Key Challenges in Managing Software Obsolescence for Industrial Product-Service Systems (IPS2)

F. J. Romero Rojo[1], R. Roy[1], E. Shehab[1], K. Cheruvu[1], I. Blackman[2], G. A. Rumney[3]

[1.] Decision Engineering Centre, Manufacturing Department, School of Applied Sciences, Building 50, Cranfield University, Cranfield, Bedford, MK43 0AL, UK.

{f.romerorojo; r.roy; e.shehab; k.cheruvu}@cranfield.ac.uk

[2.] Component Obsolescence Group, Unit 3, Curo Park, Frogmore, St. Albans, Herts, AL2 2DD, UK.

[3.] Sellafield Ltd, B582, Sellafield, Seascale, Cumbria, CA20 1PG, UK.

Abstract

The defence industry is moving rapidly towards new types of agreement such as availability contracts based on Industrial Product-Service System (IPS2) business models. Obsolescence has become one of the main problems that will impact on a system during its life cycle. Most of the research carried out has been focused on electronic components, neglecting the impact of obsolescence in other areas such as software. This paper presents the concept of software obsolescence and highlights the key challenges and mitigation strategies to managing it, which was identified through a set of semi-structured interviews with experts in defence, aerospace and nuclear industries and literature review in software and obsolescence management. It is observed that there is a lack of understanding of the software obsolescence and the impact that it has on the systems.

Keywords

Software Obsolescence, Skills Obsolescence, Obsolescence Management, Industrial Product-Service Systems.

1 INTRODUCTION

Nowadays, companies are moving from a product-oriented towards a service-based business model in which customers purchase the services provided by products rather than owning the products themselves. Many new projects are started with through life planning principles from the earliest concept stages. Capability and availability based contracts are enabled by Industrial Product Service Systems (IPS2) that have progressively increased in scale and complexity through to major infrastructure projects and large defence projects. In sectors such as the defence, military and civil aerospace, transportation and railways, the life-cycle of an IPS2 can be extended over many decades. Due to the high costs and long life times associated with technology insertion and design refresh, these systems often fall behind the technology wave [1-3]. One of the main problems that definitely these systems will face during their life-time is obsolescence. A part becomes obsolete when the technology that defines it is no longer implemented and hence that component becomes no longer available from stock of own spares or being procurable or produced by its supplier or manufacturer [4-8]. IPS2 is also shifting risks from the customer to the prime contractors and their risk sharing partners. The prime contractors are increasingly expected to take responsibility of managing obsolescence impact and as a result there is strong motivation to study obsolescence and document it for later assessment and review.

Many authors [9-13] agree that the rapid growth of the electronics industry is making electronic components obsolete at a fast pace. This is the reason why most of the research carried out on the area of obsolescence is focused on EEE (electronics, electromechanical and electrical) components. However, the problem of the obsolescence is not restricted to EEE components. There are many other areas of an IPS2 that can become obsolete such as: (Figure 1)

- EEE components
- Mechanical components
- Materials
- Software and media
- Test equipment
- Processes and procedures
- Skills

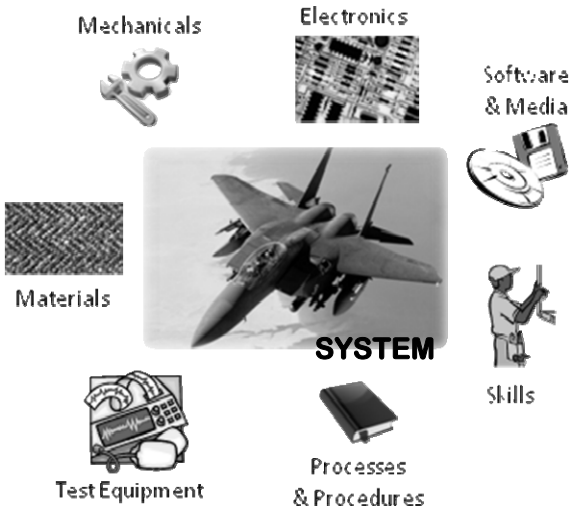

Figure 1: The Holistic View of Obsolescence

These areas have interdependencies among themselves, so it is necessary to consider them following a holistic approach rather than analysing each one independently. The research presented in this paper focuses on the obsolescence issues affecting software, media and skills. The motivation is that these areas have not been explored in depth hitherto but it is acknowledged the big impact that

