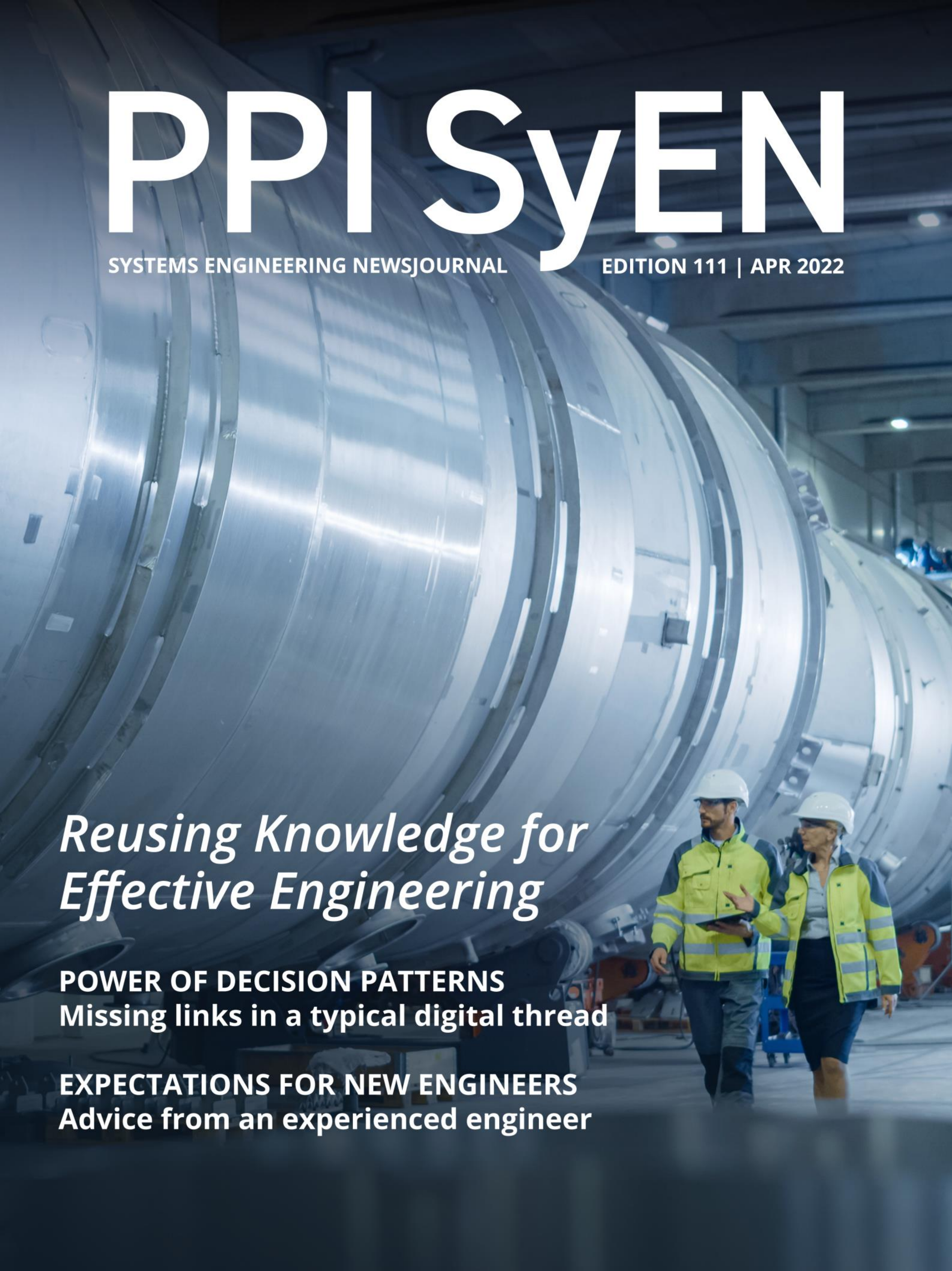


PPI SyEN

SYSTEMS ENGINEERING NEWSJOURNAL

EDITION 111 | APR 2022

Reusing Knowledge for Effective Engineering

POWER OF DECISION PATTERNS
Missing links in a typical digital thread

EXPECTATIONS FOR NEW ENGINEERS
Advice from an experienced engineer

A PROJECT PERFORMANCE INTERNATIONAL PUBLICATION | PPI-INT.COM

PPI SyEN

EMAIL: PPISyEN@PPI-Int.com

EDITORIAL STAFF

Editor
John Fitch

Editor-in-Chief
Robert Halligan

Managing Editor
René King

PRODUCTION STAFF

Marketing Manager
Benjamin Bryant

Graphic Designer
Matthew Wong

Marketing Coordinator
Rebeca Carneiro

Publishing Assistants
Trudy King
Shalani De Silva

PUBLISHER

PROJECT PERFORMANCE INTERNATIONAL

Project Performance International
2 Parkgate Drive
Ringwood, Vic 3134 Australia
Tel: +61 3 9876 7345

Tel Brazil: +55 12 9 9780 3490
Tel UK: +44 20 3608 6754
Tel USA: +1 888 772 5174
Tel China: +86 188 5117 2867

www.ppi-int.com

PPI SyEN (PPI Systems Engineering Newsjournal) is published monthly.

Archived editions and subscriptions to future editions are available for free at: https://www.ppi-int.com/syen-

WELCOME

Welcome to another edition of PPI SyEN,

As time rolls on and April comes to a close, I feel very inspired to learn more about what is happening in the systems engineering world. However, I have searched for more holistic perspectives rather than theoretical articles on narrow topics. Of course, science-oriented pieces are essential for expanding the systems engineering body of knowledge (and we on the PPI SyEN team love to dive deep); however, sometimes, it's necessary to shift perspective and look at the bigger picture. In comes this April edition of PPI SyEN!

Master decision facilitator, John Fitch, expands on his first article on Decision Patterns from PPI SyEN 107 (December 2021) with the 'So What' answer to the framework John introduced in December. From my perspective, there isn't an engineer who would not benefit from reading John's article. This is the kind of practical guidance on systems engineering techniques I have sought in my SE research expeditions this month.

If John's article gets the gears turning, Stuart Corn's article gets the engine revving with some potent and implementable advice from an experienced SE practitioner to a young engineer. Stuart's article is titled, 'One old Systems Engineer's Thoughts for New Systems Engineers', how I wish I had access to information like this at the earlier stages of my career! This is a must-read.

The Feature Articles alone make this edition worthy of bookmarking for repeat reference in the future however you know we at PPI SyEN like to cover all areas of developments in the SE world. In the news section, find out about newly formed INCOSE Chapters, opportunities to further your studies in SE, news regarding SE tools, and much more. There are some refreshing conferences on the horizon and some excellent webinars to look forward to covering topics from MBSE tool integration to cyber resilience. Several high-value products are mentioned in the resources section, including the latest INCOSE INSIGHT journal focused on Digital Engineering.

Finally, we close out with some tongue-in-cheek thoughts from Syenna regarding the impending release of the INCOSE Systems Engineering V5. Our mission is to include something for every engineer, regardless of the stage of your career or your SE area of expertise, and I believe we have succeeded with this edition. Thank you to the team for making this happen, especially our esteemed editor, John Fitch. I hope you enjoy reading this one as much as I did.

See you in May!

René

Managing Editor, PPI SyEN

PPI Systems Engineering Newsjournal (PPI SyEN) seeks:	**PPI defines systems engineering as:**
➢ To advance the practice and perceived value of systems engineering across a broad range of activities, responsibilities, and job-descriptions ➢ To influence the field of systems engineering from an independent perspective ➢ To provide information, tools, techniques, and other value to a wide spectrum of practitioners, from the experienced, to the newcomer, to the curious ➢ To emphasize that systems engineering exists within the context of (and should be contributory toward) larger social/enterprise systems, not just an end within itself ➢ To give back to the Systems Engineering community	*an approach to the engineering of systems, based on systems thinking, that aims to transform a need for a solution into an actual solution that meets imperatives and maximizes effectiveness on a whole-of-life basis, in accordance with the values of the stakeholders whom the solution is to serve. Systems engineering embraces both technical and management dimensions of problem definition and problem solving.*

SYSTEMS ENGINEERING NEWS

Recent events and updates in the field of systems engineering

INCOSE Chapter News 1Q2022

INCOSE chapters around the globe report continued progress in promoting effective systems engineering practices:

The newly-formed Latin America (LATAM) Chapter:

- Held two informational meetings (December 2021 & February 2022) with ~30 participants.
- Unveiled their new chapter logo.
- Adopted Spanish as their official language to reduce barriers to participation.
- Continued outreach to their region to grow membership and level of participation.

For more information, join their LinkedIn Group or view the chapter YouTube channel.

The Korean Society for Systems Engineering (KCOSE):

- Conducted a 2021 Fall Conference on 3-4 December 2021 to commemorate the 10th anniversary of the Society and explore the theme of "Application of Systems Engineering on ESG (Environment, Society and Government)".
- Elected Prof. JooYeoun Lee as the President of the Society for a 2-year term.

For more information, see the KCOSE home page.

The INCOSE UK Chapter:

- Announced Newcastle as the location for their Annual Systems Engineering Conference (ASEC) 2022 scheduled for 23-23 November. The conference theme is "Building toward a brighter Future".
- Released the results of the INCOSE UK 2021 Membership Survey in their latest ePreview newsletter.

For more information, see the INCOSE UK web site.

The INCOSE India Chapter:

- Is collaborating with faculty members at MIT Manipal on a four-course minor program in systems engineering.
- Has initiated a survey to understand and baseline the current state of systems engineering in India.
- Has experienced a 20% increase in membership over the past year.

For more information, see the India Chapter home page or join their LinkedIn Group.

SYSTEMS ENGINEERING NEWS

INCOSE Awards Presented

The opening plenary session of the INCOSE International Workshop (IW2022) included the presentation of multiple awards to working groups and individuals. Those honored include:

Systems and Software Interfaces Working Group
Chairs: Jeannine Siviy, Nickolas Guertin, Sarah Sheard (founding chair)

Program Management Systems Engineering (PM-SE) Integration Working Group
Co-Chairs: John Lomax, Jean-Claude Roussel, Tina Srivastava

Infrastructure Working Group
Chairs: Alain Kouassi, Marcel van de Ven, Laura Uden

Human Systems Integration Working Group
Co-Chairs: Guy-André Boy, Grace Kennedy

Systems Engineering Tools Database Working Group
Co-Chairs: John Nallon, Stephane LaCrampe, **Rene King, Robert Halligan**

Two individuals, Philomena Zimmerman and Dr. Dov Dori, received the *MBSE Propeller Hat Award* for their long-term contributions to and leadership in the disciplines of Model-Based Systems Engineering.

Congratulations to all those honored for their contributions.

Significant New Systems Engineering Guidebooks Released

In February 2022, the DoD Office of the Under Secretary of Defense for Research and Engineering (OUSD(R&E)) issued two new guidebooks for systems engineering.


Systems Engineering Guidebook

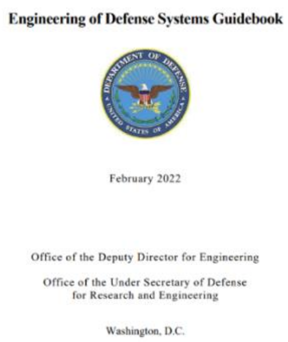
February 2022

Office of the Deputy Director for Engineering

Office of the Under Secretary of Defense for Research and Engineering

Washington, D.C.

The Systems Engineering Guidebook provides guidance and recommended best practices for defense acquisition programs. Much of this information appeared previously in the Defense Acquisition Guidebook (DAG) Chapter 3, Systems Engineering. The DAG has been canceled, and this document is intended to provide interim systems engineering (SE) guidance while the Department of Defense (DoD) is developing new Systems Engineering Modernization policy and guidance. This guidebook was prepared in cooperation with subject matter experts (SMEs) from the Military Services, Defense Agencies, industry, and academia. This guidebook is intended for Program Managers (PMs) and systems engineers and may be tailored for programs in any of the DoD Adaptive Acquisition Framework pathways (DoD Instruction (DoDI) 5000.02). Programs can use the guidebook, along with other acquisition business resources, to plan and execute program SE activities across the system life cycle.


Engineering of Defense Systems Guidebook

February 2022

Office of the Deputy Director for Engineering
Office of the Under Secretary of Defense for Research and Engineering

Washington, D.C.

