31st Annual INCOSE international symposium
virtual event
July 17 - 22, 2021

Broadening the Definition of Breadth in Systems Engineering

James R. Armstrong
Stevens Institute of Technology
400 Wildberry Ct
Millersville, MD 21108
410-987-1276
jimarmstrong29@aol.com

Abstract. Breadth versus depth is an aspect of systems engineering that is often referred to but seldom fully defined. This paper presents results of recent research that defines multiple aspects of depth that should be considered and discusses their value. The set of nine dimensions expands beyond the usual view of breadth defined as multiple systems engineering areas and can be applied to growth paths for systems engineers in their expertise development.

Introduction

One aspect included in most any discussion of systems engineering is breadth, particularly in contrast to depth as depicted in figure 1. However, definitions of breadth are limited. In general knowledge terms, breadth may be seen as knowledge of arts, science, politics, sports, or other different categories. Depth would be knowing the details to be able to note that Abraham Lincoln and Charles Darwin were born on the same day (Hoyt, 2021). In overall engineering, breadth may be determined by the number of different disciplines known such as electrical, mechanical, software, etc. Depth would be the ability to do detail design work in a specific discipline.

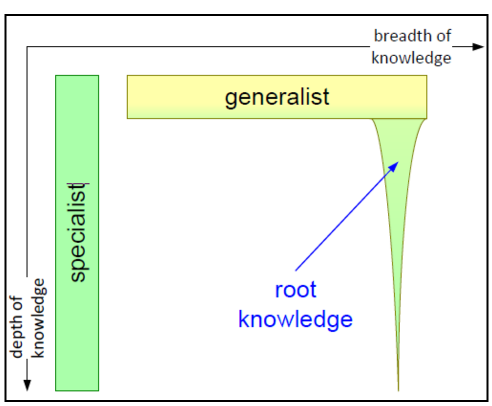


Figure 1. Breadth and Depth as Orthogonal Characteristics (Muller, 2020)

In systems engineering, one definition can be found in the Systems Engineering Professional (SEP) certification program. For certification purposes, breadth is defined as having experience in multiple systems engineering areas from 15 defined technical, management, and support areas. The requirements are three for Certified Systems Engineering professional (CSEP) (INCOSE, October 25, 2019) and six for Expert Systems Engineering Professional (ESEP) (INCOSE – December 10, 2017). Depth is addressed by the number of months in an area – 12 for CSEP and 24 for ESEP. ESEP additionally addresses professional development and leadership.

The INCOSE Systems Engineering Competency Framework (INCOSE, July 2018) addresses a very similar set of technical and management areas. It expands breadth with professional competencies and provides a set of core competencies such as systems thinking, critical thinking, and general engineering (basics of math, science, and engineering) that support the technical areas. Depth is addressed by the rating of competency in an individual area from aware to expert. By the nature of the framework, the user will define any scope of breadth and which competencies are to be addressed based on their needs. The framework also provides mappings of the competencies to the SEP certification program areas and the INCOSE Systems Engineering Handbook showing strong commonality in their content.

Expertise Development Study

A planned study described in (Armstrong, 2015) addressed development of systems engineering expertise. The study was then conducted used the Grounded Theory method (Glaser and Strauss, 1967) (Glaser, 2012a) (Glaser, 2012b). This method is specifically designed for subjective research and is based on analysis of information gathered through interviews or other data sources to develop common themes which can be further investigated. It is commonly used by social scientists. The information is coded and analyzed for common themes that are developed into the resulting theories. There are differences from quantitative methods that many technical researchers are not used to seeing. One is the use of a smaller sample set. The sample set is commonly in the low double digits. Adequacy is determined when additional data does not change the results. An analysis of the number of interviews in multiple studies (Marshall et al, 2013) recommends the number of interviews between 20 and 30 and strongly recommends not exceeding 30. Another is the absence of a control group since the purpose is to find common themes and not to demonstrate a difference. Also, the basic method does not include a review of existing literature to avoid biasing the review of the current situation. However, in this study, there was an extensive review of the many prior works on expertise development such as the work on by Bloom (1985) on development of expertise and Ericsson, et al. (1993) addressing the concept of deliberate practice.

