PROJECT PERFORMANCE
INTERNATIONAL

CERTIFICATION
TRAINING INTERNATIONAL
A PPI Company

SYSTEMS ENGINEERING NEWSLETTER

brought to you by

Project Performance International (PPI)

SyEN 63 – March 16, 2018

SyEN is an independent free newsletter containing informative reading for the technical project professional, with scores of news and other items summarizing developments in the field of systems engineering, including related industry, month by month. This newsletter and a newsletter archive are also available at www.ppi-int.com.

Systems engineering can be thought of as the problem-independent, solution technology-independent, life-cycle-oriented principles and methods, based on systems thinking, for defining, performing, and controlling the engineering effort within a technical project. The approach aims to maximize the benefit delivered to the enterprise, as influenced by the needs and values of the applicable stakeholders.

If you are presently receiving this newsletter from an associate, you may wish to receive the newsletter directly in future by signing up for this free service of PPI, using the form at www.ppi-int.com. If you do not wish to receive future Systems Engineering Newsletters, please unsubscribe by clicking on the link at the bottom of this email.

We hope that you find this newsletter to be informative and useful. Please tell us what you think. Email us at syen@ppi-int.info.

The views expressed in externally authored articles are those of the author(s), and not necessarily those of PPI or its professional staff.

IN THIS EDITION

Read More…

Conferences and Meetings

Read More…

Some Systems Engineering-Relevant Websites

Read More…

Systems Engineering Publications

- Procedia Computer Science
- Survey Report: Improving Integration of Program Management and Systems Engineering
- A Systems Approach to Managing the Complexities of Process Industries
- Results-Based Leadership
- Systems of Conservation Laws
- Systems Engineering and Analysis

Read More…

Systems Engineering Tools News

- Needs and Means
- Software related to Multiple Criteria Decision-Making

Read More…

Education and Academia

- Systems Engineering at Drexel University, Philadelphia, Pennsylvania, USA
- Systems Engineering as a VCE Subject in the State of Victoria, Australia

Read More…

Standards and Guides

- A Comparison of IEEE/EIA 12207, ISO/IEC 12207, J-STD-016, and MIL-STD-498 for Acquirers and Developers

Read More…

Definitions to Close On

Read More…

PPI and CTI News

Read More…

Upcoming PPI and CTI Participation in Professional Conferences

Read More…

PPI and CTI Events

Read More…

QUOTATIONS TO OPEN ON

"Never in the history of mankind has a literally complete set of requirements been assembled on anything. Nor is that a valid objective."

Robert John Halligan

"Purpose is that sense that we are part of something bigger than ourselves, that we are needed, that we have something better ahead to work for."

Anonymous

"If you think you can do something or think that you can't, you're probably right."

Henry Ford

EDITORIAL

The Status of the Initiative to Strengthen Integration of Program Management and Systems Engineering is in Jeopardy

by

Dr. Ralph R. Young

Practitioner, Author, Consultant, and Grandfather

An initiative was begun following the availability of a recently published book, *Integrating Program Management and Systems Engineering* (*IPMSE*), itself a collaboration of the International Council on Systems Engineering (INCOSE), the Project Management Institute (PMI), and the Consortium for Engineering Program Excellence (CEPE) at the Massachusetts Institute of Technology (USA). The research base of this book was intensive and took place over a five-year period. The purpose of the book was to investigate a concern that some systems engineers and program managers have a mindset that their respective work activities are not mutually supportive. This creates a barrier and unproductive tension[1] that negatively impacts program activities and results. Research results confirmed that a cultural barrier does exist in many organizations, particularly with respect to technical, complex programs. The research indicates that the current rate of program failures and challenges is not sustainable in an era when programs are becoming more complex in every way (technical, stakeholder, supply channel, etc.). Some organizations are deliberately attempting to improve performance by ensuring the management and technical teams assigned to the program have strongly integrated approaches to delivering a solution. Where these mechanisms are formalized and supported by the organization, the research indicated that program performance can be positively affected. **Currently, the two disciplines are in a situation in which the interests of some programs cannot be served adequately** (Rebentisch et al, Wiley, 2017). We need programs that subordinate the individual functional and organizational identities to the needs of the overall program.

All well and good that the problem of a lack of integration of program management and systems engineering activities on some programs has been identified, thoroughly researched, and well documented. Wonderful that several success stories (case studies) were found during the research where a high degree of integration of the two disciplines resulted in amazingly effective outcomes for highly complex systems programs, for example, the F/A-18 E/F Super hornet Program;[2] the Curiosity Rover component of the National Aeronautics and Space Administration's (NASA) Mars Science Laboratory;[3]

[1] The roots of unproductive tension may ultimately lie with poorly defined roles and relationships in the program and organization (*Integrating Program Management and Systems Engineering* [*IPMSE*], p. 348). However, there is a perceived power imbalance between the program management community and the systems engineering community that seems to perpetuate and amplify existing tensions. A positive step going forward would be for program management and systems engineering leaders and professionals who are involved in technical, complex programs to take special steps to integrate program management and systems engineering disciplines. Several such steps are highlighted in this article.

[2] See Chapter 7 in *IPMSE* for a description of this program and its results.

[3] See *IPMSE*, pp. 109-110.

the International Space Station;[4] a case study of Electronic Support Upgrade for the Royal Australian Navy's Anzac Class Frigate;[5] The Marriage of Systems Engineering and Program Management at Lockheed Missiles & Space Company;[6] Using Certification to Foster Integration in U.S. Government Agency Acquisition Programs;[7] Integrating Software Engineering and Program Management at Nationwide;[8] Managing Change in Engineering Program Organizations: Boosting Productivity in BMW's Engineering Department;[9] and Delivering the World's Most Complex Inner-City Infrastructure Program: Boston's Big Dig.[10]

Credit to INCOSE, PMI, and the Consortium for Engineering Program Excellence at MIT that through superb collaboration, research, and writing, "The Book" was written and is available to make leaders aware of the problems and to help program managers and systems engineers find ways to improve collaboration and integration, with or without explicit organizational change. ("The Book" is available at a discount for members of INCOSE here).

The issue, it seems to me, is that there has been little acceptance of the critical importance of the concerns resulting from the research, and even less action to address critical integration needs on the part of senior leaders.

A concerning problem over the past several decades is that some leaders will freely acknowledge the existence of a problem, yet they seem to lack any meaningful sense of its appropriate urgency. When placed in context with the daily metrics, politics, and challenges leaders now experience, strategic imperatives of many types are at risk. **The message here for senior leaders involved in program management and systems engineering for technical, complex programs is that your attention to the critical need for integration of program management and systems engineering is urgent – these communities need you to act if progress is going to be achieved in improving integration. We have already established that delivery of many effective complex systems in the future is at risk.**

Chapter 16 of "The Book" (pp. 343-363) provides a rich set of calls to action: for Academia: Help Budding Professionals Learn to Adapt; for Enterprise: Build the Right Engine for Strategy Implementation; for Policymakers: Refocus Oversight and Accountability in the Right Ways; for Industry and Professional Societies: Take an Interdisciplinary View; and for Researchers: Explore Interdisciplinary Systems. I have not observed any of these calls to action even acknowledged, much less addressed, by senior leaders in each of these areas, over the past year. Some people feel that these communities are very slow responding systems in general, and also that inertia is a powerful force to overcome. However, given that

[4] See section 9.6 of *IPMSE* for a description of this program, perhaps the most famous of all systems engineering programs, known for its convergence of science, technology, and human innovation.
[5] See section 12.5 of *IPMSE*.
[6] See Section 14.2 of *IPMSE*.
[7] See Section 14.3 of *IPMSE*.
[8] See Section 14.4 of *IPMSE*.
[9] See Section 14.5 of *IPMSE*.
[10] See Section 14.6 of *IPMSE*.

we have an abundance of hard-working, caring, concerned leaders in program management and systems engineering, clearly we can do better.

PMI recently released news that project success rates are improving[11] and that adoption of agile techniques is improving project and program management effectiveness[12]. While this may be true for many types of projects, we have a critical situation concerning large, technical, complex programs that requires urgent action.

Meanwhile, systems engineers are focused on their work, rigorous in their efforts, set in their ways, vocal concerning their views, and perhaps not sensitive to the critical need to further strengthen and improve integration of their activities with project and program managers, or else risk continued unacceptable levels of project and program success.

Another issue is that the value of systems engineering is not always recognized or appreciated. Where is the movement within the systems engineering community to change the existing mindset? As an example, in the United States, the Government Accountability Office (GAO) has noted the need for more effective application of systems engineering in technical programs, but who is pushing the United States Congress to make sure that happens? There are specific types of projects and programs that by law have to apply value engineering practices because of Congressional intervention. Why is there not the same application of systems engineering in major engineering projects and programs?

One area that was not explored in the book is where systems engineering fits within commercial engineering organizations. Is it at a level that can influence organizational systems and behaviors or does it take a back seat to more traditional disciplines? How are chief systems engineers co-leading change efforts? These and related topics are potentially fruitful areas of exploration by members of INCOSE.

Since every project and program has a manager, one would expect that person to have the duty to be the catalyst to ensure that roles and responsibilities between them and systems engineers (and others) are clearly defined and in agreement going forward. The fact that there is a book that can offer guidance and suggestions is a great step forward. The analysis in “The Book” suggests that this is an organizational issue, rather than a professional issue - Project and Program Manager (PM) roles tend to be well defined with clear responsibilities and accountabilities, whereas the Chief Systems Engineer (CSE) role is not. CSEs were clear during the research that the lack of defined roles and accountabilities impacted their ability to be effective. That is why “The Book” recommends that position descriptions with competencies, responsibilities, and accountabilities be clearly defined by the organization so that there is consistency across programs; people are clear about what is expected; and training and development can align to support the competencies and responsibilities for each discipline. Here is an opportunity for practitioners

[11] *The Pulse of the Profession*, PMI, 2018. It should be noted that this publication is a snapshot in time and that the results fluctuate from year to year. Reviewing past *Pulse* reports over the past five years, there have been years when success rates dropped sharply and use of project management techniques declined. So this year or last, depending on which *Pulse* you reference, rates may have improved or not. Government Accountability Office (GAO) reports on government program performance have similarities with reports provided in the *Pulse*. For example, a March 2015 report on select NASA programs found that NASA was making positive strides in improving its engineering program capabilities but there were still areas that required work.

[12] PMI Web site, visited 6 February 2018. See the article in the SE News section of this issue of SyEN.

including chief systems engineers and others involved in various systems engineering roles to change this situation in the spirit of continuous improvement.

I cannot help feeling that unless leaders involved in large, technical, complex programs at all levels in both disciplines aggressively seek heightened awareness of changed expectations, the mindset regarding the source of problems will not change and the critical need for increased integration will be left unaddressed in many future engineering programs. The inevitable result will be continuation of a prevalence of poorly executed complex programs. That the current situation is unsustainable is scary.

An important question: how to motivate these leaders to be proactive?

I believe that all of us involved in projects, programs, and systems engineering need to be leaders. We have to provide the leadership that makes things happen. Some of us are senior leaders, that is to say, in a position to influence the status quo. (Senior leaders must provide, in a positive and constructive way, their vision concerning how to move forward.) Some of us (practitioners?) believe we are "too busy" to get involved in addressing the problem. Perhaps practitioners do not have the bandwidth or stature to bring about the required level of change by themselves. Therefore, there needs to be a coalition for change inside organizations with the PMs and SEs leading it.

Many of the projects that PMI oversees do not have an extensive component of systems engineering. Perhaps PMI can take steps to engage project managers of those projects and programs that do have an SE component to pursue the needed continuous improvement.

Some senior managers and executives may not even be aware that the problem exists. Urgent actions are required by all of us, but especially by senior leaders. This does not need to be a confession that we have not been as effective as we could have been, but rather a recommitment to continuous improvement in our daily work and bold commitment and actions to further strengthen and improve integration of program management and systems engineering.

Here is the suggested path forward for leaders involved in large, technical, complex programs: Get the book, digest its contents, assess the relationship of its information to your personal setting, and consider and prioritize what you can do to improve the situation. Proactively providing continuous improvement in our daily work needs to be our focus.

In a 2009 paper, available here, Armstrong addressed how the worlds of systems engineering and project and program management should be integrated to combine the technical, cost, and schedule performance. The emphasis in project and program management continues to be on methods to track cost and schedule. In theory, the Work Breakdown Structure (WBS) should have technical objectives; however, in practice, this is frequently not well defined. Armstrong describes how the use of Technical Performance Measures (TPMs) can provide a better integration of technical performance into a project or program. He notes that although TPMs have been used in programs effectively for decades, it was surprising that they were observed in systems engineering training classes as the one concept that is most frequently problematic in exercises. He notes that the initial value of TPMs is to proactively forecast the technical progress to be achieved rather than only responding to technical events. TPMs are the primary known way to drive systems engineering activities using real data in a closed loop method. In order to make the technique work, there needs to be a measurement approach that will provide actionable information with sufficient granularity to allow for management response. While the concept of TPMs is straight-forward, their use has been sporadic in many programs and organizations. Armstrong's article provides guidance and information concerning how to implement the key features of TPMs in project processes, directions, and templates. If done correctly, the TPMs can provide an important link in program management and systems engineering effectiveness on a project or program. This is an opportunity for practicing systems engineers to further strengthen and improve the integration of program management and systems engineering in their daily work.