The Engineering of Defense Systems Guidebook describes the activities, processes, and practices involved in the development of Department of Defense (DoD) systems. The guidebook aligns with the engineering disciplines covered in DoD Instruction (DoDI) 5000.88, Engineering of Defense Systems, and focuses on recommended engineering best practices for the DoD Adaptive Acquisition Framework (AAF) acquisition pathways. This guidebook is intended for Program Managers (PMs), systems engineers, and other defense acquisition professionals and may be tailored for programs in any of the AAF pathways. Programs can use the guidebook, along with other acquisition business resources, to plan and execute program engineering activities across the system life cycle.

PDMA Doctoral Dissertation Proposal Competition

The Product Development & Management Association (PDMA) has announced its call for the 2022 Doctoral Dissertation Proposal Competition. The aim of the competition is to support doctoral students to develop original and impactful research on innovation and new product/service development.

Three winners will be selected and receive cash prizes and PDMA membership privileges. Winners will present their research at the 2022 PDMA Conference (12-15 November in Orlando, Florida, USA) supported by a paid conference registration and travel stipends.

Dissertation research should be on a topic related to innovation and/or new product/service development. It can examine any aspect of innovation and can take a broad range of perspectives such as the market, organization, team, manager, or consumer. Potential topics include but are not limited to:

- Design and development of new products, services, and processes
- Business model innovation
- Open innovation, platform innovation, and innovation ecosystems
- Emerging technologies driven innovation (e.g. big data, artificial intelligence, machine learning, human computer interactions, blockchain, robotics etc.)
- Interfaces between innovation and other disciplines
- Industry-specific innovation studies (e.g., innovation in healthcare, education, energy, and other sectors)

Research proposals are due on 12 August 2022. See additional details.

DecisionsFirst™ Modeler April 2022 Release

Decision Management Solutions announces its April 2022 release of DecisionsFirst™ Modeler (DFM). DecisionsFirst™ Modeler delivers enterprise-grade, Decision Modeling Notation (DMN) standard-based decision modeling that works with existing business rules, machine learning or AI platforms.

Editor's note: For more background on decision modeling using the DMN standard, please see James Taylor's excellent article in SyEN Edition 108 (January 2022).

Key Benefits:

Decision modeling with DecisionsFirst™ Modeler provides a standards-based modeling blueprint, visually friendly to the business user, for digital decisioning projects.
Maximize business engagement (thereby reducing time to market) regardless of the mix of people, business rules, machine learning and AI required in a solution.
Provides tight integrated with common development and deployment platforms - delivers increased business engagement at every stage while improving coverage and reuse.
Minimizes ongoing maintenance costs while improving impact analysis and sustaining business user engagement for the long haul.

What's new:

Performance: Improved handling for multiuser activity on Decision Requirements Diagram cache management.

New and Enhanced Features:

- Decision Service Diagram "Simulation"
- Branch management features
- Basic merge - without approval for smaller teams/projects
- Advanced merged - with approval process for larger teams/projects

Coming Soon:

- Full DMN 1.4 implementation - Supporting all FEEL (Friendly Enough Expression Language) data types and expressions.
- Guided tutorial within DFM for improving user experience, use existing confluence help guide and refer to it for more detail.
- Introduction of DFM personas with personalized UI/controls based on relevant use cases.

Check out Decision Management Solutions on LinkedIn to stay informed of the latest enhancements to DecisionsFirst™ Modeler.

"

The project manager is the systems engineer of the (mostly human) project system, the engineering manager or team leader is the systems engineer of the (mostly human) engineering system.

Robert Halligan

CONFERENCES, MEETINGS & WEBINARS

Registration Opens for INCOSE International Symposium IS2022

Registration is now open for the INCOSE International Symposium (IS2022), a hybrid conference to be held from 25-30 June, 2022. The in-person venue is Huntington Place in downtown Detroit, Michigan, USA. The theme of IS2022 is "The Power of Connection" emphasizing the opportunity to learn from other systems engineering practitioners, share your insights with your peers and network with other like-minded professionals.

Symposium attendees may choose between a dozen tutorials to be presented on 25-26 June, followed by the main program on 27-30 June consisting of 4 keynotes, 79+ technical papers and 7 panel discussions.

A diverse range of topics, covering the full systems lifecycle, will be addressed including, but not limited to:

Business or Mission Analysis	Project Assessment/Control
Complexity	Project Planning
Decision Analysis/Decision Management	System architecture/design definition
Human-Systems Integration	System Integration
MBSE	Systems of Systems
Modeling/Simulation/Analysis	Systems Thinking
Needs and Requirements Definition	Teaching and Training
Processes	Verification/Validation

These topics will span numerous application and technology domains where systems engineering disciplines have demonstrated value:

Academia	Industry 4.0 & Society 5.0
Aerospace	Information Technology/Telecommunication
Artificial Intelligence	Infrastructure
Automation	Machine Learning
Automotive	Maritime
Autonomous Systems	Oil and Gas, Energy
Biomed/Healthcare/Social Services	Rail
City Planning	Service Systems
Defense	Social/Sociotechnical and Economic Systems
Emergency Management Systems	Sustainment
Enterprise SE	Urban Transportation Systems
Environmental Systems & Sustainability	Very Small Enterprises

Register here. Discounted registration is available before 15 May for both in-person and remote participation.
See more details at the event web site.
Download the Preliminary Technical Program flyer.
Download the Book of Abstracts for the planned sessions.

Call for Presentations: PDMA 2022 Annual Conference and JPIM Research Forum

The Product Development & Management Association (PDMA) has announced two calls for content contributors to PDMA Conference and Journal of Product Innovation Management (JPIM) Research Forum. These events will be held jointly on 12-15 November, 2022 in Orlando, Florida, USA.

The Annual Conference Call for Presentations is seeking speakers to deliver engaging and interactive content for four types of conference sessions:

- 45 to 60 Minute Interactive Presentation
- 45 to 60 Minute Case Study
- 90 Minute Mini-Workshop
- 1/2 Day Workshop Sessions

Presentations and workshops should address innovation trends or tools such as:

- Sustainability and green innovation
- Inclusiveness in Innovation
- Big Data and AI
  IOT and Designing for the Edges
- 3D Printing
- Organizational Design for Innovation
- Jobs to be Done Theory
- Design Thinking
- Agile, Open and/or Lean Innovation

Case studies may address how such trends and tools are impacting key industries such as:

- Consumer Products
- Services
- Software
- High-Tech Electro-mechanical Products

Presentation proposals are due by 29 April using the online submission form. Acceptance letters will be sent by 24 June.

The 2022 JPIM Annual Research Forum seeks various types of conference submissions (e.g. competitive papers, developmental papers, special session proposals and emerging research pitches) from scholars from all disciplines who share a common interest in new product development and innovation management. Submissions should address at least one of the following tracks:

- Responsible and Social Innovation
- Open Science, Open Innovation and Innovation Ecosystems
- Individuals, Teams, and Organization of Innovation
- Innovation Adoption and Diffusion
- Design and Innovation
- New Product Development and Innovation Strategy
- Digital Innovation

Call for Abstracts: NDIA Systems and Mission Engineering Conference

The NDIA Systems and Mission Engineering Conference to be held in Orlando, Florida, USA on 1-3 November 2022 has issued a Call for Abstracts. The Systems and Mission Engineering conference brings together defense community members from industry, government, and academia to highlight ways for improving defense acquisition and system performance. This conference provides an interactive forum for Program Managers, Systems Engineers, Chief Scientists, Specialty Engineers and Managers.

Topic areas include innovative methods and lessons learned concerning:

- Agile / DevOps
- Architecture MOSA
- Digital Systems Engineering
- Education & Training
- Human Systems Integration
- Integrated Risk and Opportunity Management
- Mission Engineering / System of Systems
- Program Management
- Specialty Engineering
- Safety and Environmental Engineering
- Software Intensive Systems
- Sustainment
- Physics-Based Modeling & Simulation
- System Security Engineering
- Test & Evaluation

Abstracts must be submitted no later than 27 May 2022 via the online submission page.

Webinar: Project Huddle Overview and MBSE Tool Integration

The INCOSE Los Angeles (USA) chapter hosted a webinar on 14 April to provide an overview of Project Huddle, an Aerospace Corporation MBSE tool integration solution. Huddle enables a 1-to-1 translation of Systems Modeling Language (SysML) v1.5 model elements, relationships, and diagrams between multiple vendor tools and in-house developed tools.

Trent Severson and Karina Martinez of The Aerospace Corporation provided an overview of the approach and demonstrated how Huddle is used to provide a model translation capability.

Huddle was developed to enable the seamless integration and federation of SysML models that have been developed by different teams and organizations using different tools. After clarifying numerous relevant definitions, e.g., ontology, metamodel, XML, XMI, the speakers shared several challenges associated with SysML model interchangeability – the problem that Huddle was targeted to solve.

Using tool integration between Sparx Enterprise Architect (EA) and Cameo System Modeler as an example, Severson and Martinez described a metamodel mapping approach that produced a common schema for the two tools, the Huddle Unified Data Schema (HUDS). Model Tool Integration Plug-ins (MTIPs) were developed to enable two-way interchange between EA and Cameo that support passing of all elements, relationships and diagrams. Side-by-side comparisons of various diagrams were shared to demonstrate both the efficacy and limitations of the current approach.

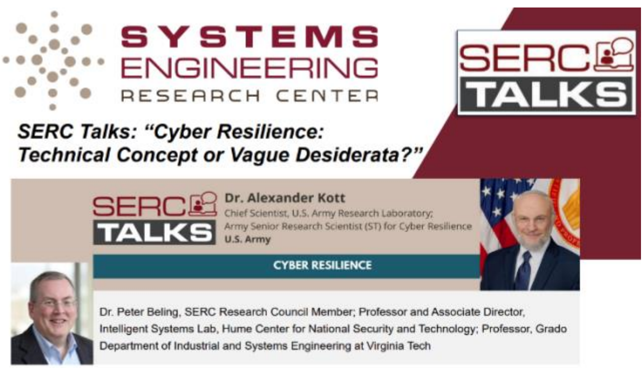
Follow-on efforts are planned to support additional metamodels (UAF, DoDAF, UML), widen the Huddle user base and participate in the OpenMBEE community.

Download the presentation here.

SERCTALKS - Cyber Resilience: Technical Concept or Vague Desiderata?

On 6 April, the Systems Engineering Research Center (SERC) hosted the second of its Spring 2022 SERC TALKS focusing on the topic of Cyber Resilience. Moderated by Dr. Peter Beling of Virginia Tech, the series hopes to stimulate an ongoing and more collaborative dialogue between academia, government and industry sectors on this important topic.

Dr. Alexander Kott of the U.S. Army Research Lab delivered the second talk, titled "***Cyber Resilience: Technical Concept or Vague Desiderata?***"



Abstract: I will begin by lamenting the fact that cyber resilience remains a subject of much confusion and vagueness. It is time for a firmer, better defined intellectual framework for cyber resilience as a technical concept. I will outline two key reasons for pursuing cyber resilience, a military one and an economic one. Along the way, I will caution against a common mistake of conflating cyber security and cyber resilience. They are not the same. Then I will sketch the differences – as well as relative advantages and disadvantages – of two classes of cyber resilience: resilience by design and resilience by intervention. I will discuss why autonomy is a key to resilience by design (and to some extent to any cyber resilience) and key features of intelligent autonomous agents for cyber defense and resilience. Regardless of the means of achieving resilience, we will not make much progress without being able to measure resilience. We cannot improve what we cannot measure. I will discuss directions towards measuring cyber resilience, and bemoan yet another harmful conflation, that of assessing and measuring.

Download slides:

Recordings are available for this talk on the SERC YouTube Channel.

Planned SERCTALKS in the Cyber Resilience series include:

- 15 June: ***Cyber Resilience*** with Ms. Melinda K. Reed, Director Resilient Systems, Office of the Under Secretary of Defense for Research and Engineering (OUSD(R&E)). Register here.

See previous SERC TALKS series and topics here.

PPI RESOURCES

PPI offers a multitude of resources available to all of our clients, associates and friends! Click on any of the links below to access these resources today.

Systems Engineering FAQ: https://www.ppi-int.com/resources/systems-engineering-faq
Industry-related questions answered by PPI Founder and Managing Director Robert Halligan.

Key downloads: https://www.ppi-int.com/keydownloads/
Free downloadable presentations, short papers, specifications and other helpful downloads related to requirements and the field of Systems Engineering.

Conferences: https://www.ppi-int.com/resources/conferences-and-meetings/
Keep track of systems engineering-relevant conferences and meeting dates throughout the year.

Systems Engineering Goldmine: https://www.ppi-int.com/se-goldmine/
A free resources with over 4GB of downloadable information relevant to the Engineering of systems and a searchable database of 7,800+ defined terms. You can expect the content of the SE Goldmine to continue to increase over time.