In the study, 24 systems engineers recognized as experts in the discipline were interviewed on their development history. The participants were either Expert Systems Engineering Professionals (16), INCOSE Fellows (5), corporate top-level systems engineers (9), or had multiples of these qualifications. Four were INCOSE Past Presidents. The gender distribution was 18 males and 6 females. Since the minimum requirement for ESEP is 20 years of experience, the ages of the sample group were primarily in the 50's and 60's. The subjects were all from the United States but several had worked internationally and all had international experience through work or INCOSE. All had experience in large aerospace companies. 19 discussed additional systems engineering experience in government, military, commercial, commercial aerospace, large and small consulting, or academic organizations.

Those interviewed were asked to discuss how they of developed systems engineering expertise and a systems view through experience, education, training, mentoring, or other methods. After coding and analysis, the results were validated by having the participants review and comment on the results. The study provided several findings concerning how the participants developed their expertise including emphasis on such methods as self-training. Further details can be found in the complete discussion of the study in (Armstrong, 2017). This paper focuses on one particularly relevant result which was the identification of multiple aspects of the breadth of the participants careers and expertise.

Breadth Dimensions

The study participants discussed a variety of experience that can be related to the concept of breadth rather than depth of systems engineering expertise. Table 3 lists the occurrences of the major categories of breadth from the interviews. The results in the study are presented in the order of number of specific mentions across the interviews. In this paper, the dimensions are presented in a different sequence with the first being those that relate to the work done and then those related to the environment in which it is done. Also, the dimension of international experience, addressed separately in the study, is added.

Table 1. Categories of Breath of Experience Discussed in Interviews (Armstrong, 2017)

Breadth Categories																									Total
A	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	√	24
B	√	√	√		√		√	√		√	√	√	√		√	√	√	√		√					15
C	√	√	√	√	√	√			√					√	√				√	√	√	√			13
D			√		√				√				√	√	√				√	√	√	√			10
E	√		√			√			√				√		√										6
F											√						√	√	√			√	√		6
G											√					√	√								3

A: SE Areas B: Management C: Product/customer D: Gov't/Commercial
E: Life Cycle F: Disciplines G: Level

Systems Engineering Areas. For INCOSE certification at the Certified Systems Engineering Professional (CSEP) and Expert Systems Engineering Professional (ESEP) levels, breadth is defined only in terms of 14 systems engineering functional (INCOSE, 2017). The current areas are Requirements Engineering, System and Decision Analysis, Architecture/Design Development, Systems Integration, Verification and Validation, Systems Operations and Maintenance, Technical Planning, Technical Monitoring and Control, Acquisition and Supply, Information and Configuration Management, Risk and Opportunity Management, Lifecycle Process Definition and Management, Specialty Engineering, and, Organizational Project Enabling Activities. These have been modified since the study and now include operations and maintenance.

For CSEP, breadth is defined as having performed in three of these areas. For ESEP, the requirement is six. Each has a minimum time requirement to provide a depth dimension to assure more than a brief experience. The intent is to assure that the CSEP or ESEP is not just a specialist in one area such as requirements but has a broader understanding of the whole discipline.

The results from the interviews are shown in Table 2. The most frequently mentioned were requirements engineering and qualification, verification, and validation. The least were quality assurance and specialty engineering.

