Product Systems Engineering Key Aspects

Product Systems Engineering Key Aspects

> 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 Author: *Ricardo Pineda*

PROJECT PERFORMANCE INTERNATIONAL

The "Product Systems Engineering" knowledge area is graciously sponsored by PPI.

☐

Contents

Acquired Products versus Offered Products

The emphasis for traditional systems engineering (TSE) is in the provisioning of products and related services that meet stakeholder needs and requirements. For acquired products, an acquirer specifies the needs and requirements, selects a supplier for development and provisioning, and then receives the needed products and services. The acquirer, after acceptance, usually owns, operates, and maintains the product and the support systems supplied by the developer. Offered products are provided by suppliers based on opportunities to develop and offer products and services to potential users of the product based on business objectives usually measured in terms of value addition to the stakeholder.

In the provisioning of product systems and related services, the enterprise owning and provisioning the product and services typically makes agreements with other suppliers to also provide elements, methods, and tools that are used during their entire life cycle. The supplying enterprises, in turn, may make further agreements with suppliers in regards to building a supply chain. The complexities of dealing with supply chains must be accounted for with respect to cost, risk, and schedule and thus can have an impact upon product or service maturity. (See articles under the Systems Engineering Organizational Strategy knowledge area (KA) in Part 5.)

Acquired Products

Specific needs for a product or service typically result in some form of an agreement between the acquirer and a supplier as specified in the agreement processes of ISO/IEC 15288 (2015). The acquirer specifies the need and requirements for the properties of the expected product or service and may or may not place specific requirements upon how the supplier plans to organize their life cycle treatment of the product or system.

The degree of formality involved with the agreement varies and is strongly influenced by whether the customer is a government entity or a commercial entity. Government contracts usually incorporate strict specifications and other unique requirements that are rarely found in commercial contracts. Government acquisition agents often specify design characteristics in addition to functional and performance specifications.

Design specifications place constraints on product systems engineering (PSE) by explicitly defining the details of a product's physical characteristics. The government acquirer may also specify how the product is to be designed and developed or how it is to be produced. Government specifications tend to be longer, more detailed, and more complex than functional specifications and much longer than specifications used in a commercial environment.

When contracting with the government or similar enterprises, the PSE must identify disagreements related to the meaning of a particular provision in a contract, and work with contracts to get a written resolution of all ambiguities and issues in the specifications. Failure to do this can lead to legal disputes and government claims of product substitution which can prevent acceptance of the product system and result in financial penalties.

Developing product systems for government customers requires PSE to do a thorough review and perform internal coordination within the enterprise to prevent it from submitting proposals that are non-compliant because the requirements are not fully understood.

Offered Products

Given an opportunity or perceived opportunity, an enterprise may decide to develop and offer products or services to a broader potential marketplace. The properties of the product or service are often determined through surveying and/or forecasting the potential market penetration. The supplier determines the structure and operation of an appropriate life cycle model for achieving the desired results (Pugh 1990).

Supply Chains and Distribution Channels

The supply of products and services to the owner of a product or service that is acquired or offered at various points during the life cycle is vital to success. It is this wider system-of-interest (WSOI) that is the outsourcing holism that must be treated properly in order to provide successful products or services. A portrayal of supply chain structure is provided in Figure 1 below.
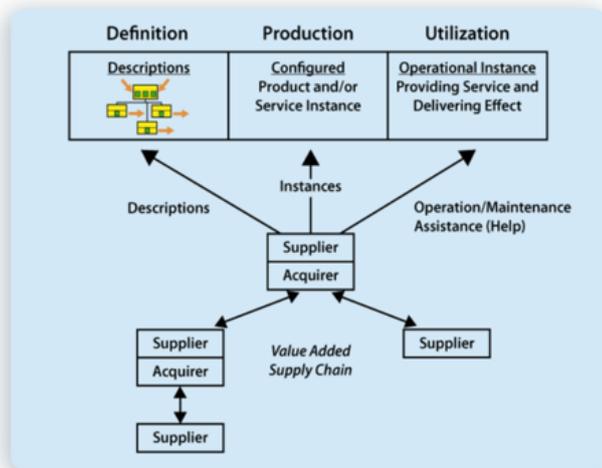
Figure 1. Supply Chain Structure (Lawson 2010). Reprinted with permission of Harold "Bud" Lawson. All other rights are reserved by the copyright owner.