Systems Engineering Tools Database (requires SEG account to log in from the Systems Engineering Goldmine): https://www.systemsengineeringtools.com/
A resource jointly developed and operated by Project Performance International (PPI) and the International Council on Systems Engineering (INCOSE). The SETDB helps you find appropriate software tools and cloud services that support your systems engineering-related activities. As a PPI SEG account holder, you have ongoing free access to the SETDB.

PPI SyEN Newsjournal (actually a substantial monthly SE publication): https://www.ppi-int.com/systems-engineering-newsjournal/
You're already reading our monthly newsjournal! However click on the link to access the history of 100+ monthly newsjournals containing excellent articles, news and other interesting topics summarizing developments in the field of systems engineering.

FEATURED ARTICLE

Decision Patterns – So What?

by John Fitch

Project Performance International

Authored for PPI SyEN

Introduction

This article is the follow-on to the *Introduction to Decision Patterns* piece that was published in SyEN edition 107, December 2021. The first article shared the author's 30+ year journey concerning the discovery, use and refinement of decision patterns, the conceptual basis (definitions, information model, methodology and elements) behind this construct, and the variety of use cases in which decision patterns have been applied (to the author's direct knowledge). The article also provided simplified examples of decision patterns for Enterprise Strategy, System/Product Design, Process Capability Design, Service Design and Curriculum/Courseware Design.

The original paper identified eight unique aspects of a decision-centric information architecture that is applicable when engineering a solution to any type of problem. These constructs include:

- Decision Breakdown Structure
- Essential information "within" each decision
- Requirements-to-decision traceability
- Decision-to-requirements derivation traceability
- Decision-to-plan traceability
- Architecture models representing alternatives
- Math/physics and lifecycle models representing alternatives
- Decision-to-roadmap traceability

This paper will elaborate on each of these information constructs, explain how they fill some "missing link(s)" in a typical digital thread and by doing so enable new and valuable system engineering capabilities. In addition to the potential benefits offered by use of these capabilities, the author will highlight current challenges to the effective use of decision patterns and experience-based tips to increase the likelihood of first-time success in applying them on a project.

Readers are encouraged to review the first article prior to diving deeper into this subject matter. In particular, note the somewhat unique definition of a decision as an element that decomposes the problem domain:

Decision = a fundamental question/issue that demands an answer/solution

Things to consider

Ultimately *all faults are decision faults*. Any failure in an engineered system or system of systems can be traced (by repeatedly asking "What went wrong?") to one or a combination of:

- Decisions overlooked
- Decisions poorly made
- Decisions poorly implemented

Because decisions and the rationale behind them are lightly captured in most projects and strategic initiatives, it is difficult to build the cause-effect, e.g., fishbone diagram back through the web of missteps to answer questions such as:

- Why did we overlook or get a late start on recognizing the importance of this decision to project success?
- What missing or untrustworthy data did we use or analysis steps did we poorly execute that led us to choose a low-value or destined-to-fail alternative?
- How did our implementation of this good idea go awry?

If your organization has difficulty in answering such questions with confidence, you may want hit the *Pause* button on your current MBSE or Digital Thread initiative in order to rethink the role that decisions play in your scheme.

Ultimately the author believes that such failures are best mitigated by a combination of process steps (methods) that populate a lean, but comprehensive decision-centric information model, all jump-started by using decision patterns tailored to problem types/domains. Software tools are also helpful in enabling effective decision management at scale by:

- Delivering decision patterns that support early identification of decisions-to-be-made.
- Capturing and visualizing decision information in an intuitive form.
- Maintaining traceability between the definition of the problem to be solved, the decisions that evaluate and select a preferred solution, the description of that solution and the consequences this solution has on the next layer of the problem (derived requirements).

To get maximum benefit from this article, readers are encouraged to:

- Compare/contrast each construct with the current practice within your organization.
- Identify which new and improved capabilities that are enabled by each construct would deliver the greatest value, if deployed successfully.
- Identify the organizational challenges that would have to be overcome before this potential value could be realized.
- Based on these assessments, identify a pilot project and would exercise several of these constructs, ideally linked together to fill in gaps in your current Digital Thread.

PPI would love to receive feedback from this self-assessment at PPISyEN@ppi-int.com.

Decision Breakdown Structure

The Decision Breakdown Structure (DBS) for a project is a decomposition of the overall problem into discrete, loosely-coupled decisions – each of which is a fundamental question/issue that demands an answer/solution. As illustrated in the previous Decision Patterns article, a project/product DBS may be built from pre-existing patterns. Such patterns include not just a flat list of decision questions, but also a topology, a logical decomposition structure that expresses how one decision naturally branches into other lower-level choices.

Although the DBS suggests a top-down sequence of decision-making, the actual decision analysis and decision-making sequence may vary.

It is common to consider a variety of critical technology decisions early (How will system function X be delivered?), select promising technologies as solution building blocks, then architect combinations of these technologies (and the components that deliver them) to create the top-level solution concept alternatives for further evaluation. Iteration between the top-level solution concept and the technology building blocks is common as the competing solution alternatives are refined and their effectiveness is better understood.

Information Model
The DBS is comprised of decision entities and the decomposition relationships among them. Attributes for each decision may include:

- Name: short title
- Description: the question to be answered
- Class: Single answer, Multiple answer, Multi-part answer
- Priority: May be captured in multiple attributes such as Impact, Innovation Opportunity and Knowledge Gap.
- Owner: Individual that is responsible for leading the decision analysis. May not be the final Decision Authority.
- Analysis Plan: Who will do what, when and how to inform the decision? May include a quantitative Analysis/Effort Budget to scale the effort invested to match the decision's relative priority.
- Start Date (of the decision analysis)
- End Date (of the decision analysis)
- Status: Identified, Planned, In-progress, Analysis complete, Approved, …

The Analysis Plan attribute may be supplemented by Analysis Task entities that break down the Decision into data-gathering and evaluation work packages, with associated responsibility assignments, start/end dates and task status.

New or Improved SE Capability
The DBS represents a formal technique for proactive decision planning and control. As a result, the DBS mitigates the "Decisions overlooked" failure mode common to projects by enabling rapid and efficient problem decomposition, explicit decision identification and framing, decision prioritization and decision analysis (trade study) planning.

The DBS gives the project the opportunity to prioritize precious analysis resources and focus them on the decisions that are deemed to be most likely to drive project success or prevent project failure. When a DBS is framed based on a proven decision pattern, there will be higher confidence that key decisions have not been overlooked. The DBS (decision questions) create a knowledge "pull" from other systems engineering processes that helps to uncover important "unknowns that need to be known", e.g., missing use cases, requirements or external interfaces, that bound the project scope. Often the Decision Blitz process that populates the DBS uncovers disagreements among stakeholders about which decisions have been made (and precisely which alternatives have been chosen) and which are yet-to-be-made during the next phase of development.

Once created, the DBS is an enabler for rolling wave planning. Decisions are made, leading to branch-by-branch elaboration of the DBS to lower-level choices. A graphical DBS provides a visualization of overall engineering status, highlights the current decision-making frontier and supports continuous replanning of the remaining project tasks to inform and realize the remaining decisions.

Finally, the DBS sets up the organizational capability to increase learning from each project. Reuse of decision models as patterns across projects and domains can have a compound interest effect on project quality and efficiency across an enterprise.

Challenges

In the author's experience, few system modeling languages and associated MBSE tools support an explicit Decision class; decisions are treated as second-tier information and dispersed among various attributes, typically as a Rationale property on physical solution elements. Decisions (questions) are frequently confused with alternatives (possible answers). Alternatives considered, but not chosen, are lost without recording why they were rejected or deferred.

Therefore, the fundamental thinking/logic that transforms a problem into a committed solution is missing-in-action. This drives up the lifecycle and change management cost of the system because of information lost to time and turnover.

Most organizations (also in the author's experience) lack a decision database that maintains a clear distinction between different aspects of the problem (decisions stated as questions), the solution alternatives considered, the chosen alternative and the rationale behind this selection.

Decisions to be made are often hidden among a project issues list, identified reactively (too late in time) and in an ad hoc and poorly framed manner. High priority decisions that are identified late often find that resource and performance budgets have been consumed by less critical choices, triggering the need for extensive and expensive design rework.

Beyond the organizational inertia challenges mentioned above, there are few, if any, commercially available tools, tool extensions or services that deliver ready-for-use decision patterns, help teams instantiate such patterns to create a project-specific DBS, visualize the DBS as a technical planning and management tool, elaborate the decision model as design progress is made and maintain traceability between decisions, requirements, solutions, etc.

Ultimately, the lack of a first-class Decision object (on par with needs, activities, functions, requirements, solution elements, risks, etc.) is the roadblock to taking advantage of the other potential benefits of a decision-centric approach to systems engineering and the vision of a Digital

Thread that includes the problem definition, solution description and the THINKING that connects the two.

Essential information "within" each decision

Within each Decision entity in a project, there is a rich set of associated information that fully captures the thought process that leads to the commitment to a specific solution. This information is essential to capture the contributions of the decision analysis team, integrate these contributions to identify the best-fit solution, communicate recommendations (decision rationale) to stakeholders and decision-makers (decision authorities) and revisit the decision efficiently, if needed.

Information Model

The essential information that supports a decision includes the following entity classes:

- Criterion: Evaluation factor; measure of effectiveness or constraint used to screen and score alternatives.
- Alternative: Potential solution to the question posed by the decision.
- Performance: Estimate of each alternative's effectiveness against each criterion. The data that populate the cells in an evaluation matrix.
- Risk: A way that an alternative may fail across its lifecycle; probabilistic event or condition that significantly reduces the alternative's overall value to stakeholders.
- Opportunity: A way that the alternative may perform better than expected/estimated; probabilistic event or condition that significantly increases the alternative's overall value to stakeholders.

It is beyond the scope of this paper to enumerate the attributes associated with each of these decision analysis classes. The relationships between these classes are shown in Figure 1.

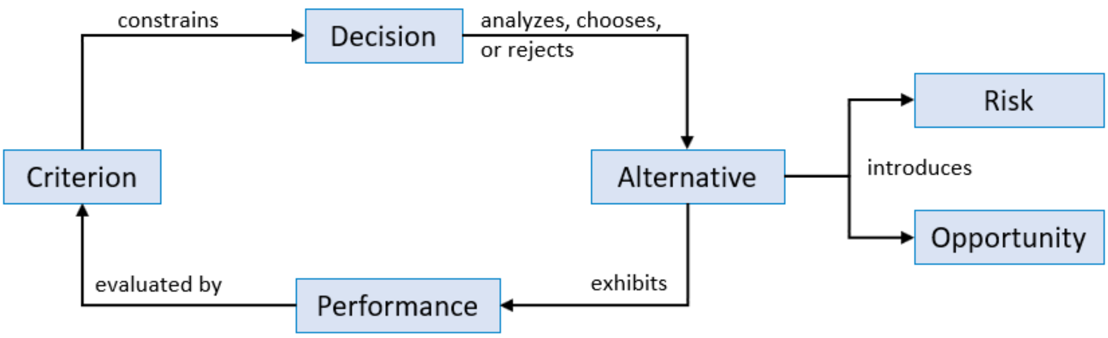


Figure 1: Decision Analysis Information Model

Each decision may have multiple criteria (N) and alternatives (M) resulting in an N x M cell evaluation matrix populated with Performance estimates. Alternatives may have (introduce) multiple risks and opportunities.

New or Improved SE Capability

The Decision Analysis information model shown in Figure 1 provides an explicit, traceable and consistently structured rationale for each decision, enabling improved decision quality, stakeholder buy-in and impact/change analysis.

By supplementing decision rationale expressed in paragraph "blobs", this model improves decision confidence by reducing the ambiguity of the "Why?" behind each decision.

The decision analysis information model provides the basis for assessing the completeness of the decision logic, i.e., detecting missing criteria, a narrow range of alternatives, incomplete or biased evaluations, overlooked risks, etc. Such detection is extremely difficult when reading rationale that is written uniquely by each engineer or analyst and buried within natural language paragraphs.

When implemented with an appropriate set of attributes for each class, these decision analysis data elements provide the basis for multiple consistently formatted viewpoints through which to gather and communicate the logic behind any choice. Where quantitative attributes are used for criteria weights and threshold/goal values, plus alternative performance estimates and uncertainties, there is the possibility of automated scoring using utility/value functions.

