Proceedings of the TMCE 2020 Symposium, 11-15 May, 2020, Dublin, Ireland, edited by I. Horváth and G. Keenaghan

ANALYSING THE IMPACTS OF SYSTEM OBSOLESCENCE BASED ON SYSTEM ARCHITECTURE MODELS

Marc Zolghadri
Quartz Lab, Supmeca
Marc.Zolghadri@supmeca.fr

Rob Vingerhoeds
ISAE-SUPAERO, Université de Toulouse
rob.vingerhoeds@isae-supaero.fr

Claude Baron
LAAS-CNRS, INSA, Université de Toulouse
Claude.Baron@laas.fr

Sophia Salas
ISAE-SUPAERO, Université de Toulouse, Quartz Lab, Supmeca
Sophia.Salas@isae-supaero.fr

ABSTRACT

Obsolescence is the fact that an entity (physical or logical) is becoming outdated or no longer useful and in this way is considered inappropriate. The objectives of this article are twofold. First, we seek to contribute to the understanding of obsolescence through two conceptual models. They enable, among others, to link the points of view of the external and internal actors of the system. This mapping is fundamental when it comes to solve the re-design problems posed by some obsolescence issues. The obsolescence management process consists of Prepare, Identify, Assess, Analyze, and Implement. The second objective is to offer two support tools for the Identify and Assess phases: House of Obsolescence and System Obsolescence Exposure Analysis. The first allows to map the changes, desired or imposed, by external actors to the critical components and functions of the system architecture. These latter are then analyzed using the second tool whose objective is to assign an obsolescence exposure index to the identified obsolescence issues in order to prioritize them for solution determination during the analysis phase. The tools make extensive use of the modeled system knowledge through the application of System Engineering. The application of the tools is then illustrated through a case study.

KEYWORDS

Obsolescence, Systems engineering, Obsolescence management process

1. INTRODUCTION

We increasingly hear about obsolescence. Sometimes, consumers are suspecting planned obsolescence fearing that companies artificially reduce the effective service life of products forcing consumers to replace them (Kreziak, Prim-Allaz, & Robinot, 2017), or (Bulow, 1986). Such planned obsolescence is reprimanded by law in France since August 2015 (https://bit.ly/2lUT2RA). The research presented in the current paper does not deal with planned obsolescence, nor does it refer to the study of knowledge obsolescence, which is also a real challenge for companies.

According to the international standard (IEC 62402, 2019), obsolescence is the "transition from availability from the original manufacturer to unavailability". Many reasons may be hidden behind this transition process such as technical, financial, legal or technological reasons. This is the case for instance for Windows 7 for which Microsoft announces that "*Support for Windows 7 is ending on January 14th 2020*" (https://bit.ly/2KHywim). Ending support does not stop the system from operating but means that no evolution of the functionalities or protection will be proposed in the future. Consequently, a system impacted by obsolescence would have an increasingly degraded performances, will not respect specific constraints, will be hard to maintain or it will soon stop operating, see (Zheng, Liyu; Terpenny, Janis P.; and Orfi, 2014). But this is not the most worrying. Such a triggered obsolescence exposes the system to various threats, among others security issues: "*... an old unpatched OS is a cybersecurity risk – the cost of an incident may be substantially higher than the cost of upgrading.",* see survey made by Kaspersky, published on 26th of August 2019 (https://bit.ly/2mntP2t).

Obsolescence and its effects can be described according to three fundamental characteristics. The first is that obsolescence has fundamentally delayed effects. This time delay is a key factor for the design of monitoring techniques in systems obsolescence management, as for example, the failure or availability of a system containing an obsolete component may not occur until well beyond its occurrence (real and proven). The second characteristic is that obsolescence can involve elements at all levels of the system structure hierarchy. "*All systems are at risk of obsolescence and Diminishing Manufacturing Sources and Material Shortages, and they may occur at the part, module, component, equipment or system level*", (SD-22, 2016). The last characteristic concerns the fact that obsolescence never remains confined because the obsolete or near-term obsolescence element interacting with others can prevent the running of the system's internal processes. Therefore, the consequences of the obsolescence, if not properly solved, may propagate to a Next-Higher Assembly or the entire system.

Understanding the obsolescence phenomenon and its propagation mechanisms is therefore essential to be able to propose decision-making support systems for obsolescence management. The objectives of our research are then twofold with 5 sub-goals:

(1) Problem-posing. The idea is to offer a sound set of concepts and models that allow to:

a) Understand and model obsolescence and its propagation.
b) Perform a system obsolescence exposure analysis.
c) Predict the consequences of obsolescence on the functional and non-functional requirements of the system, to understand whether an identified obsolescence is potentially a "showstopper" (Shuman, 2002) or "show-downgrader".

(2) Problem-solving. Once identified, solutions must be defined to deal with obsolescence, focusing on:

a) Managing obsolescence if it is unavoidable or proven, or
b) Designing systems that are resilient to the inevitable occurrence of obsolescence.

The paper addresses the points (i) and (ii). Interested readers may refer to (Zheng, Terpenny, & Sandborn, 2015) for more details about the points (iii), (iv) and (v). The understanding and modelling of obsolescence and the propagation of its consequences are addressed by the proposal of two conceptual models linking obsolescence to the fundamental concepts of system engineering.

The components or functions of a system-of-interest are not at the same level of exposure to obsolescence and are not sensitive to the occurrence of obsolescence at the same degree. Two first tools are then proposed for analyzing the health status of a system exposed to the risks for potential occurrence of obsolescence. This risk is then quantified for each component and function identified as critical by the experts of the system-of-interest.

The paper is structured as follows. Section 2 reviews the concepts related to obsolescence and the propagation of the consequences of obsolescence through the system architecture. To this purpose, some fundamental concepts of systems engineering as well as the channels for propagating the consequences of obsolescence will be highlighted through the system models. Section 3 presents conceptual models of obsolescence based on systems engineering. Section 4 then details our proposal of the first tools for health analysis of the system-of-interest facing obsolescence. These concepts are illustrated in Section 5 through an example of weather forecasting system taken from (Roques, 2017). The article concludes with a discussion of the results obtained and a presentation of the work in progress to cover the other fields of study defined in problem-posing and problem-solving.

