

Capability Updates, Upgrades, and Modernization

Capability Updates, Upgrades, and Modernization

The printable version is no longer supported and may have rendering errors. Please update your browser bookmarks and please use the default browser print function instead.

Lead Authors: *William Stiffler*, **Contributing Author:** *Brian Wells*

Modernization and upgrades involve changing the product or service to include new functions and interfaces, improve system performance, and/or improve system supportability. The logistic support of a product or service reaches a point in its life where system modernization is required to resolve supportability problems and to reduce operational costs. The *INCOSE Systems Engineering Handbook* (INCOSE 2012) and *Systems Engineering and Analysis* (Blanchard and Fabrycky 2005) both stress the importance of using life cycle costs (LCC) when determining if a product or service should be modernized. Systems can be modernized in the field or returned to a depot or factory for modification.

Design for system modernization and upgrade is an important part of the system engineering process and should be considered as part of the early requirements and design activities. Engineering change proposals (ECPs) are used to initiate updates and modifications to the original system. Product and service upgrades can include new technology insertion, removing old equipment, or adding new equipment. Form, fit, function, and interface (F3I) is an important principle for upgrades where backward compatibility is a requirement.



Contents

Topic Overview

Application to Product Systems

Application to Service Systems

Application to Enterprises

Other Topics

 The Vee Model for Modifications

Practical Considerations

 Application of Commercial-Off-the-Shelf
 Components

References

 Works Cited

 Primary References

 Additional References

Topic Overview

Product and service modernization involves the same systems engineering (SE) processes and principles that are employed during the upfront design, development, integration, and testing. The primary differences between product and service modernization are the various constraints imposed by the existing system architecture, design, and components. Modernizing a legacy system requires a detailed understanding of the product or service prior to making any changes. The constraints and the existence of design and test artifacts make it necessary for the systems engineers performing modernization to tailor the traditional development processes to fit the situation.

Product and service modernization occurs for many reasons, including the following:

1. The system or one of its subsystems is experiencing reduced performance, safety, or reliability.
2. A customer or other stakeholder desires a new capability for the system.
3. Some system components may be experiencing obsolescence, including the lack of spare parts.
4. New uses for the system require modification to add capabilities not built into the originally deployed system.

The first three reasons above are discussed in more

detail in *Applying Systems Engineering to In-Service Systems: Supplementary Guidance to the INCOSE Systems Engineering Handbook*. (INCOSE UK Chapter 2010).

The UK chapter of the INCOSE developed *Applying Systems Engineering to In-Service Systems: Supplementary Guidance to the INCOSE Systems Engineering Handbook* (INCOSE UK Chapter 2010). This guidance document applies to any system for which multiple systems are produced. These systems may be buildings, transmission networks, aircraft, automobiles or military vehicles, trains, naval vessels, and mass transit systems.

Government and military products provide a comprehensive body of knowledge for system modernization and updates. Key references have been developed by the defense industry and can be particular to their needs.

Key factors and questions that must be considered by the systems engineer when making modifications and upgrades to a product or service include:

- type of system (space, air, ground, maritime, and safety critical)
- missions and scenarios of expected operational usage
- policy and legal requirements that are imposed by certain agencies or business markets
- product or service LCCs
- electromagnetic spectrum usage expected, including change in RF emissions
- system original equipment manufacturer (OEM) and key suppliers, and availability of parts and subsystems
- understanding and documenting the functions, interfaces, and performance requirements, including environmental testing and validation
- system integration challenges posed by the prevalence of system-of-systems solutions and corresponding interoperability issues between legacy, modified, and new systems
- amount of regression testing to be performed on the existing software

Key processes and procedures that should be considered during product and service modernization include:

- legislative policy adherence review and certification
- safety critical review
- engineering change management and configuration control
- analysis of alternatives
- warranty and product return process implementation
- availability of manufacturing and supplier sources and products

Application to Product Systems

Product modernization involves understanding and managing a list of product deficiencies, prioritizing change requests, and handling customer issues associated with product usage. The *INCOSE Systems Engineering Handbook* emphasizes the use of Failure Modes, Effects, and Criticality Analysis (FMECA) to understand the root causes of product failures and provide the basis for making any product changes.