Table 2, Systems Engineering Areas Discussed (Armstrong, 2017)

Systems Engineering Areas																									Total
A	√	√	√	√	√	√	√	√	√	√	√	√	√	√			√			√	√		√	√	19
B	√	√	√	√	√	√		√	√		√	√	√			√	√	√	√	√		√	√		18
C	√	√	√		√	√	√		√	√			√	√	√	√	√		√	√	√	√	√		18
D	√	√					√				√	√	√	√	√	√	√		√	√	√	√	√	√	16
E	√	√	√	√	√	√					√		√						√	√	√	√	√		13
F	√	√	√				√		√	√	√		√			√			√			√		√	12
G	√	√				√	√		√	√	√	√		√				√		√					11
H	√	√					√	√			√		√		√	√		√		√				√	11
I	√	√	√	√	√	√					√		√							√	√	√			11
J	√	√				√					√								√		√	√	√		8
K			√		√		√			√	√									√	√		√		8
L						√				√	√						√					√	√		6
M											√		√									√			3
N		√									√		√												3

A: Requirements
B: V&V/Test
C: Training
D: Architecture/Design
E: System Integration
F: Other (Mod & Sim)
G: Analysis/Trades
H: Technical Planning
I: Process Definition
J: Baseline Control
K: Tool Support
L: Risk & Opp Mgt
M: Specialty Eng
N: Quality Assurance

The participants who were ESEPs would have identified at least six areas with two or more years of experience in their certification application. Therefore, it is apparent that not all ESEPS discussed their full breadth across all 14 areas in the interviews. This is due to asking about strengths and not a full account of each area. The average number of 6.6 areas indicated in the data would be lower than the actual. A fifteenth area of "other" is not specifically defined by INCOSE. Modeling and simulation was independently mentioned 14 times and is included in this table under the "other" category. The benefits of breadth in systems engineering areas are evident for both system level systems engineers and for those who are primarily involved in one area. At the top level, it is beneficial to have some understanding of all of the multiple actions that are part of the systems engineering process for both coordination and supervision purposes. At the area level, it helps to understand the impact on others. The systems engineering process is very interactive and has been described as a complex neural network (Armstrong, 1998).

For instance, in writing requirements, architecture and design experience helps to understand whether requirements are achievable. Also, verification experience has helped in identifying unverifiable requirements such as "The reliability of the system shall be the maximum achievable within the current state-of-the-art." In another instance, experience in contracting provided the information needed to identify a reliability test method requirement as unworkable since it required a minimum of over 30,000 hours of operations and only one piece of equipment was being manufactured with a total test period of three months.

Specialty engineering was at the bottom of the list in the study. There are multiple disciplines considered to be part of this area. Most common are generic topics such as reliability, maintainability, transportation or human factors. In a risk analysis of a plating plant design, one issue was the use of chemicals which, if combined, could produce cyanide gas. A typical bottom-up failure hazard analysis was performed on the components in the design that addressed the risk of failure causing an incident. The systems engineer with systems safety experience suggested a top-down approach using a fault tree. This approach brought the possibility of human error into view. This allowed the mitigations in the existing design to be clearly presented to the customer in a more complete and successful risk analysis.

Life-Cycle Phases. It is possible to have breadth in systems engineering areas and remain limited in experience across the life-cycle. For instance, a systems engineer can work primarily in the areas of requirements, trade studies, modeling, architecture, planning, etc. and stay in the front end of programs. Conversely, experience could focus on verification, validation, and late phase activities around the end of the program. A full range of life-cycle experience from inceptions to operations and maintenance provides a broader view of the total program and the impacts of different phases on each other.

The time spent in various phases of the life cycle varied among the interviews. However, there was a consistency in covering the full life cycle at some point during their careers. The two areas most mentioned are at opposite ends of the program life cycle were requirements and test. Most were deeply involved in front end requirements development and conceptual design. However, the value of experiencing the end result of test, deployment, operations, and support was emphasized as providing real meaning of the importance of early systems engineering activities and helped in understanding how to do them more effectively. A typical comment was “starting out in maintenance and going to installation sites was very very enlightening because it taught you what could go wrong.” In some cases, there was an organizational approach that assigned new engineers to testing first to witness problems that should be avoided before letting them get involved in early design efforts. Another approach has been to assign new engineers to attend product training with the customers to see first hand what problems they had with existing products before starting in on a new product design.