2. RELATED WORKS

2.1. Obsolescence

According to (Bartels, Ermel, Sandborn, & Pecht, 2012) obsolescence refers to "materials, parts, devices, software, services and processes that become non-procurable from their original manufacturer or supplier". It deals with a "process or condition by which a piece of equipment becomes no longer useful, or a form and function no longer current or available for production or repair" (SD-22, 2016). Many authors agree on that the "root cause of obsolescence issues in systems and products is the mismatch of the system and the components or parts lifecycles" (Zolghadri, Addouche, Boissie, & Richard, 2018).

Some principal reasons for obsolescence are:

- technology advancements (Merola, 2006) – new products appear replacing old ones,
- lack of support from vendors (Merola, 2006) – the organization is forced to modify their product to obtain the necessary updates,
- merger and acquisition of a business (Bradley & Dawson, 1998) – the acquired organization may have to change its existing system, if it is not compatible with the other system used in the acquiring organization, and/or
- incompatible products or rules.

According to (EDSTAR, 2016), the objective of obsolescence management is to ensure that obsolescence is managed as an integral part of design, development, production and in-service support in order to minimize cost and detrimental impact throughout the product life cycle. To deal with obsolescence, (SD-22, 2016) defines a process to follow: prepare, identify, assess, analyze, and implement. During the preparation phase, it is required to "develop the obsolescence strategic underpinnings and a management plan". Then it is necessary to "identify items with immediate or near-term obsolescence issues". Identification and prioritization of items most at obsolescence risk are then performed during the assessment phase. Accordingly, "a set of potential resolutions for the critical items" has to be established under cost-effectiveness constraints. Finally, the solutions have to be implemented.

This paper focuses on the identification and assessment phases and suggests an approach that allows the prioritization of obsolescence issues.

2.2. Obsolescence classification

As mentioned before, obsolescence issues may be distinguished based on a voluntary (planned, see (Kreziak et al., 2017) or (Bulow, 1986)) or involuntarily action of a company. Some obsolescence classifications use the criterion "reason or origin". For instance, (Bartels et al., 2012) define four classes:

- logistical – inability to procure,
- functional – the current product's function, performance, or reliability becomes obsolete,
- technological – advancement, and
- functionality improvement dominated obsolescence – generated to remain competitive in the market.

Another classification was proposed by (Wilkinson, 2015) who found two sources of obsolescence issues: (1) supply side, and (2) demand-side and regulation-caused. Moreover, they suggest an obsolescence fishbone diagram, proposed for avionics, which considers four sources of obsolescence: Software design (airspace requirements, commercial off-the-shelve software), Systems design (Airspace requirements, assurance standards, Regulations), Hardware manufacturing and repair (process, plant, components/sub-assemblies/materials, environmental legislations, component manufacturers), and Design tools (application, platforms and operating systems).

Finally, (SD-22, 2016) proposes a distinction based on the types of impacted items which are subdivided into 1) Software, 2) Hardware-electronic, and 3) Hardware – Materials and Structural, Mechanical, and Electrical (MaSME) items. The hardware-electronic items may become obsolete for example because of low demand, demand for new technologies, or loss of repair support expertise. Software issues are due to newer versions of the software, support termination or because of mergers and acquisitions. Hardware-MaSME obsolescence issues may be due to regulations on hazardous materials, suppliers exiting business, or unavailable tooling. In our proposal, we rely on the classification proposed by (Bartels et al., 2012).

2.3. Systems Engineering

Systems Engineering has its roots in the middle of the 20th century, when some strategic projects were being realized, in particular in defense, aeronautics and space. Several well-known standards, such as ISO/IEC/IEEE 15288 (ISO/IEC/IEEE, 2015), IEEE 1220 (IEEE, 2005) or ANSI/EIA 632 (ANSI/EIA, 1998), or guides of best practices, such as the INCOSE Systems Engineering Handbook, 4th Edition (Walden, Roedler, Forsberg, Hamelin, & Shortell, 2015) or the Guide to the Systems Engineering Body of Knowledge (SEBoK, 2015), describe today's state of the art in systems engineering. They define systems engineering as an "interdisciplinary approach and means to enable the realization of successful systems". They address the system lifecycle phases, focusing on defining customer needs and required functionalities early in the development cycle, documenting requirements, then proceeding with design synthesis and system validation while considering the complete problem: Operations, Performance, Test, Manufacturing, Cost & Schedule, Training & Support, Disposal.

Current systems engineering approaches define how the different lifecycle stages are sequenced, including different models based on linear (e.g. waterfall model, V-model), iterative (e.g. spiral model) or evolutionary approaches (e.g. set-based approaches), assuming that customer requirements are fixed throughout the system's lifecycle.

Model-Based Systems Engineering (MBSE) is a successful approach to support system requirement, design, analysis, verification and validation activities, beginning in the conceptual design phase and continuing throughout development and later life cycle phases (Kaslow, Ayres, Cahill, & Hart, 2018). Models are used to represent the systems and enable to better master the design and the verification for complex systems (Hick, Bajzek, & Faustmann, 2019)

Several languages are used for MBSE. Supported by the Object Management Group since 2006, SysML (ISO/IEC, 2017) is commonly used in systems engineering to analyse, model and design systems. It is structured around different kinds of diagrams, describing the different aspects of a system, allowing multiple views of a system.

However, there is no standard method associated with SysML. Methods have to be defined to make the use of the diagrams explicit, and to express a dedicated methodology conforming to the approach deployed. Among others, ARCADIA (Roques, 2017) is a systems engineering methodology developed by Thales; it has been first developed for Thales' purpose, and is used by the group's various divisions, but it is now more widely spread in numerous companies. ARCADIA's interest is to help structuring a systems engineering approach, while remaining compliant with systems engineering standards. A second interest is that it is supported by the freely available software tool, Capella. For these reasons, we have chosen to build our methodological proposal to address obsolescence by referring to this method and tool. Indeed, ARCADIA, even if it presents some specificities of its own, remains generic in its concepts and its development.

ARCADIA methodology is based on five different levels of analysis, model and design.

Operational Analysis (what system users need to accomplish): to analyze the operational users' needs by identifying the actors and their exchanges.

System Level Analysis (what the system must do for users): to identify the system functions required by its users (e.g. "calculate the optimal route"), under constraint of non-functional requirements.

Logical architecture (how the system will work to meet expectations): to find out the functions to perform by the system and the logical components performing them, integrating the non-functional constraints to be addressed at this level.

