

Obsolescence Risk Assessment and Management Approach for Software

Ted F Bowlds, PhD Candidate

tbowlds@gwmail.gwu.edu, 703-507-3034, abstract #18860

Department of Engineering Management
and Systems Engineering
School of Engineering and Applied
Science

The degree of obsolescence has been called the “dark side of innovation” in that the speed of innovation accelerates parts obsolescence.[1]

Outline

- *Current System Obsolescence Approach*
- *Problem Statement*
- *Software Obsolescence Research to Date*
- *Hypothesis*
- *Software Obsolescence Definition*
 - *Component (hardware) Driven Software Obsolescence*
 - *Software Driven (Increase Capability) Obsolescence*
- *Software Obsolescence Forecasting*
- *Hardware--Software Obsolescence Risk Assessment Framework*
- *Summary*

Current System Obsolescence Approach

- Many system elements affected by obsolescence
 - Electronic components
 - Materials
 - Expertise
 - Processes and techniques
 - Software
 - Test procedures and equipment
- Rapid pace of innovation in electronic components is the most visible obsolescence challenge and where most of the research has been focused
 - Efforts to date focus forecasting component obsolescence
 - Fundamental strategies exist for countering such as a life-time buy

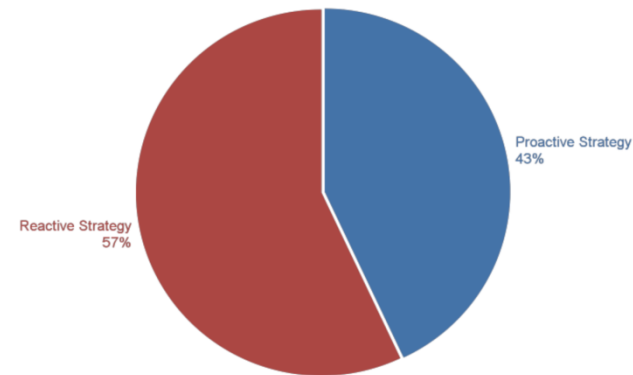
Is Software Obsolescence an Issue?

What Software Obsolescence Management Strategy do you use?

EPSRC Centre for Innovative Manufacturing
Through-life Engineering Services



One results of a 2014 online survey conducted by EPSRC Centre for Innovative Manufacturing on software obsolescence [2]



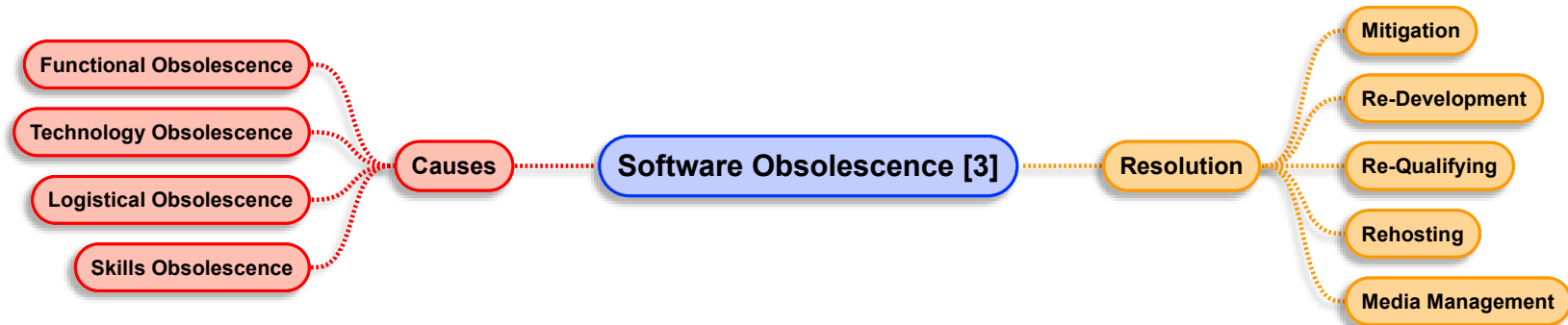
While software code can always be updated to overcome compatibility issues, rewriting is in essence a result of “obsolescence”

Problem Statement

Obsolescence analysis needs to include less researched components such as materials, expertise, processes, test equipment and software, in a more holistic approach. As a step in that direction, can the established research on electronic component obsolescence be used as a methodology for software obsolescence forecasting and thereby enable a more informed system (hardware/software) obsolescence risk assessment and management?