Challenges

Without a project DBS, there is limited basis for performing the Pareto Analysis to decide which decisions merit the full decision analysis rigor implied by the proposed information model. Imposing full rigor across the entire DBS is typically cost-prohibitive until decision analysts are highly skilled and aware of effective decision analysis shortcuts, e.g., screening out alternatives against criteria thresholds to reduce the need for the full set of N x M Performance estimates.

MBSE practitioners don't always conceptualize a wide variety of possible solution concepts (alternatives) to any design problem. The first solution model "drawn" creates a tunnel vision effect in which radically-different but potentially superior alternatives may be overlooked.

Commercial tools for decision analysis and least-common-denominator equivalents (Excel macros) are not well integrated with common MBSE tools suites.

Requirements-to-decision (R-D) traceability

A decision-centric model for solution development includes an explicit traceability link between requirements and the decisions that these requirements drive/influence.

For simplicity in this paper, I'm using the broadest definition of term "requirement" to include all types (e.g., functional, performance, resource, interface, physical, environmental, design, etc.) and also associated goals that express the relative value of delivering levels of performance beyond the threshold in a formal "shall" statement.

Information Model

As shown in Figure 2, the requirement "drives" decision relationship is implemented with more precision if the Criterion class is inserted between the requirement/goal and the decision.

Figure 2: Requirement to Decision Traceability Model

Requirements as written by stakeholders or analysts are not always "ready for prime time" use in evaluating solution alternatives for a specific decision. Some requirements will drive multiple decisions, triggering the need for the distribution of performance budgets among the criteria in all of the decisions where the requirement is relevant.

To simplify the decision analysis process by reducing the number of criteria, it is often helpful to merge multiple requirements into a single broader criterion.

The criterion object enables flexibility in modeling requirements and decisions without forcing rigid one-to-one relationships.

New or Improved SE Capability
The requirement-criterion-decision trace makes an explicit tie between requirements/goals and each design decision. Missing relationships can be detected to uncover:

- Requirements that haven't been used to influence the design, i.e., fallen through the cracks.
- Criteria for which there is no valid system requirement, i.e., gold-plating.

This trace also enables the management of budget allocation tradeoffs and roll-ups between decisions, i.e., multi-decision tradeoffs.

Decision patterns often include a corresponding criteria pattern for specific decisions. In such cases the criteria pattern creates a knowledge "pull" back to stakeholders. Requirements and decision analysts may state, "This decision typically includes a criterion for X" and ask, "Where in your requirements/goals would we find your expectations for the X requirement?". Such a stakeholder interaction provides an excellent opportunity to uncover missing requirements or the failure to identify a requirement as being relevant to a specific decision.

Challenges
In many organizations, the requirements analysis and decision analysis processes are, at best, loosely connected. At worst, they represent two independent "sets of books" being kept by dueling teams. In such situations, the focus on requirements is typically to ensure end-of-project verification rather than to ensure design quality.

Commercial tools for requirements management and decision analysis are not well integrated; the requirement -> refined by -> criterion relationship is not explicitly supported. Requirements are translated into criteria by a mix of error-prone manual entry and copy/paste actions.

Decision-to-requirements (D-R) derivation traceability

It is the author's conviction that all requirements are derived requirements, are "birthed" in and can be traced from one or more upstream decisions.

The term "derived" in this context means that requirements are the inherent consequences that flow from the definition of the solution alternative that has been chosen for implementation. The decision-to-requirements traceability construct, as shown in Figure 3, captures the thought processes by which new requirements are identified based on the chosen solution.

Information Model

In addition to the Decision, Alternative, Risk and Opportunity classes discussed previously, the D-R trace information model adds classes for (Risk) Mitigation and (Opportunity) Growth actions. Risks and opportunities have dual uses; they are part of the decision analysis process and decision-making "equation" and, once identified, they can be used to further protect and optimize the solution. Mitigation and Growth actions may be identified, then translated into specific requirements that may themselves be evaluated for their ROI.

In a sense, mitigation and growth requirements are optional; they are added based on a choice to protect or enhance the alternative that was originally defined, rather than merely assuming the risk or ignoring the opportunity.

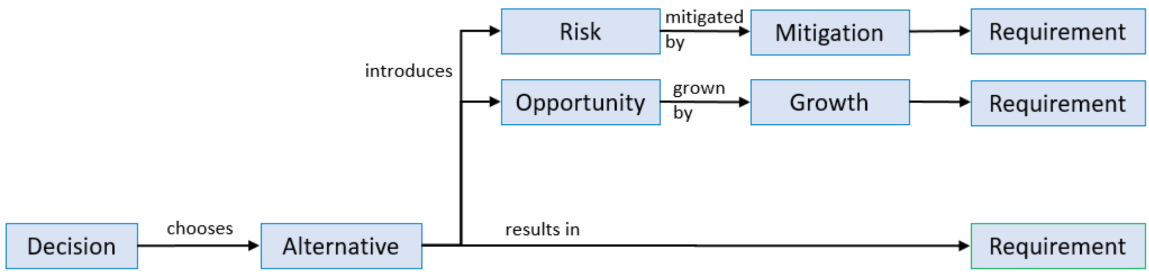


Figure 3: Decision to Requirement Traceability Model

The primary D-R derivation relationship is the Alternative -> results in -> Requirement thread. The choice of a particular solution technology may trigger a requirement for a different manufacturing or support process, new skills for users/operators, unique maintenance equipment interfaces, or tighter volume (space claim) or power budgets for other system components. The ideal time to capture these derived requirements is at the point of decision when:

- They become mandatory constraints on the rest of the solution design.
- The decision analyst who is recommending the solution has fresh and deep understanding of the definition of the alternative that has been chosen because of insights gained when evaluating the alternative.

Derived requirements that are stated from the perspective of the decision analyst will often require additional refinement to translate them into the language understood by the next-level designers.

New or Improved SE Capability

The D-R trace enables an explicit visualization of the inherent consequences (risks & mitigations, opportunities & growth actions, derived requirements) of a decision's chosen alternative. These represent the constraints imposed on the rest of the system design upon on approval of the decision.

The D-R trace enables proactive and efficient change management. If the chosen alternative doesn't perform as expected and a different alternative is later selected in its place, the requirements associated with the original solution are potentially invalidated and new requirements associated with the "replacement" solution will have to be added.

This may trigger a cascade of changes through other decisions by which an entire solution concept may collapse like a "house of cards".

The existence of the explicit D-R trace doesn't prevent such a collapse, but enables efficient management of such changes when compared to situations where such connections are retained only in the heads of a few designers.

Challenges
In the author's experience, few organizations understand the role that decisions play in creating new requirements. Traceability is often maintained only between requirements and other requirements.

Requirements management or MBSE tools don't typically recognize either the Decision, Alternative, Mitigation or Growth (action) classes. Derived requirements are inferred from engineering experience, past projects or logical or physical solution architectural models.

While each of these techniques is potentially valuable, models or engineering experience don't create requirements. Only a decision (commitment to realize a defined solution alternative) can.

Decision-to-plan (D-P) traceability

Every task in a development project plan either informs a decision or realizes (builds/integrates, verifies, deploys, validates, etc.) the solution chosen, across the full solution lifecycle. Decision-to-plan traceability is based on the recognition that the alternative chosen may affect the decisions to be made in the future and the steps that it will take to fully realize the solution.

Information Model
As shown in Figure 4, Decision-to-Plan traceability adds the Task class to the information model.

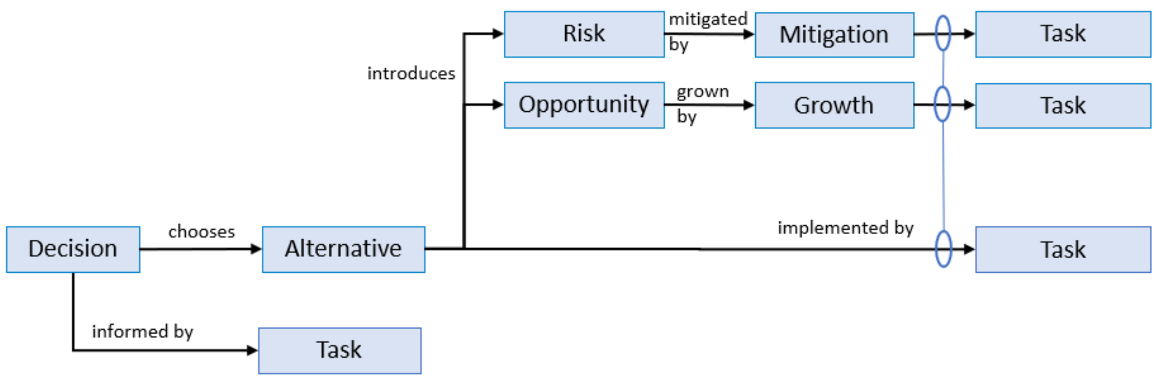


Figure 4: Decision to Plan Traceability Model

Tasks may be subclassified as:
- Analysis Tasks that *inform* the decision, e.g., stakeholder engagement to define criteria, modeling of solution alternatives within MBSE or CAD tools, estimation of alternative performance by expert inputs, vendor searches, math/physics models or simulation, screening and scoring to populate the evaluation matrix with rationale, assessment of risks/opportunities, etc.
- Risk mitigation tasks that *implement* the set of risk mitigation actions chosen to protect the solution.

- Opportunity growth tasks that *implement* the set of opportunity growth actions chosen to enhance the value of the solution.
- Implementation tasks that *implement* (fully realize) the chosen solution alternative across the rest of the solution lifecycle.

New or Improved SE Capability

The D-P trace enables the explicit visualization of the inherent consequences of each decision's chosen alternative on the project plan. Each decision may trigger an update to the rest of the project plan, either adding or refining tasks as part of rolling wave planning approach.

If done on a per-decision basis, the result can be nearly continuous alignment between the technical aspects of the project and the official project plan, schedule and resource assignments.

Challenges

Because of the lack of the Decision class in most languages and tools, the decision-to-plan trace is either not implemented or implemented with less precision by the association of tasks with physical solution components. That loss of precision may lead to overlooked analysis, risk mitigation, opportunity growth or solution implementation tasks. In such cases, the true impact of a decision on the project's cost and schedule baseline may not be accurately understood. Instead of continuous alignment between a project's technical status and its official plan, overlooked tasks may lead to loss of product value, cost overruns or delayed schedules.

Architecture models representing alternatives

The alternatives being analyzed and evaluated in most design-focused decisions may be represented by a variety of models. From the perspective of system modeling languages and MBSE tools, physical architecture and logical architecture models are the most common. Others include state models or 3D CAD models.

Information Model

Alternatives may be modeled physically as a set of Components connected to/through Interfaces.
Alternatives may be modeled logically as a set of Functions that input or output Items (matter, energy, information).

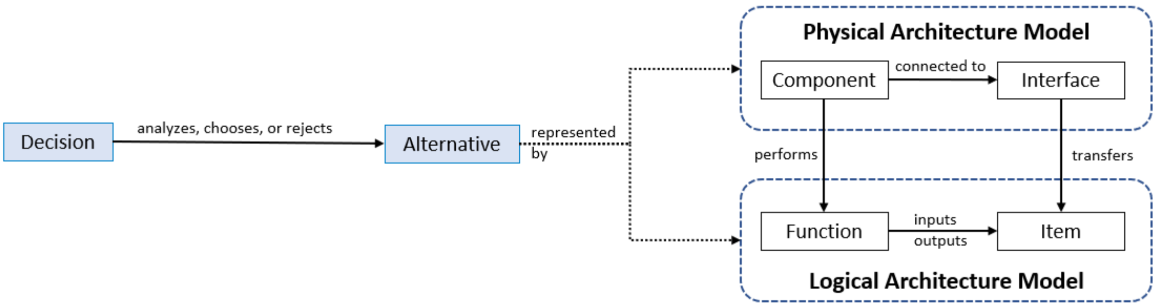


Figure 5: Architecture Models Representing Alternatives

Both the physical and logical models of an alternative must be aligned. Components perform Functions that input or output Items that are transferred across physical Interfaces.

New or Improved SE Capability
A decision-centric approach to engineering suggests that models should be created primarily to inform decisions. As such, physical and logical models should be lean, i.e., developed only to the level of decomposition and fidelity necessary to define alternatives sufficiently to evaluate them effectively and produce decision confidence. Modeling for the sake of decision confidence reduces the waste associated with modeling for modeling's sake.

However, models should be built to cover a wide range of feasible and potentially best solutions to avoid the tunnel vision that can be induced when the first model "drawn" limits solution creativity.

Challenges
Numerous tools exist to create system models. The decision-centric approach to design is less about which model is best suited to represent the structure and behavior of a range of alternatives.