Physical Architecture (how the system will be built): to find out the final architecture of the system as it must be realized (implementation and technical choices, etc.).

End-Product Breakdown Structure and Integration Contracts (what is expected of the supplier of each component): to derive from the physical architecture the conditions that each component must meet.

2.4. System requirements

Proven or anticipated obsolescence may lead to non-compliance with system specifications. It can prevent the execution of the expected functionalities (e.g. impossibility to predict the weather), can degrade the quality of the execution of these functionalities (e.g. exceeding the response time), can reduce some characteristics of the quality of the expected functionalities (e.g. unauthorized access to data). It can also lead the system to no longer comply with certain constraints (e.g. restriction on the use of certain materials, Freon for instance). It then becomes to classify the possible consequences of obsolescence. Glinz in (Glinz, 2007) defines four categories of requirements: (i) functional requirements (which relate to a functional concern), (ii) performance requirements, e.g. Timing, Speed, Volume, Throughput, (iii) requirements related to a specific quality (ilities such as usability, security, availability, etc.) of compliance with functional requirements, and (iv) constraints (which limit the space of solutions beyond what is necessary to meet specific functional, performance and data quality requirements), e.g. physical, legal, environmental, etc. We adopt this classification and group the last three categories of requirements into one referred to as non-functional requirements.

3. SYSTEM DEFINITION

This section defines two conceptual models that allow to link unambiguously the obsolescence, its possible

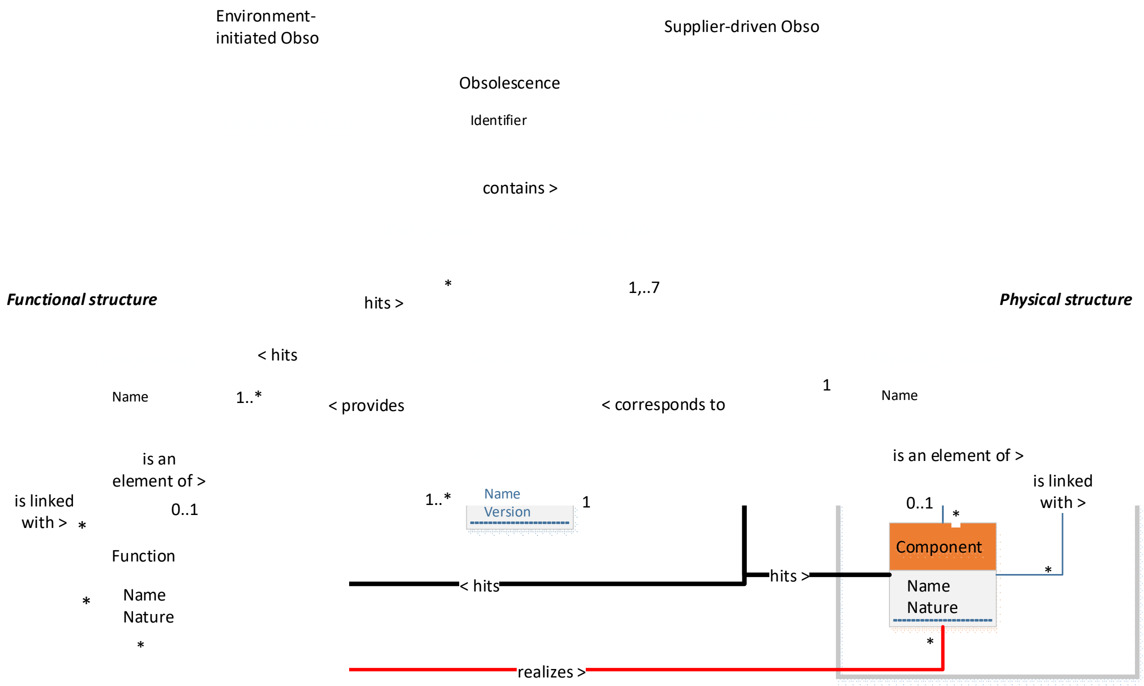


Figure 1 Architectural model of obsolescence, adapted from (Zolghadri & Couffin, 2018)

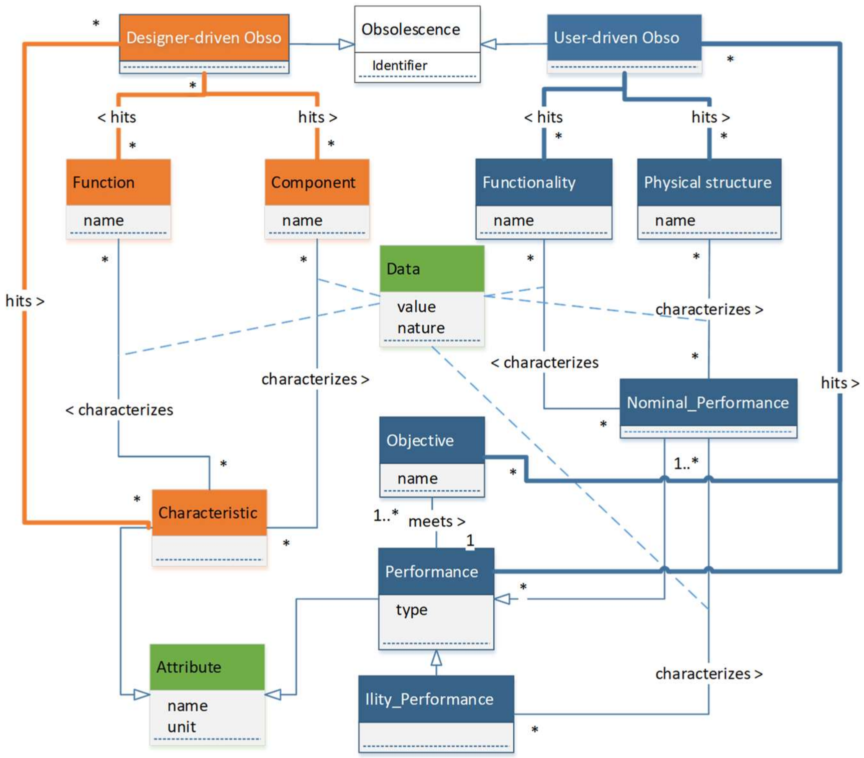


Figure 2 Detailed model of obsolescence, adapted from (Zolghadri & Couffin, 2018)

sources and the system architecture according to (ANSI/EIA 632, 1998).