In Figure 1, it is important to note that in an agreement with a supplier, the outsourcing can involve delivering complete system description solutions or portions thereof. For example, a supplier could, given a set of stakeholder requirements developed by the acquirer, develop and supply a system that conforms to the architectural solution. The supplier in turn can be an acquirer of portions of their delivered results by outsourcing to other suppliers.

In regards to production, the outsourcing agreement with a supplier can vary from total production responsibility to merely supplying instances of system elements to be integrated by the acquirer. Once again, these suppliers can be acquirers of portions of their delivery from outsourcing to other suppliers.

In regards to utilization, for non-trivial systems, outsourcing agreements can be made with a supplier to provide for operational services, for example, operating a health care information system. Further agreements with suppliers can involve various forms of logistics aimed at sustaining a system product or service or for supplying assistance in the form of help desks. Once again, suppliers that agree to provide services related to utilization can be acquirers of the services of other suppliers.

Important to all supply chains is the concept that supplying parties contribute some form of added value to the life cycle of a system-of-interest. The proper management of a supply chain system asset is a vital

part of the operations of an enterprise. In fact, the supply chain itself is an enterprise system-of-interest that is composed of acquirers and suppliers as system elements. There is definitely a structure tied together by agreement relationships. Further, the operation of the supply chain results in an emergent behavior. The supply chain system becomes a vital infrastructure asset in the system portfolios of enterprises and forms the basis for extended enterprises.

Similar to a supply chain, the distribution channels for a product system can be a complex web of relationships between the product supplier and various distributors, for example, package delivery companies, warehouses, service depots, wholesale outlets, retail sales establishments, operator training and certification organizations, and so on. The nature of the distribution channels could have a significant impact on the architecture or design of a product system.

PSE may need to include special features in the product design to accommodate for the needs of distribution channel elements, for example, heavy load tie down or lifting brackets, protective shipping packages, retail marketing displays, product brochures, installation manuals, operator certification packages, training materials, and so on. Sometimes it may be necessary to create special versions (or instances) of the product for the training of operators and users for certifying safe or secure operations, for environmental testing and qualification, for product demonstration and user testing, for patent application, for load testing and scalability demonstrations, and for interface fit checking and mass balance certification, to name some examples.

Product Lifecycle and Product Adoption Rates

The life cycle of each product follows the typical incremental development phases shown below (Wasson 2006, 59-65). A particular product to be engineered could be preceded by a previous "model" of that product as shown in the product model life cycle below, and could be superseded later by a newer model of that product. It is worth noting that there is no standard set of life cycle phases. The example below is one of many ways that the phases can be structured.
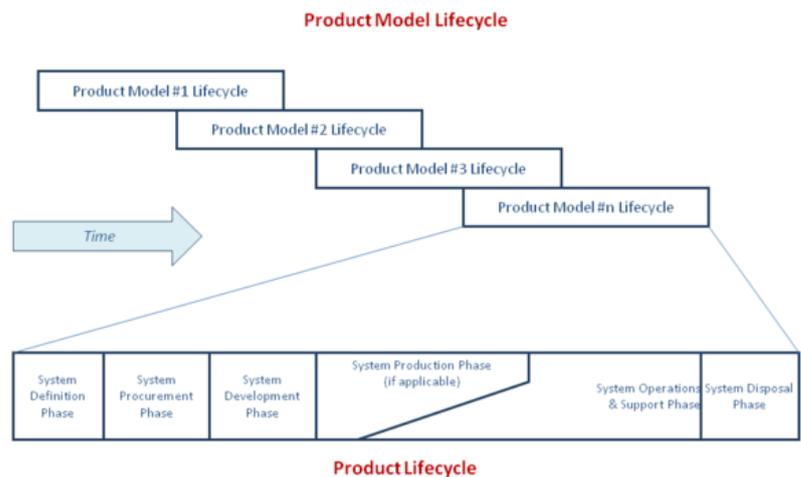
Figure 2. Product Lifecycle as Related to the Product Model Lifecycle (Wasson 2006). Reprinted with permission of John Wiley & Sons, Inc. All other rights are reserved by the copyright owner.