Rather it provides guidance on which decisions and alternatives (and therefore elements of the system) are worthy of the investment associated with explicit representations beyond natural language descriptions.

One-size-fits-all modeling practices drive up project costs beyond the value they deliver through better design decisions. Modeling a single alternative and then iterating its design is likely to lead to less-than-optimum solutions compared to modeling 2 or 3 diverse solution candidates based on a wide range of technologies and concepts.

The lack of an explicit Decision class in MBSE tools implies that those tools offer limited help in deciding which aspects of system structure and behavior should be modeled, to what level of decomposition and fidelity.

Math/physics and lifecycle models representing alternatives

Designers of solutions in stable problem domains that are addressed by a portfolio of slowly-evolving solution technologies have typically built libraries of math, physics and lifecycle models suitable to those domains. In the context of a decision-centric approach to engineering, these libraries are comprised of system modeling patterns that can provide high-confidence quantitative estimates of the performance of a variety of solution alternatives against the range of typical evaluation criteria (Measures of Effectiveness).
Such libraries are important enterprise knowledge assets that should be aligned with the frequently made and high-value decisions faced by an organization.

Information Model
Although decisions may be made using qualitative assessments of alternative effectiveness, in the discipline of systems engineering, numbers tend to rule. As shown in Figure 6, quantitative system models generate estimates of an alternative's performance against one or more evaluation criteria. Math/physics/lifecycle models are often built upon the previously-mentioned physical or logical architecture representations of an alternative.

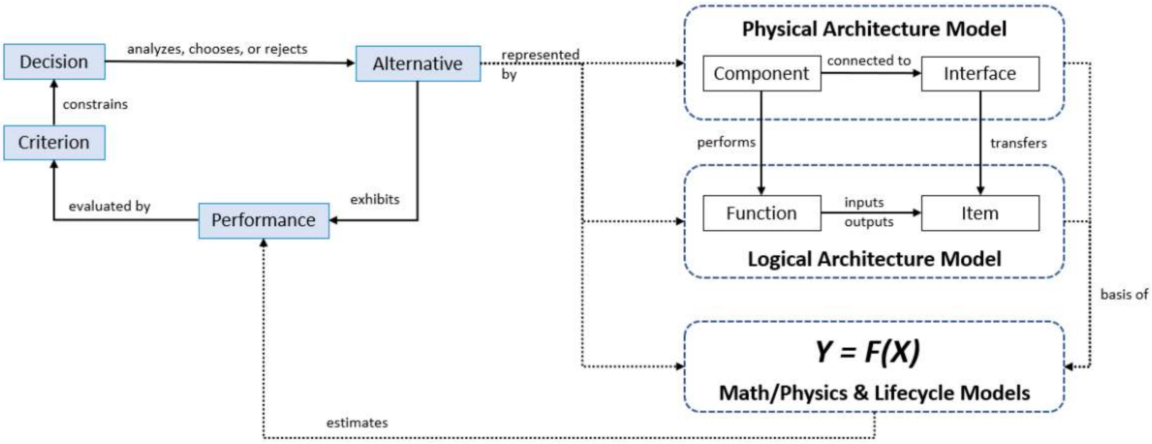


Figure 6: Math/Physics & Lifecycle Models Representing Alternatives

New or Improved SE Capability
Organizations with proven decision patterns may develop a corresponding set of quantitative modeling patterns that can be used to inform the decisions frequently faced by the enterprise. Modeling investments should be aligned with decision priorities and which criteria are the most common discriminators between solutions for each decision in the pattern.

Organizations can take advantage of their inevitable investments in system verification to identify errors or bias in their quantitative system models. For example, a performance model used early in design may predict 20% margin for a solution design against a criterion/MOE. Actual test results may show that only a 5% margin was achieved by the final solution.

Even though the solution passed the requirement threshold, this situation implies a 15% bias in the design performance model. This bias should be investigated and resolved so that future decisions can be made with higher confidence.

Challenges
Numerous model integration environments have been created in the past decade, but the author has not seen features in those tools that maintain decision pattern to model pattern alignment or help visualize gaps in an enterprise modeling portfolio. Help is required to answer the question, "Which frequently made design decisions lack sufficient modeling capabilities to yield the level of decision confidence that we need?". "Where should we invest in better modeling capabilities?

Decision-to-roadmap traceability

The best products become platforms or product lines that support a configurable mix of functionality and performance that can be adapted to meet a wide range of use cases. The discipline of creating roadmaps of strategy, capabilities, portfolios, products, features or technologies, though not typically considered as part
of systems engineering, can leverage decision patterns to extend the planning horizon and improve the long-term sustainability of an enterprise.

Information Model

Roadmaps are essentially "decisions put to time". A simple roadmap will show a portfolio of products (capabilities, features, etc.) evolving through a series of increments or releases. In the simple case, the entire roadmap represents the Product Portfolio Decision, the rows represent product lines or families and the bars represent product release Alternatives. The bar length represents the period of time that a product release will be "in the market" generating revenue and follow-on business opportunities. The end of the bar is the point at which a release is either competitively or technologically obsolete and should be retired from service.

More comprehensive roadmaps add rows for the evaluation criteria/MOEs that represent the alternatives performance against these criteria. As shown in Figure 7, such roadmaps forecast the evolution of decisions, alternatives and alternative performance/effectiveness, all in a very compact visualization.

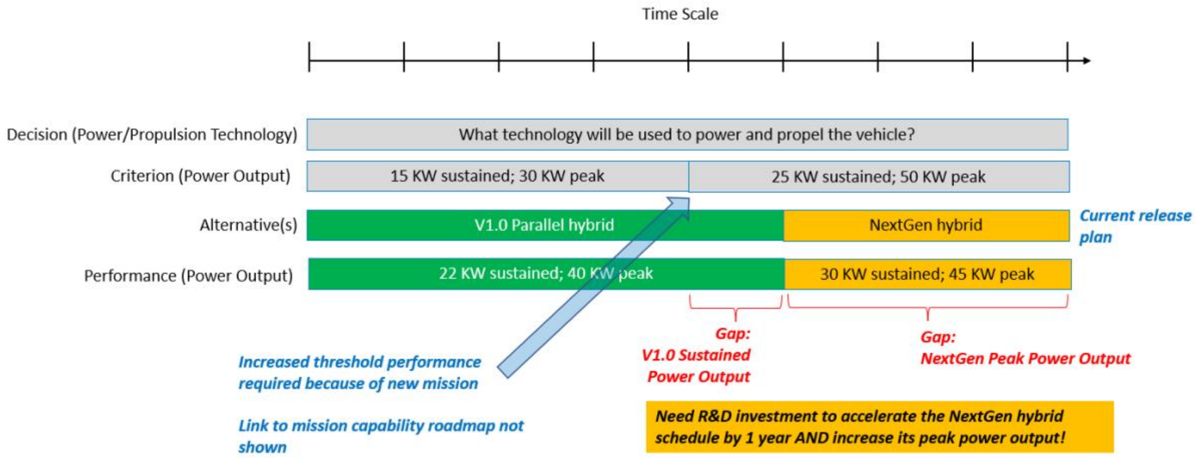


Figure 7: Decision-to-Roadmap Traceability Example

New or Improved SE Capability

The author has worked with numerous technology-focused organizations that use roadmapping for a variety of enterprise use cases. These organizations often treated roadmapping as a distinct discipline (silo) that was decoupled from strategic or product-focused decision-making.

It was common practice to appoint a roadmapping czar to create enterprise roadmaps, but these roadmaps were poorly aligned with the decision-making that was happening across business units and projects.

Recognizing that decisions and roadmaps share and can be visualized from a single source of truth, enables an enterprise to identify capability gaps, align resource investments and reduce organizational complexity.

Challenges

Significant educational outreach is needed to convince organizations that *Roadmaps = Decisions Put to Time*. Organizations with roadmap and decision-making silos may need an epiphany or competitive crisis to motivate change. Leading systems engineering tools and roadmapping tools are poorly integrated.

Emergent Properties of the Decision-Centric Approach

For simplicity, the various constructs and resulting capabilities that comprise a decision-centric approach to engineering have been described individually, but we know that there will be emergent properties of the combined decision-centric approach that may create additional capabilities and value.

Of particular interest is the entire requirements-to-decision-to-requirements (R-D-R) traceability model. This part of the Digital Thread unifies what may be perceived as a set of independent processes:

- Functional Decomposition
- Requirements Allocation
- Decision Analysis
- Design
- Requirements Derivation
- Requirements Traceability
- Design Traceability

Although experienced engineers recognize the need for recursive application of systems engineering principles and practices at every branch of system decomposition, fewer can explain the logic behind such recursion.

If it's true that requirements drive decisions and decisions create new requirements, then the traditional requirement-to-requirement traceability model is significantly flawed. The actual derivation path for any requirement is better represented with the many-to-one-to-many topology shown in Figure 8.

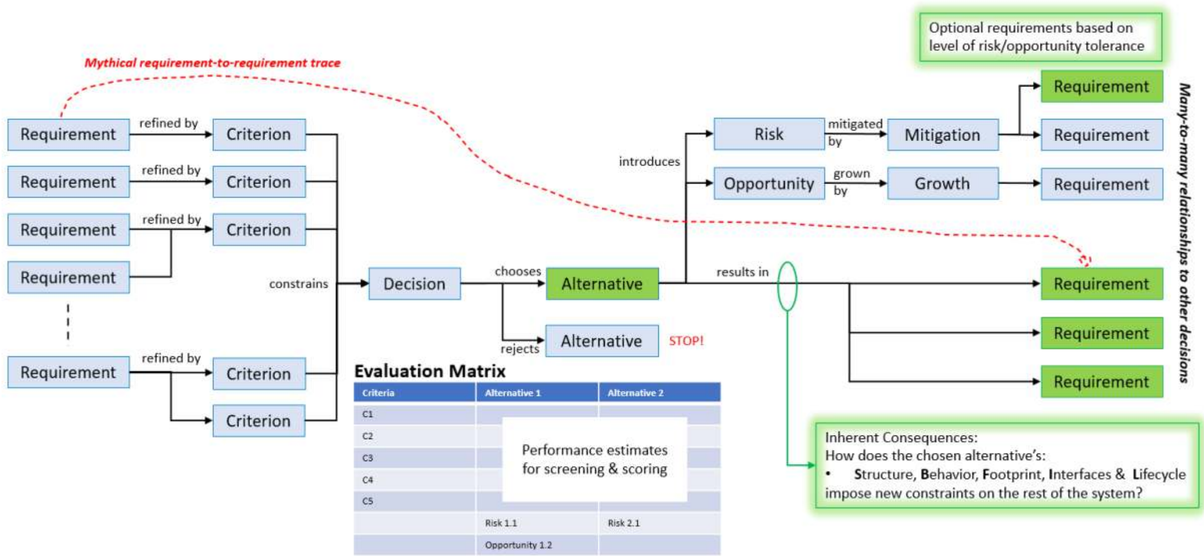


Figure 8: Requirement-Decision-Requirement (R-D-R) Traceability Model

A decision is fundamentally an integrative mechanism that balances multiple requirements and related goals (a subset of the total system requirements and value model as bounded by the decision scope) and chooses a course of action (alternative) from among many possible solutions.

The rationale behind the alternative selection is comprised of performance estimates and risks/opportunity assessments, typically visualized as an Evaluation Matrix. For a specific decision, a few criteria and perhaps risk or opportunity tie-breakers will be the discriminators that lead to commitment to a specific alternative. The alternative chosen will have inherent consequences that are part and parcel of its definition, in particular its structure, behavior, footprint, interfaces and lifecycle. These inherent consequences can be expressed as derived requirements which ***at the point of decision*** become requirements levied on the rest of the system.

Alternatives are subject to uncertainties that may be expressed as risks or opportunities. Risk mitigation actions (preventive, contingent and monitoring) and opportunity growth actions (promoting, exploiting and monitoring) may also be built back into the alternative to improve the expected value delivered by the solution. These actions can be thought of the source of optional requirements that become mandatory as soon as a decision is made to implement the actions as part of the alternative's definition.

Figure 8 shows a "mythical" R-R trace between a requirement that *drives* a decision and a requirement *derived from* the decision. As explained above, the derivation logic must consider the entire decision process to determine if a specific requirement/goal was a significant discriminator in the alternative selection. It is the entire decision that is the source of all downstream requirements.

It should give us pause when we consider that nearly all system modeling languages and supporting MBSE tools lack an explicit Decision class, given that it is the decision process and approval event that gives birth to all:

- Physical architecture elements and their combined into solution architectures.
- Allocation relationships between requirements and next-level physical architecture elements.
- Requirements derived from the chosen solution alternative.
- Risks and opportunities associated with the chosen solution alternative and the mitigation/growth requirements to enhance the value of these solutions.