I encourage all those involved in both disciplines, in whatever role, to consider actions that you might take to address these critical issues and concerns, to commit yourself to being proactive in addressing them, and to moving forward collaboratively with our colleagues in a spirit of teamwork and commitment to producing superior results on an increased number of large, technical, complex programs.

Your comments, suggestions, criticisms, and feedback are welcome, as always.

Ralph R. Young

ryoung@ppi-int.com

Integrated Trade-Off Analytics

by

Gregory S. Parnell, Ph.D.

Industrial Engineering Department

University of Arkansas

and

Matthew V. Cilli, Ph.D.

Systems Analysis Division, Systems Engineering Directorate

U.S. Army, Armament Research Development and Engineering Center

Abstract

System decision-making is challenging due to system complexity, technology and environmental uncertainties, and conflicting stakeholder objectives. Systems engineers help other engineers, program managers, and senior organizational decision makers make the difficult performance, cost, and schedule trade-offs that are required to manage risks and make new systems effective, affordable, and timely. In the past, these trade-offs have been informed by three separate groups performing separate performance, cost, and risk analyses. Today's model based systems engineering allows integrated performance, cost, and risk trade-off analytics using descriptive, predictive, and prescriptive analytics. This paper provides a framework to perform integrated trade-off analytics, the Assessment Flow Diagram to implement the framework, and the Trade-off Analytics Hierarchy which communicates the complexity of the trade-off analysis to key system subject matter experts, stakeholders, and decision makers.

Web site: https://uark.academia.edu/GregoryParnell

Email: gparnell@uark.edu

Introduction

System decision-making is challenging due to system complexity, technology and environmental uncertainties, and conflicting stakeholder objectives [4]. Systems engineers help engineers, program managers, and senior organizational decision makers make the difficult performance, cost, and schedule trade-offs that are required to manage risks and make new systems effective, affordable, and timely. In the past, these trade-offs have been informed by three separate groups performing separate performance, cost, and risk analyses. Operations analysts have used descriptive engineering data and modeling and simulation to predict potential system performance in planned system missions and scenarios; cost analysts have used descriptive engineering data and cost models to predict potential system life cycle costs; and engineers and operations analysts have used descriptive engineering data, predictive capability analysis, and risk models to assess and manage system risk. Systems engineers, operations analysts, cost analysts, and risk analysts have attempted to coordinate these separate analyses and include the results in prescriptive models to support systems decision-making.

Today's model based systems engineering (MBSE) allows integrated performance, cost, and risk trade-off analytics using descriptive, predictive, and prescriptive analytics [2]. This paper provides a framework for integrating performance, cost, and risk analysis that uses descriptive system and component design information; predictive performance from models and simulation; and prescriptive data from decision models that assess value, cost and risk.   The flow of data from design choices through intermediate performance calculations to value and cost measures is described in the integrated analysis of alternatives framework presented in the next section. The following section provides an example of the Trade-off Analytics Hierarchy for an Unmanned Aeronautical Vehicle (UAV) trade-off case study. The Hierarchy provides a summary of the descriptive, predictive, and prescriptive analytics performed in the case study which communicates the complexity of the trade-off analysis to key system subject matter experts, stakeholders, and decision makers.

Framework for Integrated Trade-off Analytics for Analysis of Alternatives

In order to help make decisions during the early life stages of system design, we created an integrated framework for analysis of alternatives [7]. Visually, this framework is shown as an influence diagram in Figure 1. An influence diagram is a concise representation of a decision opportunity [3]. Influence diagrams identity the variables and their relationships but suppress the details. Influence diagrams use four nodes: decision nodes, uncertainty nodes, constant nodes, and value nodes. A decision node signifies the decision alternatives or options and is displayed by a rectangle. An uncertainty node represents the different outcomes of an uncertain event and is depicted as an oval. A constant node symbolizes a function or number that will not change and is depicted by a diamond shape[13]. Lastly, an influence diagram has value nodes denoting the decision makers' preferences for potential outcomes. A hexagon depicts a value node. Value nodes can have different types of values such as cost or performance measures, or an affordability based on cost, performance, and service life. In the diagram, arrows are used to display influences. There are two types of influences: a probabilistic relationship and the availability of information. The time sequence of the events is from left to right. Conditional formatting is used to reduce the number of arrows shown in the influence diagram. For example, the annotation, s|r, T means the uncertain scenarios, s, given the requirements, r, and the threat assessment, T.

[13] Constant nodes are not used in Figure 1.

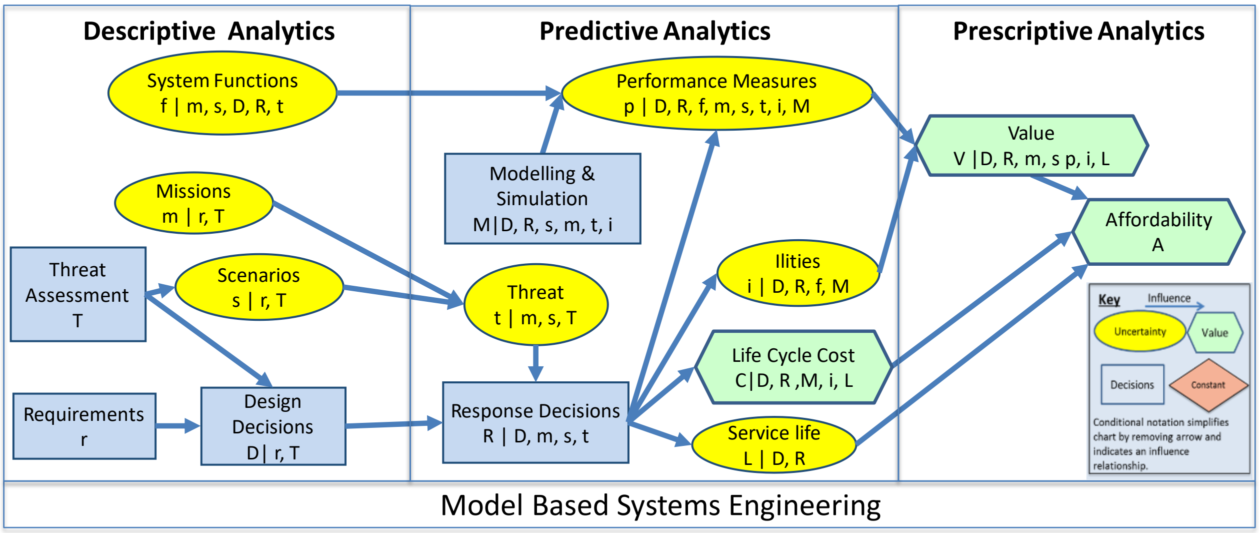


Figure 1 - Framework for Integrated Analysis of Alternatives, updated from [7].

In Figure 1 the definitions of the nodes are:

- Threat Assessment, T - a decision that identifies the anticipated adversary or environmental threats the system could face in the planned missions and scenarios.
- Requirements, r - a set of decisions stating the required minimum performance in the planned system environments and threats.
- Design Decisions, D - a set of system design decisions made with knowledge of the requirements and threat assessment.
- Scenarios, s - a chance node representing an uncertain scenario, which may or may not be in the original threat assessment or requirements analysis.
- Missions, m - a chance node representing the missions the system is actually used on, this may or may not be included in the initial threat assessment or requirements analysis.
- Modeling and Simulation (M&S), M - the decisions made which methods and techniques used to model and simulate the missions and scenarios used to predict system performance measures, ilities, and costs.
- Threat, t - a chance node representing the uncertain threat that depends on the mission. There can be different threats to different system functions. In this diagram, threat is the term used for any adverse event (environmental or adversary) that could degrade any capability of the system. This may or may not be in the original T.
- System Functions, f - a chance node determining how the system is used; it is influenced by the missions and scenarios used in the future system.

- Platform and Mission Resilience Response Decisions, R - a decision node representing mission response decisions (short-term) and platform response decisions (long-term) informed by threats during system operation.

- Performance Measures, p - a chance node depending on the function, the ilities, modeling and simulation, and resilience response decisions.

- Ilities, i - a set of chances such as reliability, survivability, availability, and others affecting the performance and cost of the system.

- Service Life, L - a chance node affected by the performance of the system, the ilities, and the resilience response decisions.

- Value, V - a value node depending on the performance for the mission for all functions and several other variables [3].

- Life Cycle Cost, C - a value node depending on the design, the produceability, the supportability, and the platform and mission response decisions [4].

- Affordability, A - a value node comparing value versus life cycle cost.

The framework is based on four important concepts. First, the framework makes use of models and simulations to create and explore in near real-time a variety of concepts and architectures for each concept. Second, the framework uses Multiple Objective Decision Analysis to convert performance measures to a multiple objective value model [4] the prescriptively defines the value tradespace. Third, the integrated framework means that every design decision is simultaneously propagated through all the performance, cost, and value models. Fourth, since the framework includes uncertainties, the framework can provide an integrated assessment of performance, value, and cost risk.

Assessment Flow Diagram

The framework in the previous section is an important first step to achieving an integrated trade-off analysis. The next step is the Assessment Flow Diagram developed by Cilli [2]. The Assessment Flow Diagram provides the explicit flow of data between the system, subsystem, and component design choices to the models and simulations used to predict the performance, ilities, and cost measures. In addition, the diagram shows the performance measures and ilities used to calculate the overall system value used in the prescriptive multiple objective decision analysis model. We illustrate diagram with an Unmanned Aeronautical Vehicle (UAV) demonstration model developed by our team from a case study provided by the Army [1]. An example of an Assessment Flow Diagram is shown in Figure 2 [8]. The bottom level shows the physical means and uncertainties. The middle level shows the model-based calculations that use the descriptive data from the physical means and ilities. The top level shows the objectives and value measures in the value model. The arrows show the flow of data.

The Assessment Flow Diagram has been shown to be very helpful in communicating to system, subsystem, and component engineers the role of descriptive system data and predictive models and simulations in obtaining the data for the prescriptive decision models.

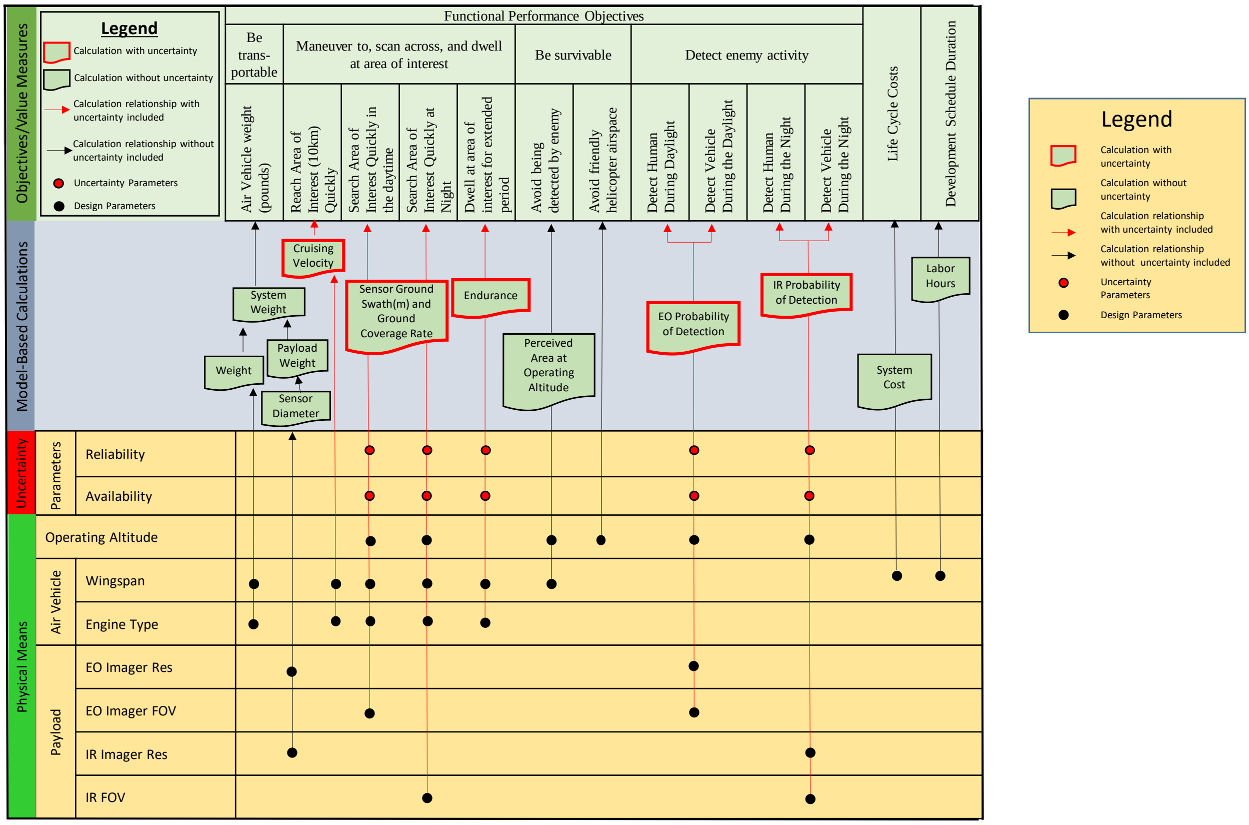

Objectives/Value Measures
Legend
Calculation with uncertainty
Calculation without uncertainty
Calculation relationship with uncertainty included
Calculation relationship without uncertainty included
Uncertainty Parameters
Design Parameters
Functional Performance Objectives
Be trans-portable
Maneuver to, scan across, and dwell at area of interest
Be survivable
Detect enemy activity
Air Vehicle weight (pounds)
Reach Area of Interest (10km) Quickly
Search Area of Interest Quickly in the daytime
Search Area of Interest Quickly at Night
Dwell at area of interest for extended period
Avoid being detected by enemy
Avoid friendly helicopter airspace
Detect Human During Daylight
Detect Vehicle During the Daylight
Detect Human During the Night
Detect Vehicle During the Night
Life Cycle Costs
Development Schedule Duration
Model-Based Calculations
Cruising Velocity
System Weight
Sensor Ground Swath(m) and Ground Coverage Rate
Endurance
EO Probability of Detection
IR Probability of Detection
Labor Hours
Weight
Payload Weight
Sensor Diameter
Perceived Area at Operating Altitude
System Cost
Uncertainty
Parameters
Reliability
Availability
Physical Means
Operating Altitude
Air Vehicle
Wingspan
Engine Type
Payload
EO Imager Res
EO Imager FOV
IR Imager Res
IR FOV
Legend
Calculation with uncertainty
Calculation without uncertainty
Calculation relationship with uncertainty included
Calculation relationship without uncertainty included
Uncertainty Parameters
Design Parameters