Software Obsolescence Research to Date

Most effort focuses on the causes and resolution of software obsolescence



- Software obsolescence issues tightly coupled to hardware obsolescence
- Hardware & software obsolescence taken as separate issues
- Limited attention paid to software obsolescence as compared to hardware obsolescence

Hypothesis

The interdependency of hardware and software enables a risk model using the established hardware obsolescence forecasting methodologies as the foundation for assessing software obsolescence. Through an integrated hardware-software forecasting strategy, an improved system obsolescence risk assessment can be determined.

Software Obsolescence Definition

Software is generally not considered to go “obsolete.” A decades old software application continues to operate (as an example, Windows XP), provided compatible hardware is available. Or a flip-phone is perfectly capable of making calls...but that’s about all.

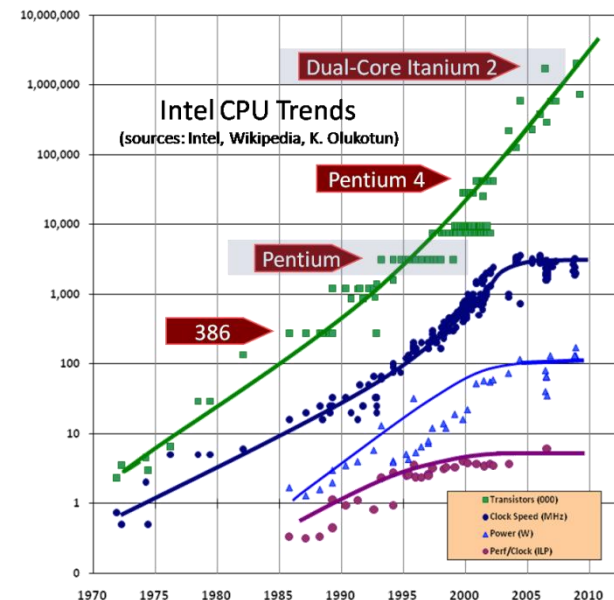
For the purpose of this research, the term “software obsolescence” refers to two incidences:

1. Hardware needed for the software to operate as designed is no longer available to be procured or is no longer supported by the OEM. As such, the software is obsolete driven by lack of compatible hardware.
2. Software enhancements or improved capability cannot be implemented because of hardware limitation. An example would be software designed for 64-bit functionality not effectively running on a 32-bit processor. Or a cyber security upgrade necessary in the software is incompatible with existing hardware.

#1 Component (hardware) Driven Software Obsolescence

Natural evolution of technology adds processing capability

- Components no longer available due to being out of production
- Changes in architecture, 32-bit to 64-bit, can impact legacy software
- Hardware driven incompatibility