The R-D-R traceability model is particularly valuable when proactively managing change. Adding 20% to the threshold value of a performance requirement may overturn a previously-made design decision to switch "best" from Alternative A to Alternative B. Alternative B may have a significantly different structure, behavior, footprint, interfaces and lifecycle which leads to invalidating most, if not all, of the requirements derived from Alternative A and adding new requirements specific to Alternative B. This creates a ripple effect as these requirement changes flow through to other decisions. Without the R-D-R trace visualizing, understanding, managing and communicating this ripple effect is difficult and prone to errors.

Conclusions

Decision-making is essential to the successful engineering of solutions to any type of problem. Decision patterns and the associated information model and methods can be applied to multiple types of projects or initiatives; common use cases include:

- Jump-starting a system/product development project.
- Technology refresh in existing system or platform.

- Crisis response: Rapid problem framing – innovation jump start.
- Entrepreneurship: New business design/launch.
- Organizational strategy refresh.
- Requirements analysis, i.e., reverse engineering stakeholder decisions to understand the source and "firmness" of requirements.

PPI can help you apply the power of decision patterns to your engineering challenges. Look for further announcements concerning our decision-focused services.

In the meantime, please inquire if you have near-term interest in a decision jump-start (blitz) for your business or project to address any of these use cases. Contact the author at jfitch@ppi-int.com or PPISyEN@ppi-int.com to communicate your interest.

References

[1] Kepner, C. and Tregoe, B. 1997. The New Rational Manager: An Updated Edition for a New World, Kepner-Tregoe, Inc., Princeton, NJ (US).

[2] Fitch, J.A. 2016. "A foundation for great thinking", https://decisiondriven.wordpress.com/2016/10/14/a-foundation-for-great-thinking/, 14 October.

[3] Mendonza, P. and Fitch, J.A. 2013. "Integrating System Models around Decisions". Paper presented at 23nd Annual International Symposium of the International Council on Systems Engineering, Philadelphia, PA (US), 24-26 June.

[4] Fitch, J.A. 2006. "Decision Driven® Design – Achieving Product and Process Breakthroughs". Workshop participant guide, delivered in multiple workshops during 2006.

[5] Fitch, J.A. 2009. "Exploiting Decision-to-Requirements Traceability, briefing to NDIA CMMI Conference, Denver, CO (US), 9 November, 2009.

[6] Mendonza, P. and Fitch, J.A. 2012. "Decision Management (DM) as the engine for scalable, cross-domain Systems Engineering". Paper presented at 22nd Annual International Symposium of the International Council on Systems Engineering, Rome, Italy, 9-12 July.

About the Author

John Fitch is a Principal Consultant and Course Presenter for Project Performance International. John brings over four decades of systems engineering, engineering management, consulting and training experience to the PPI team. In 2012, John was certified by INCOSE as an Expert Systems Engineering Professional (ESEP).

Within the field of systems engineering, John's career has focused on decision management, requirements management, risk management, systems design & architecture, product/technology road-mapping and innovation. In addition to defense/aerospace, John has guided initiatives in domains such as communications systems, software, energy, nanotechnology, medical devices, manufacturing systems, knowledge management and business process improvement.

One Old Systems Engineer's Thoughts for New Systems Engineers

by Stuart Corns

Email: stuart.g.corns@boeing.com

Authored for PPI SyEN

Introduction

When asked what might I pass on to a new systems engineer just starting out in this field, I kind of drew a blank. Having worked almost 40 years as a systems engineer (SE) at one of the largest global aerospace manufacturers, I never have really thought about it, even though I have provided training and guidance to quite a few new systems engineers over the years. New engineers, straight out of college, often ask for my expectations when they come into our SE group, but I truthfully really have none when they start. I know that they have raw potential, in that an engineering or some other technical curriculum has been completed. This demonstrates a certain intelligence, the ability to think, and the ability to solve problems. Other than some common sense and the ability to get along and play well with others, these things are really all that are needed to be successful.

To what level and how quickly individuals develop their individual systems engineering talent, skills, and abilities remains to be seen, and it will vary from person to person. It will be another three to five years of working in this field before a new engineer will add much value to a seasoned System Engineering Integration Team (SEIT). At least, that has been my experience. New personnel are an investment. It takes a lot of time and team effort to bring on new members and train them, as there is much to know both technically and administratively. In fact, at first, new members are probably going to be more of a hinderance than a help in many respects. Even when you bring on new team members that have some system engineering experience, it will take time for them to acclimate, especially when you are in the middle of a project. The transition time necessary for the individual is a little more when the new team member comes from outside the company rather than inside the company. This is mainly due to differences in company standards, processes, and procedures that will take time to learn. Something to remember: when a manager tells you that they plan to hire or add a few new engineers to your team to help you meet a project schedule, they are adding several additional tasks to your already overly ambitious workload.

I could go over in detail many system engineering activities and topics, but there are plenty of excellent materials on systems engineering that are out there to read, digest, understand, and improve upon. The basic systems engineering tenets and concepts have really not changed that much over the years; however, technologies have continued to advance at a rapid pace and have changed considerably over the years. If you are working for a large company, they will have a myriad of instructions, procedures, and guidelines that you must learn, adapt, and put into practice. Establishing priorities and time management are important skills to have when starting a new job or task. My advice would be to identify at first those things that are most relevant to the work that you are doing at the time. Learn the materials, learn how to apply them, and work very hard at it. Project experiences are different depending on your assignments, it may take several projects under your belt to have good overall perspective and appreciation of everything that needs to be done.

Your experience may come in pieces on different programs rather experiencing a single project from its start to fielded product. Things that work well on one project may not work that well on another one. Just be patient, stick with it, and continue to build on what you know and what you learn over time.

I never even heard of systems engineering when I was hired into the company in the 1980s. My degree is in electrical engineering, but I was fortunate to be placed with several outstanding older systems engineers (self-identified as system engineers at the time) in Research and Development (R&D) to receive my introduction and initial training in classical system engineering. I worked on weapons systems in the Navy which enabled me to relate well with everything that I was doing at the time. I already used military standards, specifications, handbooks, and other publications. I incorporated changes to manuals and publications distributed to the fleet, so the change processes felt familiar, and made sense. After about a year or so on the job, I saw in our company paper where the Chief Engineer for our site had said that what the company needed was a good systems engineer. I took it to heart and set out to become a good systems engineer. While my career path at the company was decided that day, it took me a while to figure out that the quote from the paper was really only a figure of speech.

I have decided not to address any specific systems engineering technical or administrative topics, methodologies, etc., in this article. But what follows, based on my experience, are a few items worth highlighting for the new engineer.

Critical Systems Thinking:

Critical systems thinking (CST) seeks to combine methods and practices from multiple engineering and supporting organizations within the design environment to develop and produce a design that will address the stakeholders' needs. Your work will be vital to other engineering teams and organizations supporting product design, development, test, and production. Learn what each team does, how they use your work, and how any changes to your work down the road might affect them. There is often overlap between individual team members in many critical skill areas. The more skills that you possess in various areas of engineering, the better positioned you will be for retention when times are tough. The people with more numerous and more critical skillsets will be the ones targeted for retention on a program more often than not.

Communication:

There is really nothing that is more important than clear and concise communication. Communication involves listening as well as speaking and writing. Create a collaborative environment that encourages input and involvement from every team and every team member. Have open and honest team discussion, and always treat everyone and their contributions to the team with respect. Teams generally comprise individuals with varying levels of experience. The most senior or loudest individual doesn't necessarily have the best solution, but then again neither does a voice that is never heard.

There is always someone in charge even in a good collaborative environment. Once team or other program decisions are made, support them to the best of your ability even if you do not agree with them. There are many paths that might lead to a desired outcome, admittedly some will be much better than others. Seek feedback. When someone or an interfacing teams says that they understand it doesn't necessarily mean that they actually do understand. The same or similar terminology often has different meaning between personnel on different teams, subtle or otherwise. Sometimes you may think you are, but may not be on the same page. Partial understanding or understanding in degrees can be problematic depending on what you are working.

When you become a team lead it may be helpful for you to have a better understanding of different personality types and corresponding communication styles. I have found Briggs Meyer or DiSC® personality type assessments for people to be useful in what to expect from people having a different personality type than my own. Engineers and Managers often have a different personality type and/or preferred style. Understanding these differences will help you work with others. Try not to be offended by criticism from others, but rather try and understand if it is warranted or if something may need to be done to address a valid criticism or concern. Many people passionate about a subject tend to go way overboard in the opposite direction just to prove their point. I have found over the years that the truth or at least an acceptable compromise can usually be found somewhere in the middle of most disputes.

Learning:

Learning is lifelong. As the saying goes, the more you learn the more you know. The more you know the more you grow. I'm getting to that forgetting stage of life now, but there is no need to go there since I plan to retire soon. I have observed the more that I learn, the more things I see to learn about. It is often said that systems engineers' knowledge is "a mile wide, but only an inch deep." I myself have often said that a systems engineer has an opportunity almost daily to show the rest of the world how much they really don't know. It does get better though, at least for me it is no longer a daily occurrence. Most of my experience is in the area of Defense and Commercial Space. The Defense Acquisition University (DAU) provides a good Design Acquisition Guidance handbook and other publications that are useful on Department of Defense (DoD) programs. I recommend reading the materials prior to the start of a project, during the project, and again at the end of the project. You may only understand a small part of certain things on a first read, but as you gain experience the subject matter will become clearer with each read, and by the end of the project most of its content should make sense to you. Thereafter, you may want to reread parts of it to reinforce your understanding of certain things from time to time.

Networking:

Networking with other systems engineers and specialty engineers on technical matters is important. Seek out the best people available and work with them to the best of your ability. I still kick technical issues around with a few people that I met while working my first program in R&D almost 40 years ago. Others that I have worked with over the years are now company leaders and technical experts in their respective fields of work.

At some point you may reach a level where very few people may have the same level of technical knowledge and experience in a certain area to discuss a topic. Imagine discussing differential equations with a first grader. How much would most 6 year olds really get out of the discussion? People must have a certain level of common knowledge and experience as a basis to understand certain concepts. Consider becoming a member of a technical society or organization such as INCOSE, AIAA, IEEE, etc. This is a good way to get additional exposure, discuss relevant topics in the field of systems engineering and provides a forum to present at technical symposiums or just to members at local chapter meetings to receive feedback or additional insight from others.

Tailoring:

The ability to tailor materials such as contracts, contract data requirements, statement of work requirements, specification requirements, data item descriptions, references and limit levels of applicability, as well as other related documents, processes, procedures is a must. Tailoring is a skill that is very important to develop, but it typically takes time and some understanding of the work involved.

The ability to do this quickly and well during contract negotiations and at the beginning of a program can save countless hours of needless work and frustration down the road. It has been my experience that one size does not fit all, and in point of fact it fits many rather poorly.

Tools:

When I first started college, we were using slide rules to crunch numbers and punch cards to input data into the compiler to run our programs on the IBM digital 1130 computing system at my university. The laptop computers that many people use today have more power than Cray supercomputers had back when I was in school. I can still use a slide rule to solve problems, but I am not nostalgic for them by any means. I can say with some level of arrogance that I can solve most any mathematical problem, old school, without a calculator or computer, given a little time.

Engineering tools are meant to save you time and free up some of your time to do other things. In my opinion; however, too many promising young systems engineers get pulled into maintaining a systems engineering tool or tools rather than learning the systems engineering behind the tool(s) that are necessary to generate quality products. There are a lot of good systems engineering tools available, but you need to know what and when as well as how to use them. We might have a lot of systems engineering tools in our bag, so to speak, but depending on what it is that you are doing you don't need to drag them all out and use them to check a box. Whilst you could use a sledge hammer to drive a nail, it makes more sense just to select and use a carpenter's hammer. If you have a pneumatic nail gun that's great, given it can be used where the nail needs to be driven and you are driving several nails.

In a similar vein, there is no need to functionally decompose a piece of Off-the Shelf (OTS) equipment just because you know how to do functional decomposition and have the tools. Because it is an existing equipment item, you could just identify the functions that it does that you are using.

There has been an alarming trend in recent years for managers and/or outside auditors to look at a list of SE work products, and demand to see where some SE product is for a system without an understanding of whether or not it needs to be done. Be prepared to rebut as they point to a myriad of company processes so as not to get saddled doing unnecessary work - to placate someone in a position of power that clearly doesn't understand what needs to be done. Sometimes it seems these days that I spend more time explaining what doesn't need to be done and why, than I do working on those things that do need to be done.