The architectural model, see Figure 1, demonstrates how obsolescence from various sources is taken into account by system designers, see section 3.1. The detailed model, see Figure 2, describes the objects of obsolescence within a system, namely the components and functions, on the one hand and the functionalities and the external physical structure on which users can trigger obsolescence on the other hand.

3.1. Architecture model

Figure 1 presents the conceptual model of the system architecture related to obsolescence, adapted from (Zolghadri & Couffin, 2018). Hereafter, the underlined words correspond to the class represented in Figure 1.

According to (ANSI/EIA 632, 1998), a system "is the object for which the developer defines the acquirer and other stakeholder requirements". A System is composed of an End System "which performs the operational functions for the system" to which are associated Enabling Systems defined as those which "perform the associated process or non-operational functions of the system".

In a hierarchical decomposition, the End System may be iteratively subdivided into a set of Next Lower Level systems. Considering simultaneously the End system and its Enabling systems allows to underline the fact that obsolescence may hit any of them.

As example, within the System "airplane", any of the components may become obsolete. In the same way, among all the Enabling Systems of the System "airplane", its development or manufacturing systems may become obsolete as well. Obsolescence is a holistic concept affecting the End and Enabling systems.

A system may be hit by obsolescence originated from customers, suppliers, environment, and internal (Soltane et al.,2018) leading to the Obsolescence class with two possible sources:

- External sources: Supplier-driven, user-driven, and environment-initiated obsolescence – imposed by any actor outside the system.
- Internal sources: Designer-driven obsolescence – sourced from the designer of the system.

This distinction also allows understanding a subtle mechanism of obsolescence. The designers are the only persons concerned with the system characteristics able to modify them if necessary. However, the requirements and constraints from external sources have to be analyzed and mapped to internal characteristics changeable by designers.

The architectural model distinguishes then those concepts manipulated by users (blue classes) from those ones which are under the responsibility of designers (orange classes).

The user-driven part of the model represents the system requirements in terms of Functionalities (or services provided by the system) and Physical Structure. The designer-driven classes model the Components and the Functions they have to perform to answer the customer needs and requirements. The classes Component, Function, Functionalities and Physical structure are all the subclasses of an abstract class called Artifact.

In Figure 1, the mapping between the components and functions, or the allocation of functions to components (Roques, 2017) is modelled by the orange link between the Functions and Components. User-driven obsolescence could be generated by the modifications of needs and requirements of the user. Designer-driven obsolescence could be generated by the designer (internal designers or designers of suppliers).

3.2. Detailed conceptual model

The second conceptual model is provided in Figure 2. The abstract class Attribute allows to define an artifact by a set of attributes, each of them identified by a name and a potential measure unit. Performance and Characteristics are its two sub-classes. The performances are used to qualify and quantify the properties of the system observable by the users. Characteristics are the internal properties of the artifact measured and used by designers. A target is associated to each performance as an objective to achieve (to reach, to optimize, or to improve). This is modelled by the class Objective.

Nominal performance defines those properties which qualifies and quantifies either the functionalities (e.g. maximum load) or the physical structure (e.g. weight of an engine) of the artifact. The -ilities_Performance characterizes the behavior of the nominal performance through -ilities (e.g. quality of service requirements, constraints, non-behavioral requirements). The association between the nominal and -ilities performances allows to link a given performance to several secondary performances. This could be for instance the meantime between two failures (MTTF) of 200h, or the service rate of 99%.

A user-driven obsolescence may be linked to the awaited functionality (suppression, adding, modifying), the physical structure (suppression, adding, modifying an artifact), objective (a new performance), or even the performance (improvement of execution speed). These possibilities are represented in Figure 2 through the links “hits” from the User-driven obsolescence to Functionality, Physical structure, Performance and Objective.

A designer-driven obsolescence may concern either a function (suppression, addition, replacement), a component (suppression, addition, replacement), or any characteristic (increasing or reduction for continuous characteristics such as weight, or change to another value for discrete characteristics). These types of obsolescence are represented by orange links named “hits” on Figure 2.

4. TOOLS FOR THE PREPAREATION AND ASSESSMENT PHASES

This section details our proposal of tools for health analysis of the system-of-interest facing obsolescence.

Note that two prerequisites must be met in order to be able to use these tools. It is first assumed that, like the SD-22 proposal, an Obsolescence Management Team is formed representing stakeholders (such as marketing and sales, manufacturing, support and maintenance, supply chain management actors).

In addition, we assume that the main system models (Operational, System, Logical and Physical Architecture diagrams) using the ARCADIA methodology exist. In fact, the determination of possible propagations of the consequences of obsolescence requires a precise mapping of dependencies within the system, because any dependency is in fact a propagation channel between dependent entities. Accordingly, the use of models defined by system engineering in general, and of the ARCADIA methodology in particular, allows to know the interdependencies between components (C-C), functions (F-F), but also the function-component assignments (C-F). These interdependencies can be partly identified in the first levels of the ARCADIA methodology, however, we rely on the physical architecture (obtained in the last modeling step, cf. figure 6) which defines these dependencies precisely.

Another advantage of using systems engineering models (here ARCADIA) is that components (or functions) that belong to different levels of detail can be studied simultaneously. The use of Capella allows the extraction of dependencies according to different granularities in a matrix form, usable later on for algorithmic processing.

Since systems are very often composed of a large number of components and modules, it is impossible to put them all under obsolescence monitoring. It is therefore necessary to carry out a first screening to identify those most at risk. The screening analysis can be done considering criteria such as those cited by (SD-22, 2016): "safety, mission criticality, Diminishing Manufacturing Sources and Material Shortages (DMSMS) - related cost, existing problems, life-cycle phase, sustainment strategy (reflects the maintenance or support concept of operations for that system), availability of data". The screening will identify an initial short-list of critical components to monitor. Using the system architecture, build upon ARCADIA, it is then possible to extract the initial short-list of critical functions performed, totally or partially, by these critical components. The critical functions can be also found by analyzing the functional chains defined in ARCADIA. A functional chain represents a sequence of functions whose fulfilment allow the achievement of an operational capability of the system.