Figure 2 - Assessment Flow Diagram for UAV Trade-off Demonstration Model, updated from [8].

Trade-off Analytics Hierarchy

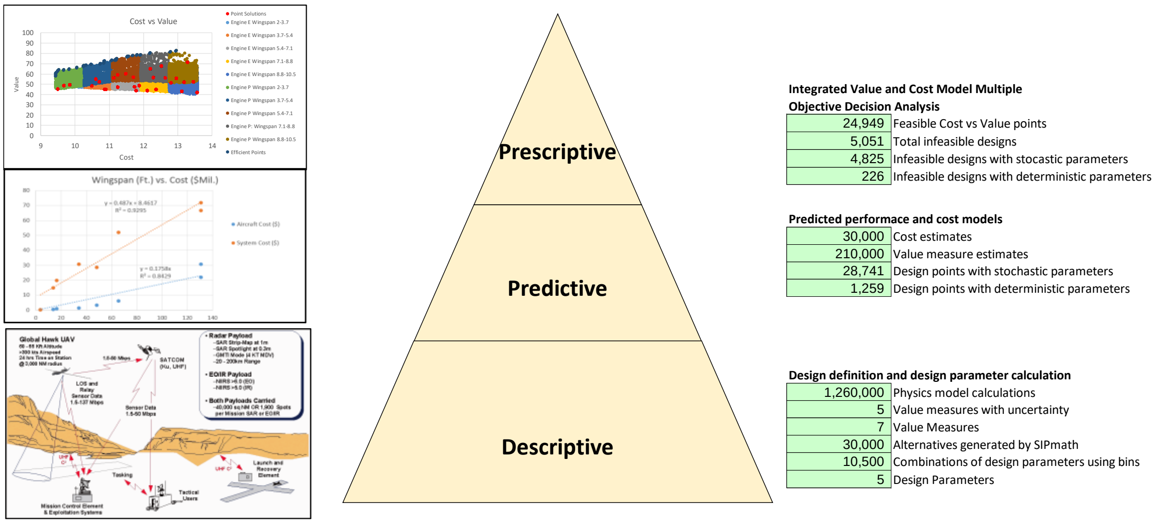
As we have noted, trade-off analytics involve descriptive, predictive, and prescriptive analytics. We have developed the Trade-off Analytics Hierarchy to summarize the use of all three types of analytics in a trade-off analysis. The Hierarchy can be used to communicate the complexity of trade-off analytics to key system subject matter experts, stakeholders, and decision makers.

We defined and explored the design space using an open source Excel add-in called SIPMath from Probability Management [6]. The design decisions are varied by performing Monte Carlo simulations for the continuous and discrete design decisions. The Trade-off Analytics Hierarchy for the above UAV demonstrations is shown in Figure 3 [8].

First, we describe the descriptive analytics. The UAV demonstration had five design parameters. Three of the design parameters were discrete: 2 engines, 15 electro-optical imagers, and 14 infrared sensors. The two continuous variables, wingspan and operating altitude, were discretized into 5 bins each. The result was 10,500 possible design alternatives with discrete design components and selection of a bin for each of the two continuous variables. To explore the design space, the simulation was run 30,000 times creating 30,000 unique possible solutions. Each design was defined by three discrete design components (engine, EO imager, and IR sensor) and two continuous parameters randomly selected from the random bins selected for wingspan and operating altitude. Ten value measures were used in the multiple objective value model developed to define and evaluate the value of the UAV designs. Physics models were used to calculate the scores on each of the value measures. The number of physics model calculations required was 1,380.000.

Second, we describe the predictive analytics. The UAV demonstration model used the physics model calculations to predict 300,000 scores (30,000 times the 10 value measures). In addition, the cost model was used to predict 30,000 costs.

Third, we describe the prescriptive analytics. Some alternatives were eliminated using feasibility checks. Using the acceptable ranges for each value measure score, 250 of the alternatives were determined to not meet the minimum requirements. The result of the prescriptive analytics was 29,750 feasible system designs with associated values and costs to be used in the affordability analysis.



Integrated Value and Cost Model Multiple Objective Decision Analysis

24,949	Feasible Cost vs Value points
5,051	Total infeasible designs
4,825	Infeasible designs with stocastic parameters
226	Infeasible designs with deterministic parameters

Predicted performace and cost models

30,000	Cost estimates
210,000	Value measure estimates
28,741	Design points with stochastic parameters
1,259	Design points with deterministic parameters

Design definition and design parameter calculation

1,260,000	Physics model calculations
5	Value measures with uncertainty
7	Value Measures
30,000	Alternatives generated by SIPmath
10,500	Combinations of design parameters using bins
5	Design Parameters

Figure 3 - UAV Trade space Tool Example of the Trade-off Analytics Hierarchy, updated from [8]

Summary and Conclusions

This paper has provided a MBSE framework for integrating performance, cost, and risk analysis that uses descriptive system and component design information; predictive performance from models and simulation; and prescriptive data from decision models that assess value, cost, and risk. This framework supports the increasing use of model based engineering in systems engineering. Implementing this framework will require significant support and participation from program managers, stakeholders, mission analysts, systems engineers, subsystem engineers, component engineers, operations analysts, risk analysts, and cost analysts. The Assessment Flow Diagram is an important tool to implement the framework.

This paper has also provided an example of the Trade-off Analytics Hierarchy for a UAV trade-off case study. The Hierarchy summarizes the descriptive, predictive, and prescriptive analytics performed in the case study which communicates the complexity of the trade-off analysis to key system subject matter experts, stakeholders, and decision makers. Furthermore, it can help subsystem and component engineers understand the important role of their models and simulations in trade-off analytics.

However, once the framework has been implemented with the Assessment Flow Diagram and the Analytics Hierarchy, the advantages will be significant. First, the framework and the Assessment Flow diagram will improve stakeholder understanding of the trade-off analysis. Second, the integrated framework will provide higher quality trade-off analyses. Third, the required analysis due to changing requirements will be much easier to perform with an existing integrated framework in place. Fourth, the largest benefit will be the selection of better system concepts.

Epilog

For further information on decision analysis, see [3]. For further information on the role of decision analysis in systems decision-making, see [4]. For an example of the integrated trade-off framework and several examples of the use of decision analysis for trade-off analytics in the systems life cycle see the recent textbook on trade-off analytics, see [2]. The framework for trade-off analytics, the Assessment Flow Diagram, and the Trade-off Analytics Hierarchy are key tools we are using in our Engineering Resilient Systems research to develop tools and techniques to improve the DoD analysis of alternatives [5]. We are also employing these tools to illustrate the potential for using set-based design to shorten the system development life cycle and identify better system concepts [9].

List of Acronyms Used in this Paper

Acronym	Explanation
MBSE	Model Based Systems Engineering
M&S	Models and Simulations
UAV	Unmanned Aeronautical Vehicle

Acknowledgements

The research reported in this paper was funded by the DoD Engineering Resilient Systems program managed by the Army’s Engineering Research and Development Center.

References

[1] Parnell, G. S., Driscoll, P. J., & Henderson, L. D. (2011). *Decision Making in Systems Engineering and Management.* John Wiley & Sons.

[2] Specking, E. A., Whitcomb, C., Parnell, G. S., Goerger, S. R., Pohl, E., & Kundeti, N. (2017, September 26-27). Trade-off Analytics for Set Based Design. *Design Sciences Series: Set Based Design*.

[3] Small, C., Parnell, G. S., Pohl, E., Goerger, S., Cottam, B., Specking, E., & Wade, Z. (2017). Engineering Resilience for Complex Systems. *15th Annual Conference on Systems Engineering Research.* Redondo Beach, CA.

[4] Parnell, G. S., Goerger, S., & Pohl, E. (2017). Reimagining Tradespace Definition and Exploration. In Ng, B. Nepal, & E. Schott (Ed.), *Proceedings of the American Society for Engineering Management 2017 International Annual Conference*, *18-21 Oct 2017.* Huntsville, AL.

[5] Probability Management. (2017). *SIPmath*. Retrieved from Probability Management: http://probabilitymanagement.org/sip-math.html

[6] Parnell, G. S., Bresnick, T. A., Tani, S. N., & Johnson, E. R. (2013). *Handbook of Decision Analysis.* Hoboken, NJ: John Wiley & Sons.

[7] Cilli, M. (2017, July 31). Decision Framework Approach Using the Integrated Systems Engineering Decision Management (ISEDM) Process. *Model Center Engineering Workshop, Systems Engineering Research Center (SERC)*.

[8] Cilli, M., & Parnell, G. S. (2017). Understanding Decision Management. In G. S. Parnell, *Trade-off Analytics: Creating and Exploring the System Tradespace* (pp. 155-202). Hoboken, NJ: John Wiley & Sons.

[9] Small, C., Parnell, G. S., Pohl, E., Cottam, B., Specking, E., & Wade, Z. (2017). *Engineered Resilient Systems and Analysis of Alternatives: A UAV Demonstration.* Univeristy of Arkansas, Industrial Engineering. Fayetteville: Center for Engineering Logistics & Distribution (CELDi).

About the Authors

Dr. Gregory S. Parnell is a Research Professor in the Department of Industrial Engineering at the University of Arkansas and Director of the M.S. in Operations Management and Engineering Management programs. He was lead editor of *Decision Making for Systems Engineering and Management,* (2nd Ed, 2011), lead author of the *Handbook of Decision Analysis* (2013), and editor of *Trade-off Analytics: Creating and Exploring the System Tradespace*, (2017). He previously taught at the West Point (Emeritus Professor), the U.S. Air Force Academy (Distinguished Visiting Professor), the Virginia Commonwealth University, and the Air Force Institute of Technology. He is a fellow of INCOSE, INFORMS, and MORs. He has a Ph.D. from Stanford University.

Dr. Matthew Cilli is the Principal Systems Engineering Manager and Data Scientist for the U.S. Army's Armament, Research, and Development Center. In this capacity, he is responsible for leading multi-organizational teams through tradespace exploration efforts to help senior leaders find system level solutions that balance competing trades of cost, schedule, and performance in the presence of uncertainty. Dr. Cilli earned his Ph.D. in Systems Engineering from Stevens Institute of Technology in December 2015 with a dissertation focused on the integration of the holistic perspective of systems engineering discipline with the mathematical reasoning prevalent throughout the operations research discipline. He was first introduced to decision analysis as part of the coursework associated with the Masters of Technology Management from the University of Pennsylvania's Wharton Business School awarded in May 1998. He also holds a Bachelor of Science degree and a Master's of Science degree in Electrical Engineering from Villanova University (1989) and NYU Polytechnic (1992) respectively. Dr. Cilli also serves as an adjunct professor in the Department of Industrial Engineering at the University of Arkansas.

ARTICLE

The Impact of System Integration on Reliability

by

James R. Armstrong

Stevens Institute of Technology

School of Systems and Enterprise

400 Wildberry Court

Millersville, MD 21108

410-987-1267

jimarmstrong29@aol.com

Abstract

Integration objectives are normally focused on whether the system of interest performs its basic functionality in accordance with stated requirements. Secondary considerations, such as effects on reliability, maintainability, or other non-functional or immediate physical attributes due to integration impacts are often left unaddressed until a problem actually arises. By looking at several instances where these impacts had significant impact on reliability, we can glean lessons to guide future integration efforts towards a more fully successful end result.