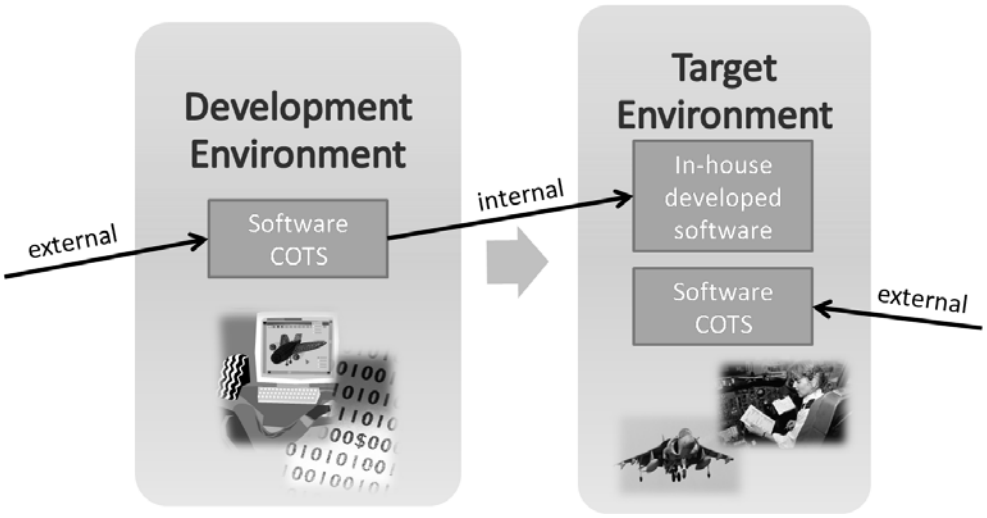

Figure 2: Software Environments

they have during the life cycle of an IPS2 in terms of availability, cost and sustainability.

The first step in this study was the review of existing literature related to obsolescence. It has been identified a growing number of publications in this area for the last 20 years. However, most of them are solely focused on the obsolescence of electronic components, disregarding the software obsolescence problem. During the last four years, a few authors [14-16] have recognised the importance of software obsolescence – especially related to Commercial off-the-shelf (COTS) Software – and hence there is a need for further research in this area to be able to manage and mitigate it properly. The Component Obsolescence Group (COG) at UK has also identified recently this necessity, and they have published a guide that gives an overview about the software obsolescence problem and provides a starting point for managing it [17].

2 RESEARCH METHODOLOGY

2.1 Sources of Information

The fact that this topic is very recent and has not been explored enough yet has been the reason why, apart from the information collected through an exhaustive literature review, it was necessary to capture information directly from industry. General information on the software obsolescence was collected through several COG meetings where obsolescence experts from all over UK gathered. Additionally, a total of eight interviews with experts on software and obsolescence from different organisations across the defence, aerospace and nuclear sectors in UK were carried out. The general perception from industry is that software obsolescence is becoming an important problem mainly because it is ignored in general. Both in US and in UK, the software obsolescence is neither been consistently managed nor mitigated proactively [16]. Contracting for availability has been studied by means of a literature review and the information collected across the major organisations in the UK defence sector [18].

2.2 Approach Adopted

The interviews were carried out using a semi-structured questionnaire in order to capture general understanding of the software obsolescence concept and then analyse in depth the key triggers of software obsolescence, the mitigation strategies that can be applied and the current practice to manage it. The information collected through the first set of interviews was systematically analysed and summarised identifying the key ideas. This summary was presented and validated at the final interview with a key expert from industry.

3 SOFTWARE OBSOLESCENCE

3.1 Software Obsolescence: An Overview

IEC 62402 [19] defines software as “programs, procedures, rules, data, and documentation associated with programmable aspects of systems hardware and infrastructure”. Some people argue that software can not become obsolete because it is not affected by degradation (and hence does not require replacement) and can be easily replicated. Their misconception is to try to apply the same reasoning to software obsolescence as to mechanical or electrical component obsolescence. It is necessary to acknowledge the different nature of the software obsolescence problem. The essence of obsolescence is that it prevents from maintaining and supporting the system. Taking this into account it is possible to identify the commonalities between hardware and software 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.

