Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 28 (2015) 197 – 201

www.elsevier.com/locate/procedia

3rd CIRP Global Web Conference

Impact of software obsolescence in defence manufacturing sectors

S. Rajagopal[a,] *, J. A. Erkoyuncu[b], R. Roy[b]

[a]*School of Applied Science, Cranfield University, Cranfield, MK43 0AL, UK*
[b]*School of Applied Science, Cranfield University, Cranfield, MK43 0AL, UK*
* Corresponding author. Tel.: +44-753-921-6931; *E-mail address:* s.rajagopal@cranfield.ac.uk.

Abstract

Software plays an important role in Defence industry. Almost every project in defence manufacturing sectors has got software with various degrees of complexity and dependencies. Whilst various research and studies has been conducted on system obsolescence, and tools developed to cost systems and components obsolescence, no major research has been undertaken to develop a framework to estimate cost of software obsolescence. Software becomes obsolete even when it is completely functional but not useful. The aim of this paper is to understand the current practices in identifying the software obsolescence, types of software obsolescence and costing of software obsolescence.

Selection and peer-review under responsibility of the International Scientific Committee of the "3rd CIRP Global Web Conference" in the person of the Conference Chair Dr. Alessandra Caggiano.

Keywords: Software Obsolescecenc; Obsolescence; System obsolescecne; Cost of software obsolescence; Software maintenance;

1. Introduction

Software is defined as programs, procedures, rules, data and documentation associated with programmable aspects of system hardware and infrastructure [IEC 62402]. In simple terms, software are those set of programs, which will make the functionality of system or hardware more simple and easy to use, and it comes with a manual.

According to NASA (Cost Estimating Handbook):

- 55% of software projects exceed budgets by at least 90%.
- Software projects at large companies are not completed by 91%.
- Of completed projects only 42% of them have original proposed features.
- Historical Cost estimates for NASA projects are under estimated by a factor of at least 2.

2. Software in Defence

Software plays an important role in defence. Almost every project in defence has got software with various degrees of complexity and dependencies. The range of software depends on the different platforms, from major complex weapon system to command and communication system, from mission planning to fighter aircraft. Software is an integral part of every project in various dependencies and complexities. [2]

3. Software Obsolescence

"The only big companies that succeed will be those that obsolete their own products before someone else does" (Bill Gates, 2006). This is true with the defence projects as MoD depends heavily on COTS and vendor's stops support to this software as soon as they deliver this software and moves on to produce different software. As soon as this happens the software becomes obsolete.

Today, the military acquires, test, and fields system of systems mixing COTS and non-COTS products [1]. These projects span for decades and the support and maintenance of these products exceed the vendor's capacity. According to a study conducted by BMT and MoD it was found that for every pound spent in developing the software £2 is spent on the maintenance of software and if it is bespoke software then you are looking at £20.

Selection and peer-review under responsibility of the International Scientific Committee of the "3rd CIRP Global Web Conference" in the person of the Conference Chair Dr. Alessandra Caggiano.
doi:10.1016/j.procir.2015.04.034

Software becomes obsolete when there is a new technological advancement; the software functionality is not required or other market factors. Vendors stop supporting when they release the new version of the same programme. Software becomes obsolete even when it is completely functional but not useful [1].

Software gets obsolete generally due to one of the three main causes [3]. *1. Functional obsolescence:* If there are changes to the hardware, system or other software in the same system. This will behave like a ripple effect as changes in one software or hardware will affect the next one and so on until it affects the entire system or capabilities. *2. Technological Obsolescence:* This happens when vendor stops supporting the products or unavailability of the software in market etc. *3. Logistical Obsolescence:* This happens when the media or the hard drive for example does not support the software. For example some of the new software will not function in old processors.

It is good to have some classification of the causes of software obsolescence but P Sandborn did not mention the human obsolescence [3] and the skills obsolescence issues. Human obsolescence plays a major role because its human nature to move from one place to another or in this case from one project to another or from one company to another. It will be impossible for a vendor to accommodate these migrations. When the employees move they take their knowledge of the product with them.

Skills obsolescence is another major factors This is a key issue as in defence most of the safety critical aviation software is written in Ada 95 and there is not enough people that can maintain these software and vendors cannot afford to keep them either.

A recent study on software obsolescence came up with a different classification to P Sandborn's classification. [3] This classification looked at skills and human obsolescence as shown in Fig. 1 [4]

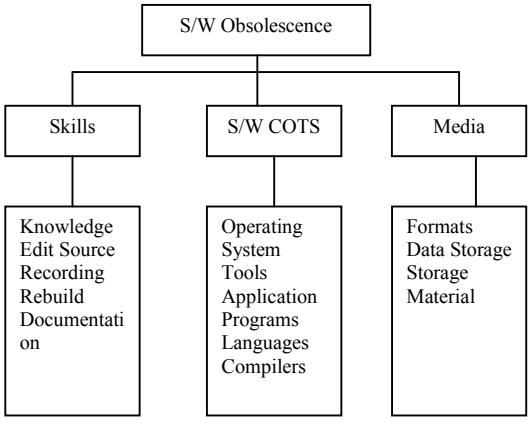

Fig. 1. Types of software obsolescence (Romero Rojo et al. 2009)

The applicable definition of software obsolescence varies depending on the system that use the software and where and how that system is being used. COTS software has got both end of sale dates and end of support dates that can be separated by a long period of time [5].

It is been argued that software will not become obsolete as it is not affected by the degradation and can be easily replicated. They are trying to apply the same reasoning to software obsolescence as mechanical or electrical component obsolescence. When an electronic or mechanical component becomes obsolete and there is no more stock available, the system cannot be maintained according to the original planning. Analogously, the software obsolescence prevents the software from being maintained accordingly [4]. Software obsolescence happens when the original developer and authorised third party cease to provide support with regular updates, upgrades or fixes or due to changes in the target environment, systems, and hardware, which makes software unusable. This study will be based on this definition of software obsolescence.

Software obsolescence plays a major role within the defence and commercial sectors and there is a lot of ground that needs to be covered. Due to this reason and to understand the software obsolescence it was important to understand the software life cycle. For this study software life cycle such as waterfall, spiral iterative/incremental life cycle was mapped across to the CADMID life cycle.

Table 1. Mapping of software life cycle with CADMID project life cycle

CADMID	Waterfall	Iterative/ Incremental	Spiral
Concept	Requirement	Initial Planning Planning Requirement	Determine Objectives
Assessment	Design	Analysis Design	Identify and Resolve Risk
Demonstration	Implementation Testing	Implementation	Development and Testing
Manufacture	Development Verification	Deployment	Prototyping
In-Service	Maintenance	Testing and Evaluation	Release Review
Dispose			

This approach provided clarity on the literature review as this allowed focuses on the individual activities in the different phases of CADMID project life cycle.

Most of the literature available on software obsolescence is in the in-service phase to mitigate the software obsolescence reactively rather than proactively. This literature looks at the mitigating action on what needs to happen when software obsolescence occurs. There is a serious knowledge gap in the scientific, commercial and academic world on the cost