how enabling products related to the deployment process have their own building blocks.

RELATIONSHIP BETWEEN THE BUILDING BLOCKS AND THE PROCESSES

The building blocks relevant to a particular system de-velopment can be “stacked” into a System Breakdown Structure (SBS). The System Design Process and Veri-fication and Validation Processes have a special rela-tionship to the SBS as shown in the figure below.

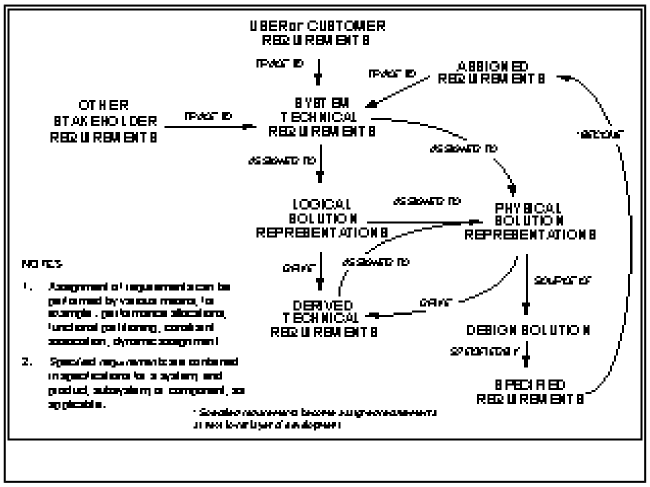
HIERARCHY OF BUILDING BLOCKS

Typically a project is developing a system of such a size that it will require more than one building block to complete the development. The figure below shows an example of how several building blocks would be related to each other within a project. Notice also how an “adjacent” Project A has building blocks that might have interfaces to the systems under development by Project B. There also may be building blocks above and below a project.



REQUIREMENTS

Requirements are especially useful for engineering a system, especially when the system consists of several building blocks. Requirements act as a kind of "decoupling mechanism" between the building blocks. They are the "terms and conditions" between the layers of development. The figure below shows the different types of requirements dealt with by the processes described in EIA 632.

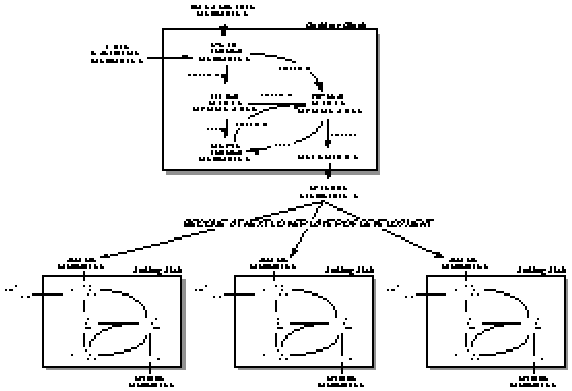


Requirements in general do not come from a particular customer. And even when they do, as is often the case in government-sponsored developments, they do not represent the full spectrum of the requirements that will eventually need to be defined for development of the products of a system.

User Requirements. User requirements are often non-technical in nature and usually conflict with each other. These need to be translated into "technical requirements" that are in the vernacular of the engineers on a project and are specific to the domain of the technology to be used in the system.

Customer Requirements. Customers will often translate their perceived user requirements into a set of requirements to be given to a development project. These also, like the user requirements, are often not technical enough and usually conflict with each other. Moreover, a particular project may have more than one customer for its products.

Stakeholder Requirements. In defining the technical requirements for the system, one must consider that there are other stakeholders for that system. For example, the manufacturing organization within your company may desire that you utilize their existing processes, tools, facilities, etc. There are constraints associated with these "enabling products." Hence, you must identify the stakeholder requirements that drive the development.



Requirements Decomposition. Once the system technical requirements are defined, further requirements are often "derived" by analyzing both the logical and physical aspects of the system. Some of these derived requirements, plus whatever system technical requirements are relevant, are "specified" for the system, its end products, and possibly subsystems of those end products.

Given these specified requirements for an item, the process starts all over again for items needing further development. In other words, another layer of building blocks may be required to develop an item far enough such that the bottom-most item can be procured from a supplier. This is illustrated in the figure to the right.

SUMMARY

The purpose of this paper is to give a high-level overview of the processes for engineering a system described in EIA 632. Further details can be found in the full publication of the standard.

REFERENCES

EIA/IS 632, "Systems Engineering." 1994.

EIA 632, "Processes for Engineering a System." To be published in 1998.

Martin, James N., "Evolution of EIA 632 from an Interim Standard to a Full Standard." INCOSE 1998 Symposium.

BIOGRAPHY

James Martin is a systems engineer at Raytheon Systems Company working on airborne and satellite communications systems. He is leader of the working group responsible for developing EIA 632, a US national standard that defines the processes for engineering a system. He previously worked at AT&T Bell Labs on wireless telecommunications products and underwater fiber optic transmission products. He has a book published by CRC Press called “Systems Engineering Guidebook”.