The initial critical components and functions are the two main inputs of the first tool called House of Obsolescence (HOO). Section 4.1 defines the set of the 3 HOOs to use. Together, they allow to map externally sourced obsolescence issues to the system architecture critical components and functions. The use of the 3HOOs may also point out at some other hidden components and functions. The output of the application of HOOs is a consolidated list of components and functions to monitor. The second tool to use targets at prioritization of obsolescence issues. It is called System Obsolescence Exposure Analysis and is presented in Section 4.2.

4.1. House Of Obsolescence, HOO

The House of Obsolescence, a concept inspired by the well-known House of Quality (Pyzdek & Keller, 2014), maps the environmental, user-driven and supplier-driven obsolescence issues to the consolidated short-list of critical functions and components. The approach is to analyze (i) the requirements coming from the environment and users, and (ii) the characteristics of the provided supplies by the suppliers, and to map them to the consolidated critical components and functions, see Figure 3.

The roof of the HOOs shows the necessary dependencies within the system, extracted from its physical architecture constructed in Capella. It allows to chain the consequences of externally sourced obsolescence issue to the critical system components and functions. Each column corresponds to a critical

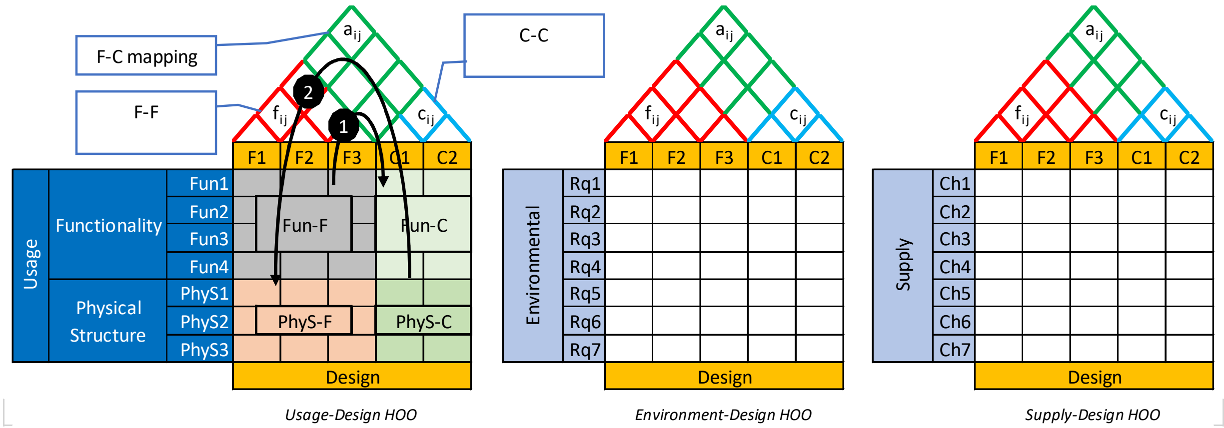


Figure 3 Usage-Design, Environment-Design and Supply-Design Houses Of Obsolescence

function or component. The C-C dependencies are mapped on the top of the components columns and show whether there are any exchanges between every couple of considered components, c_{ij}. Typically, the exchanges of data, energy and movements have to be considered. The idea is that when two components have an exchange between them (e.g. data transfer from a microcontroller to a memory), any change in the sender may affect the receiver. The exchanges define therefore the dependencies between the components. From an obsolescence point of view, the obsolescence issue of the sender could lead to modifications of receiver, and vice versa. These modifications are either "first-order hardware changes" or "first-order software changes", see (SD-22, 2016). c_{ij} can be either symbolic value (high, medium, low), or numerical: (i) Boolean (1: with or 0: without), (ii) natural values (for instance from 0 to 3 using an adopted measure scale) or (iii) real values (from 0 to 1).

The F-F dependencies are mapped on the top of the functions' columns. These dependencies are gathered through the Boolean f_{ij}. The functions dependencies define how the outputs of a function is used by others. For instance, the function "Collect Weather Data" supplies data to the function "Elaborate the Current Situation", Figure 6. Therefore, any changes in the first function may impact the second. These are the functional dependencies which are identified since the System Analysis level of ARCADIA methodology (Roques, 2017).

Finally, the partial system architecture is represented at the very top of the roof of HOO. The system architecture maps the functions to the components. It answers the question of "who does what?". The mapping is valuated through the Boolean value of a_{ij}. For instance, in Figure 6, it can be seen that the function "Acquire Temperature" is performed by the component "Sensor Holder".

It is important to notice that these models, describing the C-C, F-F and F-C dependencies, are a kind of DSM (Design Structural Matrix) and correspond to extractions from the ARCADIA models of the studied system. The latter contain in-depth knowledge of the system described through the different levels of modelling of the ARCADIA method (cf. Section 2.3).

The Usage-Design HOO allows to find out what are the possible functions and components of the system that may be hit by the user-driven obsolescence issue. Four matrices are defined there. The matrix Fun-F shows whether any requirements of the user regarding the functionalities may hit the system functions. The PhyS-C matrix shows the possible impacts of the customer requirements on the system components. Using the partial architecture of the system modelled by the HOO roof, it is possible to cascade the possible impacts on the components due to the changes in the customer functionalities requirements. These impacts are shown in the Fun-C matrix. The process is shown on the roof through the arrow "1". Similarly, the changes in the customer physical structure requirements may impact the system functions; the process represented by the arrow numbered "2".