Introduction

When the question was posed concerning the relationship of integration to reliability and maintainability, a couple of examples came up immediately in which integration had a significant negative impact. This prompted additional review of the topic to discover more examples and see what lessons might be observed. The focus in this paper is on the relationship of integration with reliability. This is interesting in part due to the nature of integration being focused on the immediate results of bringing system elements together and often verified in a relatively short term verification process. Reliability, on the other hand, is characteristically measured over a much longer time scale.

Generally, reliability is not a specific part of the focus in integration planning and execution. However, the ease in producing the examples raised the issue of whether this deserves more attention. This led to an expanded collection of cases that provide emphasis on the relationship and several lessons. Many are from personal experience and others are from research into the phenomenon. Those addressing impact on reliability are provided here along with the description of what to be aware of in planning and executing systems integration.

The relationship between reliability and integration is mostly related to allocation of mean time between failures and similar parameters. The basic assumption is that the component reliabilities are independent of each other. The reality is that there are relationships between the components and subsystems that will affect each other. The examples below provide several different considerations for addressing these interactions as part of the systems integration process.

Examples of Integration Impacting Reliability

Software impacting hardware. The first example is from a contract for computer equipment for DoD which ran into reliability issues. The prime was charged with being responsible because the equipment wasn't meeting Mean Time Between Failures (MTBF) numbers. The prime in turn passed the blame on to the hardware provider, who claimed that the high failure rate was caused by software. The lawyers wanted to know if this made any sense. When asked which hardware and which software, the answer was that a prime culprit was the hard drive and the finger was being pointed at the operating system. It was noted that the operating system could be pushing the duty cycle of the hard drive well beyond design limits. The author doesn't know what the ultimate outcome was, but this seemed a very plausible situation. Also, the selection decisions for these types of systems that rely highly on commercial-off-the-shelf (COTS) components highlight the differences in integration and reliability concerns and approaches. If the software ran on the hardware and passed functionality tests, the integration problem is considered solved. If the available component reliability data says the MTBFs are met, the reliability issue is considered solved. The attitude is often summarized by the statement that this is just a COTS program with a little bit of integration. It is likely that the hardware component of the development team had no awareness of the possible differences in operating system actions and that the software component was not well versed in impacts on hardware performance (other than immediate speed of data entry and retrieval). As a result, it is also likely that there was no early definition of this interface between the software and hardware, particularly with regards to factors that impact reliability.

Subsystem Optimization. A second example is a less complex problem that occurred during the US Defense Communications System conversion of the European microwave network from analog to digital. The engineers came up with an idea to reduce costs for the battery backup. For a large site such as Camp Darby, south of Pisa, it takes a small room full of batteries to keep the system running for the required day or two in case both commercial and generator power fail. Normally, the battery system would provide 48 volts DC and the voltage would drop through use to the lower limit of +/- 10% of the operating range of the communications equipment or down to 42.2 volts DC. An approach was developed that could use half the batteries by setting the normal operating voltage at the 52.8 high end, giving twice the voltage range to drop through under battery operations as shown in Figure 1.

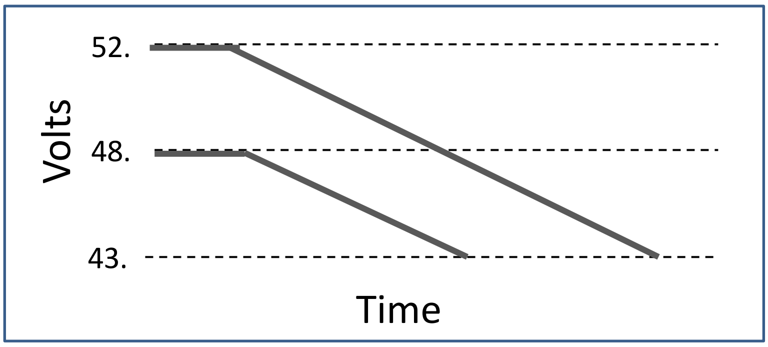


Figure 1. Increasing battery time by changing normal operation voltage setting

During the installation and checkout period, the system experienced a high rate of failures. In fact, the Army supply system stock of 2 GHz oscillator boards was totally exhausted. After some analysis, it was concluded that the culprit was the extra heat from the higher than normal power voltage. Dropping the operating voltage down less than a volt took the system off the knee in the curve and brought reliability back to acceptable limits.

The obvious lesson is that the impact of abnormal behavior by one system can have abnormal impact on another. We often consider the impact on basic performance but not necessarily the less obvious concerns such as reliability. A more general conclusion is that there is an expectation that a system or component meets all requirements in any combination of extremes. Reliability stress testing does include testing at extremes and in combinations of environmental factors with the goal of proving reliability under continuous operations in those conditions. This is particularly the case for equipment to be used in aircraft, ships, spacecraft, vehicles, or other harsher environments. Even fixed-facility equipment may be tested for environmental stress including electrical power variations. However, test methods and guidance such as that detailed in MIL-HDBK-781A vary the input by running at the high input level for 25% of the time, the low level for 25%, and the nominal level for 50%. This would not address the impact on reliability by continuous operations at the upper limit and full reliability performance should not be assumed as verified by that test. This should have been considered as part of the total systems integration of the modified design.

Classic hardware complexity. Additional research identified a typical situation in hardware complexity as shown in the case of a fishing vessel power system (Techno Fysica, n.d.). Again, the system worked well in what might be considered nominal power operations. However, the problems occurred when the fishing ship was working nets with the engine at slow speed and the prop disengaged.

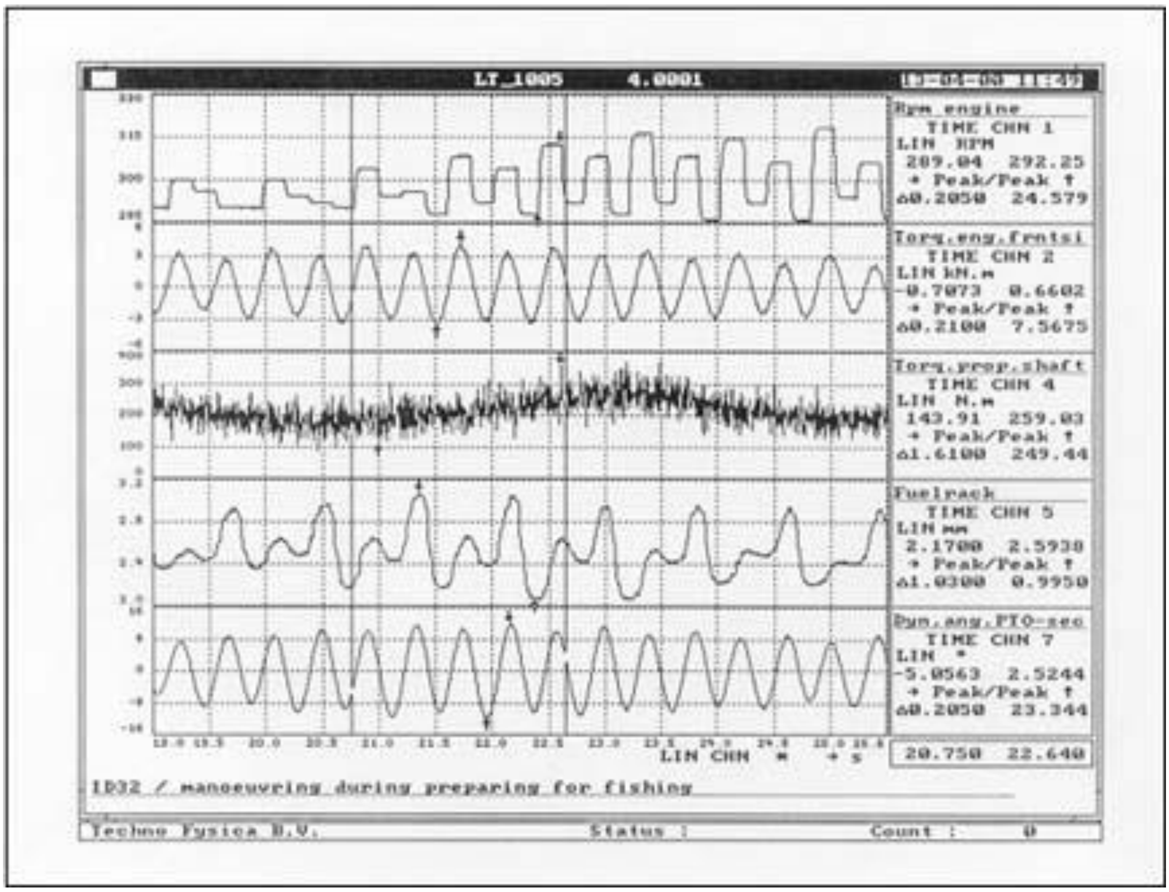


Figure 2. Low Load Engine Behavior (Techno Fysica, n.d.)

The instability of the governor during low speed engine operations, depicted in Figure 2, led to high stress on the rubber joints between the engine and the generator driving the winch for the nets. The joint was intended to dampen the vibration but stress in this mode of operation was too much. The rubber overheated and repeatedly failed. A redesigned joint was able to handle the increased load.

This is a case of needing to look not only at how the pieces fit together and work in normal operations, but also during unusual operations. At the component level, the governor and engine design were adequate to perform their functions. The joint also met its requirements for normal behavior. As in the last example, the extended low speed operations and the governor and engine variation in performance were not communicated to the joint design team. It is likely that there would have been some testing of the net operations. However, it would also likely be of relatively short duration and not enough to reveal the reliability problem. In any case, this problem would be addressed better through proper treatment of interface requirements, including a better understanding of the specific operations and extended exposure to extreme situations for inclusion in the design analysis.

Operating environment I. The integration into the operating environment is a broader topic that is generally considered an integration issue. Factors such as temperature, vibration, and humidity are normally considered in related analysis or testing. However, they may be missed in early design efforts if good systems engineering practices are not followed. An early effort to improve reliability of avionics determined that one of the principal factors in poor reliability performance was selection of components that were not able to withstand the various environments to which they were exposed. The behavior that led to their selection was a choice of alternatives based on the advertised parameters in data sheets rather than defined requirements based on the actual environments. It was noted that the data sheets tended to skip particular parameters at which the product was not excellent and competitive. This would result in less reliable performance. When asked about the missing data, the manufacturers agreed that they would be willing to provide the information if all other candidates would do the same. However, they were not particularly interested in advertising weaknesses when competitors did not. Of course, from an integration standpoint, it is of interest that there were not a lot of issues with failure to consider basic performance parameters such as signal strength, bandwidth, etc. that affect basic system integration performance issues. Environmental parameters, on the other hand, are generally easier to ignore at that level of design.

Operational environment II. There are other environmental factors to consider related to the users and their behaviors in operational situations. One such case that was related was that of night vision goggles that had a much higher failure rate than designed and tested by the producer. When tracing to the root cause, they found that the users would toss them unprotected into the back of their vehicles resulting in higher failure rates. A container had been provided with the original goggles to protect them from the environment they were exposed to. However, the container also made a great beer cooler and was not being used for its intended purpose. The solution was to provide some Styrofoam coolers.

A bit more direct situation is defined in any of multiple cases where controls have been placed where crews, particularly flight crews, would step on them climbing in and out of cockpits. There are lots of other examples of lack of consideration for the various external environments the user will be experiencing, but these are ones that directly affect reliability and lead to breakage during use.

Technology and weather environment. Brian Saucer (Saucer et. al., 2009) has developed an approach to analyzing integration readiness of new technologies. His approach is directed towards the combination of technologies within the operational system. However, the introduction of new technologies into a new external environment should also be considered and can have a strong impact on reliability. The first solid state instrument landing system, the AN/GRN-27, was based on a French design that had worked well in Europe. It was one of the first applications of solid state technology in an outdoor environment. The basic reliability testing went well as did an extended field test at multiple operational sites. However, failures started appearing at the sites in areas more prone to thunderstorms. Solid state is far more sensitive to voltage spikes on either power lines or communications and control links. What followed was not only a modification of the design but additional research to find ways to protect the systems better in the future.

Every new technology that claims to bring solutions to existing problems seems to also bring new problems. Often, they are similar to this example and something that didn't have to be considered with the old technology or wasn't as much of a concern. We need to address these "unknown unknowns" not only from an operational perspective, but also from the perspective of their impact on reliability.

Maintenance environment. One communications system had reliability problems that were site specific. It was finally determined that the difference between the sites was the frequency of floor waxing. At the sites that waxed more frequently, wax dust produced by the polishing machines would clog the cooling air filters which were located at the bottom of the racks.

These are the types of things that will often not show up in specifications or come up in requirements reviews. They can be addressed through early validation actions such as a site visit and observations of what the users are doing in all aspects of their activities, not just the operations of the system in question.