Product modernization uses the engineering change management principle of change control boards to review and implement product changes and improvements. The U.S. Office of the Under Secretary of Defense for Acquisition, Technology, and Logistics (OUSD AT&L) provides an online reference for product modernization and the use of an ECP to document planned product or service modernization efforts.

Product modernization and upgrades require the use of system documentation. A key part of the product change process is to change the supporting system documentation functions, interfaces, modes, performance requirements, and limitations. Both INCOSE (2012) and Blanchard and Fabrycky (2005) stress the importance of understanding the intended usage of the product or service documented in the form of a concept of operations.

If system documentation is not available, reverse engineering is required to capture the proper “as is configuration” of the system and to gain understanding of system behavior prior to making any changes. Seacord, Plakosh, and Lewis's *Modernizing Legacy Systems* (2003) explains the importance of documenting the existing architecture of a system, including documenting the software architecture prior to making any changes. Chapter 5 of Seacord, Plakosh, and Lewis provides a framework for understanding and documenting a legacy system (2003). The authors point

out that the product or service software will undergo a transformation during modernization and upgrades. Chapter 5 introduces a horseshoe model that includes functional transformation, code transformation, and architecture transformation (Seacord, Plakosh, and Lewis 2005).

During system verification and validation (after product change), it is important to perform regression testing on the portions of the system that were not modified to confirm that upgrades did not impact the existing functions and behaviors of the system. The degree and amount of regression testing depends on the type of change made to the system and whether the upgrade includes any changes to those functions or interfaces involved with system safety. INCOSE (2012) recommends the use of a requirements verification traceability matrix to assist the systems engineer during regression testing.

It is important to consider changes to the system support environment. Change may require modification or additions to the system test equipment and other support elements such as packaging and transportation.

Some commercial products contain components and subsystems where modernization activities cannot be performed. An example of these types of commercial systems can be seen by looking at consumer electronics, such as radios and computer components. The purchase price of these commercial systems is low enough that upgrades are not economical and are considered cost prohibitive.

Application to Service Systems

Service system modernization may require regulatory changes to allow the use of new technologies and new materials. Service system modernization requires backward compatibility to previous provided service capability during the period of change. Service system modernization also generally spans large geographical areas, requiring a phase-based change and implementation strategy. Transportation systems, such as highways, provide service to many different types of consumers and span large geographical areas. Modernization of transportation systems often requires reverse engineering prior to making changes to understand how traffic monitoring devices such as metering, cameras, and toll tags interface with the rest of the system. The California Department of Transportation's (CDOT's) Systems Engineering

Guidebook for Intelligent Transportation Systems (ITS) (2005) adds reverse engineering to the process steps for system upgrade. In addition, this reference points out the need to maintain system integrity and defines integrity to include the accurate documentation of the system's functional, performance, and physical requirements in the form of requirements, design, and support specifications.

Software modernization is discussed in the *Guide to the Software Engineering Body of Knowledge (SWEBOK)* (Bourque and Fairley, 2014).

Application to Enterprises

Enterprise system modernization must consider the location of the modification and the conditions under which the work will be performed. The largest challenge is implementing the changes while the system remains operational. In these cases, disruption of ongoing operations is a serious risk. For some systems, the transition between the old and new configuration is particularly important and must be carefully planned.

Enterprise system modernization may require coordination of changes across international boundaries. Enterprise modifications normally occur at a lower level of the system hierarchy. Change in requirements at the system level would normally constitute a new system or a new model of a system.