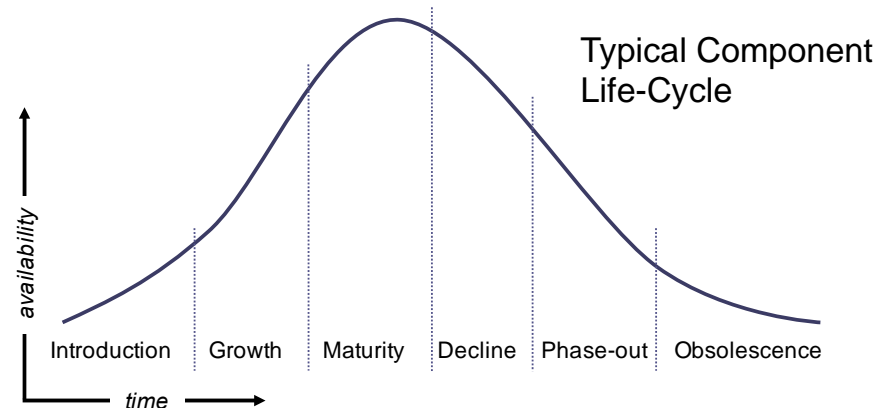


Component Life-Cycle and Obsolescence Forecasting

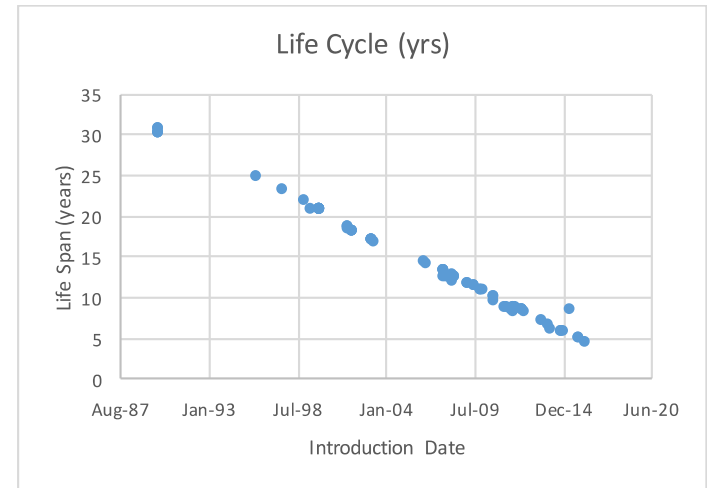
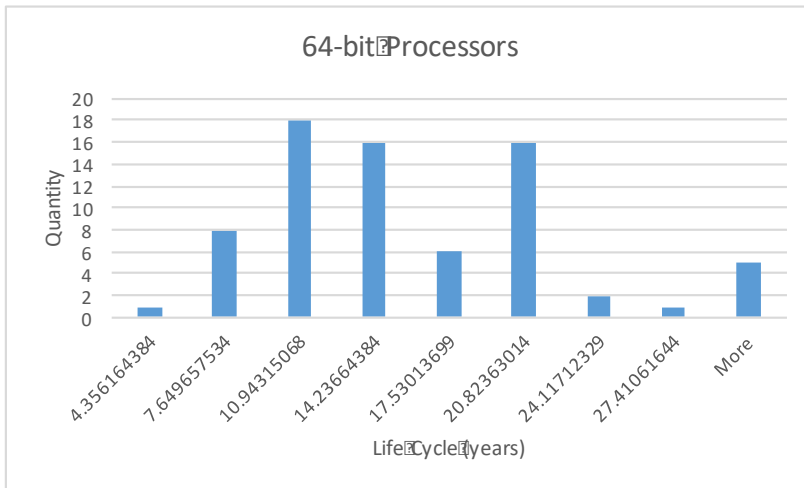
Much research and literature on forecasting component (hardware) obsolescence, clear definition.

Methodologies include [4];

- Sales data forecasting — demand driven forecasting
- Ordinal scale based approaches — technology attributes
- Leading indicator methods — component indicators of change in demand



Sample Intel 64-bit Processor Life-Cycle

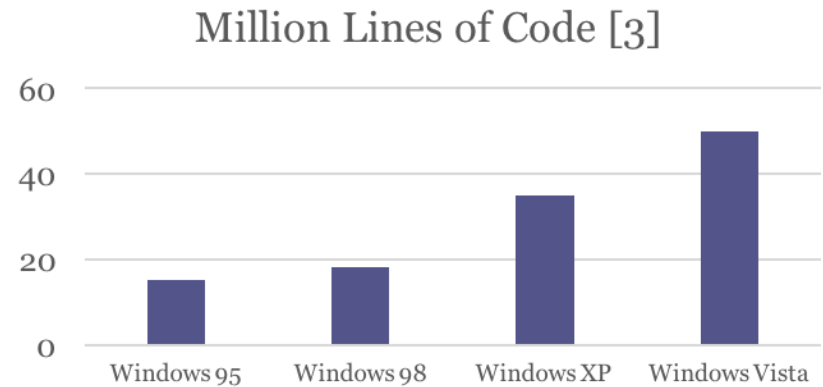


Data extracted from SiliconExpert.com -- provides a project end of life date

#2 Software Driven (Increase Capability) Obsolescence

Software enhancements as new features and capability are added

- Software lines of code (SLOC) an indicator of increased capability
- Not necessarily part of a hardware upgrade; embedded software on an aircraft mission system