In the area of computer science, the software development environment (SDE) is the “entire environment (applications, servers, network) that provides comprehensive facilities to computer programmers for software development” [20] and also for software testing. The software target environment (STE) represents the final system in which the software developed in the SDE will be ultimately run. Software obsolescence can happen in both, the development environment and the target environment, as shown in Figure 2, due to external factors (e.g. loss of support from COTS supplier) or internal factors (e.g. loss of skills). From the IPS2 business model point of view, it is important to make this distinction between environments. The reason is that an organisation may have several availability contracts, so it is in charge of supporting different systems. Therefore the organisation will have to manage obsolescence independently for each STE, at the project level, according to the terms agreed in each availability contracts. However, the obsolescence issues that happen

in the SDE have to be managed at the organisation level, and they can have an impact on the supportability of the STE as shown in Figure 2, so this strategy to manage obsolescence needs to be aligned with the availability contracts.

3.2 Types of Software Obsolescence

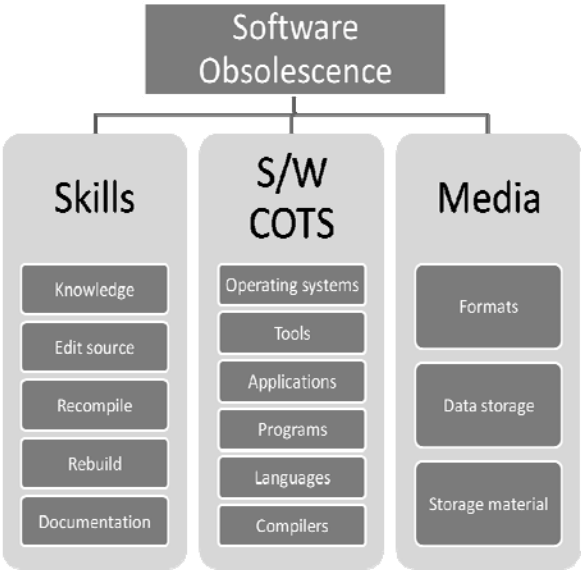

Figure 3: Types of Software Obsolescence

Software obsolescence can happen in three different areas as shown in Figure 3: skills, COTS software and media.

- Skills

It refers to the skills and information necessary to develop, support or modify software developed in-house or by a third party. The loss of required skills is regarded as skills obsolescence and inhibits the maintenance of the software. A common example of this is the difficulty to maintain legacy software (written using legacy programming languages) because the original programmers get retired and the new generations are only trained in new programming languages.

Skills obsolescence hinders the usage of the SDE to support in-house developed software hosted in the target environment.

- COTS Software

COTS Software is regarded as any commercial operating system, program, application, tool, compiler or programming language that is used in the development environment to produce in-house software or that is used directly in the target environment. COTS Software becomes obsolete when its supplier stops supporting it. This is the most common and risky software obsolescence problem because it is difficult to predict when it is going to happen and is usually beyond the control of the customer.

COTS Software can be found in the SDE in the form of tools for software development and in the STE in the form of software components or tools used for configuration at the end-user level.

- Media

It represents the data storage materials and formats used to keep the software information. If they are not properly managed and maintained, there is a risk of losing data and information because they can no longer be accessed from legacy media or legacy formats. Moreover, some forms of media have proved to be less stable and robust than expected.

4 KEY CHALLENGES TO MANAGE SOFTWARE OBSOLESCENCE FOR IPS2

The technology has been evolving rapidly over the last few decades. This has become a major issue for the support of long-life cycle IPS2 such as the availability/capability contracts in the defence and aerospace sector. Moreover, the fact that most of the electronics and software components suppliers have moved from the defence sector to a more profitable commercial market with higher volumes has exacerbated this problem [21]. In the present market, the use of COTS software is widely extended across the defence sector although they have little control over this supply chain [14]. This fact increases the risk of facing obsolescence problems because the defence interest of maintaining long-life systems over several decades clashes with the interest of COTS software providers, which is to reduce the life-cycle of their products, making the COTS software obsolete as a market strategy [14, 16].

The main problem related to software obsolescence is that it is generally ignored within the defence and aerospace sector and usually it is not included in the Obsolescence Management Plan (OMP) or just briefly mentioned without providing a detailed strategy to manage it. The current efforts in dealing with obsolescence are mainly focused on electronic components while software obsolescence is disregarded and not managed at all [16]. Apart from the lack of awareness, there is a lack of tools to assist in the software obsolescence management such as obsolescence monitoring tools (analogous to those used for electronic components such as those supplied by QinetiQ and IHS) which makes difficult the forecast of software obsolescence issues.