The *INCOSE UK Chapter Supplementary Guidance* (2010) discusses the change to the architecture of the system. In cases where a component is added or changed, this change will constitute a change to the architecture. As an example, the global positioning system (GPS) is an enterprise system implemented by the United States military but used by both commercial and government consumers worldwide. Modernization may involve changes to only a certain segment of the enterprise, such as the ground user segment to reduce size, weight, and power. Modernization may only occur in certain geographical areas of operation. For example, the air transportation system consists of multiple countries and governing bodies dispersed over the entire world. Changes can occur locally or can require coordination and integration world-wide.

Other Topics

The Vee Model for Modifications

Figure 1 below illustrates how the standard Vee model would be applied to a system modification. This Vee model is for the entire system; the key point is that if a modification is being initiated at a lower level of the system hierarchy, the Vee model must be entered at that level as shown in the figure. The figure shows three entry points to the Vee model. As the *INCOSE UK Chapter Supplementary Guidance* (2010) points out, the Vee model may be entered multiple times during the life of the system.

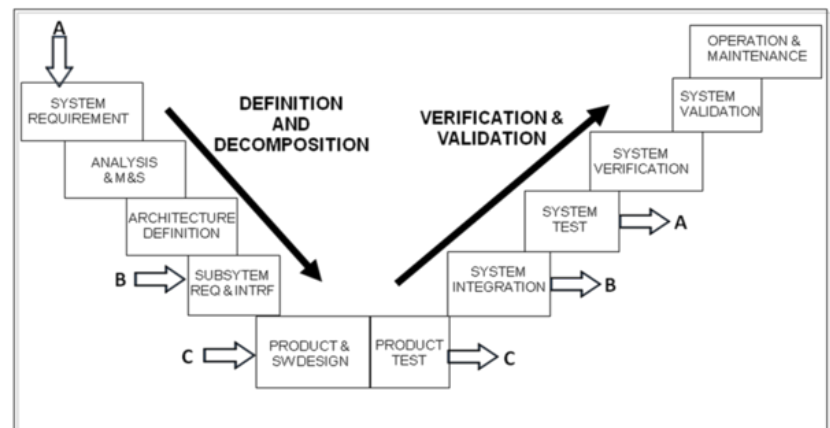


Figure 1. The Vee Model for Modifications at the Three Different Levels. (SEBoK Original)

A change to the system that does not change the system capabilities but does change the requirements and design of a subsystem that may be introduced into the process at point B on the Vee model (see Figure 1). Changes of this type could provide a new subsystem, such as a computer system, that meets the system-level requirements but has differences from the original, which necessitates modifications to the lower-level requirements and design, such as changing disk memory to solid state memory. The process for implementing changes starting at this point has been described by Nguyen (2006). Modification introduced at points B or C (in Figure 1) necessitate flowing the requirements upward through their “parent” requirements to the system-level requirements.

There are many cases where the change to a system needs to be introduced at the lowest levels of the architectural hierarchy; here, the entry point to the process is at point C on the Vee model. These cases are typically related to obsolete parts caused by changes in technology or due to reliability issues with subsystems and parts chosen for the original design. A change at this level should be F3I compatible so that none of the

higher-level requirements are affected. The systems engineer must ensure there is no impact at the higher levels; when this does occur, it must be immediately identified and worked out with the customer and the other stakeholders.

In “Life extension of Civil Infrastructural works - a systems engineering approach” van der Laan (2008) provides a maintenance process that interacts with the system engineering process, represented by the Vee model. His life extension (or modernization) process model includes reverse engineering to obtain the system definition necessary for the modernization process. Consideration of the total lifecycle of the system will result in the capture of all of the records necessary for later upgrade; however, for many reasons, the systems engineer will find that the necessary information has not been captured or maintained.