Level. Another dimension is the level of scope within the system from component to complete system. Three participants specifically described starting at the component level and progressing to higher levels of assembly within the system architecture. “Over time things started to grow. You start getting larger and larger design tasks. I started out on little electronic work and, with the success of that, I grew to larger and larger electronic things.” Their view changed from a part meeting a limited set of requirements to the entire system and environment with an expanded objective of meeting user needs. One also raised the question of why they went this path and others didn’t. Indeed, some entered at the system level and did not spend time at the component design level.

The principal benefit of having experience at multiple levels is to bring an understanding or the issues and challenges at the lower levels to the systems level oversight. Complaints from lower-level component teams in large systems is that the top-level systems integrators are giving them direction and requirements that reflect a lack of understanding of the realities at the lower levels.

A broader impact has to do with the level of a system that an organization typically works in a large system. A review of several requirements processes (Armstrong, Klue, and Stall, 2006) revealed that the content addressed different parts of the vertical trace of requirements depending on whether the organization was a systems integrator, component provider, or somewhere else in the spectrum. Figure 2. Shows how four different organizations scoped their requirements processes.

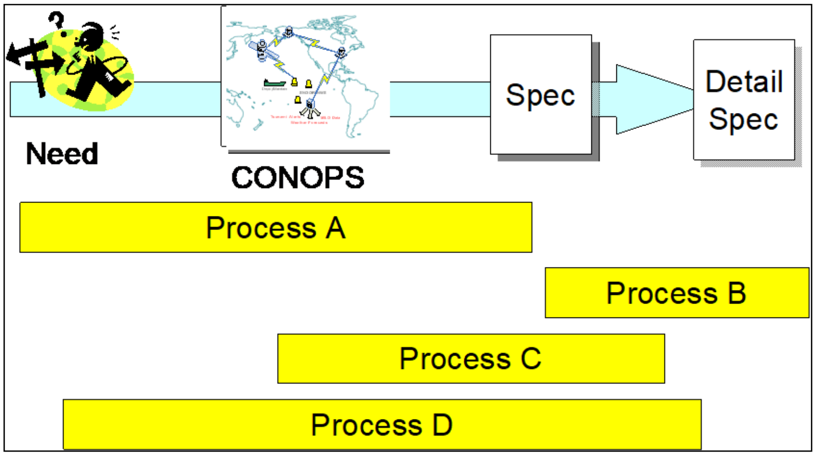


Figure 2. Scope of Various Requirements Processes (Armstrong, Klue, and Stall, 2006)

Two processes, A and D, started with the needs and concept of operations and continued to higher levels of the specification tree. These organizations were typically at the system level and passed requirements on to subcontractors. The Process B organization typically received the specification from the integrator and worked down to the detail level from there. This organization also did not address the top-level need or concept of operations since they were not involved in the preparation of those documents and, in fact, seldom saw them. An awareness of the other end of the spectrum would be beneficial to those at either end.

Technical Disciplines. Since the development of a system relies on the coordination of many different technologies, the systems engineer has to be at least conversant in multiple technical disciplines in order to help them work together. Most of those interviewed started as an electrical engineer, mechanical engineer, software developer, or in another specialized technical area. Often related to the change in levels and increased project scope, their experiences involved interfacing with or learning to perform other technical disciplines. “Now you are not just dealing with the electronics but you have to develop an understanding of the software, of the other electronic work that was being done, of the lateral mechanics of the electoral hydraulics, of the optics, everything.” They were motivated to develop at least a basic understanding of other technical disciplines than the one in which they started. In some cases, this included seeking additional education in another field or even another degree in another discipline such as electrical engineering or software development.

It is beneficial to have a basic understanding of the particular approaches and concerns of each technical discipline involved in a system. Hardware and software developers can have a fundamentally different approach to system architecture and design. Each discipline has its own techniques for analysis and often for verification. The systems engineer needs to be able to speak the various languages enough to understand how they can work together and to communicate among them.