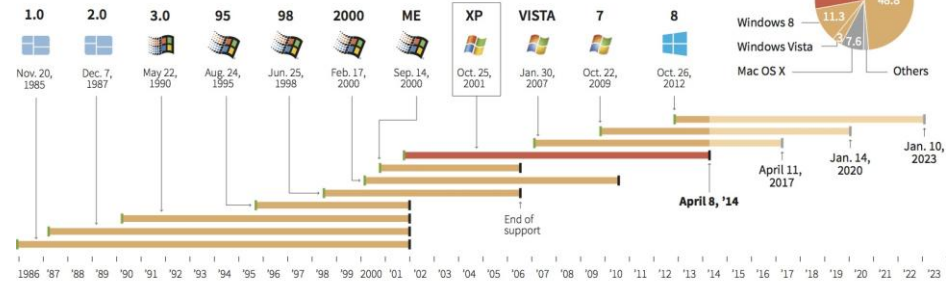
Software Life-Cycle and Obsolescence Forecasting

- Vendors stop supporting a legacy software version as newer capability is introduced
- Non-supported versions do not “stop” working, provided compatible hardware is available
- Legacy software unable to accommodate newer/updated capability or needed cyber security enhancements

Windows Operating Systems

PCs running the Windows XP operating system, which is no longer supported by Microsoft, are vulnerable to a recently disclosed Internet Explorer security bug.

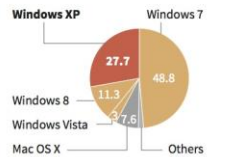
VERSION RELEASE DATE AND LIFECYCLE



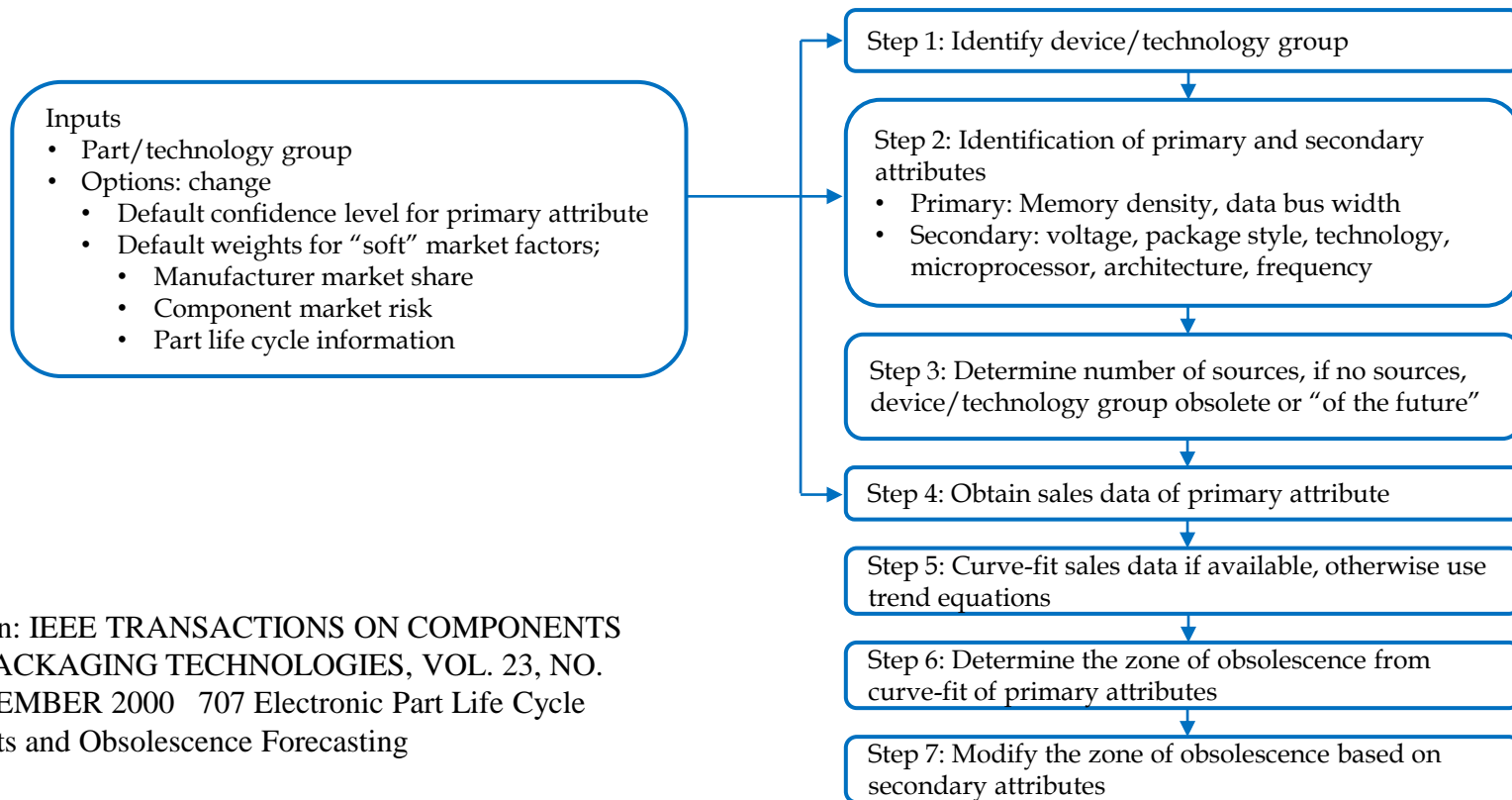
Sources: Reuters, Microsoft; Net Market Share
Staff, F. Chan, 29/04/2014