From an industry perspective, managing a product's life cycle involves more than just the engineering aspects:

> *Product lifecycle management (PLM) is the process of managing the entire lifecycle of a product from its conception through design and manufacture to service and disposal. PLM integrates people, data, processes and business systems, and provides a product information backbone for companies and their extended enterprise.* (CIMdata 2012)

There are many PLM tools and services available for facilitating the development and management of complicated product life cycles and especially for product line management (insert link to product line mgmt section here).
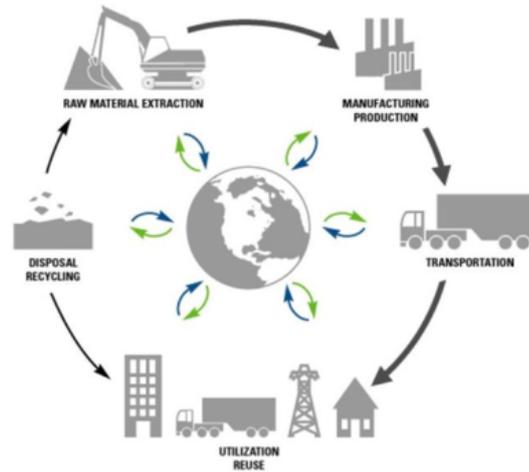
Figure 3. Product Lifecycle from an Industry Perspective.
(Source:
http://commons.wikimedia.org/wiki/File:Product%E2%80%99s_lifecy
cle.jpg#filelinks Accessed February 6, 2012. NIST Programs of the
Manufacturing Engineering Laboratory, Released by US Federal
Government, Public Domain)

The product and product model life cycles are driven by the product adoption rate, illustrated below, that is commonly experienced by most engineered products (Rogers 2003). As products reach market saturation (i.e., on the down slope of the curve below) then there would typically be a new, upgraded version of the product ready for delivery to the marketplace. PSE serves a critical role in determining the best timing for delivery of this new version and the set of features and functions that would be of the greatest value at that time.
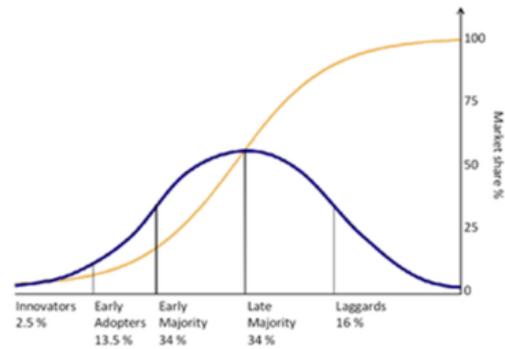
100

75

Market share %

50

25

0

Innovators
2.5 %

Early
Adopters
13.5 %

Early
Majority
34 %

Late
Majority
34 %

Laggards
16 %

Figure 4. Rogers Innovation Adoption Curve. (Source: http://en.wikipedia.org/wiki/File:Diffusionofideas.PNG Accessed February 6, 2012, Released by Tungsten, Public Domain)

Integrated Product Teams and Integrated Product Development

Product systems as discussed throughout this KA mandate the participation of different disciplines for their success during their entire lifecycle from concept to product disposal or retirement. Rapid technology innovations and market pressures in the mid '90s demanded development process (mostly input-output serial) to shorten their development time and development cost, and to improve product quality to remain competitive. For commercial enterprises, the typical development times of 18-24 months to deploy new products into markets of the '90s have in many cases been reduced to 6-12 months and even 3-6 months for the highly competitive leading edge information technology products.