The dependencies modeled in Fun-C and PhyS-F are obtained through the dependency transitivity; i.e. IF (Y depends on X) and (Z depends on Y) THEN (Z depends on X). Nevertheless, this reasoning process must take into account the following possibilities:

- the obsolescence mitigation solution applied to X may have no impact on Y (mitigation solutions 1, .., 8, see Appendix), or

Temporal horizon: N years					Obsolescence value				
System obsolescence issue		Availability/ Suitability	Obsolescence Classification	Effect	Gravity	Occurrence	Detectability	Obso exposure index	Mitigation possible solutions
a		b	c	d	G	O	D	OEI	m
Components	C1								
	C2								
Functions	F1								
	F2								
	F3								

Obsolescence Exposure Index	Criticality threshold: 100 (or 10 if G=10)		O	Almost impossible				Possible				Highly probable				No doubt, real			
				1				4				7				10			
			D	Easy	Hard	Very hard	Impossible	Easy	Hard	Very hard	Impossible	Easy	Hard	Very hard	Impossible	Easy	Hard	Very hard	Impossible
				1	4	7	10	1	4	7	10	1	4	7	10	1	4	7	10
	G	Tiny	1	1	4	7	10	4	16	28	40	7	28	49	70	10	40	70	100
		Minor	3	3	12	21	30	12	48	84	120	21	84	147	210	30	120	210	300
		Important	6	6	24	42	60	24	96	168	240	42	168	294	420	60	240	420	600
		Major	9	9	36	63	90	36	144	252	360	63	252	441	630	90	360	630	900
		Catastrophic	10	10	40	70	100	40	160	280	400	70	280	490	700	100	400	700	1000

Figure 4 Risk analysis by System Obsolescence Exposure Analysis, and OEI

- the component Y has to be modified (mitigation solutions 9 and 10) and the dependency transitivity process is cancelled on Y, or
- the mitigation solutions implemented on Y do impose modifications on Z. This is the only case where the dependency transitivity is pursued to Z.

Cutting the dependency chaining or decreasing its possible impacts are the key for controlling the obsolescence consequences propagation.

The Environment-Design HOO maps the requirements changes imposed from the environment (due to new regulations for instance) to the Functions and Components of the system.

The Supply-Design HOO finally looks to map the changes in the supply characteristics and the functions and components of the system. This is for instance the case when the "Temperature sensor" has a new operating temperature range which may have impacts on the accuracy of the measured temperature of the function "Acquire Temperature", see Figure 6.

4.2. Risk analysis: System Obsolescence Exposure Analysis

In the context of obsolescence risk analysis, the first parameter to fix is the time horizon H. The purpose is to define the time frame beyond which the risk factor estimations are too uncertain to be usable but also below which the study loses all its meaning. For example, an analysis of the risk of obsolescence for a smartphone cannot be carried out over a too short period of time (a few days since no changes can be expected in such a short period) or over a too long period of time (a few years given the speed at which technological innovations are introduced).

The choice of this horizon H depends on factors such as: the life cycle and the remaining operational time of the system (e.g. iPhone 8 phased out, yet commercially available on Apple Website, October 2019), the life cycle and the remaining operational time of the critical component under consideration (e.g. Apple A11 Bionic introduced on September 2017), and the knowledge available on possible changes in customer needs and requirements (e.g. 5G coming technology), but also the evolution of competing systems and competing technologies. Once this horizon identified, the rest of the analysis can be performed.

Lots of pieces of knowledge were collected applying the 3HOOs and ARCADIA methodology. In order to prioritize for mitigation solutions implementation, it is suggested to perform a System Obsolescence Exposure Analysis (SOEA). It is based on a table inspired from the classical Failure Mode Effects and Criticality Analysis (FMECA), see left side of Figure 4. The critical system functions and components are listed in rows. For each of them, the first analysis is to find out whether the issue is related to suitability or availability, the column "b". Remember that the suitability and availability refer respectively to obsolescence and DMSMS. This is to highlight that the solution to be deployed depends on whether the problem is of the obsolescence type (e.g. technological overrun) or DMSMS (the supplier who has stopped manufacturing the component).

Based on the obsolescence classification presented in section 2.2, the obsolescence class is identified in the column "c".

Each of the obsolescence issues could have impact on the system requirements (see section 2.4), such as functional or non-functional (time, accessibility, adaptability, etc.).

The impacted classes of system requirements are defined and reported in column "d". The following three columns, G, O and D allow to define the "Obsolescence Exposure Index" of critical functions and components based on assessment of Gravity, Occurrence, and Detectability of an obsolescence issue. The following tables defines the meanings of the used scales:

1. Occurrence. It represents how confident are the experts about the occurrence of the obsolescence issue. The discrete value is to be chosen between No doubt, Highly probable, Possible and Almost impossible.
2. Gravity. It measures the severity of the obsolescence issue on the effect (column d) of the system: Catastrophic, ..., Tiny.
3. Detectability. There are various possible ways for a company to discover the obsolescence issue. In some cases, there are documented information (published regulations or any discontinuance disclaims). In this case, the value associated with is 1; i.e. the detection is easy. However, this is not always the case, and the company should deploy efforts to discover it. This is the case when a supplier does not give any information of discontinuance, (10).

The Obsolescence Exposure Index, OEI, is then calculated: OEI = G x O x D.

It seems quite reasonable to use the same method often in exploiting the FMECA for prioritization. This is to define the two situations, see right table in Figure 4:

- Consider the issue if the gravity is 10.
- Consider the issue if the OEI is more than 100. This threshold can be adapted to the study case.

The resulting obsolescence issues are then ordered in the diminishing direction of the OEI.

5. ILLUSTRATIVE CASE: EOLE

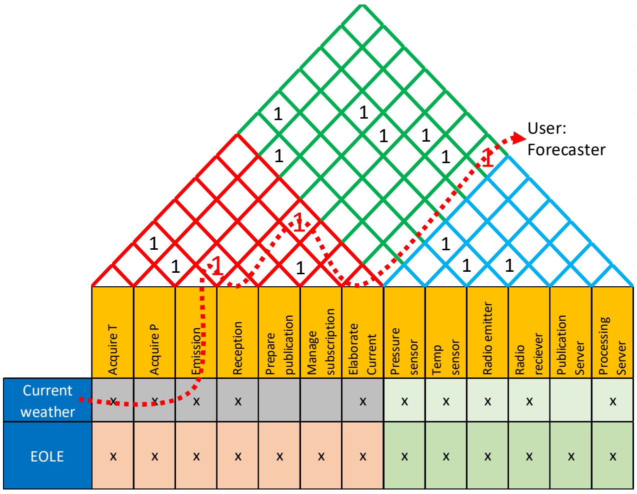
The Environment Observation Link to Earth (EOLE) case, developed by (Roques, 2017) is used to illustrate partially the obsolescence exposure index. EOLE is composed of an acquisition and a ground system. The acquisition system is a sounding balloon launched into the atmosphere in charge of data collection using sensors and data transmission to the ground system. The pressure and temperature are the two main sensors. The ground system is in charge of data acquisition planning, collecting the data from the acquisition system and finally to manage users. The system is modelled using ARCADIA methodology, supported by Capella. All details of the use case can be found in (Roques, 2017). The main goal of the obsolescence study, reported here, is to find out the most critical obsolescence issues.