Storage environment. Even structural components can have issues with integration that affect their reliability. The shelters for the instrument landing system ground electronics are quite reliable in operational use. However, the first solid state system needed to store several systems as they were produced at a faster rate than the deployment schedule could handle. During this period, they sat idle in a storage lot. After some time, moisture seeped into the walls of the shelters and turned the glue holding the aluminum sides to the frame acidic. The glue then ate away the aluminum it was in contact with until the sides fell off. In this case, the integration of the storage environment was not fully addressed or verified.

Cultural Environment. In another instance, the developer of a commercial system was experiencing difficulty with a penalty clause in its contract for reliability. The issue was that the system expected training to be provided to the operators in the manner that their normal customers did it. The new customers did not have the same training culture and the impact on the reliability was both significant and predictable. While one could argue that “it’s the customer’s problem”, such an approach is not in the best interests of maintaining good customer relations. In this case, the solution was to provide extra training working directly with the operators.

Integration with user systems. One commercial system was developed to work well with a reasonably current configuration of user systems. However, many small business users were still operating with considerably older computers than the integration addressed. The result was a significant increase in system errors and failures in addition to slow performance in general. A variant of this problem is often seen when organizations implement software solutions that only work well when the user is in the facility with direct network connections or have other access to reasonably high speed internet access. However, if one is on the road and at a hotel or other location with either a lower speed or an overused router, the software performance becomes unreliable and sometimes unusable.

Changes. Problems with changes have long been known as a problem. Apollo 13 is a classic example of problems with making a change to hardware. One example of hardware changes in the system context is evident in the DC Metro incident of January 12, 2015 (HSEMA, 2015). The problem arose when a Metro train encountered smoke in the tunnel leaving the L’Enfant station in Washington. The train was stuck in place waiting for the first responders to evacuate the passengers. During the wait time, the passengers were breathing heavy smoke resulting in multiple injuries and one death. There were several factors involved in why this situation was more problematic than it should have been. However, full integration of a new communications device was one significant contributor. The fire department personnel arriving on site faced sporadic operations of their new digital radios in the station and had to use slower methods to communicate information such as confirmation that the third rail electric power had been turned off.

The new radios had gone through a basic set of tests to verify that they worked in the Metro stations two years prior. Problems were addressed and the switch to the new radios proceeded in 2014. Additional tests at three Metro stations indicated further connectivity problems, including at L'Enfant Plaza. However, these had not been fixed by the time or the incident. System reliability had not been fully addressed in the integration of the new radios with the fire department and also the systems environment within Metro.

Software changes. The failure to properly integrate software changes is becoming a significant concern in its own right. One early instance is a software update that went wrong in the AT&T communications network in January 1990 (Carlucci, 2013). This particular modification can be considered an integration issue in two ways. First, it is the integration of a change into an existing system that results in failure of the system. Second, the problem was that the failure wasn't that the individual switch didn't work, but rather that the interaction between switches caused the failures. If one switch received a status update from another switch while it was still processing the original status message, it couldn't handle it, and crashed. That, in turn, generated status messages to other switches which got updated and crashed other switches. The chain reaction caused an extensive outage throughout the system affecting thousands of users.

More recently, American Airlines ran into a problem that canceled several flights (Boulton, 2015). An extra chart of Reagan National Airport in the database caused problems in the database software and a crash of the iPad App that the pilots use. The result was delays in multiple flights while the pilots and airlines resolved the problems.

Software complexity. One example of complexity affecting system reliability is an outage at Facebook. Hoff (2010) provides an extensive review of the details. Basically, the interaction of the applications, database, and cache layers got caught up in a series of requests and responses forcing similar requests that locked up the system similar to the AT&T example. In the commercial environment of providing frequent new features and doing so in a complex architecture, full integration testing is not likely to be accomplished. The trade is between performance now and system reliability. In systems that are not safety critical, this may be an acceptable trade to make.

Summary

In this article, we have addressed the concern: why do integration aspects frequently impact reliability? Doesn't the current guidance tell us how to avoid these problems? The answer is only partly. As noted in the introduction, a review of technical engineering standards, texts, handbooks, and other guidance results in little to counter the natural tendencies to focus on other issues in integration. For instance, the INCOSE Handbook doesn't exclude consideration of reliability impacts of integration; it just isn't included. The focus is on assembly of the system of interest. Enabling systems are mentioned as the integration support rather than the maintenance system. Emphasis is placed on interfaces associated with assembly. The eight different strategies for assembly are all in terms of alternative component assembly sequence. There is no discussion of possible secondary impacts or early analysis.

Similar views of integration can be found from review of ISO/IEC15288, EIA632, IEEE1220, the Defense Acquisition Guide, The Capability Maturity Model Integration (CMMI), and multiple textbooks including Blanchard and Fabrycky, Kassiakoff and Sweet, Rechtin, and Buede. One exception is Applied Space Systems Engineering (Larson, et al) which does include reliability as a factor to address in integration.

System integration is normally focused on the basic operational performance of the system: will the pieces fit together and operate with each other? Other impacts on factors such as reliability are not as well addressed in the analysis or testing. Maintenance is generally recognized as a critical enabling system and is often addressed better. As noted above, there are many examples of both reliability and maintenance issues that can be traced to the need for more concern as part of system integration.

From these cases, there are several lessons to be noted:

1. Expand the scope of what systems integration should address beyond simply putting the pieces together and focusing on basic functionality. By asking the questions that address the full range of system requirements, the risk of many of the problems described above can be significantly reduced.

2. Provide more emphasis on systems integration in the early phases of the program. This is the time that design decisions are made and analysis of the reliability and maintainability are performed. The author provided additional description of other problems that result from a lack of early systems integration and applies in this situation equally well (Armstrong, 2014).

3. There are several other areas that can be considered as having systems integration issues stemming from the same basic problems in limiting the thinking of what systems integration should involve. Security is certainly one of the most recently rising concerns.

4. Software brings a special concern for system reliability. In most cases, a hardware failure can be contained within a limited part of the overall system and won’t propagate. However, as the AT&T example shows, a software problem can spread through the entire system and provide the same effect as multiple hardware failures occurring simultaneously.

5. Integration with the external environment should always be the ultimate system integration concern. Several of these problems, and many more not presented here, are the result of not addressing the complete external environment. Although most of the guidance references do address deployment or transition in some form, the normal tendency is to consider the job completed when the parts of the system of interest are put together.

Conclusion

Hopefully, this discussion of examples of systems integration impacts on reliability and similar non-functional concerns such as maintainability will bring attention to an aspect of the process that is often overlooked or, at least, not given appropriate attention. There are actions we can take as systems engineering practitioners, educators, and discipline definers to address these issues. We must treat system integration as more than the assembly of components. And we must recognize that systems integration takes place throughout the entire life cycle from first concept to final disposal. In doing these two things, we can start a broader consideration of what integration should address through inclusion in early analysis, modeling, and trades related to a wider range of systems concerns such as reliability as discussed in this paper. Practitioners have some ability do this at any time on a project. Educators, standards writers, and others who define and develop the practice of systems engineering incorporate these changes to affect the current and future practice of systems integration. The various textbooks, standards, handbooks, etc. that currently define systems engineering need to incorporate these changes and considerations to provide a better foundation for the discipline. Lastly, if you have experience, examples, or ideas with regards to systems integration, whether or not in agreement with this paper or even not directly related, make it public through a paper, article, blog, contribution to a working group, or other means. Systems integration falls far behind other topics such as requirements, architecture, test, and modeling in the volume of information available as compared with the overall volume of documentation on systems engineering. The result of these actions will lead to improvements in the definition and practice of systems integration and will also result in delivery of higher quality systems.

List of Acronyms Used in this Paper

Acronym	Explanation
AT&T	American Telephone and Telegraph Company
CMMI	Capability Maturity Model Integration
COTS	Commercial Off-the-Shelf
DC	Direct Current
DC	The District of Columbia in the United States of America (Washington, D.C.)
DoD	The Department of Defense in the US
EIA	Electronics Industries Association
HSEMA	Homeland Security and Emergency Management Agency, in the United States government
INCOSE	International Council on Systems Engineering (see www.incose.org/)
ISO/IEC	International Organization for Standardization (ISO) / **International Electrotechnical Commission (IEC)**
Metro	The Washington Metro, known colloquially as Metro and branded Metrorail, is the heavy rail rapid transit system serving the Washington metropolitan area in the United States

MIL-HDBK	Military Handbook
MIL-STD	Military Standard
MTBF	Mean Time Between Failures
n.d.	no date
UK	United Kingdom
US	United States

References

Armstrong, J. R. (2014). *Systems Integration: He Who Hesitates Is Lost*. INCOSE International Symposium, Henderson, NV.

Blanchard, B. S. and Fabrycky, W. J., (2006), *Systems Engineering and Analysis*, Pearson Prentice Hall, Upper Saddle River, NJ.

Boulton, C., (April 29, 2015), *American iPad App Glitch Traced to Database Software*, The Wall Street Journal. Retrieved from: http://blogs.wsj.com/cio/2015/04/29/american-airlines-apple-ipad-glitch-traced-to-database-software/.

Buede, Dennis M., (2000), *The Engineering Design of Systems, Models and Methods*. Hoboken, NJ: John Wiley & Sons, Inc.

Carlucci, R. A., (November 1, 2013), *January 15, 1990 AT&T Network Outage.* Retrieved from: https://prezi.com/gdqfcqifvnkr/january-15-1990-att-network-outage/.

Capability Maturity Model Integration, Development, Version 1.2, Carnegie Mellon University, August, 2006.

EIA 632, Process for Engineering a System, Coordination Draft, Electronic Industries Alliance, Arlington, VA, May, 1998.

Federal Aviation Agency. (1999). *Knowledge and skill areas for systems engineers*. Washington, DC. Written by the author of this article.

Hoff, T, (September 30, 2010). *Facebook and Site Failures Caused by Complex, Weakly Interacting, Layered Systems,* Retrieved from: http://highscalability.com/blog/2010/9/30/facebook-and-site-failures-caused-by-complex-weakly-interact.html

Homeland Security and Emergency Management Agency (HSEMA), (January 23, 2015). *Initial District of Columbia Report on the L'Enfant Plaza Metro Station Incident on January 12, 2015, Executive Summary*, District of Columbia.

IEEE 1220, Standard for Application and Management of the Systems Engineering Process, IEEE, 345 East 47th Street, New York, NY 10017, 1999.

INCOSE (2015), *Systems Engineering Handbook, a Guide for System Life Cycle Processes and Activities.*

INCOSE UK Advisory Board. (2006). *Systems Engineering Competencies Framework*. London.

ISO/IEC 15288, Systems engineering – System life cycle processes, ISO, 1 Rue de Varembe, CH-1211 Geneva 20, Switzerland, 2002.

Kossiakoff, A. and Sweet, W. M., (2003), *Systems Engineering, Principles and Practices*. Hoboken, NJ: John Wiley & Sons, Inc.

MIL-STD-499B, *Systems Engineering*, May 1, 1995.

MIL-HDBK-781A, Handbook for Reliability Test Methods, Plans, and Environments for Engineering Development, Qualification, and Production, April 1, 1996.

Rechtin, E. (1991) *Systems Architecting: Creating and Building Complex Systems*. Englewood Cliffs, NJ: Prentice Hall.

Sauser, B. J., Long, M., Forbes, E. and McGrory, S. E. (2009), 3.1.1 Defining an Integration Readiness Level for Defense Acquisition. INCOSE International Symposium, 19: 352–367. doi:10.1002/j.2334-5837.2009.tb00953.x.

Techno Fysica, (n.d.), *Overloaded Elastic Coupling as a result of Governor Instability*. Retrieved from: http://www.technofysica.nl/English/case2.htm.

Weiss, T. R. (May 2, 2015), *Errant App Caused iPad Problems for Some American Airlines Pilots*, eWeek, retrieved from: http://www.eweek.com/mobile/errant-app-caused-ipad-problems-for-some-american-airlines-pilots.html.

Carlucci, R. A., (November 1, 2013), *January 15, 1990 AT&T Network Outage.* Retrieved from: https://prezi.com/gdqfcqifvnkr/january-15-1990-att-network-outage/

INCOSE (2015), *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities.* Version 4. Compiled and edited by D. E. Walden, G. L. Roedler, K. J. Forsberg, R. D. Hamelin, and T. M. Shortell. Hoboken, NJ: John Wiley & Sons, Inc.

The author requests feedback, suggestions, and ideas concerning this paper - please email them to jimarmstrong29@aol.com

ARTICLE

Integrating Program Management and Systems Engineering

by

Dr. Ralph R. Young, Editor, SyEN

"It's not that we don't know (or can't find out) what to do;

it's that we don't invest the necessary time and effort to do it."

This month we provide a summary of Chapter 8, Program Management and Systems Engineering Integration Processes, Practices, and Tools, in *Integrating Program Management and Systems Engineering* (IPMSE), a collaboration of the International Council on Systems Engineering (INCOSE), the Project Management Institute (PMI), and the Consortium for Engineering Program Excellence (CEPE) at the Massachusetts (USA) Institute of Technology (MIT). This is our tenth article in this series.

Our objective in providing this series is to encourage subscribers to leverage the research base of this book that took place over a five-year period and provided new knowledge and valuable insights. The book is available to members of INCOSE at a discount here.