Practical Considerations

As pointed out by the *INCOSE UK Chapter Supplementary Guidance* (2010) there may be multiple modifications to a system in its lifetime. Often these modifications occur concurrently. This situation requires special attention and there are two methods for managing it. The first is called the “block” method. This means that a group of systems are in the process of being modified simultaneously and will be deployed together as a group at a specific time. This method is meant to ensure that at the end state, all the modifications have been coordinated and integrated so there are no conflicts and no non-compliance issues with the system-level requirements. The second method is called continuous integration and is meant to occur concurrently with the block method. Information management systems provide an example of a commercial system where multiple changes can occur concurrently. The information management system hardware and network modernization will cause the system software to undergo changes. Software release management is used to coordinate the proper timing for the distribution of system software changes to end-users (Michigan Department of Information Technology, 2008).

Application of Commercial-Off-the-Shelf Components

Currently, a prominent consideration is the use of

commercial-off-the-shelf (COTS) components. The application of COTS subsystems, components, and technologies to system life management provides a combination of advantages and risks. The first advantage is the inherent technological advancements that come with COTS components. COTS components continue to evolve toward a higher degree of functional integration. They provide increased functionality, while shrinking in physical size. The other advantage to using COTS components is that they typically have a lower cost.

The risks associated with using COTS during system life management involve component obsolescence and changes to system interfaces. Commercial market forces drive some components to obsolescence within two years or less. Application of COTS requires careful consideration to form factor and interface (physical and electrical).

References

Works Cited

Blanchard, B.S. and W.J. Fabrycky. 2011. *Systems Engineering and Analysis*, 5th ed. Prentice-Hall International series in Industrial and Systems Engineering. Englewood Cliffs, NJ, USA: Prentice-Hall.

Bourque, P. and R.E. Fairley (eds.). 2014. *Guide to the Software Engineering Body of Knowledge (SWEBOK)*. Los Alamitos, CA, USA: IEEE Computer Society. Available at: <http://www.swebok.org>

Caltrans and USDOT. 2005. *Systems Engineering Guidebook for Intelligent Transportation Systems (ITS)*, version 1.1. Sacramento, CA, USA: California Department of Transportation (Caltrans) Division of Research and Innovation and U.S. Department of Transportation (USDOT), SEG for ITS 1.1.

INCOSE. 2012. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, version 3.2.2. San Diego, CA, USA: International Council on Systems Engineering (INCOSE), INCOSE-TP-2003-002-03.2.2.

INCOSE UK Chapter. 2010. *Applying Systems Engineering to In-Service Systems: Supplementary Guidance to the INCOSE Systems Engineering Handbook*, version 3.2, issue 1.0. Foresgate, UK: International Council on Systems Engineering (INCOSE)

UK Chapter: 10, 13, 23.

MDIT. 2008. *System Maintenance Guidebook (SMG)*, version 1.1: A companion to the systems engineering methodology (SEM) of the state unified information technology environment (SUITE). MI, USA: Michigan Department of Information Technology (MDIT), DOE G 200: 38.

Nguyen, L. 2006. "Adapting the Vee Model to Accomplish Systems Engineering on Change Projects." Paper presented at 9th Annual National Defense Industrial Association (NDIA) Systems Engineering Conference, San Diego, CA, USA.

OUSD(AT&L). 2012. "On-line policies, procedures, and planning references." Office of the Under Secretary of Defense for Acquisition, Transportation and Logistics, US Department of Defense (DoD). Accessed on August 30, 2012. Available at: <http://www.acq.osd.mil/log/>

Seacord, R.C., D. Plakosh, and G.A. Lewis. 2003. *Modernizing Legacy Systems: Software Technologies, Engineering Processes, and Business Practices*. Boston, MA, USA: Pearson Education.

van der Laan, J. 2008. "Life extension of Civil Infrastructural works - a systems engineering approach." Proceedings of the 18th annual International Symposium of the International Council on Systems Engineering, Utrecht, the Netherlands.