DESKTOP OPERATING SYSTEMS

March 2014 market share, percent

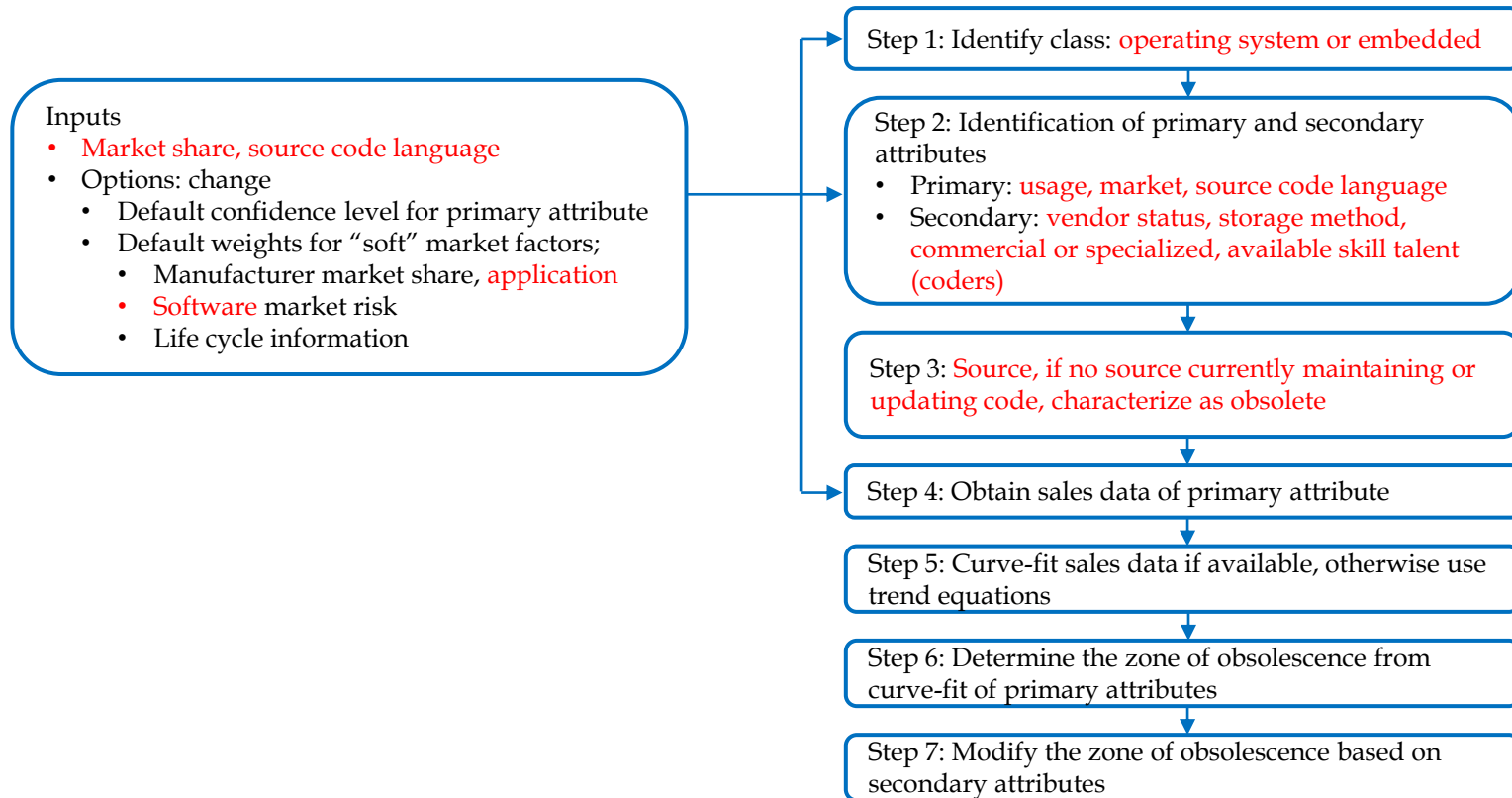


Hardware obsolescence forecasting as the foundation for software obsolescence forecasting [6]