Management. After the variety of systems engineering areas, management responsibilities were the second most reported type of breadth. 15 of the participants became technical or project managers at some point in a discipline area, team, or project. “Then, of course, you start branching out into other areas like project management and eventually into supervision and leadership and other aspects like that.” This led to the development of skills in management related tasks such as planning, budgeting, scheduling, as well as interpersonal skills.

There was a strong consensus that interpersonal skills are necessary for systems-level practice of systems engineering. One concerning result from the interviews was that those who did receive interpersonal skills training as part of their career development did so only as part of a management assignment or during a military part of their career. This is one area that the systems engineering community has paid some attention to and should continue to improve.

One tool for connecting technical and management decisions to management concerns over cost and schedule is the COCOMO cost model. By using a tool to experiment with the variables, the systems engineer can learn the impact of technical actions such as frequent requirements changes, selection of novel approaches, limiting flexibility in requirements, or locking in architectures early. Software estimation models can also show the impact of management decisions on teams, staffing, or schedule constraints on cost, schedule, and quality of the technical tasks.

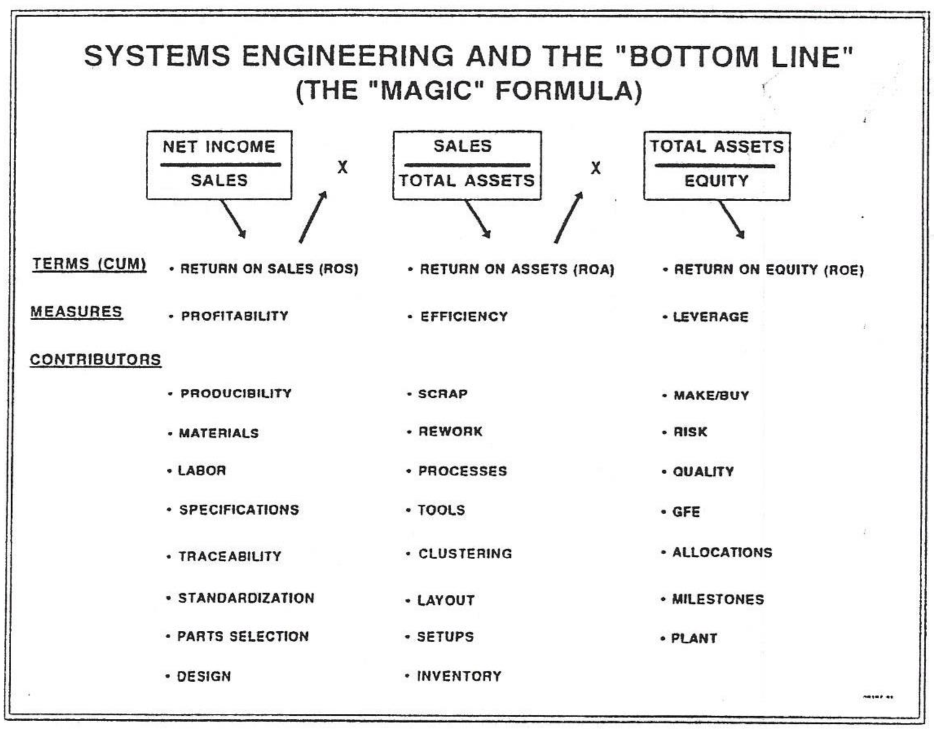


Figure 3. SE Course Dupont Formula Slide (Caver, n.d.)

Another part of the management learning is an understanding of the financial aspects of a program. There are other ways to gain this appreciation besides experience as a manager. One commercial company's systems engineering training included review of the Dupont Formula to identify the various ways systems engineering can affect profitability as shown in Figure 3. The point being that technical decisions are not independent of profitability and experience in management will help systems engineers appreciate this relationship.