Risk and Opportunity:

Understand your limitations, but don't be afraid to get outside of your comfort zone and take on work in areas where you have limited experience even if you are being forced into a challenging situation. It is necessary sometimes to fill a gap for a period of time until a Subject Matter Expert (SME) can be found so that you can step aside, or have developed the skills necessary to become that SME. You might see this as a risk at first, but it can also be viewed as an opportunity. Looking back on it, some of my best experiences happened as a result of taking an opportunity or risk working outside of my comfort zone.

There are also tracked risks and opportunities on programs that need consideration. Something to remember about program risk is that identified risks can often be mitigated to an acceptable level, but once a risk is incurred it is then considered an issue and should no longer be reported as a risk.

Peer Review:

It is a good practice to peer review systems engineering products by the whole team internally before sending them out for external review, approval, and/or release, time permitting.

While only certain individuals may have been assigned formal review responsibility, I have found it useful for the whole team to see all review inputs, redlines, comments, etc., that are received and the final update even when they cannot support the review. This practice helps less-experienced team members better understand the review process, and what to expect when they are developing similar products in the future. Sometimes technical meetings are set-up to go over review comments and what changes will be made, if any, to a review item. A SharePoint repository (or the like) that has an organized file structure with access limited to team members may be useful to the team as well.

Mentors:

I am somewhat autodidactic when it comes to systems engineering; however, over my career I have rubbed elbows with some fairly well known systems engineers. I have help start both NCoSE and INCOSE chapters and served on them. I also served as an AIAA Systems Engineering Chair about 25 years ago while working with NASA at Johnson Space Center (JSC) on the International Space Station (ISS). I have never been a mentee in a formal mentor-mentee relationship, but I have learned a lot from working with many other systems engineers over the years, including those from other countries. I have been the mentor in both formal and informal mentor-mentee relationships. I have learned a lot from mentees in those relationships as their education was more recent, different, or more extensive than my own.

Let's face it, when I graduated from college, the Mac SE was about the hottest thing around. We had one in the lockup area where I worked. Most of the senior engineers on the Research and Development (R&D) program I was working were 65 to 70 years of age, and designers still using drafting tables. I walked into a meeting one day a little miffed that I couldn't get onto our one Mac SE computer to edit a specification that I was writing. When asked what was the matter, I told them, and then added that one of these days, every engineer would have a personal computer at their desk. There were no less than thirty senior engineers and managers in the room. I think that it was probably the biggest and the longest laugh in response to something that I have said to this very day. The lead engineer in the conference room said, "yeah, like that is E-V-E-R going to happen".

Nowadays when a young engineer says something that seems a little futuristic or somewhat off kilter to me, I tend to quietly sit back and ponder just a little bit before laughing it off. The bottom line is that mentoring relationships can be good for you, but take them with a grain of salt. You need to think for yourself. I totally respect it when a mentee of mine considers what I have taught or told them, but then decides to do something a bit differently than I may have taught or advised them. Typically, I would try to identify any anticipated problems that might result from what they planned to do differently (given I was asked), or I might just simply say that I do not know what may happen as I had never done it that way. There is more than one way to skin a cat, but again there are going to be some ways that are much better than others.

Boundaries:

Being risk averse and playing it safe by keeping things the same or very similar to the tried and true is not always the best strategy. It may not be what a customer wants, needs, or even the best way to proceed to meet a requirement or need.

Don't overpromise or promise the impossible, but don't be afraid to push some boundaries that are somewhat outside of the norm. You need to understand and be upfront on potential cost and schedule risks or not overstate opportunities associated with the implementation of new processes or technologies. There are certain things that you cannot know until they are tried. You should trust the process and leadership to make the right decisions. They won't always from your perspective, but that's just life. Get used to it. Get knocked down five times, stand-up six.

Technological progress has been so rapid in recent years that parts of systems may be somewhat obsolete before a system is even fielded. Consequently, there is an immediate opportunity for technology insertion then during system sustainment. We once integrated a large screen plasma television on an aircraft to meet a customer requirement. This was before large screen LEDs televisions were available, only to then replace them a few years later. We recognized at the time that plasma technology would be affected by low cabin pressures where the LED technology would not, but at that particular point in time there were no large LED displays. These are exciting times in that we are on the brink of or at the beginning of a new technological revolution as a result of computational and information storage advancements, workforce and data interconnectivity, materials and process advancements, etc. Digital technologies are fundamentally changing business models, institutions, and society as a whole. We are only in the infancy stage of what can be accomplished using additive manufacturing, quantum computing, and many other maturing technologies.

Silence:

Don't be a silent team member. Just because you may not have as much experience as other members of a team it does not mean that your input or opinion is not valuable to the team or does not count. Active participation by all members in a team regardless of experience level is very important. Holding something inside that might be valuable does not help anyone, it is like an engine that is missing on one or more of its cylinders. An engine is much more powerful when it is hitting on all cylinders. This is true for the lead as well. Trust the team.

I once worked a project where I had certain team members working on different aspects of a project that I was trying to connect; however, I was not sure that we could ever get to the level that I envisioned. Over time the team began to see how this work might come together and be useful to the program. One day during a team meeting one team member got up, went over and locked the door so that no one could come in or go out of the room. The team members said that they had been talking with each other, and realized that I had something bigger in mind with regard to what I had each of them doing. I was then informed that I could not leave the room until I came clean. Somewhat amused, I confessed that I was holding out on them. I said that the reason that I had not shared what I was thinking with them was because (a) I was taking it slow because I didn't know if we could even do it and continue to satisfy our current work responsibilities, and (b) it included things not assigned to us that other teams should have been working at the time. I shared my vision with them. Initially they were pretty stunned by what I had shared, but at the same time totally bought what might be done and the challenge. The team divided the additional work up in relation to what each team member was already doing without any further input from me, and started investigating how to pull all of the pieces together. The only thing that I did was to share a vision that I had with my team, nothing more. I learned a lot that day. In reflection, I also realize that had they had not forced me to share my vision with the whole team that day the work never would have been accomplished. The team was later nicknamed "the BORG" by others, and received several awards for its innovative contributions to the overall program.

Stand Your Ground:

Believe in yourself. When you believe you are right stand your ground, but be open to what others may have to say. It has been said of me that he is pretty hard to convince unless you have a good argument, but if you do have a good argument he can be convinced. My message here is be confident in your abilities and decisions, but always keep fair and open mind. There may be other important aspects that you have not yet considered.

Change:

Change is just as inevitable as death and taxes. Don't be afraid of change, it is going to happen, albeit it will not always for the better. I have experienced many changes that seem to me to have been a step or two backwards. On the other hand, some changes that I did not care for initially I later determined to be quite useful. Be patient and bad changes will typically be corrected at some point sooner rather than later.

Change for the sake of change is counterproductive, disruptive, and never a good thing to have happen. Embrace those changes that are good, but either way you are going to have to deal with them. It is hard to argue with reality no matter how hard that you wish it could be different. We live in the real world, which is why I experienced that laugh about personal computers early on in my career and on that same note you can also see that things do change. In fact, you can become the change agent if you are able to initiate change at the right time and generate sufficient momentum to get to critical mass within a group or community of practice to enact a change.

Reflection:

There is no need to wait until the end of your career to reflect on what you have done. As an engineer you will be asked and will have to answer questions all of the time. You will work with missing or incomplete information and data. Questioning yourself or a little introspection from time to time comes with the territory, and it is a good thing. You cannot always wait on answers or have all of the data that you would like to make decisions. Leaders must sometimes proceed with the best data that they have available to them at the time. In certain instances, any decision is better than no decision being made at that time.

At such times you don't know what you don't know, but you do know that hindsight will be 20/20 for the critics. In hindsight it is often clear to you where you were right, wrong, lucky, unlucky, should have waited, or waited too long to do something. When I was in the Navy working with weapons, it was said that the Naval Ordnance Publications were written in blood. You may hear managers and other engineers talk of scars or lessons learned from previous programs. It is prudent to listen to those lessons learned, if they are applicable to your situation, otherwise those mistakes may be repeated.

It should be understood that analysis and iteration are inherent to the systems engineering process. It is necessary to make course corrections and/or adjustments due to requirement changes, design decisions, system analyses, systems testing, reliability, maintenance, manufacturing and many other possible considerations that could influence a system design. This has nothing to do with first time quality as new systems development typically starts out somewhat abstract or fuzzy and tends to become much more focused over time as the result of the work performed and design decisions that are made. Certain products, as they mature, are delivered multiple times during different program phases or points in time. System maturity needs to be accounted for in your metrics. Something that is considered to be of good quality at one point could be considered to be bad at another if development objectives have not been met.

Requirements

I have not attempted to cover the many things that you will have to learn to become a good systems engineer. I would feel somewhat remiss though if I did not address requirements to some extent. A systems engineer must be knowledgeable of many different requirement types and the corresponding documents types in which they are contained.

There are material requirements, standards requirements (industry and internal company), interface requirements, performance requirements, maintainability requirements, environmental requirements, material and design construction requirements, reliability requirements, safety requirements, logistics requirements, and quality requirements to name just a few.

Certain requirements should be segregated from others in documents. Contractual requirements like those appearing in a Statement of Work (SOW) identify or talk to tasks to be performed and products or services to be delivered per the Contract Data Requirements List (CDRL). The contract SOW should not contain design requirements that should be included in the Systems Specification.

Requirements should be clear, concise, and verifiable or they need to be clarified. These types of requirement issues should be identified and resolved prior to going under contract where possible, but you will learn how to recognize and deal with the bad requirements. Certain regulatory requirements are purposely written to be about as clear as tule fog, so as to be interpretable (not a good thing). Be wary when the regulator agrees it says one thing, but means another. You will learn how deal with it, but there may be scope issues depending on when such things are identified.

Systems engineering is responsible for managing system requirements and changes to requirements baselines over the life of the product. This includes maintaining bi-directional traceability between stakeholder expectations, customer requirements, technical product requirements, product component requirements, design documents, and test plans and procedures for the system.

In Conclusion:

In general, the overall objective is actually pretty simple. *Build a qualified system that will perform and meet customer requirements in its intended operating environments*. The system is designed to be maintainable and capable of being restored to full operational performance by removal and replacement of failed end items or parts without the need for specialized tools at an operating location so that the system can be quickly returned to service. Removed parts that are repairable items are then sent out for repair. Owners/Operators typically keep a number of system spare parts on hand based on the part reliability to support system operation.

The devil of how you get there is always in the details. It has been said that it is about the journey and not the destination. I would have to agree with for the most part, but it is also about the interactions, relationships and comradery with other people during the journey that occurs.

Safe travels for those just setting out,

Stuart Corns, Senior Systems Engineer

About the Author

Stuart Corns is a retired USN veteran and practicing Senior Systems Engineer for the Boeing Company. He interned for the Boeing Company prior to receiving his electrical engineering degree. He has worked for Boeing on various Independent Research and Development (IR&D), Department of Defense (DoD) projects, National Aeronautics and Space Administration (NASA) projects, the International Space Station, and various Military Commercial Derivative Aircraft (MCDA) programs as a Systems Engineer. He helped Charter both the Huntsville Regional and the Great Plains Chapters of INCOSE, and has served in various capacities. He has received numerous technical leadership awards across his career from Boeing, NASA, AIAA and other organizations that have benefited from his skills and wisdom.

SYSTEMS ENGINEERING RESOURCES

Useful artifacts to improve your SE effectiveness

DE, DE and More DE
By PPI's Robert Halligan FIE Aust CPEng IntPE(Aus)

Indications are that engineering is in the early stages of a Digital Engineering (DE) revolution. Similar to other sources, PPI defines Digital Engineering as a trans-disciplinary integrated approach to engineering that creates and uses computer-readable models and other digital data in the conduct and support of system-related activities throughout the lifecycle of a system. This emerging DE revolution is reflected in the devotion to DE of the entire March 2022 edition (Volume 25, No. 1) of the INCOSE systems engineering practitioners magazine, INSIGHT. The March 2022 INSIGHT articles are:

1. *On the Road with Digital Engineering*
   The authors provide a historical perspective of how the digital engineering concept evolved in the United States Office of the Under Secretary of Defense for Research and Engineering (OUSD (R&E)), leading to launch in 2018 of its strategy to introduce digital engineering. Insights are offered on the lessons learned.

2. *Digital Engineering Measures: Research and Guidance*
   This article describes research conducted by the Systems Engineering Research Center (SERC) in collaboration with a government/industry Digital Engineering Measures Working Group to create the first formal measurement framework for digital transformation. The authors describe the research, formation of a causal measurement model, and initial specification of candidate measures.