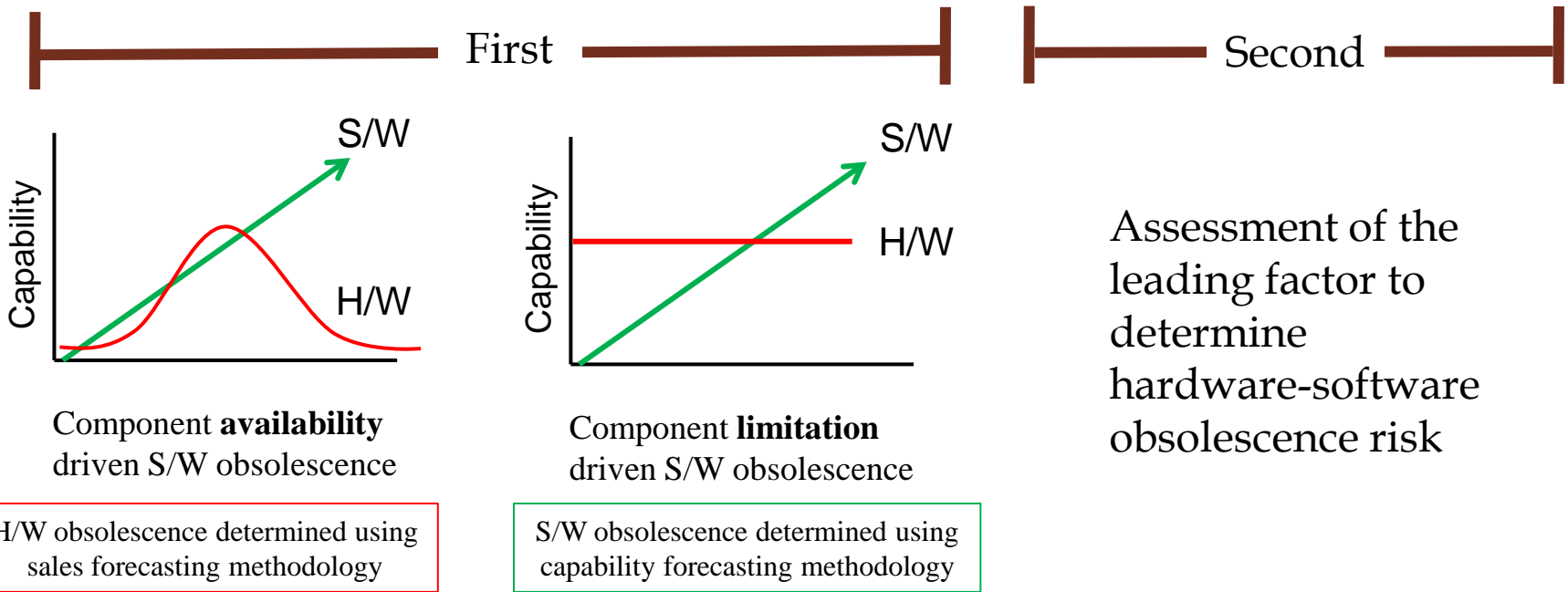


Found in: IEEE TRANSACTIONS ON COMPONENTS AND PACKAGING TECHNOLOGIES, VOL. 23, NO. 4, DECEMBER 2000 707 Electronic Part Life Cycle Concepts and Obsolescence Forecasting

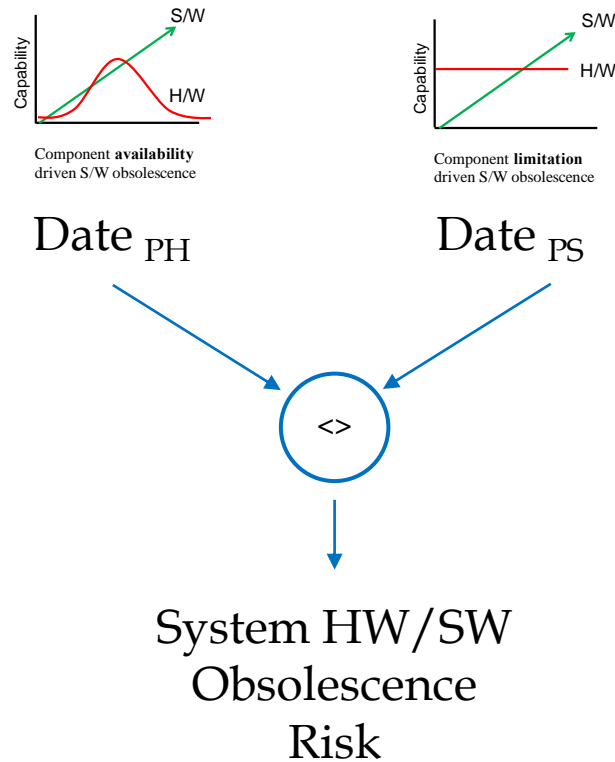
Example software obsolescence forecasting methodology based on hardware forecasting



Hardware--Software Obsolescence Risk Assessment Framework



Draft Model Construct



- Independently determine date of expected obsolescence
 - Hardware
 - Software
- Merge results to arrive at a system (hardware & software) obsolescence risk date

Summary

- From a systems perspective, tight dependency between hardware and software
- Compared to research on hardware obsolescence, little work conducted on software obsolescence
- Hardware obsolescence forecasting methodologies provide methodology for forecasting software obsolescence
- Combined, hardware and software obsolescence forecasting provides an improved risk assessment

Backup

References