Customer/Supplier. During their careers, several of the interviewed worked on both the customer and supplier side of the relationship. This provided a better understanding of the other side's issues and needs. This is one of the concerns addressed by Integrated Product Development teams that include both the customer and supplier. One aspect of the Boeing 777 program was the inclusion of suppliers even in early writing of requirements instead of just preparing a specification in-house and sending it out in the RFP. When able to ask about requirements, a supplier may learn the reasons behind the requirement. In some cases, they may learn that the customer incorrectly thought they needed the information to design the component. The supplier also had the ability to identify those missing requirements that would be helpful. The result was an improvement in quality and schedule.

The Defense Systems Management College has maintained a policy of inviting industry participants to fill 10% of the seats. While some have considered this improper use of limited training seats, it is actually an important part of the education experience. At the start of a class, each side typically expresses the perceived reasons the other side is a problem. After a while, both sides learn to recognize the challenges faced by the other.

Products. Individual experience also varied in the type of product and its technology. Products addressed ranged from simple appliances to ships and planes to large air traffic control systems. Technologies changed as a person moved from one product to another and also within the same product over time. "The change in products particularly helped develop a larger system view, a recognition of common principles, and an awareness of patterns." For example, technical experience can transfer from one technology to another, e.g., edge detection from images to manufacturing to lithography (Muller, n.d.).

One lesson learned about new technologies is that the new technology tends to bring new problems along with its new solutions. Early applications of solid-state electronics in outdoor applications such as airport Instrument Landing Systems significantly improved reliability but also brought a vulnerability to electrical spikes in communications and power lines. Fiber optics provided significant increase in bandwidth and much lighter cables for tactical use but were more difficult to splice when damaged. Such lessons can be transferred from one technology or new product to the next and don't have to be relearned each time.

Government/Commercial. All had some experience in companies with government contracts and nine had worked on the government side of the fence. Nine had experience with commercial companies of various sizes. In some cases, they actively sought out this experience to broaden their experience base and learn the differences between the two worlds.

One particular difference came to light in the writing of EIA-632. The representatives of large government contractors wanted to limit the scope of validation. Appropriately, they did not want a standard to possibly hold them to unstated and even unknown requirements that were not in the contract specification but might be discovered through validation. This would particularly be a problem in fixed price contracts. As a result, validation of the product was limited to being traced to "acquirer requirements" as depicted in Figure 4 below.

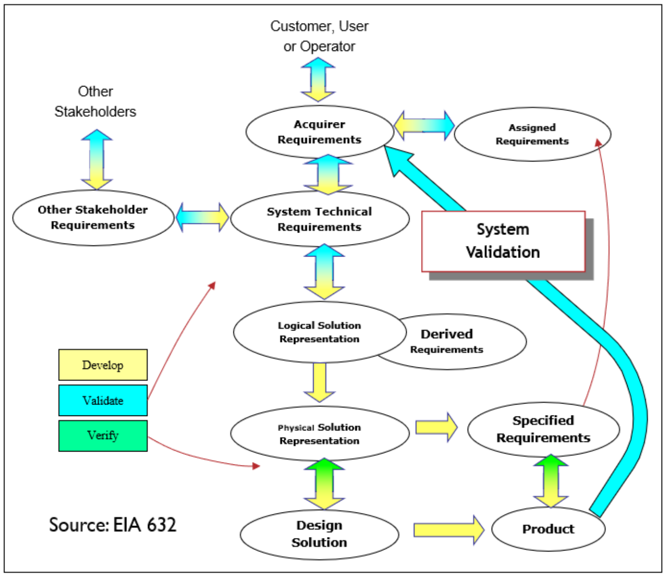


Figure 4. EIA 632 Validation Trace

However, those who were from commercial companies saw validation, early validation in particular, as critical to their success. The compromise was a Note 4 that allowed for the need for a connection between the product and the real world:

> "In addition, there can be cases where it is appropriate to validate against actual needs and expectations of end users in their environment under real-world conditions. This is called by various names: market trial, field testing, beta testing, or operational test and evaluation."

Validation has since been given more clarity in the various standards and other references. However, this example shows a difference in two business situations that should be understood by systems engineers.