An initial response to these pressures was concurrent engineering. Concurrent engineering is "... a systematic approach to the integrated, concurrent design of products and their related processes, including manufacturing and support to cause developers, from the outset to consider all elements of the product lifecycle from conception through disposal, including quality, cost, schedule and end user requirements." This definition has evolved into the integrated product development (IPD) as more descriptive of this concurrency to describe the continuous integration of the entire product team, including engineering,

manufacturing, test, and support through the life cycle. Later, as the importance of the process was recognized, the terminology was modified to integrated product and process development or IPPD (INCOSE 2012).

The INCOSE *Systems Engineering Handbook* v. 3.2.2 provides a good description of the IPT and IPDT process; the different types of IPDT; the steps in organizing and running an IPDT; good examples of IPDT, particularly for acquired systems; and a good discussion on IPDT pitfalls to avoid. (INCOSE 2012)

IPD/IPPD helps plan, capture, execute, and evaluate programs to help design, test, build, deliver, and support products that satisfy diverse stakeholder requirements. IPD/IPPD outlines the necessary infrastructure needed to deploy, maintain, evaluate and continuously improve processes and tools by aligning people (IPTs) and processes to realize product goals (customer satisfaction). The implementation of Integrated Product Development Processes (IPDP) requires an integrated approach for program planning and generally includes the following: Business Strategy, Program Management and Control, Project Planning, Product Requirements and Architecture Development, Product Design and Development, Production and Deployment, Product Verification and Validation, and Operations and Maintenance Support.

At each development stage, there is a decision gate that helps decide if the IPDP is feasible to enter the next stage of product development. IPD utilizes multi-functional IPTs to optimize the individual product and processes to meet overall cost and performance objectives. IPTs are a cross-functional group of people typically including representatives of all the relevant stakeholders in the project, who are brought together for delivering an integrated product to an external or internal customer using relevant IPDP. The main function of the IPTs is to ensure the business, technical and economical integrity and overall success of the product that is delivered to its eventual customer. IPTs carry out tailored IPDPs and follow relevant SE processes to deliver products that satisfy customer needs, overcomes external constraints, and adheres to the overall program strategy.

In the case of commercial enterprises, product development is tightly coupled with business strategies (short and long term), stakeholder value added measured in terms of return on investments (ROI), market presence/coverage, and other strategies as

defined by the business objectives. Thus, product integration teams include strategic planners, business managers, financial managers, market managers, quality assurance managers, customer representatives, and end-users, as well as other disciplines required for acquired products. Phillips (2001), Annachino (2003), and Morse (2007) provide good discussions on this topic.

Role of Architectures, Requirements, and Standards

The architectural properties of a product system are influenced by the concerns of the various stakeholders as indicated in the ISO/IEC 42010 standard (ISO/IEC 2011). The stakeholders have various views that they express based on their specific perspective. These views are vital in establishing requirements and are inputs to those responsible for defining the functions, structures, and relationships needed to achieve the desired product or service.

A number of stakeholders have been identified in the discussions of product systems. It would be possible to identify a set of important stakeholders based on the life cycle thinking provided by the ISO/IEC 15288 standard (2015), for example, one such set could consist of owners, conceivers, developers, producers, users, and maintainers as discussed by Lawson (2010). As mentioned earlier, these stakeholders should cooperate at all stages of the life cycle in specifying requirements, verifying that the requirements are met, and validating that the products produced provide needed capabilities.

In addition to the two standards that have been identified, there are a variety of standards related to specialty aspects of products, such as safety and security, as well as standards that are applicable for project management and life cycle considerations, such as requirements and quality management.