Primary References

Blanchard, B.S. and W.J. Fabrycky. 2011. *Systems Engineering and Analysis*, 5th ed. Prentice-Hall International series in Industrial and Systems Engineering. Englewood Cliffs, NJ, USA: Prentice-Hall.

Caltrans and USDOT. 2005. *Systems Engineering Guidebook for Intelligent Transportation Systems (ITS)*, version 1.1. Sacramento, CA, USA: California Department of Transportation (Caltrans) Division of Research and Innovation and U.S. Department of Transportation (USDOT), SEG for ITS 1.1.

INCOSE. 2012. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*, version 3.2.2. San Diego, CA, USA: International Council on Systems Engineering (INCOSE), INCOSE-TP-2003-002-03.2.2.

Jackson. 2007. "A Multidisciplinary Framework for Resilience to Disasters and Disruptions." *Journal of Integrated Design and Process Science*. 11(2).

OUSD(AT&L). 2011. "Logistics and Materiel Readiness On-line policies, procedures, and planning references." Arlington, VA, USA: Office of the Under Secretary of Defense for Acquisition, Transportation and Logistics (OUSD(AT&L)). <http://www.acq.osd.mil/log/>.

Seacord, R.C., D. Plakosh, and G.A. Lewis. 2003. *Modernizing Legacy Systems: Software Technologies, Engineering Processes, and Business Practices*. Boston, MA, USA: Pearson Education.

Additional References

Braunstein, A. 2007. "Balancing Hardware End-of-Life Costs and Responsibilities." Westport, CT, USA: Experture Group, ETS 07-12-18.

Casetta, E. 2001. *Transportation Systems Engineering: Theory and methods*. New York, NY, USA: Kluwer Publishers Academic, Springer.

DAU. 2010. "Acquisition community connection (ACC): Where the DoD AT&L workforce meets to share knowledge." In Defense Acquisition University (DAU)/US Department of Defense (DoD). Ft. Belvoir, VA, USA, accessed August 5, 2010. <https://acc.dau.mil/>.

Elliot, T., K. Chen, and R.C. Swanekamp. 1998. "Section 6.5" in *Standard Handbook of Powerplant Engineering*. New York, NY, USA: McGraw Hill.

FAA. 2006. "Section 4.1" in "Systems Engineering Manual." Washington, DC, USA: US Federal Aviation Administration (FAA).

FCC. 2009. "Radio and Television Broadcast Rules." Washington, DC, USA: US Federal Communications Commission (FCC), 47 CFR Part 73, FCC Rule 09-19: 11299-11318.

Finlayson, B. and B. Herdlick. 2008. *Systems Engineering of Deployed Systems*. Baltimore, MD, USA: Johns Hopkins University: 28.

IEC. 2007. *Obsolescence Management - Application Guide*, ed 1.0. Geneva, Switzerland: International Electrotechnical Commission, IEC 62302.