It is important to raise awareness of the software obsolescence problem as in most complex systems the cost of dealing with it during the life cycle is comparable to the cost of dealing with hardware obsolescence problems, or even higher [14, 16].

The IPS2 business model is triggering a shift in obsolescence risk, and now prime contractors are more responsible to manage the software obsolescence to guarantee the availability at an affordable price.

4.1 Cost Estimation of Software Obsolescence for IPS2

Nowadays in the defence sector the trend is moving towards contracting IPS2 for availability. The essence of availability contracts is that the suppliers are paid for achieving an availability target for the IPS2 (typically expressed as a percentage, e.g. “available 99.95% of the time”) and not just for the delivery of the product and spares/repairs. This helps to ensure value for money for the customer [18]. The risk of obsolescence in EEE components is progressively being included in this type of contracts. Eventually, the risk of software obsolescence will need to be included explicitly as well. The challenge is to be able to assess this risk at the bidding stage and to estimate the cost related to it for the duration of the contract. At the moment, no organisation is able to make robust cost estimations for software obsolescence.

It is acknowledged that the development of a validated cost model would facilitate the negotiation process for contracting; giving a common understanding to both parties about the risks and cost implications that software obsolescence will have on the system during its life-cycle. It would also increase the level of confidence on the software obsolescence planning through an analysis of Return on Investment (ROI) [16]. It is important that this cost model is developed at system level, so both the software and hardware obsolescence are concurrently

considered, since they are so closely linked [16]. However, there are several reasons that make the development of the cost model very challenging at this stage:

- The data related to software obsolescence problems is frequently spread over different areas such as hardware obsolescence, software defect maintenance, or program schedule slips and additional resource requirements [16]. In most of the organisations there is not a common understanding about the concept of software obsolescence and what falls in and out its scope.
- In general, there is no map of interactions across the system between hardware and software, except for high reliability applications. Typically this is due to inadequate design documentation and configuration management. The lack of this information makes very difficult the prediction of the impact that an obsolescence issue in a component will have on the rest of the system, as this will depend upon the level of interactions and dependencies.
- The organisations are not keeping systematically record of the costs associated with obsolescence events. Historical data is essential to develop cost metric that can be applied to estimate the cost of software obsolescence and include this risk in the contract. It also allows measuring the overall consequences of using different software obsolescence management strategies [16].
- The strategies deployed to manage software obsolescence are usually not included in the OMP. Nevertheless, the software obsolescence management strategy will have a critical impact on the cost.
- Unlike the electronics obsolescence area, there are no monitoring tools available in the market that can assist with the monitoring and forecasting of software obsolescence. It makes it more difficult to develop a management planning and to estimate the number and nature of the obsolescence issues expected during the contracted period.

4.2 Mitigation Strategies

By means of interviews with experts in software and obsolescence across different sectors, where they have to deal with availability contracts for long-term support it has been identified a set of mitigation strategies to reduce the risk of software obsolescence in both the probability and the impact of having an obsolescence issue. The main mitigation strategies for software obsolescence are as follows.

- Loose coupling (Decoupling). The dependencies between hardware and software can be reduced by using standard interfaces and a middleware in the system architecture design. This will mean that an obsolescence issue in a component will be less like to impact the rest of the system, and hence can be easily replaced. This mitigation strategy is especially useful to reduce the interactions between obsolescence issues in electronic components and software.
- Make the development environment flexible enough to support changes in the target environment. This is particularly important for IPS2 as it contributes to its adaptability during the life cycle [22].
- Use of Technology Roadmaps that take into account:
  - Evolution of technology
    - Maturity of technology used
    - Technology stability assessment (identify potential changes in the future)
  - Evolution of suppliers (market)
  - Evolution of customer requirements
- Proactive analysis. To carry out a risk assessment for software obsolescence based on:
  - Impact of the obsolescence issue
  - Probability of becoming obsolete

There are other mitigation strategies that can be applied specifically for each of the software obsolescence areas.

Mitigation Strategies for COTS Software Obsolescence

- Escrow agreements. It is a legal arrangement in which the software source code and the software development environment is placed by the supplier with a third party to be held in trust pending some event, upon which the software will be delivered to the user [17]. This mitigates the obsolescence issue that may happen if the software supplier goes out of business.
- Develop contract clauses to ensure lifetime support (or at least until the next midlife upgrade).
- Keep good relationships with key vendors.