Role of Modeling, Simulation, Prototyping, and Experimentation

Modeling, simulation, prototyping, and experimentation are techniques that have the purpose of improving stakeholder knowledge and shared understanding about aspects of the system to de-risk system development and operation before heavy commitment of time and funds. Examples of this are found below:

- Understanding future needs: "Warfighting experiments are the heart of the Army's warfighting requirements determination process. Progressive and iterative mixes of high fidelity constructive, virtual, and live simulations using real soldiers and units in relevant, tactically competitive scenarios provide Army leaders with future operational capability insights" (US Army 2012),

- Simulation is used to predict and optimize aspects of system performance for which there are good mathematical or logical models before committing the final physical design, and also to verify and validate the system design in scenarios where physical testing is too difficult, dangerous, or expensive, for example, checking the performance envelope of military systems in a wide range of engagement scenarios where test firing thousands of rounds to get statistically valid data is clearly unaffordable, ensuring that the safety features in a nuclear power station will operate correctly in a wide range of stressing scenarios, etc.,

- Prototyping (physical and virtual) is used in a wide variety of ways to check out aspects of system performance, usability, utility, and to validate models and simulations as part of the iterative process of converging on a final design,

- In a manufacturing context, the first units produced are often "prototypes" intended to make sure the production process is working properly before committing to high rate production, and are often not shipped to end users, but used for intensive testing to qualify the design, and

- Simulation is also used extensively for training and marketing purposes. For training, an accurate model of the human machine interface and representation of the operational context allows operators to do most of their training without putting operational hours on the real system enabling them to learn emergency procedures for combat and accident scenarios in a safe and repeatable environment; for example, airline and military pilots now train mainly on simulators. System simulators of various levels of fidelity are used to familiarize customers and end users with the potential characteristics and benefits of the system, available options and trade-offs, and integration issues early in the development and acquisition

process.

All of these methods use a variety of physical and mathematical representations of the system and its environment so modeling is an enabler for simulation, prototyping, and experimentation.

Increasing Role of Software in Product Functionality

An important trend in commercial products is the increasing importance of software in an increasingly wide range of products. Everything from phones, cameras, cars, test gear, and medical equipment now has essential functionality implemented in software. Software has had an increasing role in providing the desired functionality in many products. The embedding of software in many types of products accounts for increasing portions of product functionality. In tangible products such as cars, software helps improve functionality and usability (cruise control, climate control, etc.). In intangible products such as insurance, software helps in improving operational efficiency, data accessibility, etc.

The movement toward the internet of "things" where sensing and activating functions are incorporated is now starting to permeate. The use of various software products in proving service is also described in the Service Systems Engineering article.

Recent advancements in IT and software have assisted in their increased use in PSE. Although software development is already a very complex field, the role of software in the development and functionality of products is growing larger each day.

There is a need to broaden the horizons of software engineers to think of problem solving not only in software terms, but also using the systems thinking approach. For this purpose, software engineers need to be able to think critically about the problem and also the possible solutions to the problem or opportunity and its implication for business objectives.

Product Integration and Interface Control

Integration is "the set of activities that bring together smaller units of a system into larger units" (Eisner

2008). Products may consist of several systems, subsystems, assemblies, parts, etc., which have to work together as a whole to deliver the offered product's functionalities at specified performance levels in the intended operations environment. Product integration entails not only the working together of hardware and software components, but also the organization, processes, people, facilities, and the resources for the manufacturing, distribution, maintenance, customer support, sales channels, etc. Grady (2010) groups the above information into three fundamental integration components: functional organization, product integration, and process integration.

PSE plays an important role to ensure well defined interfaces, interactions, relationships, information exchange, and processes requirements between product components. These requirements are baseline, documented, traced, verified, and validated for the end-to-end Product integration and to maintain and ensure product offering integrity during its life cycle. The systems engineering hierarchical decomposition level allows requirement definition and allocations at different levels of abstraction to define the building blocks of the product architecture; these building blocks are assigned to integrated product development teams (IPDTs) for detailed design and development. The IPDTs or the systems engineering integration team (SEIT) must interact with all involved players to generate appropriate architectural block specifications at the lower tier of development for a product's architectural configuration and configuration tracking. As the building blocks are put together, interface requirements, information exchange, and interaction and relationships among entities are verified against the baseline. Once a configuration item has been built and tested against the baseline, test and verification at higher levels are conducted to obtain the final product configuration; the final product configuration can only be changed by a formal approval from a configuration control board (CCB). Note: the acronym CCB is often used to mean the change control board that, in addition to configuration control, makes decisions of any aspect of a project or an enterprise.