All aspects of integration are about individuals and how they coordinate the application of their collective knowledge, expertise, and capabilities to deliver results. Effective integration efforts are accomplished by concerned and motivated individuals through the application of processes, practices, and tools that help to enable several important abilities:

- Enable communication and common understanding related to the key objectives and activities to accomplish those objectives.
- Provide frameworks for defining specific work activities.
- Establish expectations of each person's contribution.
- Document approaches for coordinating and tracking work efforts.
- Identify critical points where individual and group work efforts must come together.
- Facilitate problem identification and resolution.
- Apply generally accepted approaches that have demonstrated effective results under similar circumstances in the past.
- Support and accelerate the accomplishment of specific work activities.

Some processes, practices, and tools are designed for individual use, while others may be structured for group activities. Both uses have appropriate application within complex programs. It is incumbent on the user or users to apply them in a way that facilitates integration across disciplines within a team. If users apply the processes, practices, and tools in ways that focus on their respective functions at the expense of collaboration with other disciplines, then there is likely to be only limited integration. On the other hand, if processes, practices, and tools not only define the work to do, but also do it in a way that embeds collaboration, communication, and shared decision-making in the tasks, then integration is much more likely.

No matter which integrating processes, practices, and tools are used, all will likely need to be tailored to the specific program context. This chapter makes the point that processes, practices, and tools should be deliberately designed and implemented in the program ("tailored") as part of efforts to improve integration between the program management and systems engineering disciplines.

The processes, practices, and tools discussed in this chapter are organized by the timeline of their impact on integration: episodic or pervasive. Episodic integration emerges as the need requires. It is driven by periodicity in that it arises at points along the program timeline and is typically a result of overlay processes governing the program life cycle. Pervasive integration tends to be synchronous with the day-to-day work of the program or its component projects. Here the opportunity for integration runs contiguously.

Episodic integration mechanisms are applied occasionally to certain activities or at specific intervals within a program. These mechanisms are not daily drivers of integration between program management and systems engineering, but rather represent periodic forcing functions that require program managers and chief systems engineers to work together closely to produce successful outcomes.

Examples of episodic integration mechanisms include:

- **Program Gate Reviews.** Gate reviews require that all program aspects, such as cost, schedule, performance, risk, requirements, and testing, be presented in their current state of maturity at each individual gate. These details allow the governance body to evaluate the overall program viability and make appropriate decisions. Gate reviews require that the program managers and chief systems engineers work together to prepare the case for the program's advancement to the nest gate. Gate reviews are, therefore, one of the few integration mechanisms utilized by most organizations. In higher performing organizations, the reviews provide a sanctioned vehicle for close collaboration and subject matter expert review that is embraced by the various functions to assess where program performance is in relation to plan. Ideally, the reviews are coupled with the other processes, practices, and tools discussed in this chapter. In the highly successful Super Hornet case study that was summarized in SyEN 62 (February 2018), the "12 days of August" gate review provided the opportunity to gather all the disciplines together to make the necessary trade-offs among competing requirements and interests. This resulted in a tight set of requirements that could be evaluated for cost and turned into system specifications for the subsequent phase of the program. This bold action by the Program leadership was key to the success of the Program.

- **Joint Planning.** Concept development and planning for engineering programs combines three critical components, each of which must integrate with the others:
    - The **product concept** attempts to define the interrelationships between the value of the program, customer needs, and product requirements associated with the strategic opportunity.
    - The **business plan** validates the strategic opportunity by evaluating the product's alignment with business strategy, the market for the product, the level of investment required for the development and production, and the return to the organization on its investment.
    - **Program organization and processes** outline how the organization will develop and produce the product, including critical program activities, the associated human and other resources, stakeholders who will be engaged, and governance.

In addition, each of the above components identifies potential risks that could impact the program both positively, in terms of new opportunities, and negatively, in terms of threats that could hinder success. Research consistently indicates that effective execution of these planning and scoping activities at the start of a program can improve its overall performance and its ability to deliver the desired business benefits. The lack of inclusive and coordinated planning has been recognized as one of the most common sources of problems in programs, leading to unproductive tensions across different areas and between team members involved in the product development process.

Dedicated Team Meeting Space. The creation and use of dedicated team meeting space and standup meetings is a proven process in a variety of domains. Toyota helped popularize this concept for managing programs in recent years through its use of the "obeya".[14] This approach has spread across industry sectors and has gained even broader acceptance as "agile" approaches for managing programs and projects has grown. An obeya provides a dedicated common space for teams and sub teams to meet. The name comes from a Japanese term that translates into English as "big room" or "war room" and refers in traditional practice to a room where a cross-functional team meets to figuratively or literally break down the product completely and investigates changes to it in real time. This allows the team to make rapid trade-off decisions that are acceptable to the multiple perspectives of the team. On the walls, teams usually affix a summary of the program goals, key milestones, deliverables, and a key performance indicator dashboard containing metrics and graphics that describe the current state of the program. It is also quite

[14] Obeya or Oobeya refers to a form of project management used in Asian companies and is a component of lean manufacturing and in particular the Toyota Production System. Analogies have been drawn between an obeya and the bridge of a ship, a war room and even a brain. Wikipedia · Text under CC-BY-SA license

common to have prototypes of the product or parts of the product, drawings of the product, and charts depicting the system architecture, risks, issues, the status of action items, and so on. The obeya thus serves as a communications channel for all disciplines involved in the development activities of the program, which encourages integration of processes, methods, teams, sub teams, and on.

- **Pulsed Product Integration and Iterative Development.** Drawing from concepts from Agile development and the fast prototyping approach, pulsed product integration and iterative development is sometimes described as the "daily build" of product components into more complex components or into complete products. Iterative development comprises the use of short cycles to create and deliver product increments, parts of the product, or other deliverables related to a program. It is time-boxed, which means the length of the short cycles is the same throughout the program. Systems engineering ensures that the diverse elements come together to produce viable systems, and program management ensures that the viable systems will produce the benefits desired for the program.

Examples of pervasive integration mechanisms include:

- **Standards, Methods, and Assessments.** Recent studies have shown that programs with greater integration and better performance often present common characteristics related to the proper use of standards[15] to build an integrated program development methodology. A *methodology* is the means by which teams within an organization apply a level of consistent discipline to their activities and a documented approach for integrating interacting or interdependent practices, techniques, procedures, and rules to determine how best to plan, develop, control, and deliver a defined objective. As the methodology is implemented, executive leaders and users must evaluate the specific practices and the extent to which the organization is adhering to its methodology. Such assessment leads to continuous improvement in the methodology as its user community moves closer to being "best-in-class" performers.

- **Integrated Product and Process Development.** Integrated product and process development, also known as simultaneous engineering or design-build, uses multidisciplinary teams in design to jointly derive requirements and schedules with equal emphasis on product (i.e., design) and process (i.e., manufacturing) development. This approach uses multifunctional, integrated teams that are preferably co-located. The integrated team includes the primary functions involved in the design process; technical process specialties such as quality, risk management, and safety; business groups such as finance, legal, procurement, and other nontechnical support; and customer or market representatives (or advisors) that will be the "voice of the customer".

[15] A *standard* is a document, established by consensus and approved by a recognized body, which provides, for common and repeated use, rules, guidelines or characteristics for activities or their results, aimed at the achievement of the optimum degree of order in a given context. Source: Eric Rebentisch, Editor-in-Chief, *Integrating Program Management and Systems Engineering: Methods, Tools, and Organizational Systems for Improving Performance*, 2017.

- **Work Design Processes.** Work design processes such as configuration management can help to increase communication and collaboration across the program. Another work design process is standardized work, for example, rigorous design standardization supports platform reusability. Standardized work can help improve the flow of work within a program. Some organizations view their integrated teams and continuous improvement of their standardized processes as a competitive advantage. The key point in this discussion of work design processes is that work processes may either isolate functional disciplines from one another or integrate those disciplines. The best examples demonstrate that work processes are deliberately designed so that integration is a natural outcome of the work itself. These tailored work packages should be intimately connected with standards, methodologies, and assessments.

- **Requirements Management.** Requirements management is another pervasive mechanism that forces conversation between program managers and chief systems engineers. Effective requirements management practices help program managers and chief systems engineers align their work so that customers receive preferred solutions and desired program benefits, and value is realized for the business. Requirements management is also one area of potential conflict between program managers and chief systems engineers. The program manager is pressured to keep activities on track and the chief systems engineer is challenged to elicit, document, and validate good requirements for design and development ("evolve the real requirements").[16] If the two cannot effectively collaborate with customers and other stakeholders to ensure there are stable requirements, both may share responsibility for program failures associated with cost, schedule, performance, and solutions.

- **Risk Management.** Every engineering program features some level of uncertainty and risk that must be managed so the program manager delivers the solutions customers expect within established parameters. Effective risk management must also ensure that the sponsoring organization realizes its desired benefits. That is why risk management practices must be pervasive and integrated at the program level. It is also important that the program manager and chief systems engineer work together in identifying and managing risks. It is easy to fall into the trap that the program manager focuses on business risk and the chief systems engineer focuses on technical risk, but these risks are often interrelated such that risks in one area may have implications for risk in others. Using risk management as an opportunity for better integration will often result in surfacing risks that, in isolation, may be missed totally.

- **Technical Performance Measurement (TPM).** There are numerous decision-making tools, processes, and methods for use by program managers in collaboration with chief systems engineers, but very few techniques call for integration through technical performance

[16] See *Effective Requirements Practices* (Addison-Wesley Information Technology Series) for an elaboration on ten proven effective requirements practices and a detailed discussion and process for evolving the real requirements. Available at https://www.amazon.com/Effective-Requirements-Practices-Ralph-Young/dp/0201709120/ref=sr_1_1?s=books&ie=UTF8&qid=1517837716&sr=1-1&keywords=effective+requirements+practices

measurement. Such methods are useful for identifying potential risks in an integrated and collaborative approach using various parameters such as technical, cost, and schedule. The technical performance measures (TPMs) serve as a key program management/systems engineering integration point to ensure program success – both business and technical. A well-thought-out program of technical performance measures provides an early warning system for review of technical problems and supports assessments of the extent to which operational requirements will be met as assessments of the impacts of proposed changes in system performance. TPMs can help identify tradespace for the program manager, develop the program's test plan, and provide key inputs into major program decisions.

- **Governance.** Governance is a structured mechanism through which individuals with oversight responsibility and authority provide guidance and decision-making for important organizational activities. Within the program and project domains, an organization's governance structure reinforces an integrated management approach between program management and systems engineering.

What's Different about an Integrated Approach?

Subscribers may be interested in a comprehensive summary of what is different about an integrated approach – access and download this file.

A Concern Regarding the Status of the Initiative to Strengthen Integration of Program Management and Systems Engineering

The editorial provided in this issue expresses a perspective concerning this initiative that should be considered by everyone involved in program management and/or systems engineering, or serving in a management or executive role concerning these disciplines.

Summary

This chapter provides a wealth of information and advice for the systems engineer. We believe that your time digesting it and applying the suggested practices, methods, and tools will serve both you and your organization well. You might consider:

- How might the processes, practices, and tools presented in this chapter be useful to a program with which you are currently involved?

- How might you use one or more of the processes, practices, and tools discussed in this chapter on your project, program, or in your organization as a means by which to encourage greater collaboration between program management and systems engineering?

- Reflecting on programs you have been involved with in the past, how might your organization have benefited from greater collaboration between program management and systems engineering?

- Do you see evidence of integration processes, practices, and tools that are being practiced between program management and systems engineering in your organization? Is a mechanism in place to share and leverage these best practices among projects, programs, and organizations?

- That which you can do, personally, to foster continuous improvement in your daily work.

References

[1] Rebentisch, Eric, Editor-in-Chief. *Integrating Program Management and Systems Engineering: Methods, Tools, and Organizational Systems for Improving Performance*. Hoboken, New Jersey USA: John Wiley & Sons, 2017.

[2] Conforto, E. C., M. Rossi, E. Rebentisch, J. Oehmen, and M. Pacenza. *Survey Report: Improving Integration of Program Management and Systems Engineering*. Whitepaper presented at the 23rd INCOSE Annual International Symposium, Philadelphia PA (USA). A downloadable copy is available at https://dspace.mit.edu/bitstream/handle/1721.1/79681/Conforto%20et%20al%202013%20-%20PMI%20INCOSE%20MIT%20Survey%20on%20Integration%20of%20Program%20Management%20and%20Systems%20Engineering.pdf?sequence=1

[3] Lucae, Sebastian. Improving the Fuzzy Front-end of Large Engineering Programs – Interviews with Subject Matter Experts and Case Studies on Front-end Practices. Diploma thesis, Nr. 1392. Retrieved from http://cepe.mit.edu/wp-content/uploads/2014/04/SL_Improving-the-fuzzy-front-end-of-engineering-programs_print.pdf on February 7, 2018.

[4] Reiner, Thomas. *Determination of Factors to Measure the Effective Integration between Program Management and Systems Engineering.* Rheinisch-Westfalische Technische Hochschule (RWTH), Aachen, Germany: Master's Thesis, 2015.