IEEE. 2010. *IEEE Standard Framework for Reliability*

- Prediction of Hardware*. New York, NY, USA: Institute of Electrical and Electronics Engineers (IEEE), IEEE 1413.
- IEEE. 1998. *IEEE Standard Reliability Program for the Development and Production of Electronic Systems and Equipment*. New York, NY, USA: Institute of Electrical and Electronics Engineers (IEEE), IEEE 1332.
- IEEE. 2008. *IEEE Recommended practice on Software Reliability*. New York, NY, USA: Institute of Electrical and Electronics Engineers (IEEE), IEEE 1633.
- IEEE. 2005. *IEEE Standard for Software Configuration Management Plans*. New York, NY, USA: Institute of Electrical and Electronics Engineers (IEEE), IEEE 828.
- INCOSE. 2010. "In-service systems working group." San Diego, CA, USA: International Council on Systems Engineering (INCOSE).
- INCOSE UK Chapter. 2010. *Applying Systems Engineering to In-Service Systems: Supplementary Guidance to the INCOSE Systems Engineering Handbook, version 3.2, issue 1.0*. Foresgate, UK: International Council on Systems Engineering (INCOSE) UK Chapter: 10, 13, 23.
- Institute of Engineers Singapore. 2009. "Systems Engineering Body of Knowledge, provisional," version 2.0. Singapore.
- ISO/IEC. 2003. "Industrial Automation Systems Integration-Integration of Life-Cycle Data for Process Plants including Oil, Gas Production Facilities." Geneva, Switzerland: International Organization for Standardization (ISO)/International Electro technical Commission (IEC).
- Jackson, S. 2007. "A Multidisciplinary Framework for Resilience to Disasters and Disruptions." *Journal of Design and Process Science*. 11(2): 91-108, 110.
- Jackson, S. 1997. *Systems Engineering for Commercial Aircraft*. Surrey, UK: Ashgate Publishing, Ltd.
- Koopman, P. 1999. "Life Cycle Considerations." In Carnegie-Mellon University (CMU). Pittsburgh, PA, USA, accessed August 5, 2010. Available at: http://www.ece.cmu.edu/~koopman/des_s99/life_cycle/index.html.
- Livingston, H. 2010. "GEB1: Diminishing Manufacturing Sources and Material Shortages (DMSMS) Management

Practices.” McClellan, CA, USA: Defense MicroElectronics Activity (DMEA)/U.S. Department of Defense (DoD).

Mays, L. (ed). 2000. "Chapter 3" in *Water Distribution Systems Handbook*. New York, NY, USA: McGraw-Hill Book Company.

MDIT. 2008. *System Maintenance Guidebook (SMG)*, version 1.1: A companion to the systems engineering methodology (SEM) of the state unified information technology environment (SUITE). MI, USA: Michigan Department of Information Technology (MDIT), DOE G 200: 38.

NAS. 2006. *National Airspace System (NAS) System Engineering Manual*, version 3.1 (volumes 1-3). Washington, D.C., USA: Air Traffic Organization (ATO)/U.S. Federal Aviation Administration (FAA), NAS SEM 3.1.

NASA. 2007. *Systems Engineering Handbook*. Washington, DC, USA: National Aeronautics and Space Administration (NASA), NASA/SP-2007-6105, December 2007.

Nguyen, L. 2006. "Adapting the Vee Model to Accomplish Systems Engineering on Change Projects." Paper presented at 9th Annual National Defense Industrial Association (NDIA) Systems Engineering Conference, San Diego, CA, USA.

Reason, J. 1997. *Managing the Risks of Organizational Accident*. Aldershot, UK: Ashgate.

Ryen, E. 2008. *Overview of the Systems Engineering Process*. Bismarck, ND, USA: North Dakota Department of Transportation (NDDOT).

SAE International. 2010. "Standards: Automotive--Maintenance and Aftermarket." Warrendale, PA, USA: Society of Automotive Engineers (SAE) International.

Schafer, D.L. 2003. "Keeping Pace With Technology Advances When Funding Resources Are Diminished." Paper presented at Auto Test Con. IEEE Systems Readiness Technology Conference, Anaheim, CA, USA: 584.

SEI. 2010. "Software Engineering Institute." In Software Engineering Institute (SEI)/Carnegie-Mellon University (CMU). Pittsburgh, PA, USA, accessed August 5, 2010. <http://www.sei.cmu.edu>.

SOLE. 2009. "Applications Divisions." In The International Society of Logistics (SOLE). Hyattsville, MD, USA, accessed August 5, 2010. <http://www.sole.org/appdiv.asp>.

< [Previous Article](#) | [Parent Article](#) | [Next Article](#) >

SEBoK v. 2.6, released 20 May 2022

Retrieved from

"https://www.sebokwiki.org/w/index.php?title=Capability_Updates,_Uppgrades,_and_Modernization&oldid=65006"

This page was last edited on 18 May 2022, at 09:51.