Interface agreements, specifications, and interface designs are usually documented through the interface control documents (ICD) and the interface design descriptions (IDD); in some instances, depending on the complexity of the product and the type of internal and/or external interfaces, an interface control working group (ICWG) is created to analyze and baseline changes to an

interface for further recommendation to the CCB.

A configuration item (CI) may be hardware (HWCI), software (SWCI), firmware, subsystems, assemblies, non-development items, commercial off-the-shelf (COTS) items, acquirer furnished equipment, and/or processes. Please see Wasson (2006), Grady (2006), and INCOSE *SE Handbook* v. 3.2.2 for a more detailed description of configuration and interface control.

A product may experience hundreds of changes during its life cycle due to new product releases/enhancements, repair/replacement of parts, upgrades/updates in operating systems, computer infrastructure, software modules, organizational changes, changes in processes and/or methods and procedures, etc. Thus, strong mechanisms for bookkeeping and activity control need to be in place to identify, control, audit, account and trace interfaces, interactions, and relationships between entities that are required to maintain product configuration status (Eisner 2008). The product configuration and CI's are then controlled through the configuration management process.

Configuration Management and Risk Management

Configuration management (CM) deals with the identification, control, auditing, status accounting, and traceability aspects of the product, and broadly covers the book-keeping and control activities of the systems engineering process (Eisner 2001). Any product configuration changes to the baseline (configuration item, operational baseline, functional baseline, behavior baseline) or product baseline are submitted to a configuration control board (CCB) through an engineering change request (ECR) and/or a configuration change request (CCR). The CCB then analyzes the request to understand CI impacts and the feasibility (time and cost) of authorization or rejection of change request(s). The lack of proper control and tracking of CI and product baselines may result in a loss of features, functionality, data, interfaces, etc., leading to backtracking and CI version losses which may affect the offered product. All approved changes will have to be baselined, documented, and tested for backward compatibility and to ensure compliance with the integrated product functionality. Thus, successful implementation and life cycle management of the product mandates a highly disciplined CM process that maintains proper control over the product and its

components. Please see the INCOSE *Systems Engineering Handbook* v. 3.2.2 (2012) for a detailed description of the CM Process.

Risk management deals with the identification, assessment, and prioritization of technical, cost, schedule, and programmatic risks in any system. Almost all engineered systems are designed, constructed, and operated under some level of risks and uncertainty while achieving multiple, and often conflicting, objectives. As greater complexities and new technologies are introduced in modern systems, the potential of risks have significantly increased. Thus, the overall managerial decision-making process should involve an extensive cost-benefit analysis of all identified, qualified, and evaluated risks (Haimes 2008). Risk management involves the coordinated and most cost-effective application of resources to minimize, monitor, and control the probability and/or impact of all identified risks within the systems engineering process. The risk management process requires the involvement of several disciplines and encompasses empirical, quantitative, and normative judgmental aspects of decision-making. Furthermore, risk assessment and management should be integrated and incorporated within the broader holistic approach so technology management can help align the risk management requirements to the overall systems engineering requirements. Thus, the inclusion of a well defined risk management plan that deals with the analysis of risks, within the systems engineering master plan is vital for the long term and sustained success of any system (Blanchard and Fabrycky 2011).

References

Works Cited

Annachino, M. 2003. *New Product Development: From Initial Idea to Product Management.* Amsterdam, Netherlands: Elsevier.

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.

Eisner, H. 2008. *Essentials of Project and Systems Engineering Management*, 3rd ed. New York, NY, USA: John Wiley & Sons.

Grady, J. 2006. *System Requirements Analysis.* New

York, NY, USA: Academic Press.

Haimes, Y. 2008. *Risk Modeling, Assessment, and Management*, 3rd ed. New York, NY, USA: John Wiley & Sons.

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.

ISO/IEC/IEEE. 2015.*Systems and software engineering - system life cycle processes*.Geneva, Switzerland: International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC), Institute of Electrical and Electronics Engineers.ISO/IEC 15288:2015.