The ARCADIA application allowed to identify the Sounding balloon and Ground station; the yellow boxes in the middle of the physical architecture, see Figure 6. The Sounding balloon is composed of two modules: Sensor holders and Nano-computer. The Ground station is made of Publication and Processing Servers. The deep blue boxes define the mappings between the functions and the components of the system. The external entities and their exchanges with EOLE are represented in clear blue: Earth atmosphere, Weather operator, etc. They contain their respective functions. The exchanges between the internal and external entities are represented by oriented arcs.

The issue of obsolescence is triggered by a need for improvement identified internally within the company. The screening step is not necessary due to the simplicity of this case study. Only the Usage-design HOO is applied to determine the concerned functionalities and components of the EOLE, Figure 5.

Obsolescence issue rationale. The VHF technology has shortcomings comparing to UHF (Ultra High Frequency 430MHz). The advantage of UHF is the reduction of interference due to a more accessible frequency spectrum.

1) Usage-design HOO. Remind that the goal is to map functionalities to system functions and components.



Temporal horizon: 2 years				Obsolescence value				
System obsolescence issue	*Availability/ Suitability*	*Obsolescence Classification*	*Effect*	*Gravity*	*Occurrence*	*Detectability*	*Obso exposure index*	*Mitigation possible solutions*
a	*b*	*c*	*d*	*G*	*O*	*D*	*OEI*	*m*
Radio Emitter, Receiver	S	Tech	Serv ice	9	7	1	63	2, 6 or 7

Figure 5 Usage-design HOO and SOEA of EOLE

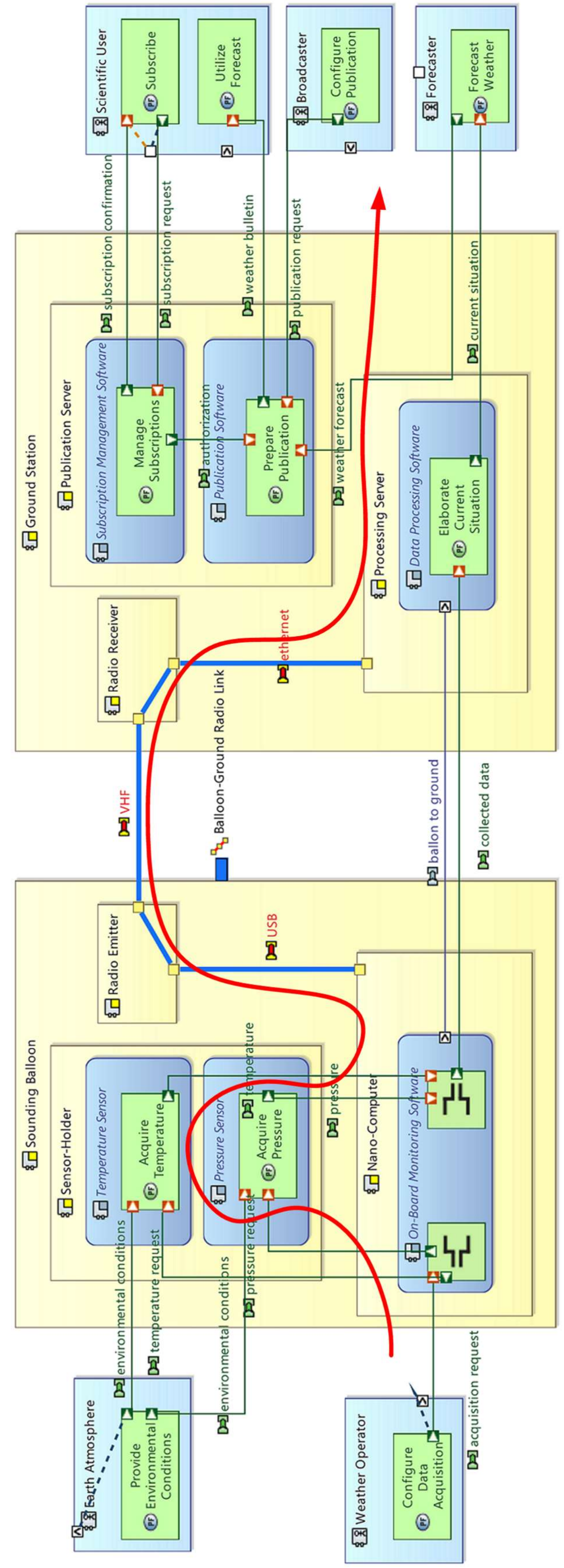


Figure 6 An extract of the Physical Architecture of EOLE

We consider the "Current Weather" functionality consisting of the execution of:

1. Acquire Temperature & Acquire Pressure (by Sounding balloon)
2. Transmission from Sounding Balloon to Ground station (by Radio Emitter et Receiver)
3. Elaborate Current Situation (by Ground Station).

This functionality can be identified by the red line in the ARCADIA model (Figure 6) and its extraction in the HOO roof (Figure 5). The two components, the radio emitter and the radio receiver, are central to this functional chain; the obsolescence mitigation may have a direct impact on them. Therefore, they are then critical and should be monitored. It is therefore necessary to estimate its OEI through the use of the SOEA matrix.

2) SOEA application. Remind that the goal is to assess the criticality of the obsolescence issue, and to brainstorm for mitigation solution. This obsolescence issue is related to the suitability, column b of the SOEA matrix. It corresponds to a technological obsolescence according to (Bartels et al., 2012); column c. The main effect (column d) is the possible message errors due to the interferences. This means the system provided service may be degraded.
The computation of the OEI is obtained based on the expert assessment of gravity, occurrence, and detectability. Suppose that the analysis is performed on 2019 and the horizon of the study is 2 years due to the newer version of EOLE that is predicted to be sold.
G = 9: the obsolescence will degrade the service usability of the whole system.
O = 7: the UHF technology is already available; the obsolescence of VHF is highly probable.
D = 1: the state-of-the-art of radio transmission is easily available; there is no need for any specific effort to detect it.
The OEI of this obsolescence issue is then about 63.

The obsolescence mitigation should be chosen among the possible strategies listed in the Appendix. According to the specificity of the problem, an approved item, a Simple substitute or Complex substitute seem the best solutions to upgrade the system and to mitigate the obsolescence issue. The final choice of the solution to implement requires more technical definition which is out of scope of this example.