Mitigation Strategies for In-house Developed Software Obsolescence

- Maintain the supporting infrastructure [17].
- Collaboration across different departments to minimise problems of integration/interactions.
- Consider the use of COTS software instead.
- Ensure skills do not become obsolete (apply mitigation strategies for skills obsolescence listed as follows).

Mitigation Strategies for Skills Obsolescence

- Standardisation (Use of “Preferred Technology”) [17]. Minimise the number of programming languages/compilers/software components used across the organisation.
- Maintain people with skills and knowledge required (even after retiring) [17]. So they can continue supporting the system as consultants or they can transfer their skills and knowledge by training other people.
- Use a “Skill register” database to monitor experts and their skills.
- Develop training schemes to preserve skills and knowledge required, proactively identifying potential skills shortages.
- Implement knowledge management systems within the organisation.
- Make sure that the human resources department is aware of potential skills shortages, so new experts can be hired promptly.
- Consider outsourcing the maintenance or development of software. This may be a more cost effective solution than trying to keep the skills in-house for the maintenance or development of software. However, this decision may increase the uncertainty of having an obsolescence issue due to the loss of control over the supplier.

Mitigation Strategies for Media Obsolescence

- Keep structured documentation, formats and data storage systematically, and up to date.
- Plan the upgrades of media, formats and data storage.
- Outsource the media management.

5 CONCLUSIONS

Obsolescence has become one of the main problems that will impact IPS2 offering during its life-cycle so it needs to be considered and managed proactively. Obsolescence can have an impact on different areas of the physical artefact, such as electronic and mechanical components, materials, test equipment, processes and procedures, skills, software and media. The research carried out so far has been mainly focused on the obsolescence of electronic components and many tools and mitigation strategies have been developed to assist in the management of these issues and make it more proactive. However, there is little awareness about the obsolescence problems affecting software, skills and media; although the impact that it may have on the system is comparable to that from electrics obsolescence.

The IPS2 business model is causing a shift in obsolescence risk from the customer to the prime contractor. This is bringing prime contractors to a new scenario where they are more responsible to manage the software obsolescence to guarantee the availability at an affordable price. However, it has been identified a lack of understanding about the concept of software obsolescence and how it can be managed. Usually software obsolescence management is very briefly included, if at all, in the Obsolescence Management Plan (OMP), which is mainly focused on EEE components obsolescence. This lack of understanding about the software obsolescence concept and the lack of supporting tools (i.e. obsolescence monitoring tools) for the prediction of obsolescence issues are the main challenges to manage software obsolescence.

The software obsolescence issues can arise in both the software development environment (SDE) and the software target environment (STE). It is necessary to differentiate them in order to manage the software obsolescence properly. Moreover, software obsolescence can happen in three different areas: skills, COTS software and media. Each area has different characteristics and hence different management strategies should be applied to deal with each one.

A set of general software obsolescence mitigation strategies have been suggested, such as: decoupling, make the development environment more flexible, use of technology roadmaps that take into account the evolution of technology, the suppliers and the customer requirements, and risk assessment for software obsolescence. Additionally, a set of mitigation strategies have been suggested to deal with obsolescence in each of the following areas: COTS software, in-house developed software, skills, and media.

Finally, it has been identified that currently there are no models for the cost estimation of software obsolescence. The main reasons that make this development very challenging are mainly related to the lack of understanding of the problem, the lack of historical information about software obsolescence issues, the lack of software obsolescence management tool and the lack of information about the interactions between hardware and software. Future research on this area should be focused on the development of a model for the cost estimation of software obsolescence, tools for the monitoring, managing and predicting software obsolescence issues. Additionally, it is required to explore the correlation between hardware and software obsolescence due to the high level of interdependencies between them.

6 ACKNOWLEDGMENTS

This project was funded by the EPSRC/Cranfield IMRC and the industrial collaborators: BAE Systems (Insyte), Lockheed Martin (Insys), GE Aviation, UK Ministry of Defence (MoD), Rolls-Royce, Association of Proposal Management Professionals (APMP), Society of British Aerospace Companies (SBAC), Galorath International and Cognition Europe.

Special thanks to the first four industrial collaborators aforementioned, MBDA, SELEX Galileo, Sellafield Ltd. and the Component Obsolescence Group (COG) for contributing actively, sharing information and data.

7 REFERENCES

[1] Sandborn, P., Mauro, F. and Knox, R., 2007, A data mining based approach to electronic part obsolescence forecasting, IEEE Transactions on Components and Packaging Technologies, 30; 3; 397-401.