Another difference between commercial and government systems is the different approaches to funding and budgets. The DSMC teaches government students how to read the financial data in corporate annual reports and gives them an understanding of the basics such as the Dupont formula. For the industry students, an introduction to the government funding process and rules for spending different categories of funds can be eye opening. Although not a technical topic, these factors on both sides can have serious impact on systems development decisions.

International. In terms of interview data, international involvement was not part of the data coding. The sample was from the United States and only one participant talked about work outside the United States. Two discussed involvement in international standards development. However, twenty-two were involved in international systems engineering organizational leadership, symposia, working groups, or other activities through INCOSE. Work on international standards was described by one participant as having "…fostered an understanding from a world perspective of engineering and how they do it the same and how they do it differently."

One of the first issues to arise in international programs is language It is difficult enough to agree on terminology in one language. An interesting class exercise has been to have students ask multiple people at work what validation means. There is always a wide variety of responses. In an early revision of the CMMI®, the author team had problems agreeing on whether a plan implements a strategy or a strategy implements a plan. The result was to eliminate a requirement for an integration strategy as the first practice and start with definition of the assembly sequence. When international usage is added, it becomes more difficult. Working with both some US and UK companies, the relationship between project and program, which is part of the other, is reversed. When we change languages, things get even more difficult as several companies have learned that their brand names have negative meanings, some of which would not be appropriate to mention here. Certainly, attention has to be paid to the actual consistency of meaning in translating technical information from one language to another or translating INCOSE products to other languages.

Another lesson to be learned is the differences in cultures and customs. The human aspect of the system may not be optimal in a different culture. Even things as basic as dimensions for human interfaces or language familiarity are likely to vary. Cultural norms concerning the relationship between levels of seniority or gender can play in the functionality of systems design as well. Also, some cultures look at systems as a means of providing employment while others want less use of human resources.

Customs can also refer to the import process for international trade. The actual paperwork, approvals, procedures, delays, cost, and other factors can be and have been a surprise to engineers who are not aware of them. One site survey team intending to take the scenic route from Germany to Italy instead of the main road was surprised to be asked for a $100,000 duty at the Austrian border station that was not used to handling this specific situation. In another case, two flatbeds of trucks to support installation teams were separated in route. The recipients were dismayed that the one that arrived first could not be unloaded because customs would not release them until the whole shipment was delivered.

Different laws can present additional lessons. One US program was surprised to learn that their FCC frequency approval was not valid in Germany. Other programs had to insert activities to obtain type approval on equipment connecting to the host country's communications network at the last minute in a tight schedule.

Conclusion

One of the results of the source study is the identification of several dimensions to breadth in systems engineering in addition to the traditional view of systems engineering technical and management areas as noted in the SEP program, capability framework, and the various systems engineering standards, texts, and handbooks. While these additional dimensions may not be seen as new information to experienced systems engineers, their identification as a set of concerns to be addressed has not been previously recorded.

Breadth in systems engineers certainly starts with expanding experience in a larger number SE areas but can be expanded in other dimensions to aid in the development of effective systems engineers. A summary of the dimensions in this paper follows:

- SE Areas – having a broader view of the overall process and how the areas work together, e.g., requirements and verification to assure that stated requirements are verifiable.
- Life cycle – not staying in a comfort zone of one phase of a project such as only working with requirements and architecture to understand the impacts each phase has on the others.
- Level – learning both the detail design issues and the overall, top-level system view and issues
- Technical Disciplines – having an awareness of several of the technical disciplines that are part of the design, not being limited to only hardware or software.
- Management – understanding the business side of the organization, having interpersonal skills, leading.
- Customer/Supplier – seeing the relationship from both sides.
- Product – having experience with other products and being able to bring solutions from outside the immediate work to the project.
- Government/Commercial – having knowledge of the differences between working to a specification and statement of work and creating a system that is only successful if it sells, e.g., the impact on validation.
- International – knowing the positives and negatives of the variations in cultures, laws, languages, environments, etc.