ISO/IEC/IEEE. 2011. *Systems and software engineering - Architecture description*. Geneva, Switzerland: International Organization for Standardization (ISO)/International Electrotechnical Commission (IEC)/Institute of Electrical and Electronics Engineers (IEEE), ISO/IEC/IEEE 42010.

Kass, R. 2006. "The logic of warfighting experiments." DOD Command and Control Research Program (CCRP). August 2006. Accessed 23 April 2013 at http://www.dodccrp.org/files/Kass_Logic.pdf.

Lawson, H. 2010. *A Journey Through the Systems Landscape*. London, UK: College Publications.

Morse, L., and D. Babcock. 2007. *Managing Engineering and Technology*. International Series in Industrial and Systems Engineering. Upper Saddle River, NJ, USA: Prentice Hall.

Phillips, F. 2001. *Market Oriented Technology Management: Innovating for Profit in Entrepreneurial Times*. New York, NY, USA: Springer.

Pugh, S. 1990. *Total Design: Integrated Methods for Successful Product Engineering*. Englewood Cliffs, NJ, USA: Prentice Hall.

Reinertsen, D. 1997. *Managing the Design Factory: A Product Developers Tool Kit*. New York, NY, USA: Simon & Schuster Ltd.

Rogers, E.M. 2003. *Diffusion of Innovations*, 5th ed. New York, NY, USA: Free Press.

Smith, P., and D. Reinertsen. 1997. *Developing products in half the time – new rules, new tools*, 2nd ed. Hoboken, NJ, USA: John Wiley & Sons.

US Army. 2012. "Chapter 2, section A.4" in *Army Science and Technology Master Plan*. Accessed January 12, 2012. Available at: http://www.fas.org/man/dod-101/army/docs/astmp/c2/P2A4.htm.

Wasson, C.S. 2006. *System Analysis, Design, and Development*. Hoboken, NJ, USA: John Wiley & Sons.

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.

Eisner, H. 2008. *Essentials of Project and Systems Engineering Management*, 3rd ed. New York, NY, USA: John Wiley & Sons.

Wasson, C.S. 2006. *System Analysis, Design, and Development*. Hoboken, NJ, USA: John Wiley & Sons.

Additional References

Annachino, M. 2003. *New Product Development: From Initial Idea to Product Management*. Amsterdam, Netherlands: Elsevier.

Haimes, Y. 2008. *Risk Modeling, Assessment, and Management*, 3rd ed. New York, NY, USA: John Wiley & Sons.

Kass, R. 2006. "The logic of warfighting experiments." DOD Command and Control Research Program (CCRP). August 2006. Accessed 23 April 2013 at http://www.dodccrp.org/files/Kass_Logic.pdf.

Lawson, H. 2010. *A Journey Through the Systems Landscape*. London, UK: College Publications.

Morse, L., and D. Babcock. 2007. *Managing Engineering and Technology*. International Series in Industrial and Systems Engineering. Upper Saddle River, NJ, USA: Prentice Hall.

Phillips, F. 2001. *Market Oriented Technology Management: Innovating for Profit in Entrepreneurial*

Times. New York, NY, USA: Springer.

Pugh, S. 1990. *Total Design: Integrated Methods for Successful Product Engineering*. Englewood Cliffs, NJ, USA: Prentice Hall.

Reinertsen, D. 1997. *Managing the Design Factory: A Product Developers Tool Kit*. New York, NY, USA: Simon & Schuster Ltd.

Rogers, E.M. 2003. *Diffusion of innovations,* 5th ed. New York, NY: Free Press.

Smith, P., and D. Reinertsen. 1997. *Developing products in half the time – new rules, new tools*, 2nd ed. Hoboken, NJ, USA: John Wiley & Sons.

US Army. 2012. "Chapter 2, section A.4" in *Army Science and Technology Master Plan*. Accessed January 12, 2012. Available at: http://www.fas.org/man/dod-101/army/docs/astmp/c2/P2 A4.htm.

< Previous Article | Parent Article | Next Article >
SEBoK v. 2.11, released 25 November 2024