[5] Armstrong, James R. "Tracking the Technical Critical Path: Systems Engineering Meets Project Management". Paper presented at the 2009 INCOSE/APCOSE Conference. Download this paper here.

SYSTEMS ENGINEERING NEWS

PMI Acclaims Benefits of an Agile Project Management Approach

Agility is the ability to quickly sense and adapt to external and internal changes to deliver relevant results in a productive and cost-effective manner. Being agile is a mindset based on a set of key values and principles designed to enable collaborative work and deliver value through a people-first approach. Agile transformation is an ongoing, dynamic effort to develop an organization's ability to adapt rapidly within a fast-changing environment and achieve maximum value by engaging people, improving processes and enhancing culture.

In today's accelerated market, a culture of organizational agility that enables flexible use of the right approach for the right project is an essential strategy. As the leading association for more than three million project, program and portfolio management professionals around the world, PMI has long been an advocate for organizational agility. PMI believes practitioners should consider the full range of project management approaches, from predictive to agile, in determining which method will deliver the best project outcomes.

Varied approach

Being agile is a topic of growing importance in project management. The most forward-thinking organizations are embracing a continuum of practices that range from predictive to agile, well defined to iterative, and more to less controlled. Approximately a quarter of organizations use hybrid or customized approaches that match techniques to the needs of the project and stakeholder group. Another approach to project delivery is to take a hybrid approach. Hybrid approaches use a combination of agile and predictive elements, such as a gate review process for continued funding decisions and Scrum for development work.

PMI believes that agile and predictive approaches, as well as other methods, are effective in specific scenarios and situations, a belief that is supported by the company's research. Organizations with higher agility reported more projects successfully meeting their original goals and business intent – whether they use hybrid (72 percent), predictive (71 percent) or agile (68 percent) approaches – than those with low agility using the same methods. Higher organizational agility supports more projects in meeting their original goals and business intent – one of the key measures of project success.

More information

INCOSE Website Features Members' Profile Spotlights

A feature available on the INCOSE website is the member profile spotlight. It provides a photo of the member, and one can click on it to access information including a profile of the individual that identifies the person's residence, number of years involved in systems engineering, role in INCOSE, the person's insight and perspective about INCOSE, and personal perspective such as how the person's organization has changed its engineering practices over the years.

Object Management Group and International Council on Systems Engineering Partner to Advance Standards for CubeSats

The Object Management Group® (OMG®), an international, open membership, not-for-profit technology standards consortium, signed a memorandum of understanding (MOU) with the International Council on Systems Engineering (INCOSE), the largest organization in the world dedicated to systems engineering. The MOU between OMG and INCOSE formalizes the working relationship between the two parties as they develop and distribute the INCOSE CubeSat Reference Model (CRM) to advance standards for CubeSats.

A 1U CubeSat is 10x10x10 centimeters with a mass of about 1.3 kilograms. CubeSat units can be joined to form a larger satellite such as 2U, 3U and 6U. Their small payload makes standalone launch cost-prohibitive but allows CubeSats to join launches from other organizations at a significantly reduced cost. The Nanosatellite and CubeSat Database reports that CubeSat launches tripled from 2016 to 2017 to nearly 300 launches. The number of CubeSats and the number of organizations developing CubeSats have increased dramatically. However, there is no common reference model to design a CubeSat, thereby limiting interoperability, slowing development times and forcing constant research to remain current on the latest innovations.

The CRM provides the logical architecture of a CubeSat space and ground system. The CRM logical components are intended to be used as a starting point for a mission-specific CubeSat logical architecture, followed by the physical architecture and the CubeSat development. The CRM is also sufficiently flexible to accommodate different logical architectures.

William Hoffman, president and COO of OMG and Garry Roedler, president of INCOSE, signed the MOU. Under the terms, the INCOSE CRM will go through the OMG standardization process. In addition, OMG will issue a request for proposals regarding the CubeSat Reference Model. OMG CRM status, progress and deadlines can be found here.

“Both OMG and INCOSE share a mutual commitment to support and accelerate knowledge growth in the systems engineering arena,” said Hoffman. “Developed by INCOSE using the OMG Systems Modeling Language™ (SysML®), the CRM is already in a model form, providing a fill-in-the-package framework and identifies subsystems to consider and other agencies to coordinate prior to launch. Space-mission-procuring organizations will use the CRM to capture mission requirements in a standard format that is readily accessible and understandable by vendors and developers.”

“Today’s announcement reinforces our joint cooperation with OMG to deliver a CubeSat Reference Model that applies a model-based systems engineering (MBSE) method and will be reusable and extensible to a variety of specific space missions and payloads,” said Roedler.

Learn more about CRM through INCOSE’s Space Systems Working Group or join OMG to influence the CRM and other related technology standards like SysML.

More information

Call for Best Business Process Management (BPM) Dissertation Award

The Steering Committee of the International Conference on Business Process Management announces the new Best BPM Dissertation Award.

All dissertations that have been officially completed after 01 January 2017 are eligible as candidates. Dissertations from a broad spectrum of topics that relate to BPM are invited. These topics include, but are not limited to:

- BPM and strategy
- BPM and innovation
- BPM and governance
- BPM methods
- Process mining
- Process modelling and analysis
- BPM and decision management
- BPM enactment architectures and process execution
- BPM techniques and algorithms
- BPM and organizational culture
- BPM and various stakeholders
- BPM and organizational routines
- BPM and IT implementation
- BPM and process-aware information systems
- BPM and new types of coordination and collaboration
- BPM and new technologies

Any type of research no matter which research methods have been used is welcome. This means that dissertations that have been conducted using formal methods, empirical methods or design methods are equally welcome.

More information

FEATURED ORGANIZATIONS

Society for Industrial and Applied Mathematics

The Society for Industrial and Applied Mathematics is an academic association dedicated to the use of mathematics in industry. SIAM is the world's largest professional association devoted to applied mathematics, and roughly two-thirds of its membership resides within the United States. Founded in 1951, the organization began holding annual national meetings in 1954, and now hosts conferences, publishes books and scholarly journals, and engages in lobbying in issues of interest to its membership. The focus for the society is applied, computational, and industrial mathematics, and the society often promotes its acronym as "Science and Industry Advance with Mathematics". Members include engineers, scientists, and mathematicians, both those employed in academia and those working in industry. The society supports educational institutions promoting applied mathematics.

More information

Editor's note: See the article provided in the SE Publications section concerning SIAM's book, *Numerical Methods for Conservation Laws*, by Jan S. Hesthaven.

CONFERENCES AND MEETINGS

For more information on systems engineering related conferences and meetings, please proceed to our website.

SOME SYSTEMS ENGINEERING-RELEVANT WEBSITES

Society for Industrial and Applied Mathematics (SIAM) Computational Science & Engineering Series

The SIAM series on Computational Science and Engineering (CS&E) publishes research monographs, advanced undergraduate- or graduate-level textbooks, and other volumes of interest to an interdisciplinary CS&E community of computational mathematicians, computer scientists, scientists, and engineers. The series includes both introductory volumes aimed at a broad audience of mathematically motivated readers interested in understanding methods and applications within computational science and engineering and monographs reporting on the most recent developments in the field. The series also includes volumes addressed to specific groups of professionals whose work relies extensively on computational science and engineering.

SIAM created the CS&E Series to support access to the rapid and far ranging advances in computer modeling and simulation of complex problems in science and engineering, to promote the interdisciplinary culture required to meet these large-scale challenges, and to provide the means to the next generation of computational scientists and engineers.

epubs.siam.org/action/showBookSeries?seriesCode=cs

National Defense Industrial Association (NDIA)

This site contains numerous presentations from the NDIA 20th Annual Systems Engineering Conference, Springfield, Virginia (USA), October 23 - 26, 2017.

https://ndia.dtic.mil/2017/systems/2017systems.html

SYSTEMS ENGINEERING PUBLICATIONS

Procedia Computer Science

Launched in 2009, Procedia Computer Science is an electronic product focusing entirely on publishing high quality conference proceedings. Procedia Computer Science enables fast dissemination so conference delegates can publish their papers in a dedicated online issue on ScienceDirect, which is then made freely available worldwide. Conference proceedings are accepted for publication in Procedia Computer Science based on quality and are therefore required to meet certain criteria, including quality of the conference, relevance to an international audience and covering highly cited or timely topics.

More information

Survey Report:

Improving Integration of Program Management and Systems Engineering

Results of a Joint Survey by PMI and INCOSE

Whitepaper presented at the 23rd INCOSE Annual International Symposium

Philadelphia, Pennsylvania USA

Research was conducted sponsored by a strategic alliance of the International Council on Systems Engineering (INCOSE) and the Project Management Institute (PMI) to investigate a concern that some systems engineers and program managers have a mindset that their respective work activities on some programs are not mutually supportive. This creates a cultural barrier and unproductive tension and negative impacts on program activities and results. A total of 680 Chief Systems Engineers and Program Managers responded to the survey.

Briefly, the research results were that:

- More than half of the program managers and chief systems engineers indicated that their programs were only somewhat integrated or not at all integrated.

- Lack of integrated planning is the main source of the tension.

The results of the research and analysis validated that a cultural barrier does exist in many organizations.

A downloadable copy of the report is available here.

A Systems Approach to Managing the Complexities of Process Industries

A Systems Approach to Managing the Complexities of Process Industries
Edited by Fabienne Salimi
IChemE

Image source

by

Fabienne Salimi and Frederic Salimi

Book description (from the Amazon website):

A Systems Approach to Managing the Complexities of Process Industries discusses the principles of system engineering, system thinking, complexity thinking and how these apply to the process industry, including benefits and implementation in process safety management systems. The book focuses on the ways system engineering skills, PLM, and IIoT can radically improve effectiveness of implementation of the process safety management system.

Covering life cycle, megaproject system engineering, and project management issues, this book reviews available tools and software and presents the practical web-based approach of Analysis & Dynamic Evaluation of Project Processes (ADEPP) for system engineering of the process manufacturing development and operation phases. Key solutions proposed include adding complexity management steps in the risk assessment framework of ISO 31000 and utilization of Installation Lifecycle Management. This study of this end-to-end process will help users improve operational excellence and navigate the complexities of managing a chemical or processing plant.

Presents a review of Operational Excellence and Process Safety Management Methods, along with solutions to complexity assessment and management;

Provides a comparison of the process manufacturing industry with discrete manufacturing, identifying similarities and areas of customization for process manufacturing; and

Discusses key solutions for managing the complexities of process manufacturing development and operational phases

More Information

Results-Based Leadership

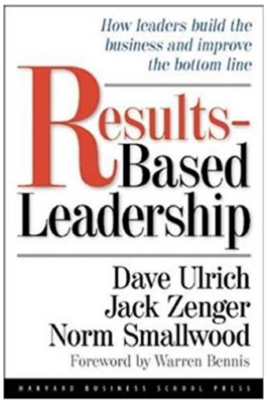


Image source

by

Dave Ulrich, Jack Zenger, and Norm Smallwood

Book description (from the Google store website):

A landmark book, *Results-Based Leadership* challenges the conventional wisdom surrounding leadership. Authors Ulrich, Zenger, and Smallwood - world-renowned experts in human resources and training, argue that it is not enough to gauge leaders by personal traits such as character, style, and values. Rather, effective leaders know how to connect these leadership attributes with results. Results-Based Leadership shows executives how to deliver results in four specific areas: results for employees, for the organization, for its customers, and for its investors. The authors provide action-oriented guidelines that readers can follow to develop and hone their own results-based leadership skills. By shifting our focus to the connection between the attributes and the results of leadership, this perceptive new guide fundamentally improves our understanding of effective leadership. Results-Based Leadership brings a refreshing clarity and directness to the leadership discussion, providing a hands-on program to help executives succeed with their leadership challenges.

More information

Editor’s note: Of 26 reviewers of this book on Amazon’s Website, 73% rated it five stars. One reviewer commented, “*The text was written in 1999, but the information concerning leadership is still relevant for 2018. Leaders should read the book, digest the information, and put it in force at their organizations and see what positive changes can be made. If we had results-based leadership today, many companies would not be going under. Know your resources and capabilities and know how to use them.”*

Systems of Conservation Laws

by

Jan S. Hesthaven

A chapter in *Numerical Methods for Conservation Laws* in the *Computational Science & Engineering Series*

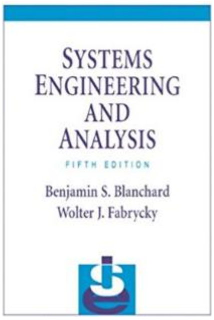
Having gained an understanding of the mathematical properties of the scalar conservation law, let us now turn to the more interesting and, unfortunately, considerably more complex case of systems of conservation laws. In this chapter conservation laws of the form are discussed:

$$\frac{\partial u(x,t)}{\partial t}+\frac{\partial f(u,x,t)}{\partial x}=0,\ \ x\in\Omega_x\subset \mathrm{R}, \qquad (3.1)$$

subject to initial conditions $u(x,O) = u_o(x)$. Here $u : \Omega_x \times \Omega_t \rightarrow \mathrm{R}^m$ and $f : \mathrm{R}^m \times \Omega_x \times \Omega_t \rightarrow \mathrm{R}^m$. Hence, systems with m equations are considered but attention to problems posed in one spatial dimension is restricted. Concluding the chapter, some of the challenges associated with the extension to multidimensional problems for which substantial parts of the fundamental theory remain incomplete are briefly highlighted.