This list is not intended to provide a magic minimum number of dimensions or set that solves the issue of breadth in each instance. However, it does provide additional guidance for systems engineers to consider in their development. It can be used in the development of career development paths by both individuals and organizations.

References

Armstrong, JR, (1998), 'Systems Engineering as a Neural Network and its Impact on Integrated Product Development", INCOSE International Symposium, Vancouver, BC, Canada

Armstrong, J., Klue, G., and Stall, A. (Nov, 2006) 'Integration of Processes on a Multiorganization Program', Systems and Software Consortium, Herndon, VA.

Armstrong, JR, and Wade, J. (2012) 'Developing Expertise in Systems Engineering: Lessons from Other Disciplines', INCOSE International Symposium, Rome, Italy

Armstrong, JR (2017), 'Development of Systems Engineering Expertise', PhD thesis, Stevens Institute of Technology (Hoboken, NJ, US).

Bloom, B. S. (1985). *Developing talent in young people.* New York. Ballantine Books. Pp. 32, 305-306, 328

Caver, T, n.d., ‘Systems Engineering and the Bottom Line’, Systems Management Development Corporation, Springfield, VA.

Ericsson, K. A., Krampe, R. T. and Tesch-Romer, C. (1993). ‘The role of deliberate practice in the acquisition of expert performance’. Psychological Review. Vol. 100. No. 3, 363-406

Glaser, B. G. and Straus, A. L. (1967). *The Discovery of Grounded Theory: Strategies for qualitative research*. Aldine Transaction, Transaction Publishers, Piscataway, NJ.

Glaser, B. G. (2012a) *Constructivist Grounded Theory?*. The Grounded theory Review, vol. 11, no. 1. Pp 28-38.

Glaser, B. G. (2012b) *Stop. Write! Writing Grounded Theory*. The Grounded theory Review, vol. 11, no. 1. Pp 2-11.

Hoyt, D. L., (March 24, 2021), ‘Jumble Crossword’, Washington Post

INCOSE, (December 10, 2017), ‘Form 42 – Instructions for Completing Form 41 (Application for INCOSE ESEP)’, INCOSE, San Diego, CA

INCOSE, (July 1 2017), ‘Individual Application for CSEP’, INCOSE, San Diego, CA

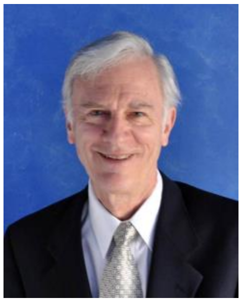
INCOSE, (July 2018), ‘INCOSE Systems Engineering Competency Framework’, TP-2018-002-01.0, INCOSE, San Diego, CA

INCOSE, (October 25, 2019), ‘Form 2 – Instructions for Completing Form 1 (Application for INCOSE CSEP)’, INCOSE, San Diego, CA

Marshall, B., Cardon P., Podar A., and Fontenot R., (2013), ‘Does Sample Size Matter in Qualitative Research?: A Review of Qualitative Interviews in IS Research’, Journal of Computer Information Systems

Muller, G. (2020), *The Awakening of a System Architect version 1.1*, Retrieved on 11/12/2020 from http://gaudisite.nl/MaturitySlides.pdf, University of South-Eastern Norway, Notodden, Norway

Biography



James R. Armstrong. Jim Armstrong has practiced systems engineering for 54 years, performing various roles including configuration management, test, deployment, chief engineer, program manager, and program element monitor. For the last 30 years, he taught, consulted, and appraised systems engineering in industry and government. Also, he was on the author teams for several of the systems engineering standards and models. He is also certified in the use of the Meyers-Briggs Type Indicator. He has a BS in Mechanical Engineering from Rensselaer Polytechnic Institute, an MS in Systems Management from the University of Southern California, and a PhD in Systems Engineering from Stevens Institute of Technology. He has an INCOSE Expert Systems Engineering Professional certification.