[2] Singh, P. and Sandborn, P., 2006, Obsolescence Driven Design Refresh Planning for Sustainment-Dominated Systems, The Engineering Economist, 51; 2; 115-139.

[3] Madisetti, V., Jung, Y., Khan, M., Kim, J. and Finnessy, T., 2000. On upgrading legacy electronics systems: methodology, enabling technologies and tools, VHDL International Users Forum Fall Workshop, 2000. Proceedings, Orlando, FL, USA, 18-20 Oct.: 7-14.

[4] Singh, P., Sandborn, P., Lorenson, D. and Geiser, T., 2002, Determining Optimum Redesign Plans for Avionics Based on Electronic Part Obsolescence Forecasts, Proceedings World Aviation Congress, November 2002, Phoenix, AZ, SAE International.

[5] Meyer, A., Pretorius, L., Pretorius, J. H. C., 2004. A model to manage electronic component obsolescence for complex or long life systems. Engineering Management Conference, 2004 Proceedings IEEE International, 3; 1303-1309.

[6] Pecht, M. G. and Das, D., 2000. Electronic part life cycle. IEEE Transactions on Components and Packaging Technologies, 23; 1; 190-192.

[7] Sandborn, P., 2007a. Designing for Technology Obsolescence Management. Proceedings of the 2007 Industrial Engineering Research Conference, 19-23 May 2007, Nashville, TN.

[8] Solomon, R., Sandborn, P. and Pecht, M., 2000. Electronic part life cycle concepts and obsolescence forecasting. Components and Packaging Technologies, IEEE Transactions on [see also Components, Packaging and Manufacturing Technology, Part A: Packaging Technologies, IEEE Transactions on], 23; 4; 707-717.

[9] Feldman, K. and Sandborn, P., 2007. Integrating Technology Obsolescence Considerations into Product Design Planning. Proceedings of the ASME 2007 International Design Engineering Conferences & Computers and Information in Engineering Conference, Sept. 2007, Las Vegas, NV.

[10] Howard, M. A., 2002. Component obsolescence – It's not just for electronics anymore. Proc.FAA/DoD/NASA Aging Aircraft Conference, San Francisco, CA, 16-19 Sept.

[11] Craig, R. W., Co, B. and St Louis, M. O., 2002. A methodology for addressing support equipment obsolescence. Aerospace and Electronic Systems Magazine, IEEE, 17; 5; 20-25.

[12] Frank, B. and Morgan, R. L., 2007. Obsolescence mitigation and management of electronics test equipment. Autotestcon, 2007 IEEE: 527-536.

[13] Mont, O., 2004. Product-service systems: Panacea or myth (Doctoral Dissertation thesis), The International Institute for Industrial Environmental Economics (IIIEE), Lund University: Lund, Sweden.

[14] Sandborn, P., 2007b. Software Obsolescence - Complicating the Part and Technology Obsolescence Management Problem. IEEE Transactions on Components and Packaging Technologies, 30; 4; 886-888.

[15] Sandborn, P. and Plunkett, G., 2006. The Other Half of the DMSMS Problem - Software Obsolescence. DMSMS Knowledge Sharing Portal Newsletter, 4; 4; 3-11.

[16] Merola, L., 2006, The COTS software obsolescence threat, Fifth International Conference on Commercial-off-the-Shelf (COTS)-Based Software Systems, 2006. San Diego, CA.

[17] Rumney, G. A., 2007, The Software Obsolescence Minefield: A Guide for Supporting Software and Electronic Information, 1st ed., COG International Ltd., UK.

[18] Romero, F. J., Roy, R., Shehab, E. and Wardle, P., 2009. Obsolescence Challenges for Product-Service Systems in Aerospace and defence Industry. The 1st CIRP Industrial Product-Service Systems (IPS2) Conference, 1st-2nd April 2009, Cranfield University, UK.

[19] IEC 62402:2007 - Obsolescence Management - Application Guide.

[20] Software Development Environment, available at: http://en.wikipedia.org/wiki/Software_development_environment (last accessed on 5/11/09).

[21] Shuman, L. L. (2002), Cost-Benefit Analysis Tools for Avionics Parts Obsolescence, Research Report AU/ACSC/02-103/2002-04, Air Command and Staff College, Air University, Maxwell Air Force Base, Alabama.

[22] Roy, R. and Cheruvu, K.S., 2009, A Competitive Framework for Industrial Product-Service Systems. Int. J. Internet Manufacturing and Services.