3. *Systems Modeling Language (SysML v2) Support for Digital Engineering*
   This article contains much valuable information on SysML v2 that we have not seen in print before.

4. *Being Digital: Why Addressing Culture and Creating a Digital Mindset are Critical to Successful Transformation*
   The paper stresses the importance in digital transformation of cultural change occurring in synchrony with the inherent technological changes.

5. *Constructing an Authoritative Source of Truth in a Changing Information Landscape*
   This article describes many non-obvious issues including those related to requirements, best practices, and available tool capabilities in building and maintaining an Authoritative Source of Truth (ASoT), an ASoT being a fundamental ingredient of a Digital Thread implementation within Digital Engineering.

6. *Creating the Digital Thread*
   The author describes approaches to creating the digital thread, including detailed explanations of many types of trace links and their properties. An example digital thread well supports the descriptions.

7. *Distributed Cross-Domain Link Creation for Flexible Data Integration and Manageable Data Interoperability Standards*
   The author offers a strategy described as providing a more flexible, user-friendly approach than contemporary standardization efforts such as SysML v2 and ISO 10303-243:2021 - MoSSEC for engineers to specify cross-domain links between data in different application and technology domains of engineering. The application and technology domain-independent principles and tools of systems engineering, in the view of PPI, belie the claimed benefits of the approach; the reviewer for one would much rather work in a sound, easy-to-use language such as SysML v2 than in various languages specific to application and technology domains. PPI also believes that the former approach is much more conducive to team learning and sound, integrative engineering practices being used within a team.

8. *Realizing the Value Promise of Digital Engineering: Planning, Implementing, and Evolving the Ecosystem*
   This article summarizes an aid to analyzing and understanding, planning, implementation, and ongoing improvement of what is described as the Innovation Ecosystem or its components, based on a configurable model-based formal pattern created by the INCOSE MBSE Patterns Working Group.

9. *Digital Twin: Reference Model, Realizations, and Recommendations*
   Members of the American Institute of Aeronautics and Astronautics (AIAA) Digital Engineering Integration Committee (DEIC) address the development of standardized methodologies to realize the full potential of digital twins and increase their adoption across a wider range of disciplines and applications.

10. *Versatile Test Reactor Open Digital Engineering Ecosystem*
    This paper provides an interesting case study in the use of digital engineering principles to reduce risk and cost and improve schedule in the design of a 300-MW sodium-cooled fast nuclear reactor.

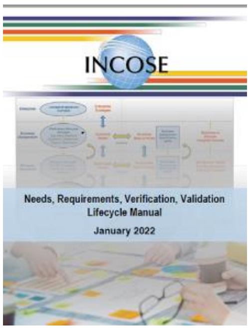
11. *Digital Engineering Measures Correlated to Digital Engineering Lessons Learned from Systems Engineering Transformation Pilot*
    This novel article describes digital engineering success measures (DESMs) and their correlation with results observed during a pilot that applied digital engineering methods and tools using an authoritative source of truth (ASoT). The pilot correlated ratings from 17 lessons learned categories to 22 DESMs grouped into four metrics categories.

12. *Acquirer Driven Digital Engineering Transformation.*
    The authors advocate the adoption of digital engineering practices by acquisition agencies.

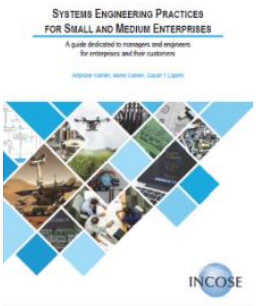
INCOSE members may access INSIGHT Volume 25, No. 1 through the Wiley Online Library or INCOSE Connect.

New INCOSE Products Released

INCOSE has released two new products, available to INCOSE members and non-members, through the INCOSE Connect.



The *Needs, Requirements, Verification, Validation Lifecycle Manual* (NRVVLM), published in January, 2022, presents systems engineering lifecycle concepts from the perspective of needs, requirements, verification, and validation (NRVV) definition and management across the system lifecycle. Acquire the manual here. (free for INCOSE members).



Systems Engineering Practices for Small and Medium Enterprises may be purchased here. (INCOSE members receive a 50% discount).

Object Management Group – March 2022 Technical Meeting Resource Hub

The Object Management Group (OMG) standards development organization conducts quarterly technical meetings to make progress on and take official actions concerning various standards. OMG has set up a Technical Meeting Resource Hub to capture the results of the 25 March Technical Meeting.

Meeting content has been captured for the following groups:

Platform Task Forces:

- Analysis & Design Platform Task Force
- Architecture-Driven Modernization Task Force
- Artificial Intelligence Platform Task Force
- Middleware and Related Services Platform Task Force
- System Assurance Platform Task Force

Platform Special Interest Groups:

- Blockchain Platform SIG
- Data Distribution Service SIG
- Ontology Platform SIG

Domain Task Forces:

- Business Modeling & Integration (BMI) Domain Task Force
- C4I Domain Task Force
- Government Information Sharing and Services Domain Task Force
- Manufacturing Technology and Industrial Systems (ManTIS) Domain Task Force
- Retail Domain Task Force
- Robotics Domain Task Force
- Space Domain Task Force

Domain Special Interest Groups:

- Systems Engineering Domain SIG

In addition to providing an overview of the issues addressed and results (decisions made) by the various groups, the links above provide contact information (group emails & URLs) through which additional information may be available. Given the impact of standards on both engineering processes and solution designs, SyEN readers are encouraged to stay current on developments in their areas of interest.

Updated Cyber Resilient Weapon Systems Body of Knowledge (BoK)

In order to better address the increasing threat of cyber-attacks and cyber warfare, the U.S. DoD has developed a curated collection of specialized knowledge designed to advance the engineering of cyber resilient weapon systems. The updated Cyber Resilient Weapon Systems Body of Knowledge (CRWS-BoK) portal was released as Version 1.3 in February 2022.

The CRWS-BoK provides resources to protect system elements through cyber resilient engineering organized under and searchable across Areas to Protect (Technology, Data and Information, Mission and System Function) and Technical Process (Architecture Design, Design Development, Requirements Management, Risk Management, Stakeholder Requirements, System Analysis and Verification). The portal includes links to glossaries of cyber resilience and defense terminology.

Congratulations to Melinda Reed, the Director of Resilient Systems, and the team from the Office of the Under Secretary of Defense for Research and Engineering (OUSD(R&E)) Office of Strategic Technology Protection Exploitation (STP&E) for completing this important work. Ms. Reed's 15 June SERCTALK on Secure Cyber Resilient Engineering (SCRE) has been addressed earlier in this SyEN edition. Register here.

Upcoming PPI Live-Online™ Systems Engineering Five Day Courses

Click here to view the full schedule or register for an upcoming course.

P006-876-1	Asia UTC +8:00 (SGT 6:00) PPI Live-Online	09 May - 13 May 2022
P006-876-2	Oceania UTC +10:00 (AEST 8:00) PPI Live-Online	09 May - 13 May 2022
P006-877-1	Europe UTC +2:00 (CEST 9:00) PPI Live-Online	16 May - 20 May 2022
P006-877-2	United Kingdom UTC +1:00 (BST 8:00) PPI Live-Online	16 May - 20 May 2022
P006-877-3	South Africa UTC +2:00 (SAST 9:00) PPI Live-Online *(Only available in South Africa)*	16 May - 20 May 2022
P006-878-1	North America UTC -4:00 (EDT 8:00) PPI Live-Online	16 May - 20 May 2022
P006-878-2	South America UTC -3:00 (BRT 9:00) PPI Live-Online *(Only available in South America)*	16 May - 20 May 2022
P006-879-1	Europe UTC +2:00 (CEST 9:00) PPI Live-Online	13 Jun - 17 Jun 2022
P006-879-2	United Kingdom UTC +1:00 (BST 8:00) PPI Live-Online	13 Jun - 17 Jun 2022
P006-879-3	South Africa UTC +2:00 (SAST 9:00) PPI Live-Online *(Only available in South Africa)*	13 Jun - 17 Jun 2022
P006-880-1	Asia UTC +8:00 (SGT 6:00) PPI Live-Online	20 Jun - 24 Jun 2022
P006-880-2	Oceania UTC +10:00 (AEST 8:00) PPI Live-Online	20 Jun - 24 Jun 2022

Spotlight: PPI Systems Engineering Five Day Course// North and South America 16-20 May 2022

PPI's 5-Day Systems Engineering Course is an excellent opportunity for those wanting to master new techniques to enhance their project outcomes, reduce rework and advance their professional development. This is not your typical systems engineering training; participants have informed us how how this course changes careers and companies.

This highly practical course, which has previously trained 11,600 students from 38 countries, is designed for every practising engineer and engineering manager, regardless of job title, application domain, or technology orientation. Whether you are a 40-year experienced principal engineer or a recent graduate, if you do not agree that you learnt new and valuable things by participating in the training, we will refund your course fee, no questions asked. This course will be taught online in the America's over 16 - 20 May 2022 by PPI's John Fitch.

FINAL THOUGHTS FROM SYENNA

Dear Reader,

It is now the anniversary of my first article for your prestigious publication, and if you are reading this the editors have seen fit to accept another contribution.

I am fortunate to be in touch with April, who is one of the authoring team for the 5th Edition of the INCOSE Systems Engineering Handbook (H5). She has generously given me some insight into what we might expect to see when it is published – see the following table.

I use the abbreviation "H5" because every time I type "SEH" into a Microsoft application, Bill Gates turns it into "SHE". I take this as subliminal support for my theory that women are naturally great systems engineers, whilst men have to work at it.

Change No	Description of change	Rationale
1	After the necessary context setting, **Value Engineering** will be the first chapter in the book, followed by **Decision Management**	Engineering is all about decision-making. Every decision should be made with respect to an agreed Value model. Having this at the forefront of our thinking doesn't guarantee good decisions, but at least we can make a respectable attempt. This topic is the last thing to be covered in H4, when eyelids are giving up the battle against gravity. The last shall now be first. Systems Engineers have to be business-savvy, and it's not a bad idea to get over this shock at the start of the book.
2	Next will come **Tailoring**	People get depressed reading H4, thinking that they will never be able to follow all those processes with full rigor. After a couple of weeks or so they get to tailoring. Then they discover that they only need execute the processes to the extent justified by the relevant business case. The usual reaction is "if only they had told me that in the first place". It's best not to tell them that tailoring could result in them *increasing* the rigor.
2	The processes will appear in alphabetical order	In H4 people are desperate to read time into what is a set of **logical** process descriptions (what and how but not when). They get trapped into thinking that the sequence of execution has to be the sequence of printing, which is only the case for a sequential life cycle approach. In alphabetical order, **Disposal** (for example) will come before **System requirements definition**, which isn't a bad idea since most requirements are rubbish.

Change No	Description of change	Rationale
3	**Specialty Engineering** will be re-named **Precious Engineering**	People might spend their whole lives building a career around a precious discipline such as aesthetics, human factors, or safety. They don't get much chance to be on stage, so they have to make the most of it when it arrives. And they can rightly sulk if not invited to the performance. The leading lady (the system architect) has to learn how to integrate all of the precious disciplines into a winning production. As per H4, the specialty disciplines will appear in alphabetical order because we don't want to upset anyone by putting their subject at the bottom of the list.
5	The new process of **System disintegration** will be introduced, reducing the scope of **Disposal**	Intended system disintegration should be planned for. (Less capable organizations will also have a lot of unplanned system disintegration to do). When I did my own car maintenance, the manual used to describe in detail how to strip things down, concluding with "reassembly is the reverse process". How rarely was that the case. When we disintegrate a system, we lose its magical emergent properties, which is what happens when we disintegrate its definition on the left-hand side of the V (reductionist thinking in action). When we integrate it, they appear, and we pray that they will be the ones we want.
6	**Complexity**	This topic will be separated from the description of Systems of Systems (SoS). A non-SoS can be pretty complex. Complexity is not an attribute of the system; rather it is an attribute of our human (in)ability to predict its emergent properties. A system that was complex 100 years ago might be considered merely complicated today. Similarly, what is complex to one person or organization may be complicated to another.
7	**Systems of Systems**	A new label will be found for this, which is likely to delay publication by 3 years or so. Any system that has more than one level in its hierarchy is a system comprised of systems. It's *not about* the structure of the system; it's about the structure of *decision-making for the system*. This should span all life-cycle stages, not just the creation and operation suggested in H4.

If you have any further suggestions for April, please email her at April@fo.ol. Contributions must be submitted by noon on 1st April. Failing that, please do email ppisyen@ppi-int.com with your thoughts.