More Information

Systems Engineering and Analysis, Fifth Edition



by

Benjamin S. Blanchard and Wolter J. Fabrycky

Book description (from the Amazon website):

For senior-level undergraduate and first and second year graduate systems engineering and related courses. *Systems Engineering and Analysis, 5th Edition,* provides a total life cycle approach to systems and their analysis.

This practical introduction to systems engineering and analysis provides the concepts, methodologies, models, and tools needed to understand and implement a total life cycle approach to systems and their analysis. The authors focus first on the process of bringing systems into being - beginning with the identification of a need and extending that need through requirements determination, functional analysis and allocation, design synthesis, evaluation, and validation, operation and support, phase-out, and disposal. Next, the authors discuss the improvement of systems currently in being, showing that by employing the iterative process of analysis, evaluation, feedback, and modification, most systems in existence can be improved in their affordability, effectiveness, and stakeholder satisfaction.

More information

SYSTEMS ENGINEERING TOOLS NEWS

Needs and Means

Needs & Means, developed by RSBA Technology Ltd in 2015, is the first purpose-built Planguage requirements authoring and management application.

Planguage, which is short for Planning Language, was developed by Tom Gilb and is a specification language and a set of related methods for systems engineering. It is used to specify requirements, designs, and project plans. Planguage consists of a set of defined concepts, a set of defined parameters and grammar, and a set of icons. Gilb’s methods, including Planguage and Evolutionary Development (Evo), are referred to in our Feature Article in this issue. They are devoted to achieving 'stakeholder satisfaction'.

Gilb has applied them successfully in many complex projects. References 1 and 2 at the end of Gilb's article provide much more information and guidance.

The Needs & Means tool is built on Meteor, an open source platform for web, mobile, and desktop JavaScript development and using the MongoDB document database. The chosen technology enables Needs & Means to offer real-time collaboration with updates immediately visible to other users viewing the same content. Both Meteor and MongoDB are running on Amazon's cloud infrastructure.

If you are already using Planguage in your company or projects, Needs & Means will certainly support your efforts.

If you aren't yet familiar with Planguage and the use of Evo in practice, the instances where Gilb mentions the Needs & Means tool in his article on pages 10 and 11 will provide context.

More information

Software related to Multiple Criteria Decision-Making

While reading through the draft articles lined up for SyEN 63, my eye caught the paper on Integrated Trade-Off Analytics by Parnell & Cilli. Of particular interest was their reference to Multiple Objective Decision Analysis (MODA), or as it is also known, Multiple Criteria Decision-Making (MCDM). Having been challenged with MCDM problems before, I knew that the availability of tools to support such decision-making can be hugely beneficial.

My very first search for MCDM tools produced a link to the website of the International Society on MCDM. It turns out that MCDM has quite a history, as can be seen from the "Short MCDM History" page.

Of particular interest is the impressive list of Software Related to MCDM. The very last link on this page contains a collection of Multiple Criteria Decision Support Software by Dr. Roland Weistroffer.

Not often one comes across such a handy resource.

EDUCATION AND ACADEMIA

Master's Degree in Management of Innovation and Design for Industry, Graduate School in Systems Engineering and Innovation, Université de Lorraine

The Université de Lorraine in France has more than 55,000 students, more than 1,800 PhD students, and 60 research laboratories. It promotes innovation through the dialog of knowledge, taking advantage of the variety and strength of its scientific fields, and aiming at the promotion of knowledge transfer to irrigate innovation and economic growth as well as the progress of fundamental science. The University has a large spectrum of international activities – it has 7,900 foreign students representing 137 nationalities.

About 1,500 of its students spend at least one semester abroad during their studies. The laboratories, departments, faculties and schools of the Université de Lorraine have hundreds of international partnerships around the world.

More information

Systems Engineering as a Victorian Certificate of Education (VCE) Subject in the State of Victoria, Australia

Yes, systems engineering is a pre-University secondary education elective subject. And it is the same systems engineering, more or less!

Scope of Study

Victorian Certificate of Education (VCE) Systems Engineering involves the design, creation, operation and evaluation of integrated systems, which mediate and control many aspects of human experience. Integral to Systems Engineering is the identification and quantification of systems goals, the development of alternative system designs concepts, trial and error, design trade-offs, selection and implementation of the best design, testing and verifying that the system is well built and integrated, and evaluating how well the completed system meets the intended goals.

This study can be applied to a diverse range of engineering ends such as manufacturing, land, water, air and space transportation, automation, control technologies, mechanisms and mechatronics, electro technology, robotics, pneumatics, hydraulics, and energy management. Systems Engineering considers the interactions of these systems with society and natural ecosystems. The rate and scale of human impact on the global ecology and environment demands that systems design and engineering take a holistic approach by considering the overall sustainability of the systems throughout their life cycle. Key engineering goals include using a project management approach to attain efficiency and optimization of systems through innovation. Lean engineering and lean manufacturing concepts and systems thinking are integral to this study.

Rationale

VCE Systems Engineering promotes innovative systems thinking and problem-solving skills through the Systems Engineering Process, which takes a project-management approach. It focuses on mechanical and electro technology engineered systems.

The study provides opportunities for students to learn about and engage with systems from a practical and purposeful perspective. Students gain knowledge and understanding about, and learn to appreciate and apply technological systems.

VCE Systems Engineering integrates aspects of designing, planning, fabricating, testing and evaluating in a project management process. It prepares students for careers in engineering, manufacturing and design through either a university or a Technical and Further Education (TAFE) vocational study pathway, employment, apprenticeships and traineeships. The study provides a rigorous academic foundation and a practical working knowledge of design, manufacturing and evaluation techniques. These skills, and the ability to apply systems engineering processes, are growing in demand as industry projects become more complex and multidisciplinary.

Aims

This study enables students to:

- develop an understanding of the Systems Engineering Process and the range of factors that influence the design, planning, production, evaluation and use of a system;
- understand the concepts of and develop skills in the design, construction, fault-finding, diagnosis, performance analysis, maintenance, modification, and control of technological systems;
- acquire knowledge of mechanical, electrical/electronic and control systems and apply this knowledge to solve technological problems;
- develop an understanding of how technologies have transformed people's lives and can be used to solve challenges associated with climate change, efficient energy use, security, health, education and transport;
- acquire knowledge of new developments and innovations in technological systems;
- develop skills in the safe use of tools, measuring equipment, materials, machines and processes, including using relevant information and communications technologies, and understand the risk management processes;
- acquire knowledge of project management, and develop problem-solving and analytical skills; and
- gain an awareness of quality and standards, including systems reliability, safety and fitness for the intended purpose.

More information

A Comparison of IEEE/EIA 12207, ISO/IEC 12207, J-STD-016, and MIL-STD-498 for Acquirers and Developers

by

Lewis Gray

The presentation slides are available here.

DEFINITIONS TO CLOSE ON

System

A system is a set of interacting or interdependent component parts forming a complex/intricate whole. Every system is delineated by its spatial and temporal boundaries, surrounded and influenced by its environment, described by its structure and purpose and expressed in its functioning.

Source: Wikipedia

Elements of a system

A system has three basic elements - input, processing and output. The other elements include control, feedback, boundaries, environment, and interfaces.

- Input: Input is what data the system receives to produce a certain output.
- Output: What goes out from the system after being processed is known as Output.
- Processing: The process involved to transform input into output is known as Processing.
- Control: In order to get the desired results, it is essential to monitor and control the input, Processing and the output of the system. This job is done by the control.
- Feedback: The Output is checked with the desired standards of the output set and the necessary steps are taken for achieving the output as per the standards, this process is called as Feedback. It helps to achieve a much better control in the system.
- Boundaries: The boundaries are the limits of the system. Setting up boundaries helps for better concentration of the actives carried in the system.
- Environment: The things outside the boundary of the system are known as environment. Change in the environment affects the working of the system.

- Interfaces: The interconnections and the interactions between the sub-systems are known as the interfaces. They may be inputs and outputs of the systems.

Source: Systems Analysis and Design Blog

System integration

In engineering, system integration is defined as the process of bringing together the component subsystems into one system and ensuring that the subsystems function together as a system. In information technology, systems integration is the process of linking together different computing systems and software applications physically or functionally, to act as a coordinated whole.

Source: Wikipedia

Maintainability

Maintainability is the ease with which a product can be maintained in order to: isolate defects or their cause, correct defects or their cause, repair or replace faulty or worn-out components without having to replace still-working parts, prevent unexpected breakdowns, maximize a product's useful life, maximize efficiency, reliability, and safety, meet new requirements, make future maintenance easier, or cope with a changed environment. In some cases, maintainability involves a system of continuous improvement - learning from the past in order to improve the ability to maintain systems, or improve reliability of systems based on maintenance experience. In telecommunication and several other engineering fields, the term Maintainability has the following meanings: (1) A characteristic of design and installation, expressed as the probability that an item will be retained in or restored to a specified condition within a given period of time, when the maintenance is performed in accordance with prescribed procedures and resources. (2) The ease with which maintenance of a functional unit can be performed in accordance with prescribed requirements.

Source: Wikipedia

Mean time between failures

Mean time between failures (MTBF) is the predicted elapsed time between inherent failures of a system during operation. MTBF can be calculated as the arithmetic mean (average) time between failures of a system. The MTBF is typically part of a model that assumes the failed system is immediately repaired (mean time to repair, or MTTR), as a part of a renewal process.

Source: Wikipedia

PPI AND CTI NEWS

Systems Engineering for Business Seminar Presented by Robert Halligan in Warsaw

It was the pleasure of Project Performance International's Managing Director Robert Halligan to present an introductory seminar on systems engineering to the INCOSE Poland chapter and guests on 10th March, 2018 in Warsaw.

In the presentation and discussion, Robert made the business case for systems engineering as a tool for reduced costs, shorter timescales and increased product value delivery, embracing the results of a number of recent, soundly-conducted studies on the subject.

Robert also assessed the current state of systems engineering practice against typical success criteria, using objective data to draw conclusions. He then put forward thoughts on the likely evolution of systems engineering practice over the next decade.

Here, Robert is being interviewed for a STEM promotional video by Karolina Majdziñska of the Center for Innovation and Technology Transfer Management of Warsaw University of Technology. (www.cziitt.pw.edu.pl)

UPCOMING PPI AND CTI PARTICIPATION IN PROFESSIONAL CONFERENCES

PPI will be participating in the following upcoming events.

The 12th Annual IEEE International Systems Conference

(Sponsoring)

23 - 26 April 2018

Vancouver, Canada

INCOSE IS2018

(Exhibiting)

7 – 12 July 2018

Washington, DC, USA

INCOSE SA 2018

(Exhibiting & Sponsoring)

3 - 5 October 2018

Pretoria, South Africa

PPI AND CTI EVENTS

Systems Engineering 5-Day Courses

Upcoming locations include:

- London, United Kingdom
- São José dos Campos, Brazil
- Wellington, New Zealand

Requirements Analysis and Specification Writing 5-Day Courses

Upcoming locations include:

- Adelaide, Australia
- Ankara, Turkey
- London, United Kingdom

Systems Engineering Management 5-Day Courses

Upcoming locations include:

- Amsterdam, the Netherlands
- Pretoria, South Africa
- Melbourne, Australia

Requirements, OCD and CONOPS in Military Capability Development 5-Day Courses

Upcoming locations include:

- Melbourne, Australia
- Pretoria, South Africa
- Amsterdam, the Netherlands

Architectural Design 5-Day Course

Upcoming locations include:

- Pretoria, South Africa
- Adelaide, Australia
- London, United Kingdom

Software Engineering 5-Day Courses

Upcoming locations include:

- London, United Kingdom
- Pretoria, South Africa
- Stellenbosch, South Africa

Human Systems Integration Public 5-Day Courses

Upcoming locations include:

- Melbourne, Australia

CSEP Preparation 5-Day Courses (Presented by Certification Training International, a PPI company)

Upcoming locations include:

- Denver, Colorado, United States of America
- Pretoria, South Africa
- Austin, Texas, United States of America

Kind regards from the SyEN team:

Robert Halligan, Editor-in-Chief, email: rhalligan@ppi-int.com

Dr. Ralph Young, Editor, email: ryoung@ppi-int.com

Suja Joseph-Malherbe, Managing Editor, email: smalherbe@ppi-int.com

Project Performance International

2 Parkgate Drive, Ringwood North, Vic 3134 Australia Tel: +61 3 9876 7345 Fax: +61 3 9876 2664

Tel Brasil: +55 12 3937 6390

Tel UK: +44 20 3608 6754

Tel USA: +1 888 772 5174

Web: www.ppi-int.com

Email: contact@ppi-int.com

Tell us what you think of SyEN. Email us at syen@ppi-int.info.