1. Plotkin, E., & Porter, K. (2016, April 6). *Four obsolescence management myths that kill defense programs*. Retrieved April 26, 2016, from Military Embedded Systems: <http://mil-embedded.com/articles/four-myths-kill-defense-programs/>
2. Rajagopal, S., Erkoyuncu, J. A., & Roy, R. (n.d.). Software Obsolescence Online Survey Results
3. Sandborn, P. (2007). Software obsolescence-Complicating the part and technology obsolescence management problem. *IEEE Transactions on Components and Packaging Technologies*, 30(4), 886–888
4. Zheng, L., Nelson, R., Terpenney, J., & Sandborn, P. (2012). Ontology-Based Knowledge Representation for Obsolescence Forecasting. *Journal of Computing and Information Science in Engineering*, 13(1), 014501. doi:10.1115/1.4023003
5. Lohr, Steve, & Markoff, John (2006, March 27). *Windows Is So Slow, but Why?* . The New York Times <http://www.nytimes.com/2006/03/27/technology/27soft.html?8hpib>
6. Soloman, R., Sandborn, P. A., & Pecht, M. G. (2000). Electronic Part Life Cycle Concepts and Obsolescence Forecasting. *IEEE TRANSACTIONS ON COMPONENTS AND PACKAGING TECHNOLOGIES*, 23(4), 707–717. doi:10.1109/6144.888857