6. CONCLUSIONS AND PERSPECTIVES

This paper addressed the fundamentals of obsolescence. It results that obsolescence is a problem that all businesses face or will face. The paper proposed two conceptual models that allow to understand the key mechanisms of obsolescence, and how an external source of obsolescence may be linked to the critical components and functions of the system. Two tools were presented so to support decision-making during the first phases of the obsolescence management process.

The application of the proposed tools allows to lead the team towards a targeted consideration of the system components and functions likely to be impacted by obsolescence. The HOO seeks to guide towards a mapping of the modifications of the external elements to the components and functions of the system that the designers may need to modify. Once these main components and functions have been identified, the use of SOEA should lead the team to estimate the effects of the problem but also the possible solutions. These two tools are therefore the first to be used to delimit the scope of the problems to be solved. The use of systems engineering models is fundamental to the operation of these tools.

However, the use of these models in solving obsolescence problems goes far beyond these tools. In particular, they allow to obtain predictive models (probabilistic graphs) that can be used in the determination of components and functions impacted by obsolescence. The authors are currently working on the exploration of these tools.

REFERENCES

ANSI/EIA. (1998). *Standard for Processes for Engineering a System, ANSI/EIA STD 632*.

Bartels, B., Ermel, U., Sandborn, P., & Pecht, M. G. (2012). *Strategies to the prediction, mitigation and management of product obsolescence* (Vol. 87). John Wiley & Sons.

Bradley, M., & Dawson, R. J. (1998). An analysis of obsolescence risk in IT systems. *Software Quality Journal*, *7*(2), 123–130.

Bulow, J. (1986). An economic theory of planned obsolescence. *Quaterly J. Econ.*, *101*(4), 729–750.

EDSTAR. (2016). *Expert Group 21 : Obsolescence Management Final Report*.

Glinz, M. (2007). On non-functional requirements. *15th IEEE International Requirements Engineering Conference (RE 2007)*, 21–26. IEEE.

Hick, H., Bajzek, M., & Faustmann, C. (2019). Definition of a system model for model-based development. *SN Applied Sciences*, *1*(9), 1074.

IEC 62402. (2019). *IEC 62402, Obsolescence management*.

IEEE. (2005). *IEEE standard for application and management of the systems engineering process, 1220*. IEEE.

ISO/IEC/IEEE. (2015). *Systems and software engineering– System life cycle processes, IEEE STD 15228*. International Organization for Standardization Geneva.

ISO/IEC. (2017). Information technology -- Object management group systems modeling language. Retrieved from https://www.iso.org/standard/65231.html

Kaslow, D., Ayres, B., Cahill, P. T., & Hart, L. (2018). A model-based systems engineering approach for technical measurement with application to a CubeSat. *2018 IEEE Aerospace Conference*, 1–10.

Kreziak, D., Prim-Allaz, I., & Robinot, E. (2017). *Des tiroirs pleins de telephones remplaces : consommateurs et objets a obsolescence perçue. Rapport de recherche du projet COOP, Consommateurs et Objets à Obsolescence Programmée.*

Merola, L. (2006). The COTS software obsolescence threat. *Fifth International Conference on Commercial-off-the-Shelf (COTS)-Based Software Systems (ICCBSS'05)*, 7--pp.

Pyzdek, T., & Keller, P. A. (2014). *The Six Sigma handbook: a complete guide for green belts, black belts, and managers at all levels*. McGraw-Hill Companies.

Roques, P. (2017). *Systems Architecture Modeling with the Arcadia Method: A Practical Guide to Capella*. Elsevier.

SD-22. (2016). *Diminishing Manufacturing Sources and Material Shortages (DMSMS) A Guidebook of Best Practices for Implementing a Robust DMSMS*.

SEBoK. (2015). System Requirements. Retrieved from https://sebokwiki.org/wiki/Guide_to_the_Systems_Engineering_Body_of_Knowledge_(SEBoK)

Shuman, L. (2002). *Cost-benefit analysis tools for avionics parts obsolescence*. Master thesis, Maxwell Air Force Base

Wilkinson, C. (2015). *Obsolescence and life cycle management for avionics*. report DOT/FAA/TC-15/33,

Zheng, Liyu; Terpenny, Janis P.; and Orfi, N. (2014). An ontological model and method for obsolescence resolution and management. *IMSEC*.

Zheng, L., Terpenny, J., & Sandborn, P. (2015). Design refresh planning models for managing obsolescence. *IIE Trans.*, *47*(12), 1407–1423. https://doi.org/10.1080/0740817X.2014.999898

Zolghadri, M., Addouche, S.-A., Boissie, K., & Richard, D. (2018). Obsolescence prediction: a Bayesian model. *Procedia CIRP*, *70*, 392–397.

Zolghadri, M., & Couffin, F. (2018). One-off or batch upgrading of complex products and systems: concepts, data model and analysis framework. *Research in Engineering Design*, *29*(2), 263–284. https://doi.org/10.1007/s00163-017-0268-1

APPENDIX

The obsolescence mitigation methods (SD-22, 2016).

	Resolution	***Definition***
1	No solution required	Existing stock will satisfy future demand.
2	Approved item	The issue is resolved by the use of items already approved on and still in production.
3	End-of-need buy	A sufficient quantity is purchased to sustain the product until its next technology refreshment or the discontinuance of the host assembly.
4	Repair	The issue is resolved by: Repair, Reclamation of items from marginal, out-of-service, or surplus materials, , … to ensure continued support.
5	Extension of produc. or support	The supplier is incentivized to continue providing the obsolete items.
6	Simple substitute	The item is replaced with an existing item that meets all requirements without modification to either the item or its Next-Higher Assembly and requires only minimal qualification.
7	Complex substitute	A replacement item that has different specifications but requires no modification of the source product or the NHA, is researched and validated.
8	Dvp of a new item or source	A replacement product is developed that meets the requirements of the original product without affecting the NHA.
9	Redesign-NHA	The affected item's NHA must be modified. Only the NHA is affected, and the new design will not affect anything at a higher level.
10	Redesign–complex/ system replacement	A major assembly redesign affects assemblies beyond the obsolete item's NHA and may require that higher level assemblies, software, and interfaces be changed.